



**République Algérienne Démocratique et Populaire**

**Ministère de L'Enseignement Supérieur et de la Recherche Scientifique**

**Université A.MIRA-BEJAIA**

**Faculté de Technologie**

**Département de Génie Electrique**

## **Projet de fin d'étude**

**Pour l'obtention du diplôme de Master en Electronique**

**Option : Instrumentation**

**Thème :**

---

### *Conception d'une régulation de la température dans un bac d'eau chaude au niveau de CEVITAL*

---

**Préparé par :**

Bouchema Sonia

**Dirigé par :**

Mr : TAFININE Farid

Mr : MAHMOUDI Nadir

**Le membre de jury :**

Mme : Idjedaren

Mr : Hanfoug

**Année Universitaire : 2018/2019**



# REMERCIEMENT

C'est avec humilité et gratitude que nous reconnaissons ce que nous devons :

Nous commençons par remercier et rendre grâce à Dieu le tout puissant de nous avoir donné le courage et la volonté de mener à bon terme ce travail.

Nos remerciements vont en particulier à Mr TAFININE. Farid, notre promoteur de l'université, qui nous a dirigé et beaucoup aidé afin de mener à bien ce travail.

Nos remerciements vont en particulier à Mr MAHMOUDI. Nadir, notre promoteur de l'entreprise CEVITAL, qui nous a dirigé et beaucoup aidé afin de mener à bien ce travail.

Nos remerciements aussi les membres de jury qui nous ont fait l'honneur d'accepter le jugement de notre travail.

Nous tenons aussi à remercier tous les enseignants de l'option Electronique sans exception.

Nous tenons à remercier également l'équipe maintenance raffinerie d'huile.

Que tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail, trouvent ici nos sincères reconnaissances.





# DÉDICACES

J'ai le grand honneur de dédier ce travail :

A la femme qui a consacré sa vie pour que la mienne soit meilleure, par l'aide et le soutien qu'elle m'a réservé, pour que je puisse continuer mes études, du fond du cœur, je te remercie, toi ma mère.

A mon père qui m'a apporté toute l'aide dont j'avais besoin.

A mes frères (dada Nabil, Brahim, Soufiane, Hemza, Fares).

A ma très chère sœur Kenza.

A toute ma grande famille.

A mes amis (Basma, Lynda, Amel, Kenza, Youcef, Kamal, Karim).

Et tous ceux qui m'ont aidé dans mon travail.



# Liste des abréviations

## A

**API** Automates Programmable Industriel

**AP** Automates Programmable

## C

**CPU** Central Processing Unit

## P

**P** Proportionnel

**PI** Proportionnel Intégral

**PID** Proportionnel Intégral Dérivé

**PLC** Programmable Logic Controllers

## S

**SPA** Système programmable automatisé

## T

**TOR** Tout Ou Rien

# Liste des figures

1.1	Les étapes de raffinage de l'huile.....	5
1.2	Les différentes étapes de recyclage d'eau .....	6
1.3	La vanne régulatrice .....	10
1.4	Clapet papillon.....	11
1.5	La vanne interne.....	13
1.6	Cavitation .....	14
1.7	Aspect extérieur d'une pompe centrifuge.....	14
1.8	Principe de fonctionnement d'une pompe centrifuge.....	15
1.9	Vanne pneumatique.....	16
1.10	Indicateur de pression.....	17
1.11	Indicateur et transmetteur de pression.....	17
1.12	Transmetteur de pression.....	18
1.13	Capteur de niveau.....	19
1.14	Le bac d'eau.....	20
2.1	Notion de correcteur PID.....	24
2.2	Le rôle des paramètres PID.....	25
3.1	Structure d'un système automatisée.....	29
3.2	Situation d'un automate dans un système automatisée de production.....	33
3.3	Les différents types d'automate.....	33
3.4	Structure interne d'un automate programmable.....	34
3.5	Cycle d'exécution d'un programme.....	37
3.6	Le temps de repense total (TRT).....	38
3.7	Structure modulaire d'un API S7-300.....	40
4.1	Nouveau projet sur STEP 7.....	48
4.2	Choix du rack.....	49
4.3	Choix de la CPU.....	49
4.4	Choix des modules d'entrée/sortie.....	50
4.5	Structure d'un projet .....	50

4.6	Le bloc OB1.....	51
4.7	Réalisation du programme.....	51
4.8	Chargement du programme dans la CPU-1.....	51
4.9	Chargement du programme dans la CPU-2.....	52
4.10	Ouverture du simulateur sur STEP 7.....	53
4.11	Simulateur du programme.....	53
4.12	Compilation sous WinCC Flexible Runtime.....	55
4.13	Liaison entre la station S7-300 et la station HMI.....	55
4.14	Simulateur S7-PLC SIM.....	63
4.15	Système global.....	64
4.16	Vue de régulateur PID.....	64



# Sommaire

<b>Introduction générale</b>	<b>1</b>
<b>1 Description du processus de raffinage d'huile</b>	<b>2</b>
1.1 Introduction .....	2
1.2 Description du processus de raffinage d'huile .....	2
1.2.1 Définition du raffinage .....	2
1.3 Processus de raffinage au niveau du complexe CEVITAL de Bejaia .....	3
1.3.1 Présentation de la raffinerie.....	3
1.3.2 Les différentes étapes.....	3
1.4 Description du système actuel.....	6
1.5 Cahier de charges .....	7
1.5.1 Description du problème .....	7
1.5.2 La solution proposée au problème.....	7
1.5.3 Le fonctionnement de la vanne de remplissage.....	8
1.5.4 Le fonctionnement de la vanne d'arrosage.....	8
1.5.5 Le fonctionnement de la pompe.....	8
1.5.6 Le fonctionnement de la vanne régulatrice.....	8
1.6 Description des équipements du système existant.....	14
1.6.1 La pompe centrifuge.....	14
1.6.2 La vanne pneumatique Tout Ou Rien... ..	15
1.6.3 Les capteurs .....	16
1.6.4 Le bac d'eau .....	19
1.7 Conclusion.....	20
<b>2 Généralités sur la régulation</b>	
2.1 Introduction .....	21
2.2 Définition de la régulation automatique.....	21
2.3 Objectifs de la régulation.....	22

2.4	Types de régulation automatique.....	23
2.4.1	Régulation tout ou rien (TOR).....	23
2.4.2	Régulation analogique.....	23
2.4.3	Régulation numérique .....	23
2.6	Critères de performance d'une régulation .....	23
2.7	Définition de la régulation PID .....	24
2.8	Notion de correcteur PID.....	24
2.8.1	Le FB 41 CONT-C.....	25
2.8.2	Le rôle des paramètres PID .....	25
2.8.3	Description .....	25
2.8.4	Etats de fonctionnement .....	26
2.8.5	Information d'erreur .....	26
2.9	Conclusion .....	27
<b>3 Les automates programmables industriels (API)</b>		
3.1	Historique.....	28
3.2	Automatisation .....	28
3.2.1	Définition .....	28
3.2.2	Objectifs de l'automatisation.....	28
3.2.3	Structure d'un système automatisé.....	29
3.2.3.1	Partie opérative PO.....	30
3.2.3.2	Partie commande PC.....	30
3.2.3.3	Poste de contrôle .....	30
3.3	Les automates programmables industriels .....	31
3.3.1	Historique.....	31
3.3.2	Définition.....	31
3.3.3	Domaines d'emploi des automates.....	31
3.4	Les fonctions d'un automate programmable.....	32
3.4.1	La nature des informations traitées par l'automate.....	32
3.4.2	Architecture d'un automate programmable industriel.....	33

3.4.2.1 Aspect extérieur.....	33
3.4.2.2 Structure interne.....	34
3.4.2.2.1 Une unité de traitement ou processeur.....	34
3.4.2.2.2 Une mémoire de programme.....	35
3.4.2.2.3 Une mémoire de données.....	35
3.4.2.2.4 Une interface d'entrée.....	35
3.4.2.2.5 Une interface de sortie.....	35
3.4.2.2.6 Un module d'alimentation.....	35
3.4.2.2.7 Principales fonctions.....	36
3.4.2.2.8 Traitement du programme automate.....	37
3.5 Critères de choix d'un automate.....	38
3.5.1 Présentation de l'API S7 -300.....	39
3.5.2 Description de S7-300.....	39
3.5.3 Structure modulaire du S7-300.....	40
3.6 Conclusion.....	43
<b>4 Simulation et résultats</b>	
4.1 Introduction.....	44
4.2 Présentation du logiciel STEP7.....	44
4.3 Utilisation de STEP7.....	45
4.3.1 Présentation des fonctionnalités supplémentaires.....	45
4.3.2 Les blocs de codes et les blocs de système.....	45
4.3.2.1 Bloc d'organisation OB.....	45
4.3.2.2 Bloc fonctionnel FB.....	46
4.3.2.3 Fonction FC.....	46
4.3.2.4 Bloc de données DB.....	46
4.3.2.5 Bloc système pour fonctions standard et fonction système.....	46
4.3.2.6 Bloc fonctionnel système SFB.....	46
4.3.2.7 Fonction système SFC.....	47
4.3.2.8 Donnée système SDB.....	47

4.4 Création d'un projet sous STEP7.....	47
4.4.1 Configuration matérielle.....	48
4.4.1.1 Etapes 01.....	48
4.4.1.2 Etapes 02.....	49
4.4.1.3 Etapes 03.....	50
4.5 Le principe de la programmation sur STEP7.....	50
4.6 Chargement du programme dans la CPU.....	51
4.7 Simulation du programme dans l'automate.....	52
4.7.1 Présentation de PLC SIM.....	52
4.8 Présentation du WinCC flexible.....	54
4.8.1 Logiciel exécutif SIMATIC WinCC Flexible Runtime.....	54
4.8.2 Intégration dans SIMATIC Step7.....	55
4.9 Partie programmation.....	56
4.9.1 Programmer le mode manuel et le mode automatique.....	56
4.10 Simulation .....	63
4.10.1 Simulation de programme sous STEP7 avec PLCSIM.....	63
4.10.2 Simulation du programme sous WinCC.....	64
4.10.3 Vue de régulation PID.....	64
4.10.4 Résultats.....	65
4.11 Conclusion.....	66
<b>Conclusion générale.....</b>	<b>67</b>

## **ANNEXES**

## **Introduction générale**

Les entreprises de production dans ses nombreux secteurs d'activité (agro-alimentaire, pétrochimie, automobile,...) sont de plus en plus soumises à un environnement de concurrence et de progression permanente. Pour assurer leurs avenir, les entreprises doivent s'adapter rapidement et efficacement à ces conditions. Elles doivent aussi offrir des produits et des services de qualité dans des délais courts et des prix compétitifs.

Parmi les solutions pour remédier ces contraintes, il existe les automates programmables industriels (API), qui représentent une structure permettant d'automatiser les procédés, avoir un bon rendement et une maintenance rapide.

Pour des raisons économiques et environnementales les responsables du complexe agro-alimentaire de Bejaia ont proposé la réalisation d'un système automatisé pour le recyclage d'eau osmose et la réutiliser dans l'industrie.

L'automatisation et la supervision du système de recyclage qui est le sujet de note travail, consiste à comprendre le fonctionnement du processus et étudier toutes ses parties.

Dans ce but, notre travail sera partagé en quatre chapitres :

Dans le premier chapitre on fera un aperçu sur le complexe agro-alimentaire de Bejaia, puis on donnera une description fonctionnelle de système étudié.

Le second chapitre en la présentation d'une généralité sur la régulation, objectif de la régulation et son rôle dans l'industrie.

Le troisième chapitre est consacré aux automates programmables industrielles (API).

Le dernier chapitre est consacré à l'automatisation et la supervision du système, où il a lieu de définir les organes de contrôle et de commande utilisés, puis on passera à l'élaboration du programme du fonctionnement automatique du processus à l'aide du logiciel de base Step7 et pour sa validation on utilisera le logiciel de simulation d'automate S7-PLCSIM, intégré dans le Step7. La dernière étape de ce chapitre consiste en l'élaboration d'une plateforme de supervision par l'interface homme-machine WinCC flexible, qui permettra de suivre l'évolution de procédé en fonction du temps et qui simplifie la tâche de contrôle pour l'opérateur.

Nous clôturerons notre travail par une conclusion générale.

# Chapitre 1

## Description du processus de raffinage d'huile

### 1.1 Introduction

Dans ce premier chapitre nous donnons en premier temps une description du complexe agroalimentaire CEVITAL et le processus de raffinage de l'huile et ses différentes étapes. Nous évoquons le processus de la raffinerie au complexe CEVITAL de Bejaia, en particulier la méthode manuelle de récupération d'huile issue des différentes étapes de raffinage d'huile brute, dite : fosse de relevage [1].

### 1.2 Description du processus de raffinage d'huile

#### 1.2.1 Définition du raffinage

Le raffinage est l'ensemble des opérations qui servent à transformer l'huile brute en un produit comestible en éliminant les impuretés qui le rendent impropres à la consommation. En effet, les huiles contiennent de nombreux composés : certains sont très utiles (vitamines, insaponifiables, etc.), d'autres sont nuisibles à leurs qualités (gommes, acides gras libres, pigments, agents odorants, etc...). Le raffinage consiste donc à éliminer au mieux ces composés afin d'obtenir une huile aux qualités organoleptiques et chimiques satisfaisantes. Il comprend plusieurs opérations [1] :

**-Démucilagination (ou dégomme)** : elle permet de débarrasser les huiles des gommes après leur hydrolyse par un acide.

**-Neutralisation** : les acides gras libres sont les impuretés les plus représentées dans les huiles à raffiner. L'étape de neutralisation sert à éliminer ces composés susceptibles d'accélérer l'oxydation de l'huile. Elle se fait à l'aide de la soude caustique et suivie d'un lavage à l'eau et d'un séchage.

**-Décoloration** : elle sert à éliminer les pigments contenus dans l'huile.

-**Filtration** : cette étape permet d'obtenir une huile limpide après élimination de la décolorante.

-**Désodorisation** : cette étape permet de débarrasser l'huile de son odeur désagréable par distillation sous vide poussé à une température élevée (180°C-200°C).

## 1.3 Processus de raffinage au niveau du complexe

### CEVITAL de Bejaia

#### 1.3.1 Présentation de la raffinerie

La capacité de production de la raffinerie d'huile du complexe CEVITAL de Bejaia est de 1800 tonnes par jours, qui se font en trois lignes de production :

-Deux lignes symétriques (A et B) de capacité de 400 tonnes/jours chacune qui constitue 800 tonnes par jour.

-Une ligne C de capacité 1000 tonnes par jour.

Pour plus d'information sur le complexe CEVITAL voir l'annexe 1 [1].

#### 1.3.2 Les différentes étapes

##### Neutralisation et démucilagination

L'huile brute provenant des bateaux arrive dans des bacs de stockage puis passe ensuite par un moteur à brosse afin d'éliminer les particules physiques qui pourra prendre lieu. Elle est ensuite soutirée à l'aide d'une pompe (PU), puis elle passe dans un échangeur de chaleur pour la chauffer de 25°C à 90°C, ensuite, dans un mélangeur (M1) en lui rajoutant un dosage de l'acide citrique afin d'éliminer les phospholipides. Afin d'accélérer la réaction chimique, le mélange (Acide citrique et huile brute) est transfère vers un réacteur R1 qui permet une agitation pendant 20 minutes (réaction de démucilagination), en second lieu, on ajoute la soude caustique dans le mélangeur M2 afin d'éliminer les acides gras libre (AGL).

L'huile neutralisée et démucilaginée passe à travers des séparateurs qui permettent la séparation de l'huile des pates de neutralisation par centrifugation à 90°C. La phase lourde (pate) passe vers les parois des séparateurs et la phase légère (l'huile) reste au centre.

Ensuite, l'huile séparée subira un lavage à l'eau chaude et d'un dosage d'acide citrique pour éliminer le reste des savons, puis on effectue un séchage sous vide afin d'éliminer l'humidité. A la fin de la neutralisation, l'huile est envoyée dans un bac pour la décoloration [1].

### **Décoloration :**

Dans un réacteur sous vide, on ajoute la terre décolorante à l'huile neutralisée en le chauffant avec de la vapeur. Le mélange est maintenu sous agitation durant 30 minutes pour que la terre absorbe les pigments. L'étape suivante consiste à faire passer le mélange à travers des filtres dite (filtre NIAGARA) qui vont piéger la terre contenant les pigments.

### **Désodorisation :**

Cette opération consiste à injecter de la vapeur sèche dans l'huile décolorée maintenant sous vide à une température de 225°C à 240°C. L'huile sort de l'économiseur à une température de 125°C puis passe vers un échangeur pour atteindre 110°C, enfin l'huile désodorisée est refroidie à 35°C en passant sur un économiseur, puis sur un refroidisseur.

### **Remarque :**

La figure ci-dessous représente toutes les étapes du raffinage d'huile, en précisant les auxiliaires principales de fabrication ainsi que les constituants indésirables éliminés. Notons que les déchets de la neutralisation vont être envoyés à la section de traitement des déchets qui s'effectue selon deux étapes : prétraitement et saponification.

Pendant l'étape de saponification, on réchauffe le mélange qui sera transféré dans un réacteur pour décomposition pendant une heure. Ce qui entraîne la formation de plusieurs phases. Phase eau-acide, phase huile-acide et une phase intermédiaire non traitée, elle sera envoyée à la fosse pour recyclage.

Après cette étape de neutralisation, l'huile sera lavée et l'eau de lavage est envoyée dans des canalisations entraînant une quantité d'huile dans la fosse [1].



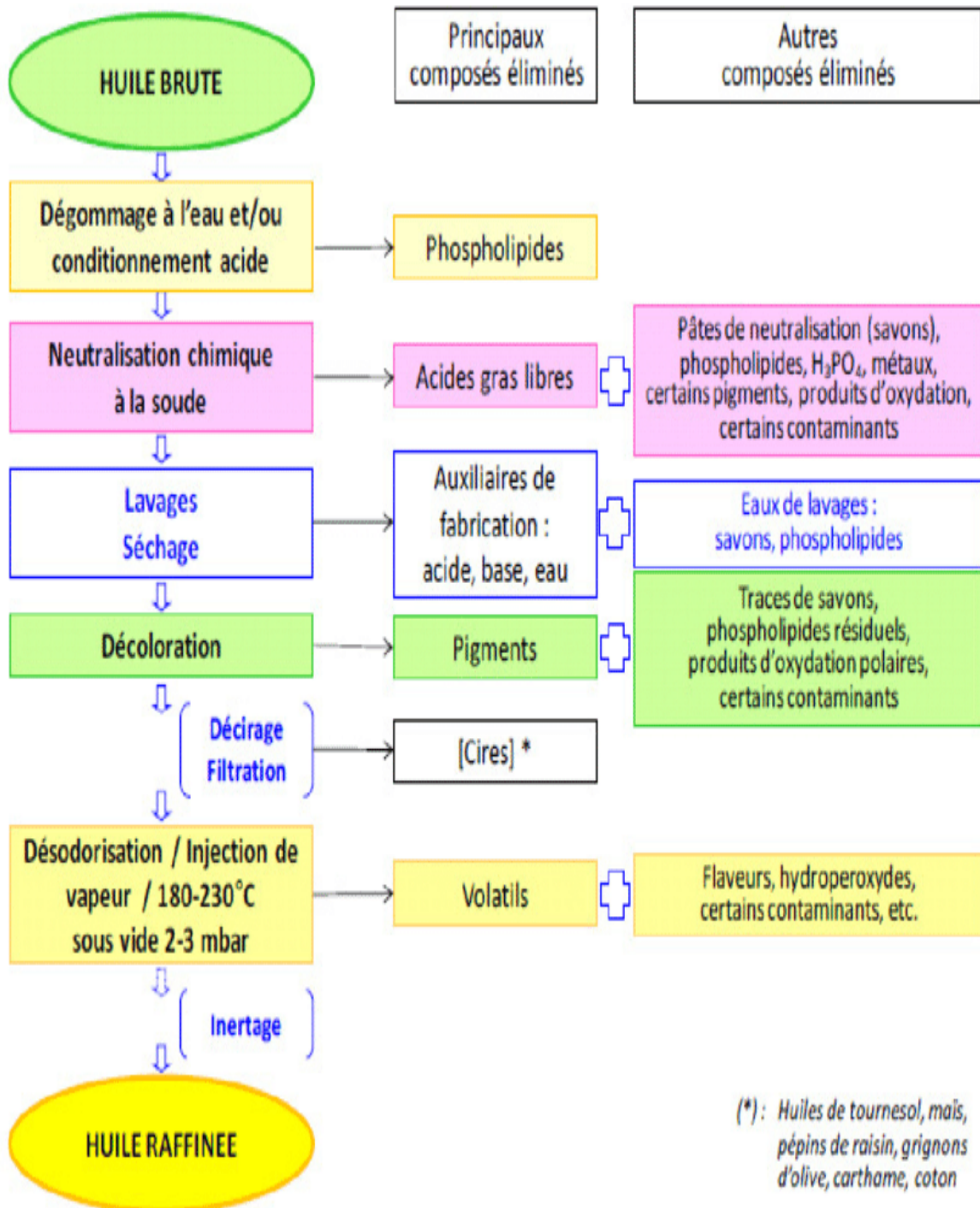


Figure 1.1- les étapes de raffinage de l'huile [1].

## 1.4 Description du système actuel

Le bac d'eau chaude sera rempli via la vanne de remplissage par l'eau osmose. Cette dernière, sera réchauffée par la vapeur de 9bar à la température désirée via une vanne régulatrice. La pompe va aspirer, par la suite, l'eau osmose qui sera dirigée vers le PROCESS (échangeurs, séparateurs...).

Cette eau revient vers le bac après un échange thermique avec une température diminuée à la consigne désirée de la vanne régulatrice. Cette dernière s'ouvre pour atteindre la température consignée.

Le schéma ci-dessous explique le fonctionnement de remplissage d'eau dans le bac via la vanne de remplissage.

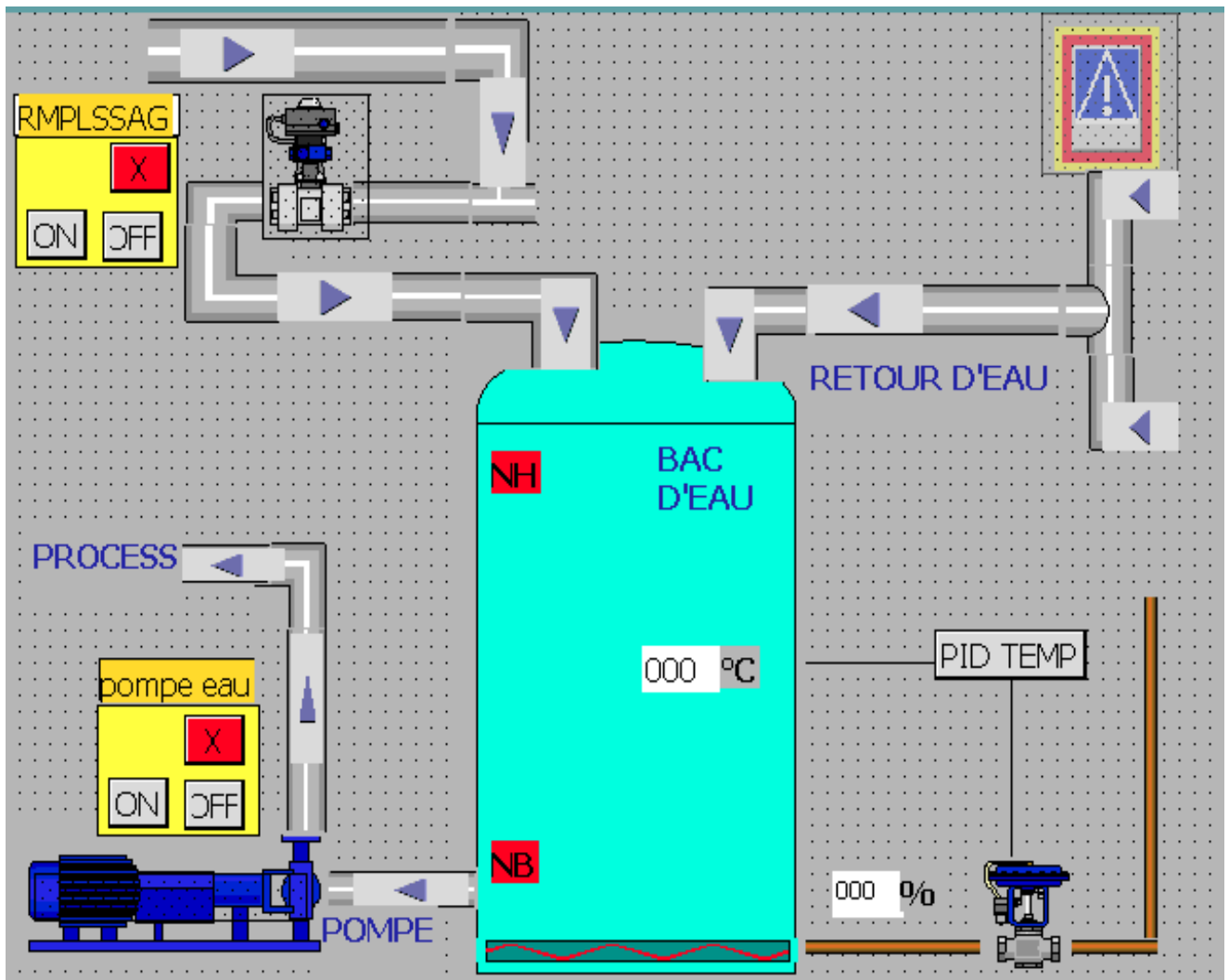


Figure 1.2-Les différentes structures de réchauffement d'eau.

## 1.5 Cahier des charges

- Nous avons élaboré ce cahier de charge afin de pouvoir remplir le bac avec la bonne régulation de température en automatisent la vanne de remplissage.
- A titre d'information, le bac est équipé d'un transmetteur de température.
- Un détecteur de niveau.
- Une vanne de remplissage.
- Une vanne régulatrice.

### 1.5.1 Description du problème

Au départ notre système est composé de :

- Un bac d'eau, une pompe qui démarre à l'indication d'absence de niveau bas.
- Une vanne régulatrice qui suit la consigne.
- Une vanne manuelle de remplissage.

Dans la conduite de retour vers le bac, on a ajouté un piquage des retours des condensats.

Dans cette conduite de retour des condensats, il ya des fuites de vapeurs qui sont dû aux purgeurs d'eau, les vapeurs seront chauffées plus que la température désirée et atteindre la température d'ébullition, ce qui formera de nuages de vapeurs, sachant que la pompe est dans l'impossibilité d'aspirer la vapeur se trouvant dans le bac mais capable uniquement d'aspirer de l'eau, tel est le problème posé.

### 1.5.2 La solution proposée au problème

Pour remédier à ce problème, c'est-à-dire trouver une solution qui permet le bon fonctionnement de remplissage, de régulation et une meilleure aspiration de la pompe, on a rajouté une électrovane d'arrosage automatique. Cette dernière permet de condenser la vapeur se trouvant dans le bac. 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.

### 1.5.3 Le fonctionnement de la vanne de remplissage

- La vanne doit être ouverte automatiquement à l'indication d'un niveau bas.
- La vanne doit être fermée automatiquement à l'indication d'un niveau haut.
- La vanne peut être ouverte ou fermée manuellement.

### 1.5.4 Le fonctionnement de la vanne d'arrosage

- La vanne s'ouvre en cas de température élevée, ce qui va provoquer, naturellement, la présence de la vapeur dans le bac. La vanne a pour rôle, la condensation de la vapeur, autrement dit, la vapeur se transforme en eau.

### 1.5.5 Le fonctionnement de la pompe

En cas d'indication d'un niveau bas d'eau dans le bac, la pompe s'arrête automatiquement. Elle s'ouvre, à l'indication d'un niveau supérieur au niveau bas.

### 1.5.6 Le fonctionnement de la vanne régulatrice

La vanne régulatrice est utilisée comme organe de réglage dans différents types de boucles de régulation. Elle permet de contrôler le débit dans une canalisation, en modifiant les pertes de charge de celle-ci [1].

### Contraintes dues au fluide et à l'environnement :

La vanne de réglage devra être conçue et fabriquée de manière à fonctionner correctement, avec un minimum d'entretien, malgré un certain nombre de problèmes posés par le fluide et par son environnement. Le fluide qui passe dans la vanne de réglage peut être [1].

- Corrosif (attaque chimique des matériaux) ;
- Chargé de particules solides (érosion, encrassement de la vanne) ;
- Chargé de bulles gazeuses, ou constitué d'un mélange de liquides et de gaz non homogènes ;
- Visqueux (exemple de l'huile) ;
- Inflammable ou explosif en présence de l'air, d'une étincelle ;
- Toxique, donc dangereux en cas de fuite ;

- Dangereux, car il peut se transformer chimiquement tout seul (polymérisation) ou réagir avec d'autres produits, parfois violemment,
- Un liquide qui se solidifie lorsque la température baisse (cristallisation) ;
- Un liquide qui se vaporise lorsque la température augmente ou que la pression diminue ;
- Une vapeur qui se condense lorsque la température baisse ou que la pression augmente ;
- Sous forte pression ou sous vide.

L'analyse approfondie et la résolution de ces problèmes doivent permettre d'assurer la sécurité du personnel et des installations, ainsi que le bon fonctionnement de la vanne. L'ambiance extérieure peut poser les problèmes suivants :

- Atmosphère explosive, corrosive, sèche ou humide, poussiéreuse, chaude ou froide ;
- Vibrations, dues par exemple à une machine voisine ;
- Parasites, dus à des appareils demandant une grande puissance électrique.

### **Eléments constituant la vanne de régulation**

La vanne est constituée de deux éléments principaux [1]:

- Le corps de vanne : c'est l'élément qui assure le réglage du débit ;
- Le servomoteur : c'est l'élément qui assure la conversion du signal de commande en mouvement de la vanne.

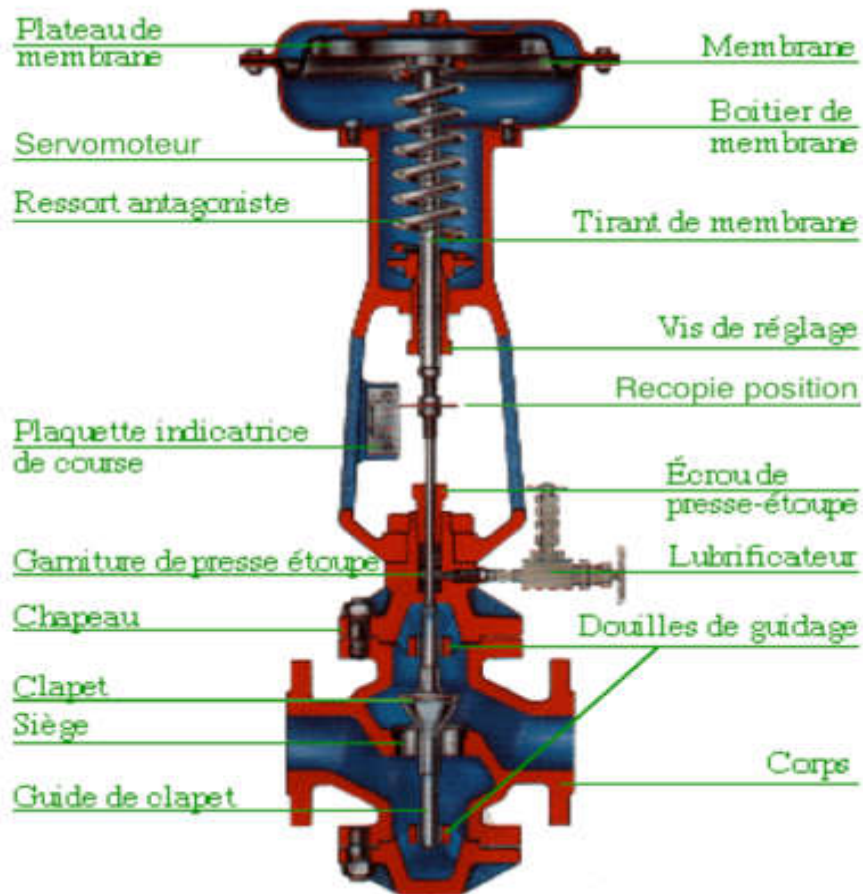


Figure 1.3 : La vanne régulatrice [1]

Et aussi d'un certain nombre d'éléments auxiliaires [1].

- Un contacteur de début et de fin de course ;
- Une recopie de la position ;
- Un filtre détendeur ;
- Un positionneur : il règle l'ouverture de la vanne en accord avec le signal de commande.

### Forme du corps de vanne

On distingue les différents corps de vannes [1].

- Le corps droit : l'entrée et la sortie sont dans le même axe.
- Le corps d'angle : l'entrée et la sortie sont dans deux plans perpendiculaires.
- Le corps mélangeur : il possède deux entrées et une sortie afin de permettre le mélange de deux fluides.

- Le corps de dérivation (répartiteur) : il possède une entrée et deux sorties afin de permettre la séparation du fluide suivant deux directions.

## Clapet papillon

L'obturateur est un disque dont le diamètre est égal au diamètre intérieur de la conduite. A la fermeture, ce disque a une surface perpendiculaire au sens du passage du fluide. La variation de la section de passage se fait par inclinaison de ce disque par rapport à la verticale. La tige de l'obturateur effectue un mouvement de rotation, ce qui est nettement préférable pour la presse étoupe (meilleure étanchéité). Cette rotation est souvent limitée à un angle d'ouverture de  $60^\circ$  à cause de l'importance du couple exercé par le fluide.

Ce type de vanne n'est réalisable que pour des grands diamètres  $DN > 4''$ .vue la surface de l'obturateur et la forme de celui-ci, il ne peut être utilisé pour des pressions très élevées. Du fait de la grande longueur de portée du papillon sur le corps (qui forme aussi le siège), l'étanchéité à la fermeture est délicate à obtenir, donc le liquide qui plaque la tige d'obturateur contre la garniture (effort transversal) [1].

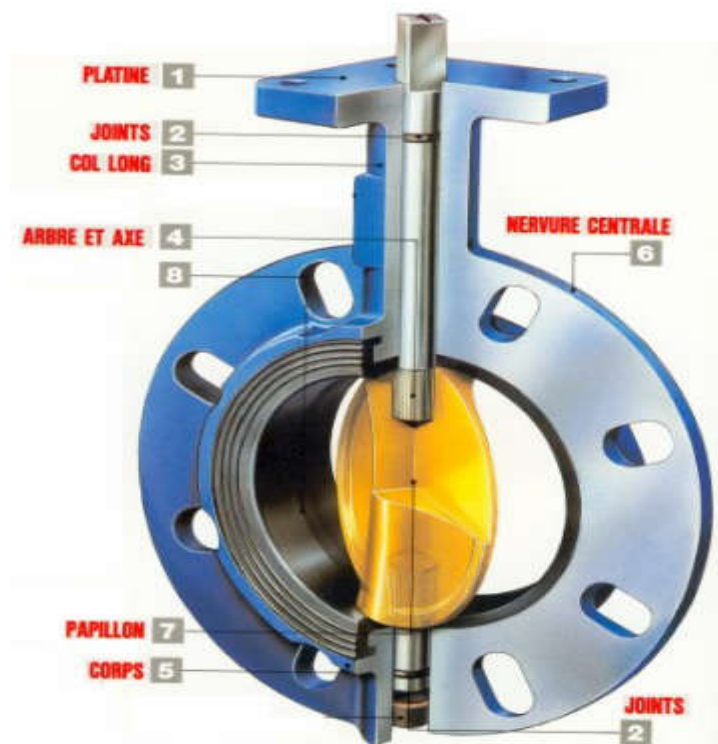


Figure1.4: Clapet papillon [1].

## Les servomoteurs

Le servomoteur est l'organe permettant d'actionner la tige de laper de la vanne. L'effort développé par le servomoteur à deux buts [1] :

- Lutter contre la pression agissant sur le clapet ;
- Assurer l'étanchéité de la vanne.

Ces deux critères conditionnent le dimensionnement des servomoteurs. Le fluide moteur peut être ; de l'air, de l'eau, de l'huile,.....).

En général, le fluide: est de l'air et la pression de commande varie de 0.2 bar à 1 bar (3-15 PSI). On distingue :

- Le servomoteur classique à membrane, conventionnel (à action direct ou inverse) ou réversible (on peut changer le sens d'action).
- Le servomoteur à membranes déroulante, surtout utilisé pour les vannes rotatives.
- Le servomoteur à piston. Utilisé lorsque les efforts à fournir sont très importants. La pression de commande peut être importante. Le fluide moteur peut être de l'air, de l'eau ou de l'huile.
- Le servomoteur électrique, utilisé pour les vannes rotatives. On associe à un moteur électrique un réducteur de vitesse permettant ainsi d'obtenir des couples très importants.

## Cavitation et vaporisation

La restriction de la section de passage présentée par la vanne et son opérateur provoque une augmentation de la pression dynamique.

Il en résulte une diminution de la pression statique plus ou moins importante selon [1]:



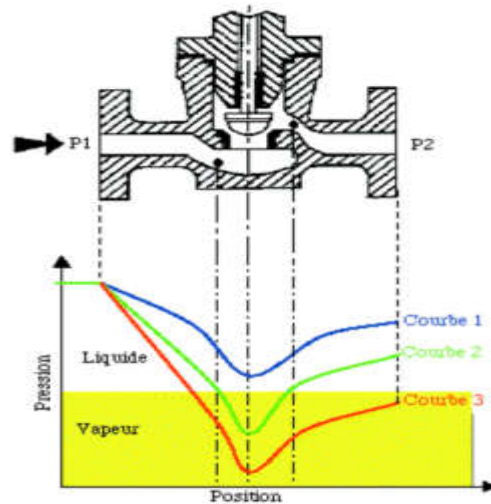


Figure 1.5: La structure interne de la vanne [1].

- La géométrie interne de la vanne.
- La valeur de la pression statique en aval de la vanne.

Cette diminution de la pression statique de la vanne doit être comparée à la tension de vapeur du liquide à la température d'écoulement, car il peut en résulter des phénomènes nuisibles à la qualité du contrôle et à la tenue du matériel.

## Cavitation

Lorsque la pression statique dans la veine fluide décroît et atteint la valeur de la tension de vapeur du liquide à la température d'écoulement, le phénomène de cavitation apparaît (formation de petites bulles de vapeur au sein du liquide, courbe 2 sur la figure 1.6).

Quand la pression statique s'accroît à nouveau (diminution de la vitesse par élargissement de la veine fluide). Les bulles de vapeur se condensent et implosent [1].



Figure 1.6 : Le phénomène de cavitation [1].

## 1.6 Description des équipements du système existant

Le système est composé des éléments : bac d'eau, des vannes, une pompe, deux capteurs de niveau, capteur de température.

### 1.6.1 La pompe centrifuge

Une pompe centrifuge transforme la puissance mécanique du moteur électrique en énergie hydraulique fournie au fluide et qui se traduit par des débits et des pressions. C'est le type de pompe industrielle le plus commun [2].

Le débit pompé est essentiellement en fonction :

- De la différence de pression entre aspiration et refoulement.
- De la vitesse de rotation de la roue. Du diamètre de la roue (vitesse périphérique).

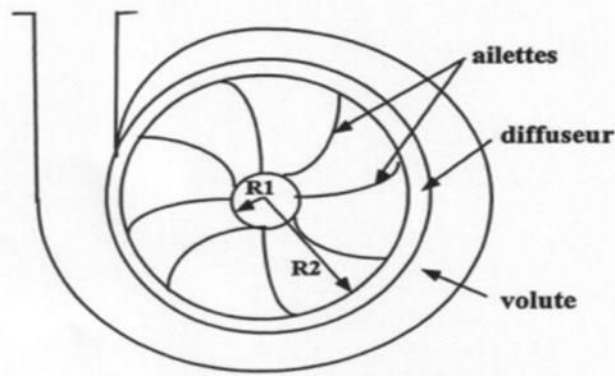


Figure 1.7 : Aspect extérieur d'une pompe centrifuge.

## Principe de fonctionnement

La pompe centrifuge est une machine à passage radial. L'eau, aspirée dans une tubulure d'aspiration, entre dans la roue axialement et déviée à une direction radiale et rencontre les aubes ou ailettes.

Un espace annulaire est placé à la sortie périphérique de la roue, qui peut être un diffuseur lisse, un diffuseur à ailettes, un colimaçon ou une volute qui se termine par un cône divergeant, permet de transformer une partie de l'énergie cinétique en pression, en réduisant la vitesse absolue de sortie de la roue, puis le courant se rassemble dans la volute en forme d'escargot. La volute se comporte comme collecteur des filets fluides pour les ramener à une bride sortie constituant le raccordement à la tuyauterie de refoulement [4], [3].



**Figure 1.8 :** principe de fonctionnement d'une pompe centrifuge [4].

### 1.6.2 Vanne pneumatique Tout Ou Rien (TOR)

Une vanne « Tout ou Rien » est utilisée pour contrôler le débit des fluides en tout ou rien. Elle exécute une action discontinue qui prend deux positions ou deux états 0 et 1 (ou 0 et 100%), c'est-à-dire ouverte ou fermée.

Les vannes tout ou rien sont utilisées pour la commande des systèmes ayant une grande inertie ou la précision de la régulation n'est pas importante [5].



**Figure 1.9:** Vanne pneumatique [5].

### 1.6.3 Les Capteurs

Un capteur est un organe de prélèvement d'information qui élabore, à partir d'une grandeur physique, une autre grandeur de nature différente (très souvent électrique). Cette grandeur représentative de la grandeur prélevée est utilisable à des fins de mesures ou de commande.

Un transmetteur est le dispositif qui convertit le signal de sortie du capteur en un signal de mesure standard, il fait le lien entre le capteur et le système de contrôle.

#### a) Indicateurs de pression

Les indicateurs de pression sont employés pour la mesure de pression des liquides et des gaz dans le cas où ces derniers ne sont pas fortement visqueux ou cristallisés.



Figure 1.10 : Indicateur de pression [6].

On peut résumer le fonctionnement d'un indicateur de pression comme suit : La pression du milieu agit directement sur le tube du bourbon dont l'extrémité libre fait tourner l'indicateur [6].

### b) Indicateur et transmetteur de pression

Ces capteurs sont employés pour la mesure et la transmission de pression. L'élément de bourbon rétréci sous pression, son mouvement est converti par un capteur inductif en signal électrique. Les transmetteurs de pression conviennent à tous les liquides et les gaz qui ne sont pas fortement visqueux, et sont appropriés aux médias et en atmosphères corrosives [6].



Figure 1.11: Indicateur et transmetteur de pression [6].

### c) Transmetteurs de pression

Les transmetteurs de pression mesurent la pression absolue ou relative, selon la version, dans les gaz, vapeurs et liquides. Elles incorporent une jauge de contrainte d'une couche

Épaisse comme moyen de mesure. La pression est convertit en un signal électrique, ils sont utilisés dans tous les domaines des procédés industriels [6].



Figure 1.12: Transmetteur de pression [6].

#### **d) Le capteur de niveau (contacte a seuil magnétique pour indicateurs de niveau magnétique type BGU**

Les contacts à seuil magnétique type BGU servent à détecter les seuils de niveau de remplissage sur des indicateurs de niveau magnétique. Ils produisent un signal binaire qui peut être transmis aux équipements de signalisation ou de commande installés en aval. Les versions bistables permettent le stockage de signaux.

Les contacts à seuils magnétique sont installés directement sur l'indicateur de niveau magnétique ou moyen d'un collier de serrage au sur l'afficheur magnétique au moyen de blocs coulissants [8].

Un capteur de niveau est conçu pour la détection de niveau de liquides de toutes natures, un capteur de niveau pour liquides est parfaitement fiable même en présence de dépôts, leur avantage est très vaste comme la protection contre la marche à vide des pompes.

Le principe de mesure est préconisé dans des conditions extrêmes où les autres principes de mesure ne conviennent pas ; il est équipé d'un détecteur synthétique souple pour la mesure continue de niveau ou la mesure d'interface sur des cuves rondes ou coniques [7].



**Figure 1.13:** Capteur de niveau [8].

### Référence

On utilise un détecteur de niveau de type : BGU

### Caractéristique

- Courant 4-20 mA
- Il possède une tige de longueur  $L= 400\text{mm}$ .
- Hautement résistant jusqu'à  $+200^{\circ}\text{C}$  et facile à nettoyer.

### 1.6.4 Le bac d'eau

Le volume de bac d'eau est 15 mètre cube il contient d'un capteur de niveau de type BGU, des vannes, une pompe, des capteurs et des transmetteurs.



Figure 1.14 : Le bac.

## 1.7 Conclusion

Dans le premier chapitre, nous avons évoqué la gestion de l'eau osmose au niveau de la raffinerie d'huile de l'entreprise CEVITAL. L'expérience nous a révélé que pour une assurance d'une alimentation régulière et continue de l'eau au niveau de la raffinerie et du bon fonctionnement de cette dernière, le recyclage de l'eau osmose s'avère la méthode la plus appropriée pour remédier au problème.



# Chapitre 2

## Généralités sur la régulation

### 2.1 Introduction

La régulation automatique, actuellement rebaptisée (automatique) est noyée dans les techniques modernes de commande-robotique, productique etc., en raison surtout de l'apparition de l'électronique, puis vers les années 60 du microprocesseur et donc de l'informatique.

Un régulateur est un dispositif capable de maintenir une grandeur physique à une valeur la plus proche possible de la valeur désirée. Mais il est utile de souligner que les vieilles techniques de régulation classiques restent encore très utilisées dans l'industrie car la théorie en automatique avance bien plus vite que l'application car les moyens informatiques sont plus (performants) que la connaissance du système à traiter c'est-à-dire le modèle.

### 2.2 Définition de la régulation automatique

La régulation automatique regroupe l'ensemble des moyens matériels et techniques mis en œuvre pour maintenir automatiquement (pas d'intervention Manuelle) une ou plusieurs grandeurs physiques (installation de production, robot, alimentation électronique stabilisée, etc..) en vue d'en imposer le comportement.

Cette prise de contrôle s'effectue par l'intermédiaire de certains signaux (grandeurs physique) qu'il est alors nécessaire de mesurer afin de déterminer l'action à entreprendre sur le système.

Lorsque des perturbations ou un changement de consigne se produisent, la régulation automatique provoque une action correctrice sur une autre grandeur physique, parmi les grandeurs d'entrée du procédé (grandeur réglant), afin de ramener la grandeur réglée vers sa consigne initiale (cas de perturbations) ou vers sa nouvelle consigne (cas de changement de consigne c'est-à-dire changement de point de fonctionnement).

Les méthodes de l'automatique offrent donc la possibilité de modifier le comportement statique et dynamique d'une grandeur physique, afin qu'elle évolue conformément aux exigences de l'application [9], [10].

## 2.3. Objectifs de la régulation

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 dans le temps, pour celles qui ne présentent que 2 états (système binaire ou TOR), tel les feux de signalisations, les commandes d'ascenseurs, de transfert de pièces par convoyeurs, etc...), on utilise une autre approche différente à la structure de boucle utilisée dans la plupart des systèmes y compris notre système (réservoir).

Les systèmes automatiques assurent en fait deux type de fonctions :

-Maintenir la grandeur commandée, ou grandeur réglée, à une valeur de référence malgré les variations de conditions extérieures ; on parle de la régulation en sens strict.

-Répondre à des changements d'objectif, ou à un objectif variable tel-que la poursuite de cible, on parle d'un fonctionnement d'asservissement [11].

Le rôle de l'automaticien (chargé d'obtenir un système régulé) sera multiple :

-Instrumenter le système : choisir capteurs et actionneurs en fonction des besoins physiques, de coût et de performance demandées au système.

-Déterminer les relations entré-sortie du système, des capteurs et des actionneurs. On parlera des lors de :

- Modéliser quand on s'attachera à déterminer la structure mathématique de ces relations.
- Identifier quand on s'intéressera à calculer les coefficients du modèle.
- Synthétiser une loi de commande (un correcteur) afin d'obtenir un système performant : précis, rapide et stable tout en s'affranchissant des influences néfastes des perturbations [4].

## 2.4 Types de régulation automatique

### 2.4.1 Régulation tout ou rien (TOR)

Ce mode d'action est essentiellement discontinu. Sa réalisation impose de se fixer une limite inférieure et une limite supérieure lorsque la mesure atteint la limite inférieure l'actionneur prend une position particulier (ouverture ou fermeture pour une vanne) de façon analogue le fait d'atteindre la limite supérieure place l'actionneur dans la position contraire. Ça oscille donc entre ces deux valeurs extrêmes et sa variation prend une allure en dents de scie. Ce réglage est simple ou le signal de commande ne prend que deux valeurs soit 0 ou 1 [13].

### 2.4.2 Régulation analogique

C'est le type de régulation ou le signal du régulateur et la mesure variant 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.

### 2.4.3 Régulation numérique

Le principe de la régulation numérique est que le régulateur prend la forme d'un algorithme programmé sur microprocesseur et exécuté en temps réel, i.e. impérativement à chaque période d'échantillonnage.

## 2.6 Critère de performance d'une régulation

Les performances d'une régulation peuvent se définir à partir de l'allure du signal de mesure suite a un échelon de consigne. Notons toutefois que les critères de performances classiques peuvent se résumer comme suit :

- **Stabilité** cette condition est impérative avec un certain degré de stabilité (marge de sécurité).en générale on impose une marge de gain de 2 a 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 nécessite 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 suivre la consigne.

## 2.7 Définition de la régulation PID

Le régulateur standard le plus utilisé dans l'industrie est le régulateur PID (proportionnel intégral dérivé), car il permet de régler de ses trois paramètres les performances (amortissement, temps de réponse,...) d'un processus modélisé par un deuxième ordre. Nombreux sont les systèmes physiques qui, même en étant complexe, ont un comportement voisin de celui d'un deuxième ordre.

Par conséquent, le régulateur PID est bien adapté à la plupart de processus de type industriel et est relativement robuste par rapport aux variations des paramètres du procédé.

Si la dynamique dominante du système est supérieur à un deuxième ordre, ou si le système contient un retard important ou plusieurs modes oscillants, le régulateur PID n'est plus adéquat et un régulateur plus complexe (avec plus de paramètres) doit être utilisé, au dépend de la sensibilité à la variation des paramètres du procédé [9].

## 2.8 Notion de correcteur PID

Sous forme d'un tableau récapitulatif, on résume les avantages et les limitations des actions de base des régulateurs PID :

Action	Point forts	Point faible
<b>P</b>	Action instantanée	Ne permet pas d'annuler une erreur statique mais permet de la réduire
<b>I</b>	Annule l'erreur statique	Action lente Ralentit le système (effet déstabilisant)
<b>D</b>	Action très dynamique	Améliore la rapidité apporte un effet stabilisant, sensibilité aux bruits forte sollicitation de l'organe de commande

Table 2.1- Notion de correcteur PID [9].

### 2.8.1 Le FB 41 CONT-C

Le bloc FB 41 CONT-C (Continuos Controller) sert à réguler des processus industriels à grandeurs d'entrée et de sortie continues sur les automates programmables SIMATIC S7. Le paramétrage permis d'activer ou désactives des fonctions partielles du régulateur PID et donc d'adapter ce dernier au système régulé [14].

### 2.8.2 Le rôle des paramètres PID :

<b>P</b>	L'action proportionnelle applique une correction instantanée pour tout écart entre la mesure et la consigne, plus 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.
<b>I</b>	Cette composante 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.
<b>D</b>	Cette action permet d'anticiper la réponse de la régulation en cas de 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.

#### 2.2- Le rôle des paramètres PID

### 2.8.4 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 à la main la valeur de réglage.

Selon le type de CPU, il sera mis en œuvre grâce au FB41 (pour les CPU 300) 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.

### 2.8.5 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 [15].

### 2.8.6 Information d'erreur

Ce bloc ne procède pas à un contrôle interne d'erreur. Le mot indicateur RET-VAL n'est pas employé.

Pour plus d'information sur le FB41 CONT-C consulter l'annexe 3.

## 2.9 Conclusion

Dans ce chapitre, nous avons présenté l'objectif de la régulation, une description générale des systèmes industriels ainsi que quelque méthode de régulation numérique, ce qui résume dans le rôle des paramètres des régulateurs PID, toute en essayant de répondre au cahier des charges élaboré.

## Chapitre 3: Automate programmable industriel (API)

### 3.1. Historique

Les automates programmable industriels sont apparus à la fin des années soixante, a la demande de l'industrie automobile américaine (GM), qui réclamait plus d'adaptabilité de leurs systèmes de commande.

Les coûts de l'électronique permettant alors de remplacer avantageusement les technologies actuelles.

### 3.2. Automatisation

#### 3.2.1. Définition

L'automatisation de la production consiste à transférer toute ou une partie des tâches de coordination, auparavant exécutées par des opérateurs humains, dans un ensemble d'objets techniques appelé partie commande.

La partie commande mémorise le savoir faire des opérateurs pour obtenir la suite des actions à effectuer sur les matières d'œuvre afin d'élaborer la valeur ajoutée.

Elle exploite un ensemble d'informations prélevées sur la partie opérative pour élaborer la succession des ordres nécessaires pour obtenir les actions souhaitées [12].

#### 3.2.2 Objectifs de l'automatisation

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

- Accroître la productivité du système c'est-à-dire augmenté la quantité de produits élaborés pendant une durée donnée. Cet accroissement de productivité exprime un gain de valeur ajoutée sous forme :
  - ✓ D'une meilleure rentabilité, d'une meilleure compétitivité.
  - ✓ Améliorer la flexibilité de production ;
  - ✓ Améliorer la qualité du produit grâce à une meilleure répétabilité de la valeur ajoutée.
  - ✓ S'adapter à des contextes particuliers :
    - Adaptation à des environnements hostiles pour l'homme (milieu salin, spatial, nucléaire...),
    - Adaptation à des tâches physiques ou intellectuelles pénibles pour l'homme (manipulation de lourdes charges, tâches répétitives parallélisées...),



- Augmenter la sécurité, etc...

D'autres objectifs, à caractères sociaux, financiers... peuvent s'ajouter à ceux-ci [12].

### 3.2.3 Structure d'un système automatisé

Tout système automatisé comporte :

- une partie opérative (P.O.) procédant au traitement des matières d'œuvre afin d'élaborer la valeur ajoutée,
- une partie commande (P.C.) coordonnant la succession des actions sur la partie opérative avec la finalité d'obtenir cette valeur ajoutée [12].

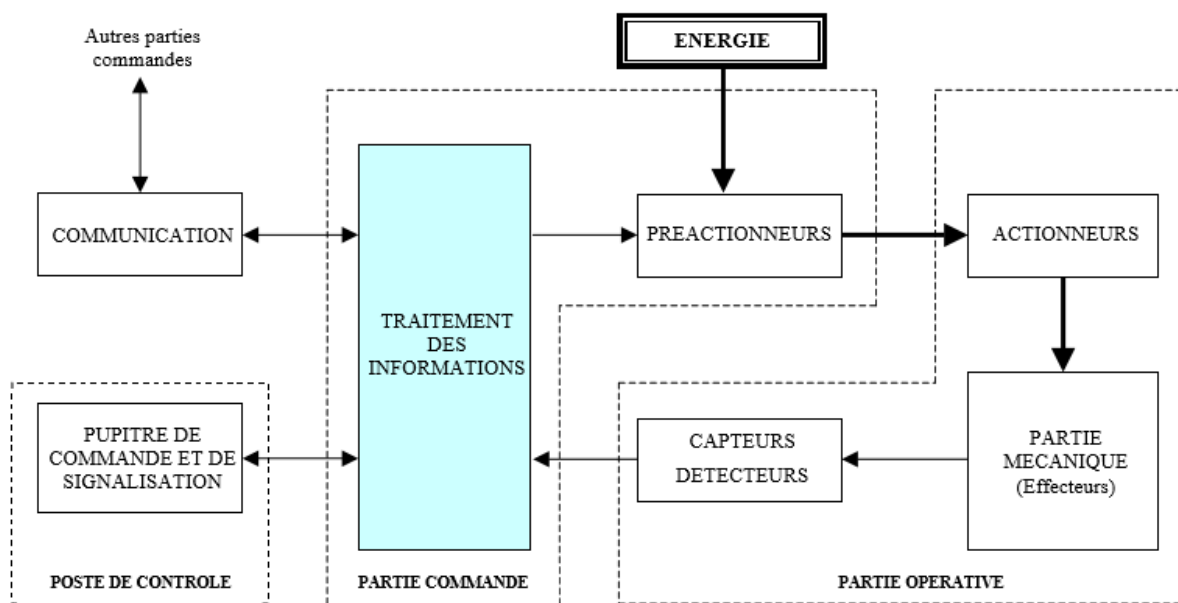


Figure 3.1 : Structure d'un système automatisée [13]

### 3.2.3.1 Partie opérative PO

La partie opérative est le processus physique à automatiser. Elle opère sur la matière et les produits entrants en transformant, par exemple, des pièces brutes en pièces usinées. Les opérations sont obtenues lorsque les ordres donnés par la partie commande sont exécutés dans des bonnes conditions [3].

La (P O) comporte en générale d'une part, les outillages et moyens divers mettant en œuvre le processus d'élaboration et d'autre part, les différents actionneurs destinés à mouvoir ou à mettre en œuvre ces moyens par exemple, moteur électrique pour actionner une pompe [3].

### 3.2.3.2 Partie commande PC

La partie commande (P.C) élabore d'une part, des ordres qui seront destinés à la (P.O) et d'autre part, des signaux de visualisation en fonction des comptes rendus venant du processus physique commandé et des consignes qu'il reçoit en entrée et ceci afin de coordonner toutes ces actions à effectuer [3].

La partie commande d'un système est un ensemble de composant et de constituants de traitement de l'information, destiné a :

- ✓ coordonner la succession des actions sur la partie opérative;
- ✓ surveiller son bon fonctionnement;
- ✓ gérer les dialogues avec les intervenants;
- ✓ gérer les communications avec d'autres systèmes;
- ✓ assurer le traitement des données et des résultats relatifs au procédé, aux matières d'œuvre, aux temps de production, à la consommation énergétique [12].

### 3.2.3.3 Poste de contrôle

Composé des pupitres de commande et de signalisation, il permet à l'opérateur de commander le système (marche, arrêt, départ cycle,...). Il permet également de visualiser les différents états du système à l'aide de voyants, de terminal de dialogue ou d'interface homme-machine (IHM) [13].

A fin d'automatiser un système désirée, il est indispensable d'utiliser en premier lieu des automates programmable industriels (AP I).

### 3.3. Automate programmable industriel (API)

#### 3.3.1. Historique

Les automates programmables industriels sont apparus à la fin des années soixante, à la demande de l'industrie automobile américaine (GM), qui réclamait plus d'adaptabilité de leurs systèmes de commande.

Avant ils utilisent des relais électromagnétiques et des systèmes pneumatiques pour la réalisation des parties commandes c'est-à-dire logique câblée qui présentés les Inconvénients suivants : cher, pas de flexibilité, pas de communication possible alors ils ont trouvé comme solution l'utilisation des systèmes à base de microprocesseurs permettant une modification aisée des systèmes automatisés autrement dit (logique programmée).

Les ordinateurs de l'époque étant chers et non adaptés aux contraintes du monde industriel, les automates devaient permettre de répondre aux attentes de l'industrie [13].

#### 3.3.2. Définition

L'Automate Programmable Industriel (API) est un appareil électronique programmable, adapté à l'environnement industriel, qui réalise des fonctions d'automatisme pour assurer la commande de pré actionneurs et d'actionneurs à partir d'informations logique, analogique ou numérique [13]. Un automate programmable comporte une mémoire programmable par un utilisateur automaticien à l'aide d'un langage adapté, par exemple :

- Logique séquentielle et combinatoire
- Temporisation
- Comptage, décomptage, comparaison
- Calcul arithmétique

#### 3.3.3. Domaines d'emploi des automates

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 [13].

### 3.4 Les fonctions d'un automate programmable

On distingue les fonctions principales suivantes [14] :

- la détection depuis des capteurs répartis sur la machine,
- la commande d'actions vers les pré-actionneurs et les actionneurs,
- le dialogue d'exploitation,
- le dialogue de programmation,
- le dialogue de supervision de production.

#### 3.4.1 La nature des informations traitées par l'automate

Les informations peuvent être de type :

Tout ou rien (T.O.R.) : l'information ne peut prendre que deux états (vrai/faux, 0 ou 1). C'est le type d'information délivrée par un détecteur, un bouton poussoir,....

Analogique : l'information est continue et peut prendre une valeur comprise dans une plage bien déterminée. C'est le type d'information délivrée par un capteur (pression, température ...).

Numérique : l'information est contenue dans des mots codés sous forme binaire ou bien hexadécimale. C'est le type d'information délivrée par un ordinateur ou un module intelligent [13].

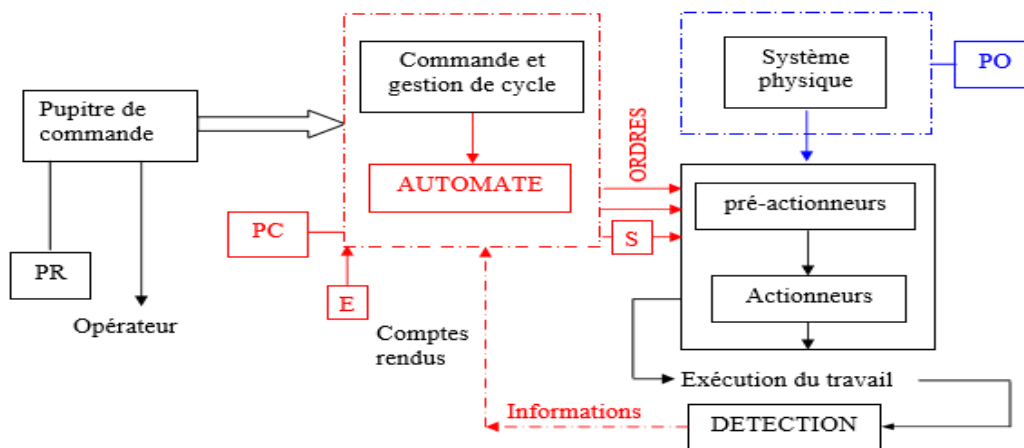


Figure 3.2: Situation d'un automate dans un système automatisé de production [16].

### 3.4.2. Architecture d'un automate programmable industriel

#### 3.4.2.1 Aspect extérieur

Les automates peuvent être de type compact ou modulaire.

- De type compact, on distinguera les modules de programmation (LOGO de Siemens, ZELIO de Schneider, MILLENIUM de Crouzet ...) des micro-automates. Il intègre le processeur, l'alimentation, les entrées et les sorties. Selon les modèles et les fabricants, il pourra réaliser certaines fonctions supplémentaires (comptage rapide, E/S analogiques ...) et recevoir des extensions en nombre limité. Ces automates, de fonctionnement simple, sont généralement destinés à la commande de petits automatismes.
- De type modulaire, le processeur, l'alimentation et les interfaces d'entrées / sorties résident dans des unités séparées (modules) et sont fixées sur un ou plusieurs racks contenant le "fond de panier" (bus plus connecteurs). Ces automates sont intégrés dans les automatismes complexes où la puissance, capacité de traitement et flexibilité sont nécessaires [13].

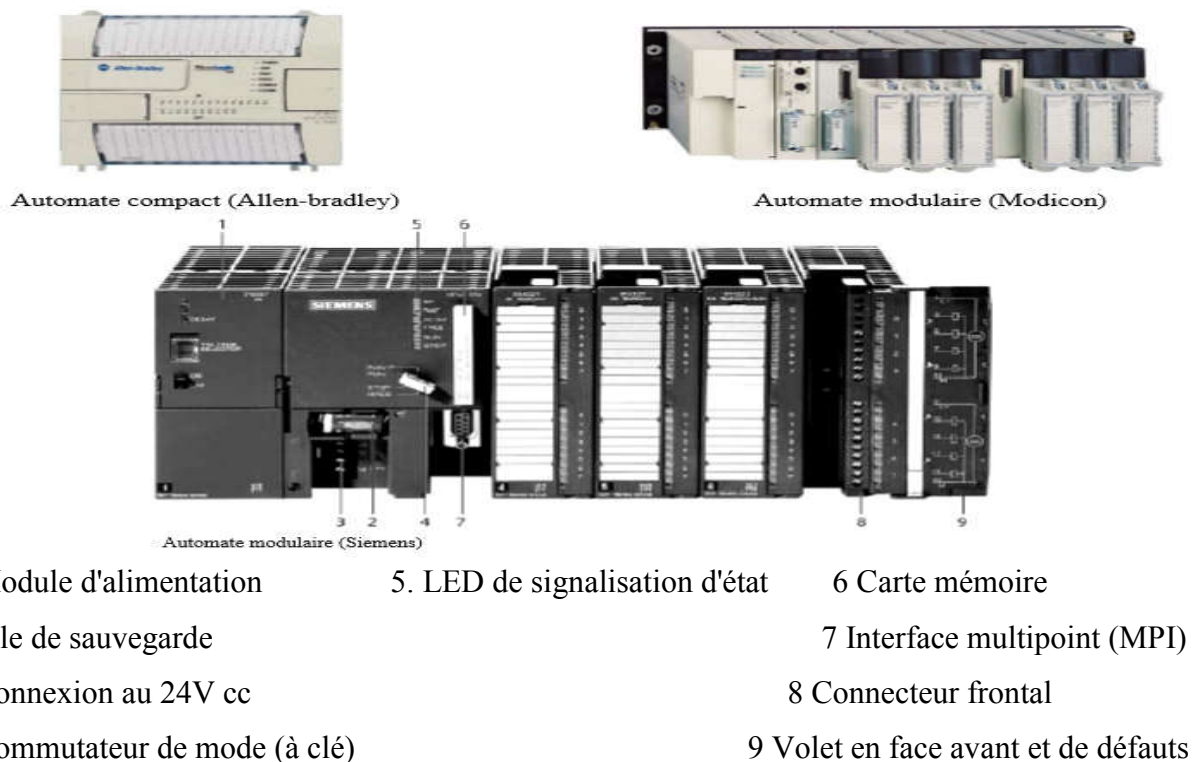


Figure 3.3 : Les différents types d'automate [13]



Leur principales fonctions sont : pour les automates compacts permettent de commander des sorties en T.O.R et gèrent parfois des fonctions de comptage et de traitement analogique. Et pour les automates modulaires permettent de réaliser de nombreuses autres fonctions grâce à des modules intelligents que l'on dispose sur un ou plusieurs racks. Ces modules ont l'avantage de ne pas surcharger le travail de la CPU car ils disposent bien souvent de leur propre processeur [13].

### 3.4.2.2 Structure interne

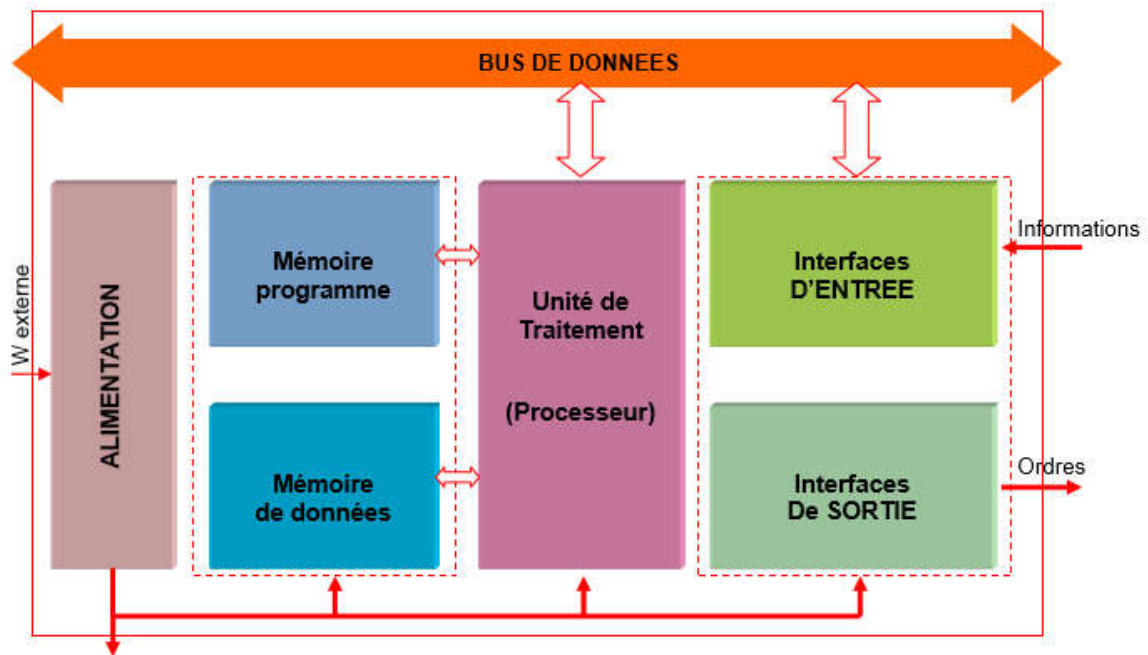


Figure3.4: structure interne d'un automate programmable [14].

Un A.P.I. se compose de :

#### 3.4.2.2.1 Une unité de traitement ou processeur

À base de microprocesseur, elle réalise toutes les fonctions logiques, arithmétiques et de traitement numérique (transfert, comptage, temporisation ...) ; en assurant [13], [14].

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





### 3.4.2.2.2 Une mémoire de programme

La mémoire programme de type RAM contient les instructions à exécuter par le processeur afin de déterminer les ordres à envoyer aux pré-actionneurs reliés à l'interface de sortie en fonction des informations recueillies par les capteurs reliés à l'interface d'entrée [14].

### 3.4.2.2.3 Une mémoire de données

La mémoire de donnée permet le stockage de [14]:

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

### 3.4.2.2.4 Une interface d'entrée

L'interface d'entrée permet la connexion à l'API d'une multitude de capteurs pouvant être :

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

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

### 3.4.2.2.5 Une interface de sortie

L'interface de sortie permet la connexion à l'API d'une multitude de pré actionneurs pouvant être [14] :

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

### 3.4.2.2.6 Un module d'alimentation

Le module d'alimentation transforme l'énergie externe provenant du réseau en la mettant en forme afin de fournir aux différents modules de l'API les niveaux de tension nécessaires à leur bon fonctionnement. Plusieurs niveaux de tension peuvent être utilisés par les circuits internes (3v, 5v, 12v, 24v...) il sera dimensionné en fonction des consommations des différentes parties [14].



### 3.4.2.2.7 Principales fonctions

- **Cartes d'entrées / sorties** : Au nombre de 4, 8, 16 ou 32, elles peuvent aussi bien réaliser des fonctions d'entrées, de sorties ou les deux. Ce sont les plus utilisées et les tensions disponibles sont normalisées (24, 48, 110 ou 230V continu ou alternatif ...). Les voies peuvent être indépendantes ou posséder des "communs". Les cartes d'entrées permettent de recueillir l'information des capteurs, boutons qui lui sont raccordés et de la matérialiser par un bit image de l'état du capteur. Les cartes de sorties offrent deux types de technologies : les sorties à relais électromagnétiques (bobine plus contact) et les sorties statiques (à base de transistors ou de triacs).
- **Cartes de comptage rapide** : elles permettent d'acquérir des informations de fréquences élevées incompatibles avec le temps de traitement de l'automate.
- **Cartes de commande d'axe** : Elles permettent d'assurer le positionnement avec précision d'élément mécanique selon un ou plusieurs axes. La carte permet par exemple de piloter un servomoteur et de recevoir les informations de positionnement par un codeur. L'asservissement de position pouvant être réalisé en boucle fermée.
- **Cartes d'entrées / sorties analogiques** : Elles permettent de réaliser l'acquisition d'un signal analogique et sa conversion numérique (CAN) indispensable pour assurer un traitement par le microprocesseur. La fonction inverse (sortie analogique) est également réalisée. Les grandeurs analogiques sont normalisées : 0-10V ou 4-20mA.

#### Autres cartes

- Cartes de régulation PID
- Cartes de pesage
- Cartes de communication (Ethernet ...)
- Cartes d'entrées / sorties déportées [13].

### 3.4.2.8 Traitement du programme automate

Tous les automates fonctionnent selon le même mode opératoire :

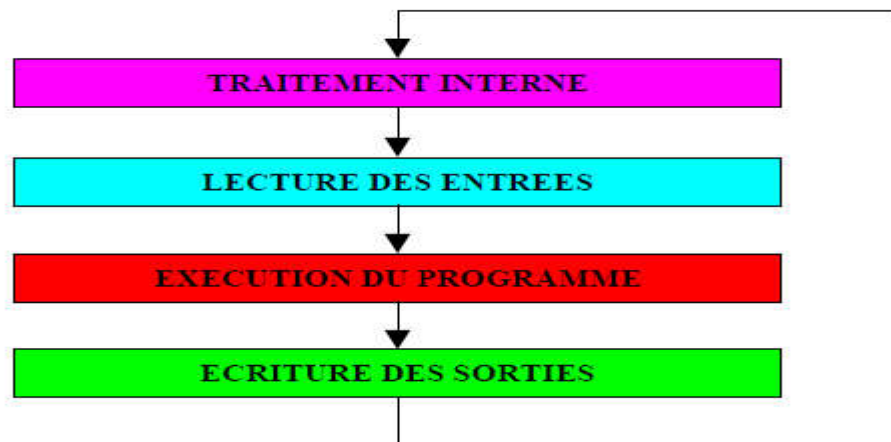


Figure3.5: cycle d'exécution d'un programme [13].

- **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.

- Ces quatre opérations sont effectuées continuellement par l'automate (fonctionnement cyclique).

On appelle scrutation l'ensemble des quatre opérations réalisées par l'automate et le temps de scrutation est le temps mis par l'automate pour traiter la même partie de programme. Ce temps est de l'ordre de la dizaine de millisecondes pour les applications standards [13].

- **Le temps de réponse total (TRT)**

C'est le temps qui s'écoule entre le changement d'état d'une entrée et le changement d'état de la sortie correspondante :

Le temps de réponse total est au plus égal à deux fois le temps de scrutation (sans traitement particulier) [13].

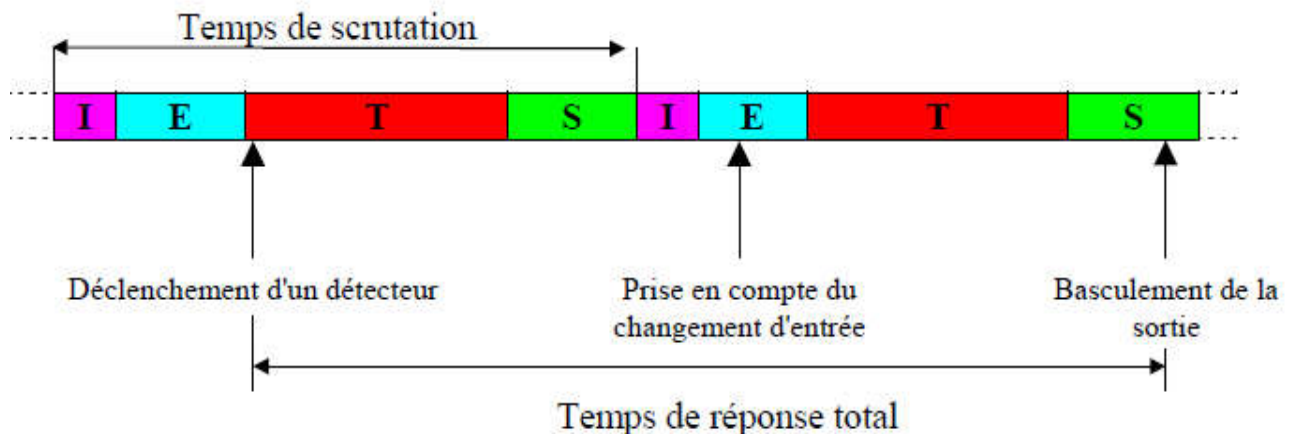


Figure3.6: Le temps de repense total (TRT) [13].

### 3.5 Critères de choix d'un automate

Le choix d'un automate programmable est en premier lieu le choix d'une société ou d'un groupe et les contacts commerciaux et expériences vécues sont déjà un point de départ.

Les grandes sociétés privilégieront deux fabricants pour faire jouer la concurrence et pouvoir "se retourner" en cas de "perte de vitesse" de l'une d'entre elles.

Le personnel de maintenance doit toutefois être formé sur ces matériels et une trop grande diversité des matériels peut avoir de graves répercussions. Un automate utilisant des langages de programmation de type GRAFCET est également préférable pour assurer les mises au point et dépannages dans les meilleures conditions.

La possession d'un logiciel de programmation est aussi source d'économies (achat du logiciel et formation du personnel). Des outils permettant une simulation des programmes sont également souhaitables.



Il faut ensuite quantifier les besoins :

- Nombre d'entrées / sorties : le nombre de cartes que peut avoir une incidence sur le nombre de racks dès que le nombre d'entrées / sorties nécessaires devient élevé.
- 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.
- Fonctions ou modules spéciaux : certaines cartes (commande d'axe, pesage ...) permettront de "soulager" le processeur et devront offrir les caractéristiques souhaitées (résolution, ...).
- 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 ...) [13].

Dans notre cas en a choisie un automate siemens s7-300 qui satisfait le critère du choix d'un automate.

(Dans notre cas nous allons travailler avec).

### 3.5.1 Présentation de l'API S7-300

L'automate S7 est constitué d'une alimentation, d'une CPU et des modules d'entrée ou de sortie (modules E/S). L'automate programmable contrôle et commande une machine ou un processus à l'aide du programme s7. Les modules d'entrées / sorties sont adressés dans le programme s7 via les adresses d'entrée (E) et adresse de sortie (S). L'automate est programmé à l'aide du logiciel step7 [17].

### 3.5.2 Description de S7-300

L'automate S7-300 est le modèle de base de la gamme des API siemens qui comprend aussi les S7-200 (modèle compacte) et les S7-400 (modèle utilisé en régulation) [18].

L'automate Siemens S7-300 est une plate-forme d'automatisation universelle pour des applications avec des architectures centralisées et décentralisées, orientée sécurité, motion control ou avec interface ETHERNET/PROFINET intégré. L'automate Siemens SIMATIC S7-300 peut s'intégrer également dans des solutions compactes ou dans des têtes de station pour

Traitement intelligent décentralisé [19]. Il se programme avec le logiciel STEP 7 de Siemens [18].

### 3.5.3 Structure modulaire du S7-300

Le système S7-300 est modulaire et on y trouve les types de module suivant :

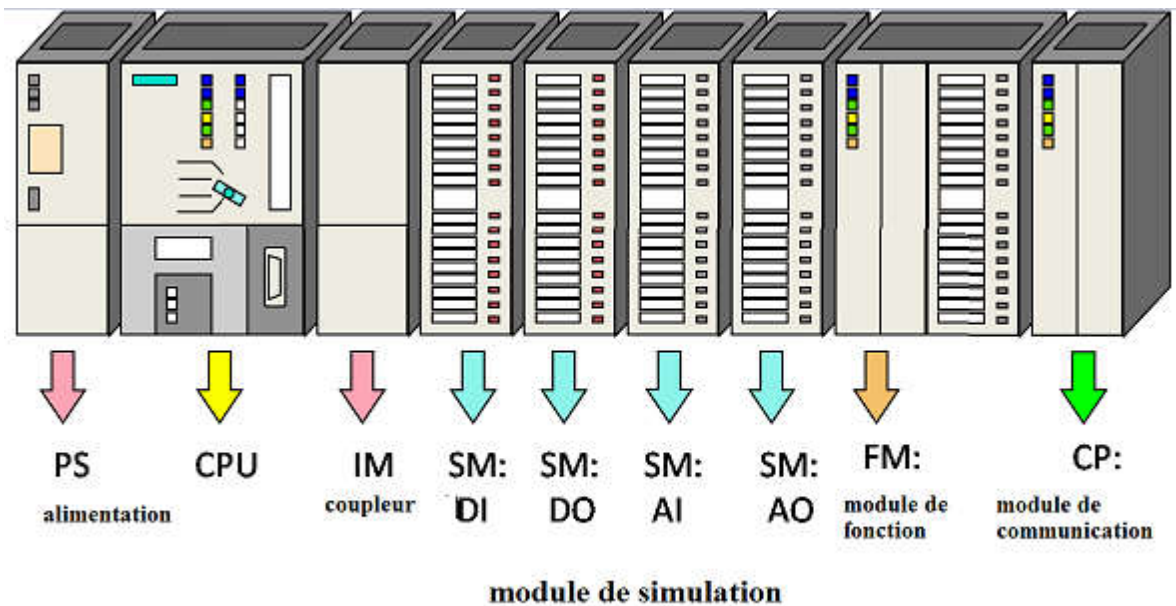


Figure3.7: Structure modulaire d'un API S7-300 [25].

- **Module d'alimentation PS :**

Tout réseau 24V peut être utilisé pour alimenter la CPU des S7-300. Les modules suivants (voir le tableau N° :1) de la gamme S7 sont prévus pour être utilisés avec le S7-300.

**Tableau 1: les modules d'alimentation de la gamme S7-300 :**

Désignation	Courant de sortie	Tension à la sortie	Tension à l'entrée
PS 307	2 A	DC 24 V	AC 120/230 V
	5 A	DC 24 V	AC 120/230 V
	10 A	DC 24 V	AC 120/230 V





- **Unité central CPU**

L'unité centrale c'est le cerveau de l'API. Ce dernier liée aux étapes des signaux d'entrée exécute le programme utilisateur et commande-les sorties.

Le S7-300 dispose d'une large gamme de CPU à différents niveaux de performance, on compte les versions suivantes :

- ✓ CPU à utilisateur standard : CPU 313, CPU 314, CPU 315 et CPU 316.
- ✓ CPU avec fonction intégrées : CPU 312 IFM et la CPU 314 IFM.
- ✓ CPU avec interface PROFILBUS DP : CPU 315-2DP, CPU 316-2DP et CPU 318-2DP.

- **Coupleur IM**

Les coupleurs sont les cartes électroniques qui assurent la communication entre les E/S (périphériques ou autres) et l'unité centrale. L'échange de l'information entre la CPU et les modules d'E/S s'effectue par l'intermédiaire d'un bus interne (liaison parallèle codée).

Les coupleurs ont pour rôle le raccordement d'un ou plusieurs châssis au châssis de base.

Pour l'API S7-300, les coupleurs disponibles sont :

- IM 365 : pour les couplages entre les châssis d'un mètre de distance max.
- IM 360 : pour les couplages allant jusqu'à 10 mètres de distances.

- **Module de simulation SM**

Le module de simulation SM 374 est un module spécial qui offre à l'utilisateur la possibilité de tester son programme lors de la mise en service en cours de fonctionnement. Dans le S7-300, ce module se monte à la place d'un module d'entrée ou de sortie TOR. Il assure plusieurs fonctions telles que :

- La simulation des signaux de capteurs aux moyens d'interrupteurs.
- La signalisation d'état des signaux de sorties par des LEDS.

- **Module de fonction FM**

Ces modules réduisent la charge de traitement de la CPU en assurant des tâches lourdes de calculs. On peut citer les modules suivants :

- FM 354 et FM 357 : modules de commande d'axe pour servomoteur.
- FM 353 : module de positionnement pour moteur pas à pas.
- FM 355 : module de régulation.
- FM 350-1 et FM 350-2 : module de comptage.



- **Module de communication CP**

La communication homme-machine ou machine-homme ainsi que l'échange de données avec d'autres appareils sont assurées par des processeurs de communication qui permettent :

- La conduite et l'observation des machines ou processus.
- La signalisation et la consignation des états des machines et des installations.
- **Les interfaces AS-I et MPI**
  - **AS-I** : c'est un système de connexion employé pour le premier niveau du processus dans les installations d'automatisation. Il permet de relier des capteurs et actionneurs échangeant des données binaires.
  - **MPI** : c'est une interface multipoint pour SIMATIC S7 et peut servir à la mise en réseau de plusieurs CPU pour l'échange de petites quantités de données (jusqu'à 70 octets).
- **Rémanence sans pile du S7-300**

Pour sauvegarder les temporisations, compteurs et mémentos S7-300 n'a pas besoin de pile et le contenu des blocs y est également sauvegardé en cas de coupure du courant. Les unités centrales de SIMATIC S7-300 ont une mémoire tampon de secours ne nécessitant aucune maintenance qui sauvegarde les opérandes et les données qui ont été déclarées permanentes [5].

### 3.6 Conclusion :

Dans ce chapitre, on a pu voir les principales définitions du système automatisé et l'automate programmable industriel ainsi que leur structure, ce qui nous a aidé à la conception du fonctionnement ainsi que le rôle joué par l'automate dans un milieu industriel.

Donc selon le besoin et en tenant compte de la tâche que doit réaliser un automate programmable l'utilisateur doit faire un bon choix dans cette variété des API.

asservissement, régulation, Etc....

Il permet de commander, mesurer et contrôler au moyen de modules d'entrées et de sorties (logiques, numériques ou analogiques) différentes sortes de machines ou de processus en environnement industriel. La compacité, la robustesse et la facilité d'emploi des automates programmables industriels (A.P.I.) font qu'ils sont très utilisés dans la partie commande des systèmes industriels automatisés [14].

## Chapitre4 : Simulation et résultats

### 4.1. Introduction

Un algorithme est la description d'un enchaînement de calculs et de tâches élémentaires en vue de la réalisation d'un traitement automatique de données. Ces données doivent être représentées par des structures appropriées. L'algorithme peut ensuite être codé dans un langage de programmation pour donner un programme. La programmation regroupe ces activités : algorithmique, structuration de données et codage, les deux premières étant certainement les plus importantes.

Contrairement à ce que pourrait croire un novice, la tâche principale de la conception d'un programme n'est pas l'écriture de celui-ci dans un langage informatique donné. La réalisation d'un programme se décompose en effet en plusieurs phases. De manière simplifiée, on présente ainsi la genèse d'un programme [20] :

- établissement d'un cahier des charges : le problème posé ;
- analyse du problème et décomposition de celui-ci en sous-problèmes plus ou moins indépendants et plus simples à résoudre ;
- choix de structures de données pour représenter les objets du problème ;
- mise en œuvre des différents algorithmes à utiliser pour résoudre les sous-problèmes ;
- codage des algorithmes et création du programme dans un langage de programmation donné ;
- tests (et corrections) et validation auprès des utilisateurs.

### 4.2. Présentation du logiciel STEP 7

STEP 7 est le progiciel de base pour la configuration et la programmation de systèmes d'automatisation SIMATIC. Il fait partie de l'industrie logicielle SIMATIC. Le progiciel de base STEP 7 existe en plusieurs versions :

- STEP 7-Micro/DOS et STEP 7-Micro/Win pour des applications autonomes simples sur SIMATIC S7 - 200.
- STEP 7 pour des applications sur SIMATIC S7-300/400, SIMATIC M7-300/400 et SIMATIC C7.



### 4.3. Utilisation de STEP 7

STEP 7 permet l'accès « de base » aux automates siemens. Il permet de programmer individuellement un automate en différents langages. Il prend également en compte le réseau des automates, ce qui permet d'accéder à tout automate du réseau pour le programmer, et éventuellement aux automates de s'envoyer des messages entre eux. Il ne permet pas d'incorporer les ordinateurs dans le réseau durant le fonctionnement, il n'y a pas de dialogue entre les pc et les automates, donc pas de supervision du processus par un logiciel centralisé, comme ce serait possible sous WINCC [5].

#### 4.3.1 Présentant des fonctionnalités supplémentaires

Le logiciel de base vous assiste dans toutes les phases du processus de création de vos solutions d'automatisation :

- la création et la gestion de projets,
- la configuration et le paramétrage du matériel et de la communication,
- la gestion des mnémoniques,
- la création de programmes, par exemple pour les systèmes cible S7,
- le chargement de programmes dans des systèmes cible,
- le test de l'installation d'automatisation,
- le diagnostic lors de perturbations de l'installation.
- Possibilité d'extension grâce aux applications proposées par l'industrie logicielle SIMATIC
- Possibilité de paramétrage de modules fonctionnels et de modules de communication
- Forçage et fonctionnement multiprocesseur
- Communication par données globales
- Transfert de données commandé par événement à l'aide de blocs de communication et de blocs fonctionnels
- Configuration de liaisons

#### 4.3.2 Les blocs de codes et les blocs de système

##### 4.3.2.1 Bloc d'organisation OB

Un OB est appelé cycliquement par le système d'exploitation et constitue donc l'interface entre le programme utilisateur et le système d'exploitation, il est directement appelé par le système.





L'OB contient des instructions d'appels de blocs indiquant à l'unité de commande de l'automate l'ordre dans lequel il doit traiter les blocs. Ils déterminent la structure du programme utilisateur.

#### **4.3.2.2 Bloc fonctionnel FB**

Le FB dispose d'une zone de mémoire qui lui est affectée en propre. Il est possible d'affecter un bloc de données DB au FB à l'appel du bloc. Il est possible d'accéder aux données du DB d'instant via les appels contenus dans le FB. On peut affecter plusieurs DB à un FB. Il est possible d'appeler d'autres FB et FC dans un bloc fonctionnel via des instructions d'appels de blocs.

#### **4.3.2.3 Fonction FC**

Une FC ne possède pas une zone mémoire propre. Les données locales d'une fonction sont perdues après son exécution. Il est également possible d'appeler d'autres FB et FC dans une fonction via des instructions d'appels de blocs.

#### **4.3.2.4 Bloc de données DB**

Les DB sont utilisés pour la mise en disposition d'espace mémoire pour les variables types données. Il existe deux types de blocs de données. Les DB globaux dans lesquels tous les OB, FB et FC peuvent lire les données enregistrées ou écrire des données et les DB d'instance qui sont affectés à un FB donné [5].

#### **4.3.2.5 Bloc système pour fonctions standard et fonction système**

Sont des fonctions prêtes à l'emploi stockées dans la CPU. Ces blocs peuvent être appelés par l'utilisateur et utilisées dans le programme, mais on ne peut pas changer ni accéder à leurs programmes.

On dispose dans STEP 7 des blocs système suivant :

#### **4.3.2.6 Bloc fonctionnel système SFB**

Bloc fonctionnel stocké dans le système d'exploitation de la CPU et pouvant être appelé par l'utilisateur.

#### 4.3.2.7 Fonction système SFC

Fonction stockée dans le système d'exploitation de la CPU et pouvant être appelée par l'utilisateur.

#### 4.3.2.8 Donnée système SDB

Zone de mémoire dans le programme configurée par différentes applications de STEP 7 (par exemple S7 configuration, communication, configuration...), pour le stockage des données dans le système d'automatisation [5].

### 4.4 Création d'un projet sous STEP 7

Un projet permet de regrouper l'ensemble des programmes et données nécessaires à réaliser une tâche d'automatisation. Ces données englobent en particulier :

- les données de configuration pour la configuration matérielle et les données de paramétrage pour les modules.
- les données de configuration pour la communication par réseau.
- les programmes pour modules programmables.

La tâche principale dans la réalisation d'un projet, consiste à préparer ces données et à effectuer la programmation.

Le plus simple pour créer un nouveau projet, c'est d'avoir recours à l'assistant "Nouveau projet". Pour l'appeler, choisissez la commande « Fichier » « Assistant "Nouveau projet" ». Il vous posera les questions nécessaires dans des boîtes de dialogue et créera le projet pour vous. Vous disposez d'une grande liberté d'action pour la suite du traitement de votre projet. Une fois votre projet créé, vous pouvez par exemple poursuivre votre travail en configurant d'abord le matériel, puis en créant le logiciel correspondant ou en créant d'abord le logiciel indépendamment d'un matériel configuré [21].





Figure 4.1: Nouveau projet sur STEP 7.

#### 4.4.1 Configuration matérielle

Pour réaliser la configuration d'un automate programmable vous allez utiliser deux fenêtres :

- la fenêtre de station dans laquelle vous allez disposer les profilés support/châssis pour la configuration de la station.
- la fenêtre "Catalogue du matériel" dans laquelle vous allez sélectionner les composants matériels requis, comme par exemple les profilés support ou châssis, les modules et cartouches interface [22].

On procède selon les étapes suivantes :

##### 4.4.1.1 Étape 01

Le choix du rack en tenant compte des dimensions de l'armoire électronique et de nombre des modules entrée / sortie utilisé dans le projet d'automatisations.

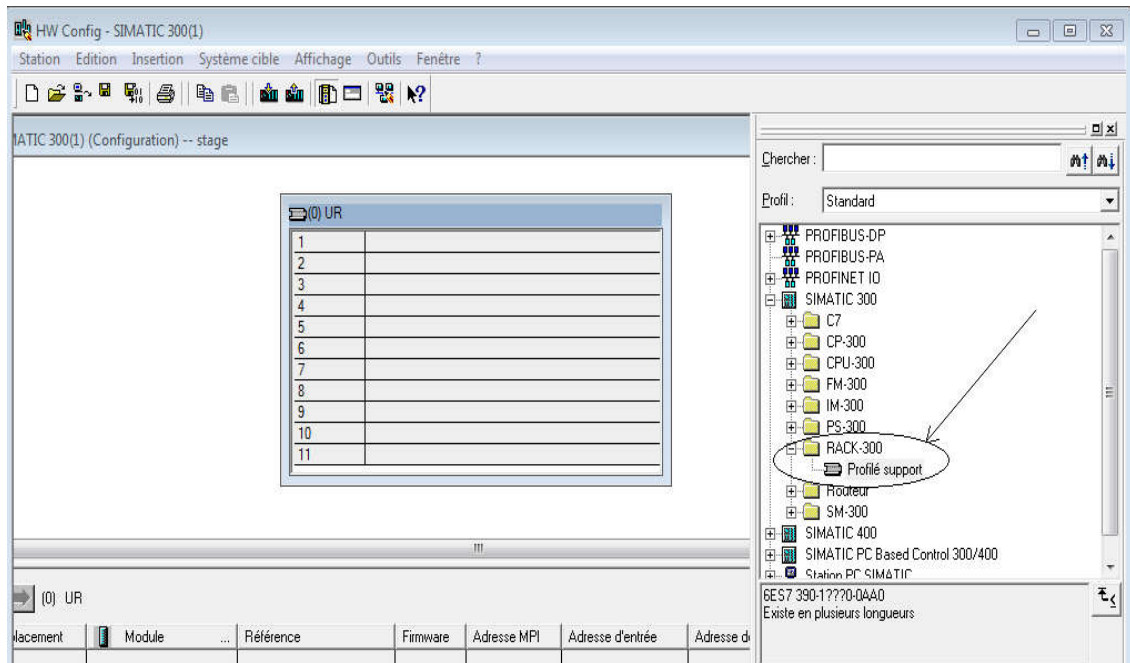


Figure 4.2: Choix du rack.

#### 4.4.1.2 Étape 02

Choix de la CPU utilisée pour le projet.

Choisir l'adresse MPI pour la CPU (CPU avec réseau PROFIBUS\_DP), l'adresse MPI est réglé par défaut à la valeur 2.

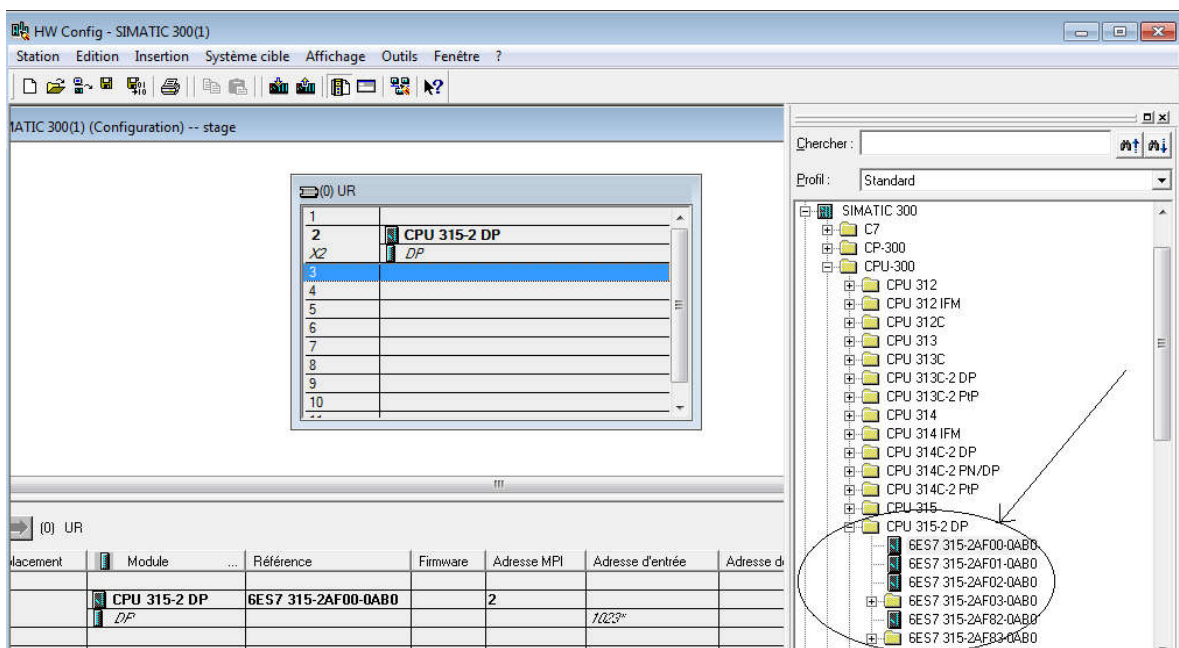


Figure 4.3: Choix de la CPU.



### 4.4.1.3 Étape 03

Choisir les modules d'entrée sortie selon le nombre d'entrée sortie qu'on possède

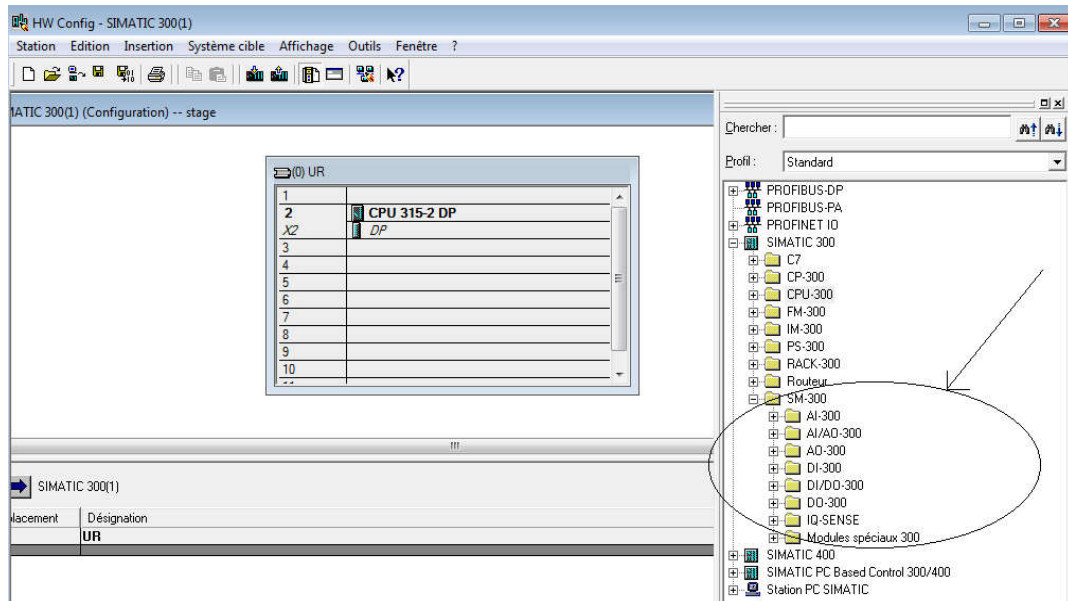


Figure 4.4: Choix des modules d'entrée / sortie.

## 4.5 Le principe de la programmation sur STEP 7

Après avoir configuré le matériel l'enregistrer et le compiler, on va faire la programmation de processus de marche de notre système.

Cliquer sur « programme S7 » ce qui nous donne les items « sources », « mnémoniques » et « blocs » comme le montre la figure qui suit :

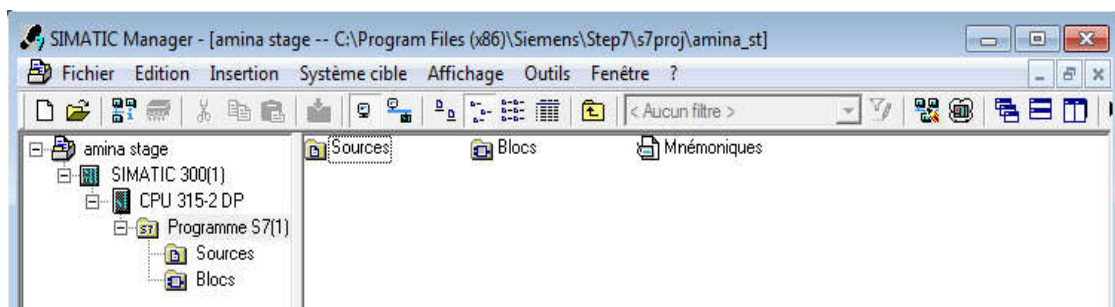


Figure 4.5 : Structure d'un projet.



Cliquer sur le répertoire « blocs »

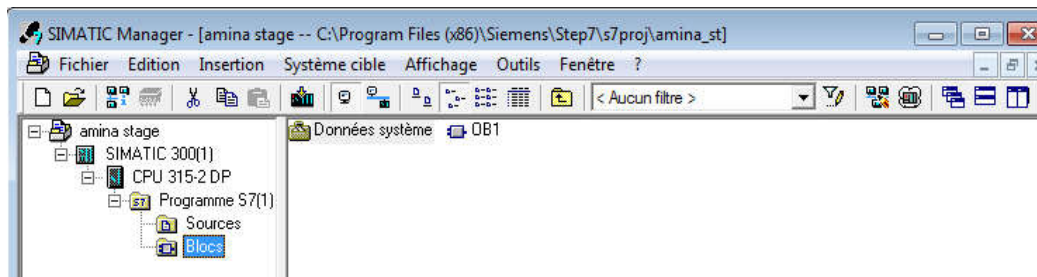


Figure 4.6: Le bloc OB1.

Double cliquez sur « OB 1 » pour choisir le programme à utiliser ainsi que pour lancer le logiciel « STEP 7 » la fenêtre de l'éditeur apparaît.

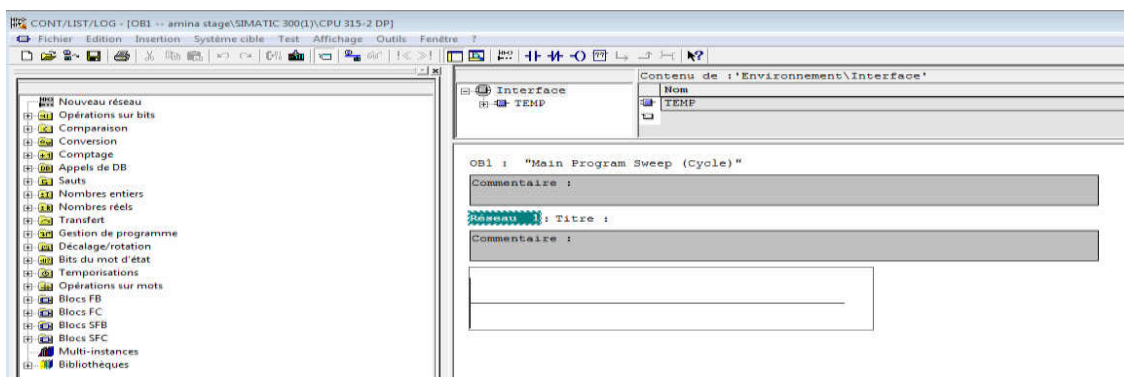


Figure 4.7: Réalisation du programme.

#### 4.6. Chargement du programme dans la CPU

On fait le transfert du programme vers la CPU en appuyant sur « système cible » puis « charger » ou directement sur l'item charger comme le montre les deux figure qui suit.

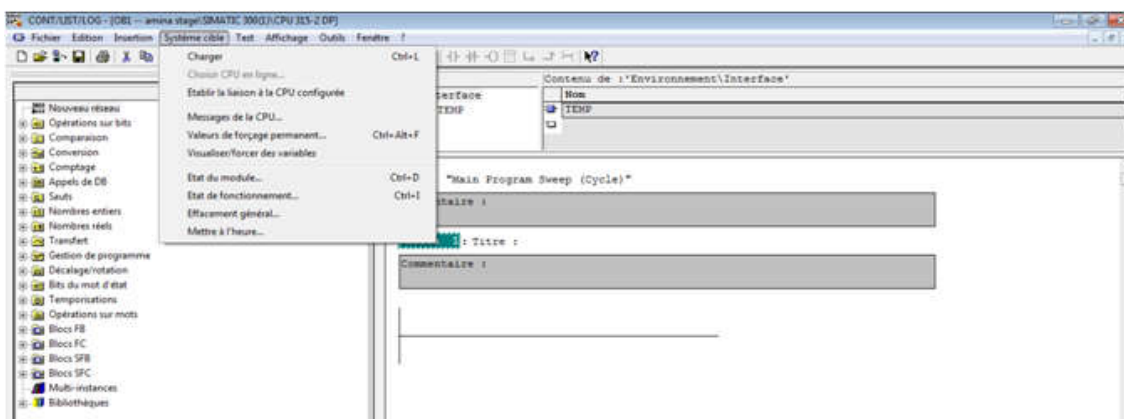


Figure 4.8: Chargement du programme dans la CPU-1



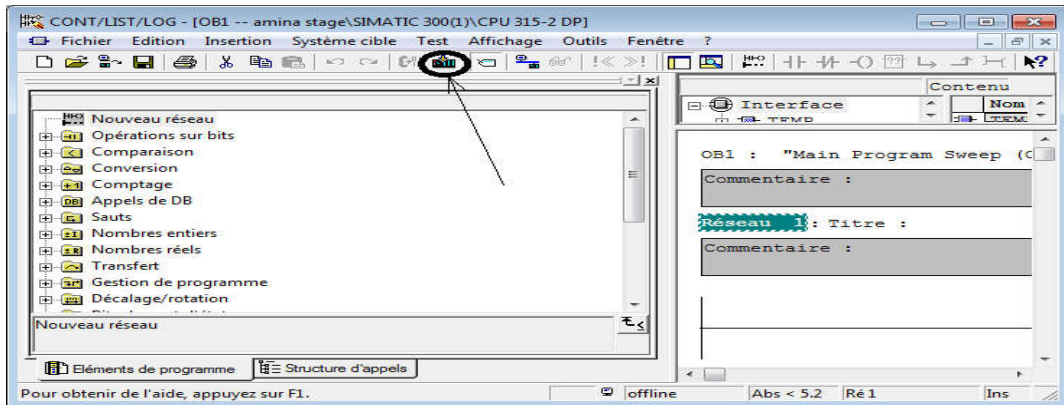


Figure 4.9 : Chargement du programme dans la CPU-2.

## 4.7 Simulation du programme dans l'automate

Le fonctionnement du programme peut être visualisé via PLC SIM.

### 4.7.1 Présentation de PLC SIM

Dans S7-PLCSIM, on peut exécuter notre programme STEP 7 et l'essayer dans un automate programmable simulé. Cette simulation s'exécute sur notre PC ou console de programmation. Avec S7-PLCSIM, nous pouvons simuler des programmes utilisateur STEP 7 qui ont été développés pour les automates S7-300, S7-400 et WinAC.

S7-PLCSIM offre une interface simple au programme utilisateur STEP 7 servant à visualiser et à modifier différents objets tels que les variables d'entrée et de sortie. Pendant que notre programme est traité par la CPU simulée, vous pouvez recourir au logiciel STEP 7 [23]

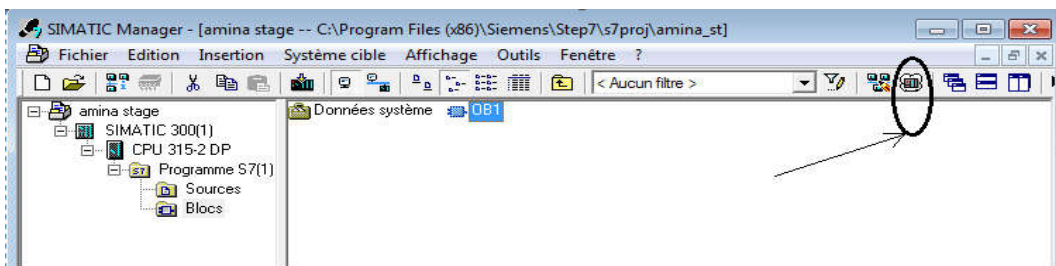


Figure 4.10 : Ouverture du simulateur sur STEP 7.

Après avoir ouvert le simulateur on passe aux étapes suivantes :

- Faire appel aux entrées et aux sorties selon notre besoin (adressage).
- Maitres à « 0 » ou à « 1 » l'état des entrées en cochant la case correspondante
- On coche la case RUN ou RUN-P du simulateur comme le montre la figure.

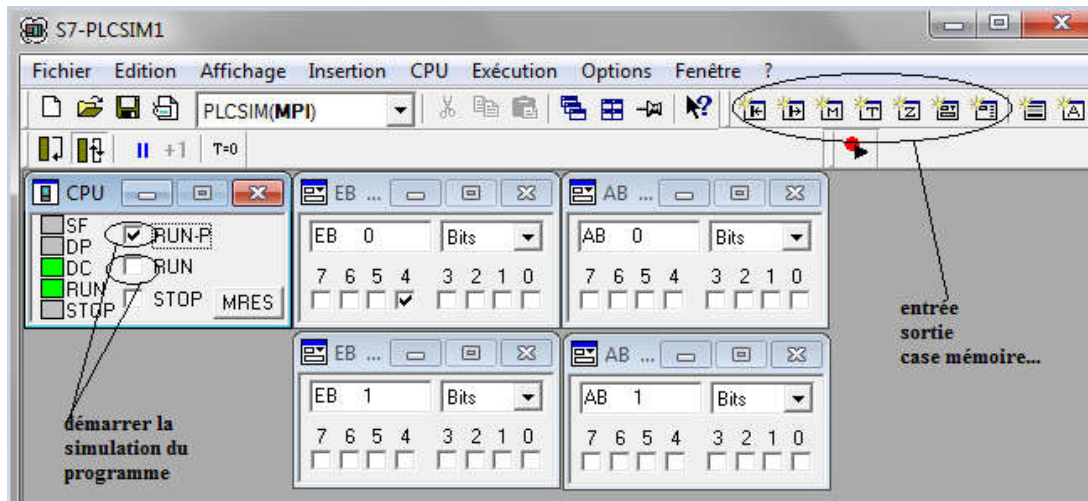


Figure 4.11 : Simulateur du programme.

- ✓ Double clique sur le bloc OB1 l'éditeur est alors lancé et le contenu du programme est affiché.
- ✓ Pour visualiser le changement d'état des variables on appuis sur la paire de lunette ainsi l'élément actif s'affiche en trait vert gras et un élément inactif s'affiche en trait pointillé en bleu.

## 4.8 Présentation du Wincc flexible

Wincc flexible est l'interface homme-machine (IHM) idéale pour toutes les applications au pied de la machine et du processus dans la construction d'installations, de machines et de machines de série.

Le Wincc flexible permet de disposer d'un logiciel d'ingénierie pour tous les pupitres opérateur SIMATIC HMI, du plus petit Micro Panel jusqu'au Multi Panel ainsi que d'un logiciel de supervision Runtime pour solutions monoposte basées sur PC et tournant sous Windows XP, 7 et Vista.

Les projets peuvent être portés sans conversion et sont exécutables sur diverses plateformes IHM. Grâce à des logiciels et à des projets multilingues, Wincc flexible peut être utilisé dans le monde entier.

### 4.8.1 Logiciel exécutif SIMATIC Wincc Flexible Runtime

La partie exploitation (Runtime) est embarquée sur tous les terminaux SIMATIC HMI, les fonctionnalités IHM et les capacités fonctionnelles dépendent de la configuration matérielle. Wincc Flexible Runtime est disponible pour les PC en différentes variantes qui se différencient par le nombre de Power Tags utilisés (seules les variables qui possèdent une liaison procès avec l'automate sont comptabilisées comme Power Tags). En plus, de ces Power Tags, le système peut gérer des variables internes (sans liaison au processus), des seuils constants ou variables et des messages (jusqu'à 4000) comme options additionnelles du système. Avec le SIMATIC WinCC Flexible Runtime, nous pouvons simuler notre plateforme d'en moins deux manières :

- ✓ En utilisant le S7-PLCSIM pour la manipulation des variables (lancer Runtime);
- ✓ En utilisant la table de simulation qui permet d'entrer les valeurs des variables (lancé WinCC flexible avec la simulation).

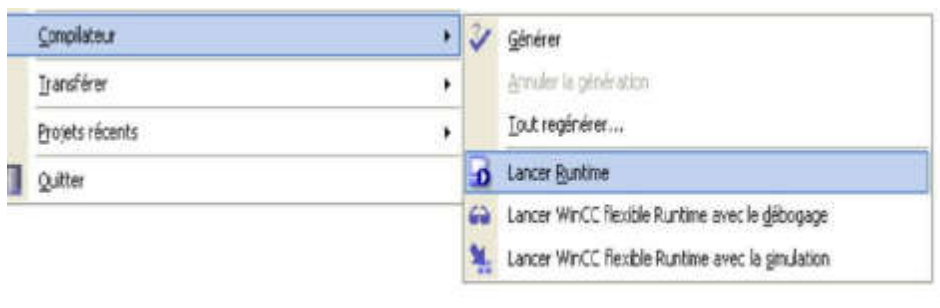


Figure 4.12 : Compilation sous WinCC flexible Runtime.

### 4.8.2 Intégration dans SIMATIC Step 7

L'intégration dans l'environnement de configuration de SIMATIC Step 7 autorise la gestion de projets WinCC flexible à l'intérieur de Step 7 et l'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 Step 7. Le SIMATIC Manager

nous donne accès à tous les objets de WinCC flexible. On peut créer, copier ou effacer des terminaux IHM tout comme des objets IHM individuels, tels que des images ou recettes.

Pendant la configuration, on accède directement à la table des mnémoniques et aux blocs de données Step 7 qu'on a défini au moment d'établissement du programme d'automate. Il n'est donc pas nécessaire de définir des variables procès dans WinCC flexible. La figure suivante montre la liaison entre la station S7-300 et la station de supervision HMI.

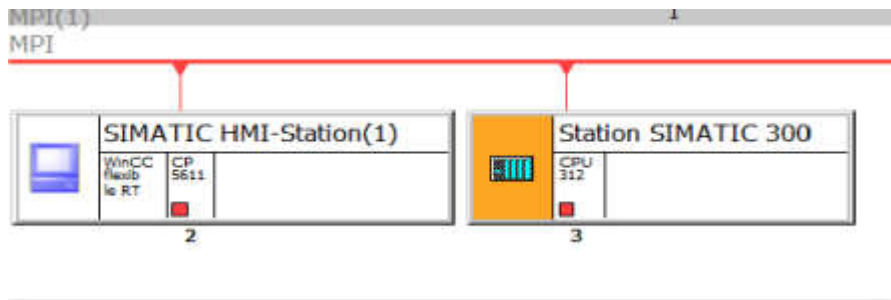
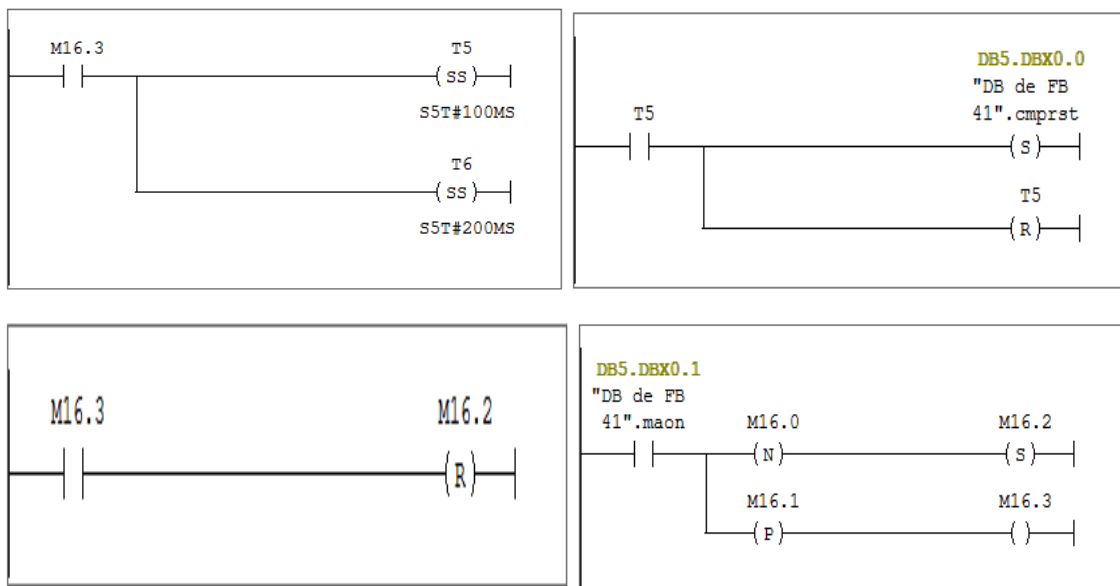


Figure 4.13 : Liaison entre la station S7-300 et la station HMI.

## 4.9 Partie programmation

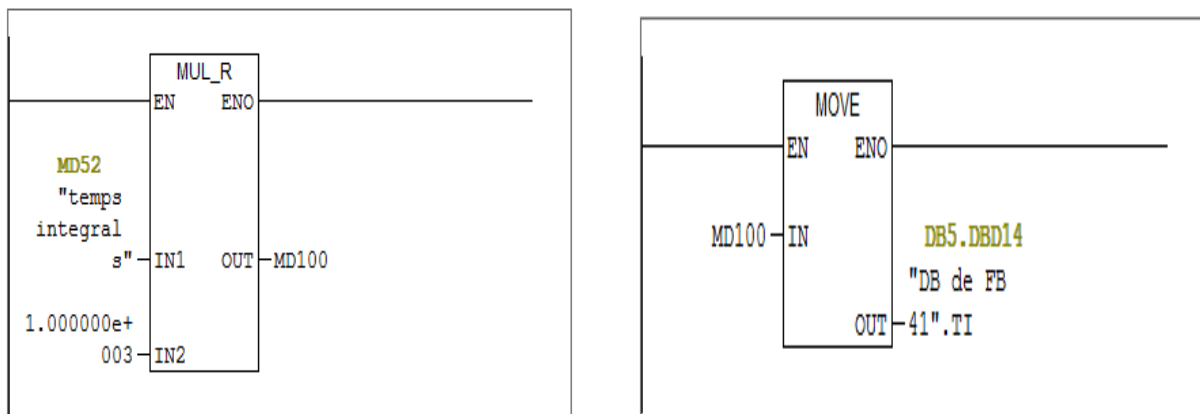
### 4.9.1 Programmer le mode manuel et le mode automatique

Pour le mode automatique le fonctionnement se fait automatiquement sans l'intervention de l'opérateur, la commande de marche et de l'arrêt vient de l'automate par contre pour le mode manuelle, il y a l'intervention de l'operateur c'est-à-dire forcer le démarrage ou l'arrêt d'un équipement en cliquant sur marche ou arrêt du mode manuelle à partir de la salle contrôle.



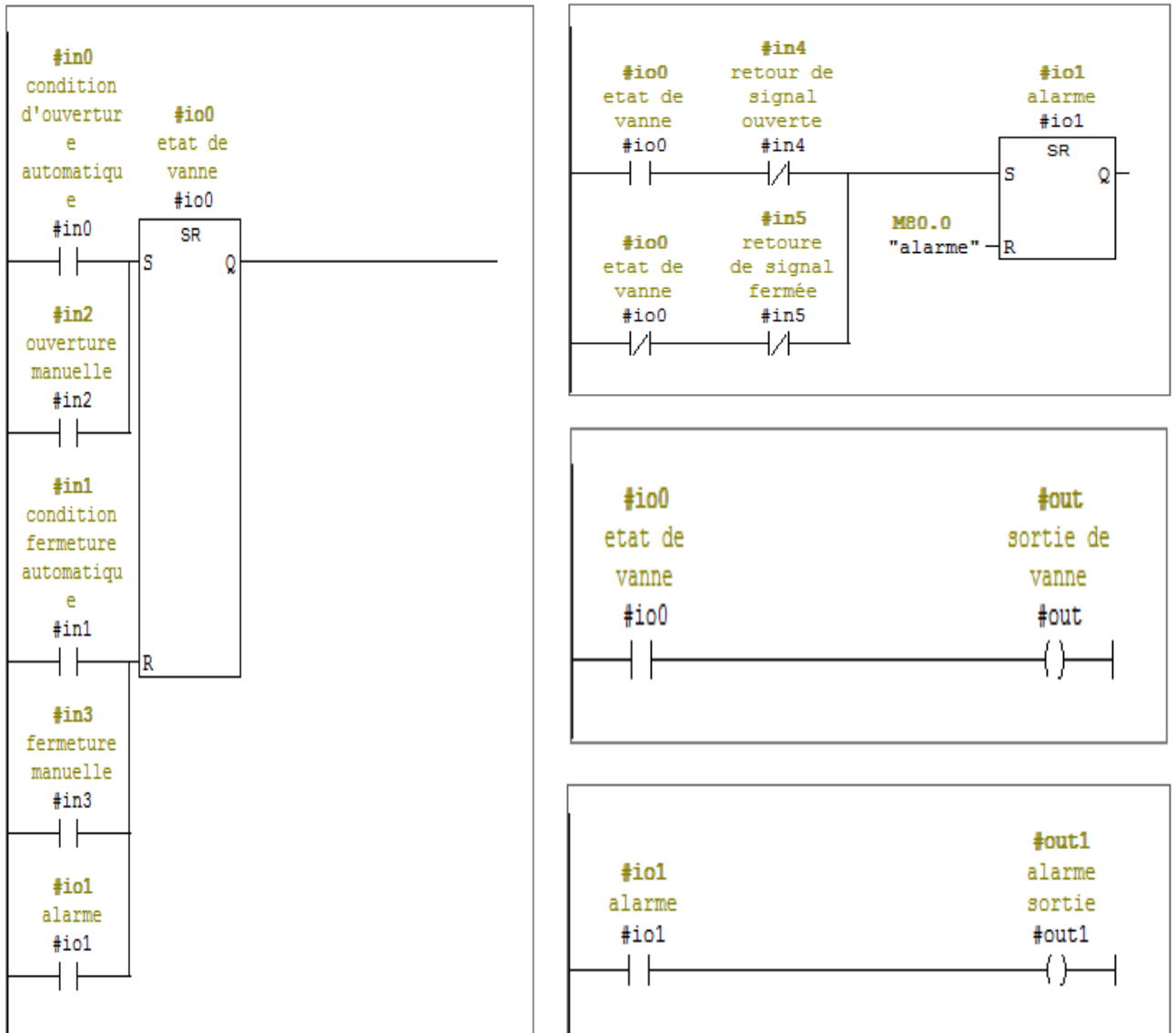
### Le temps d'intégrale

C'est le temps d'intégrale qu'est existe sur le bloc FB41.



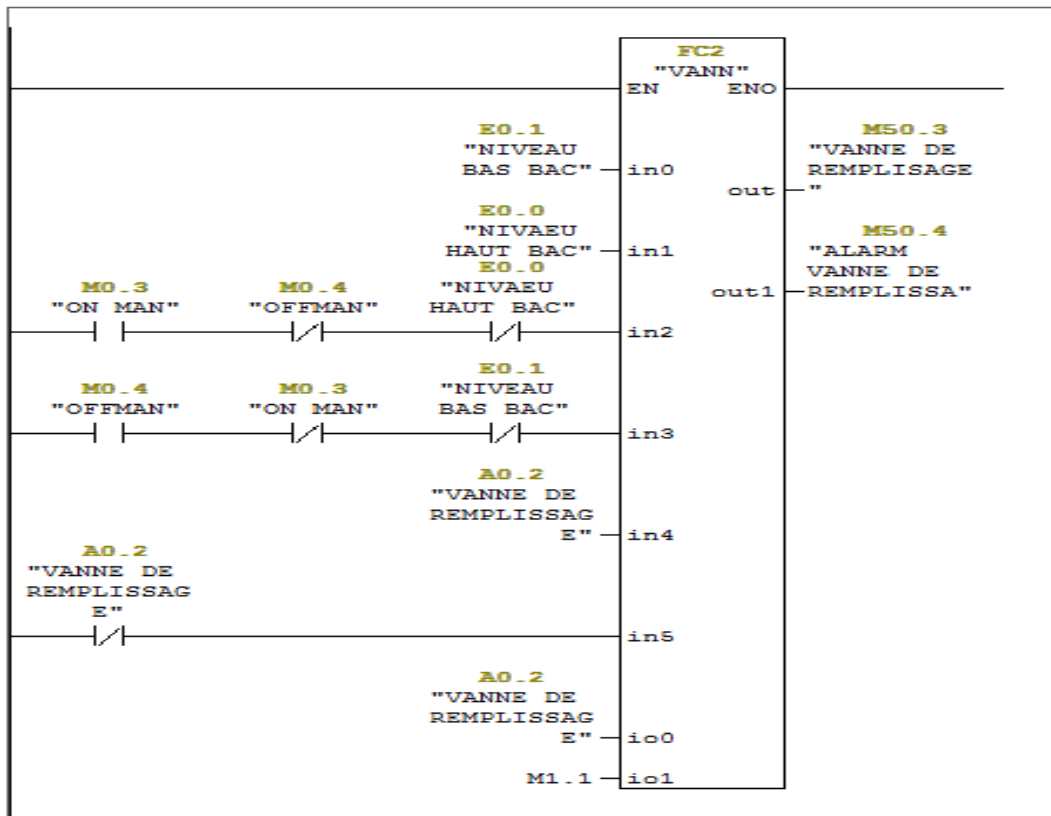
## Le fonctionnement de la vanne de remplissage

Le fonctionnement de la vanne de remplissage est programmé comme suit dans le bloc FC2.



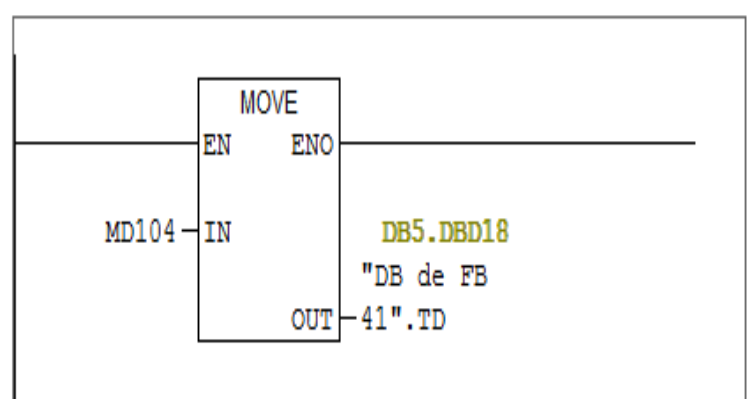
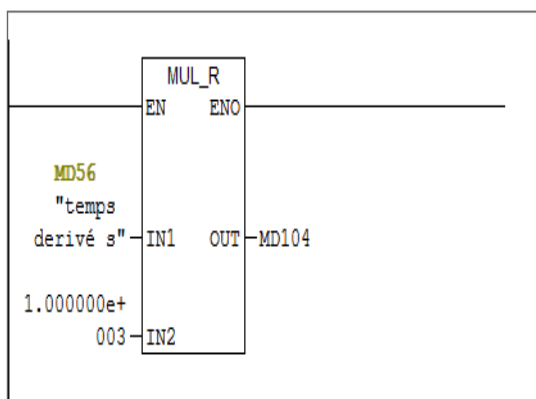


On fait l'appelle le bloc FC2 dans le bloc FC1 comme suit :



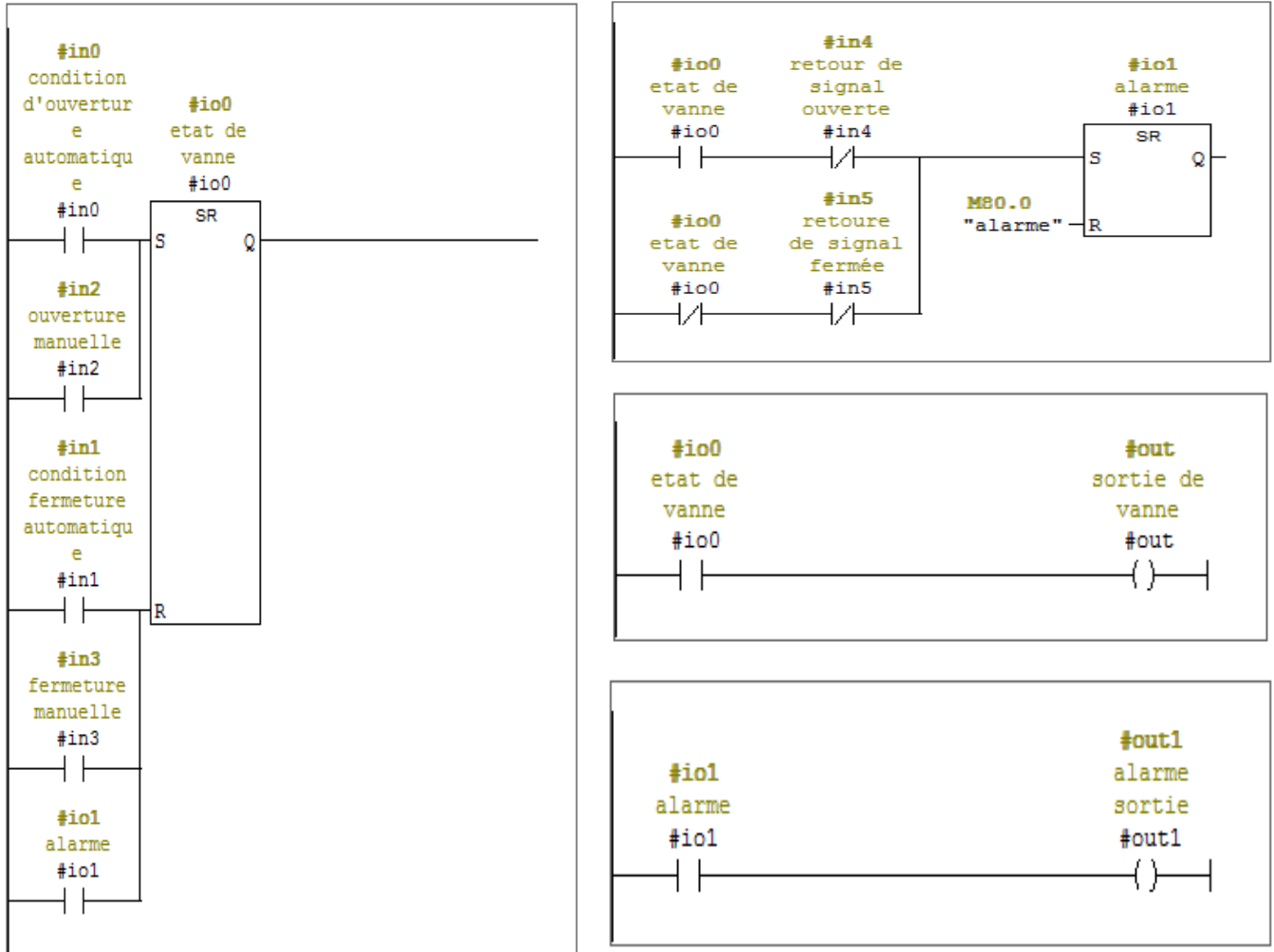
### Le temps de dérive

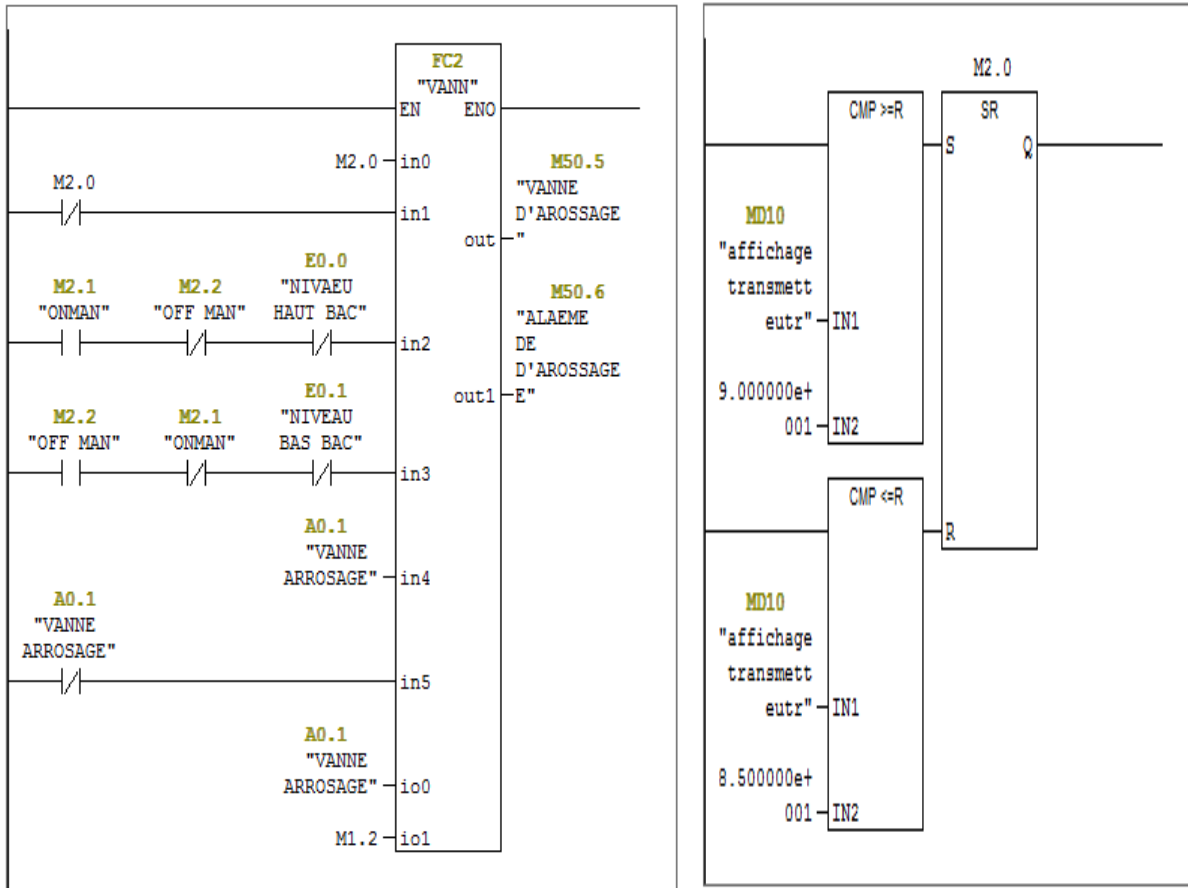
C'est le temps dérive qui existe dans le bloc FB41.



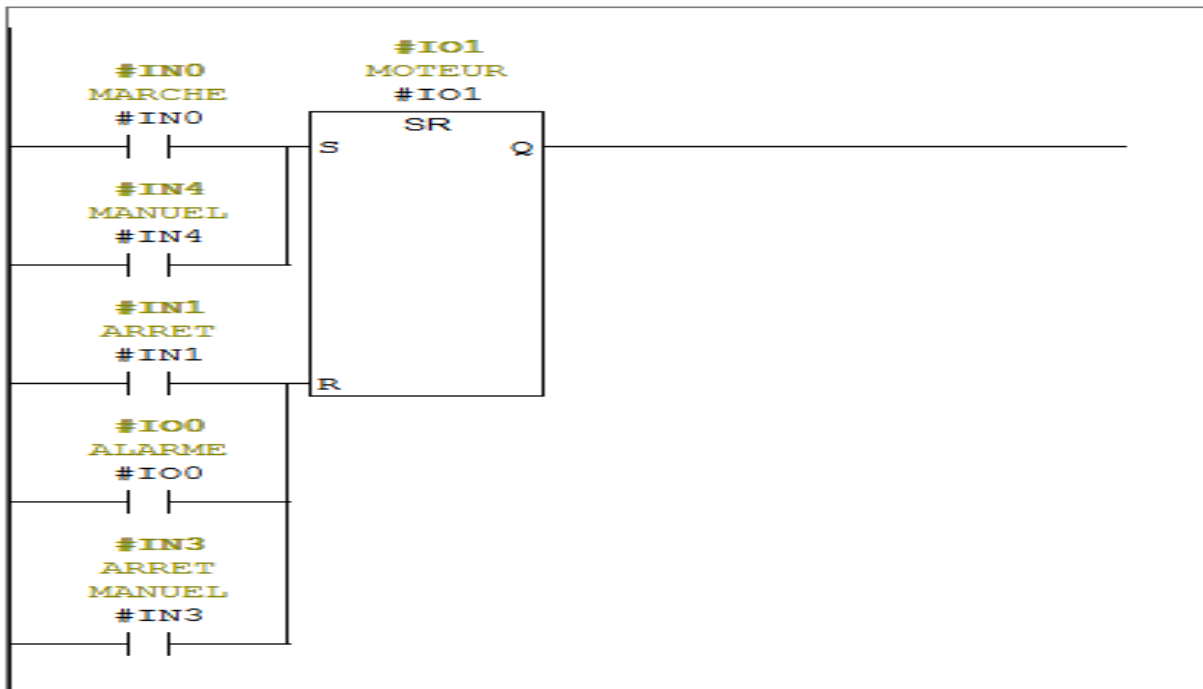
## Le fonctionnement de la vanne d'arrosage

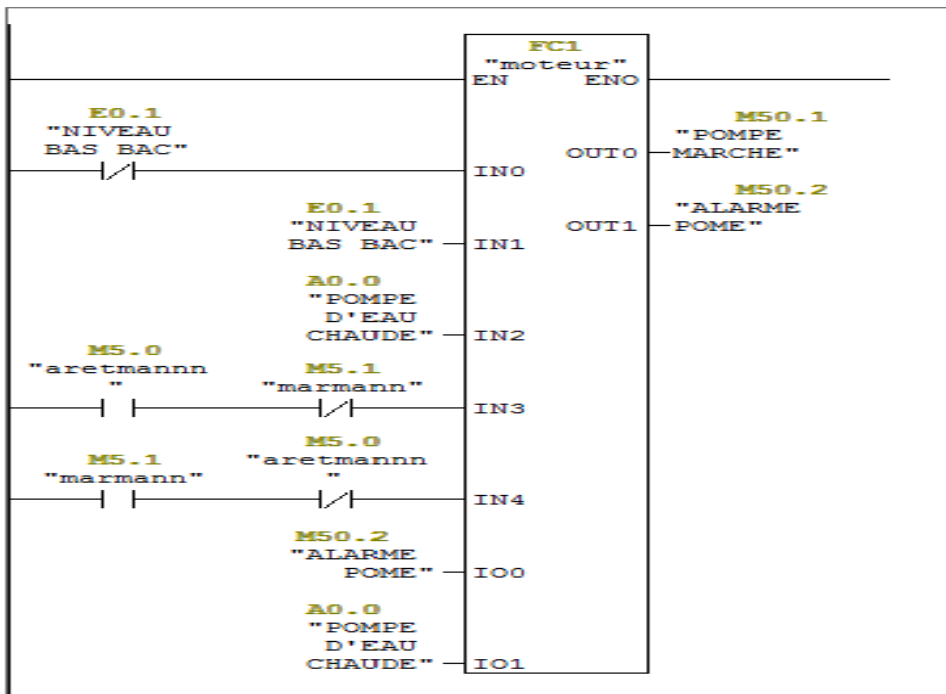
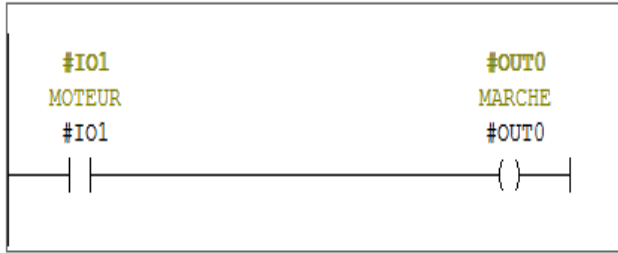
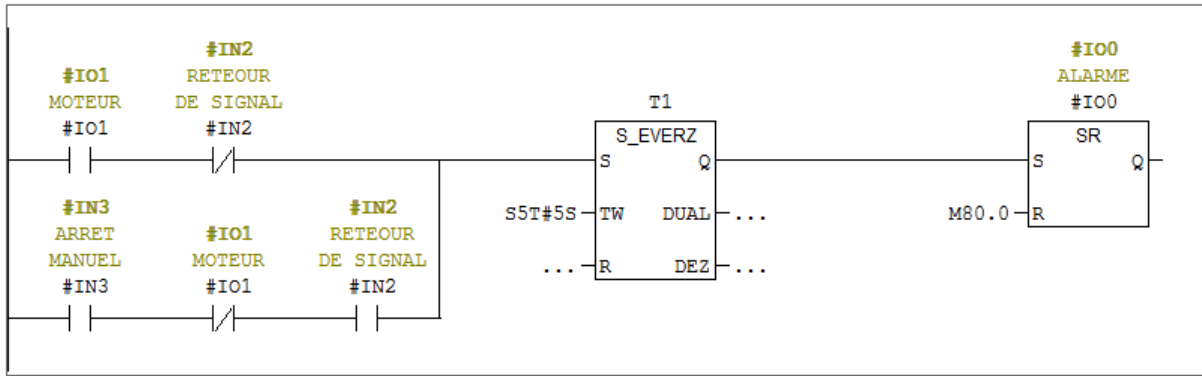
La fonction de vanne de remplissage dans FC2 :



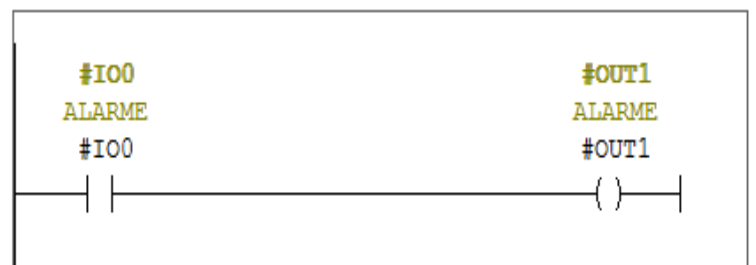
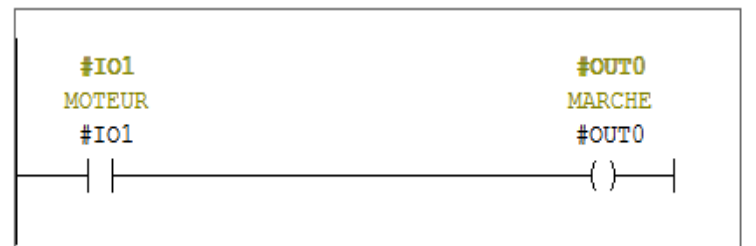
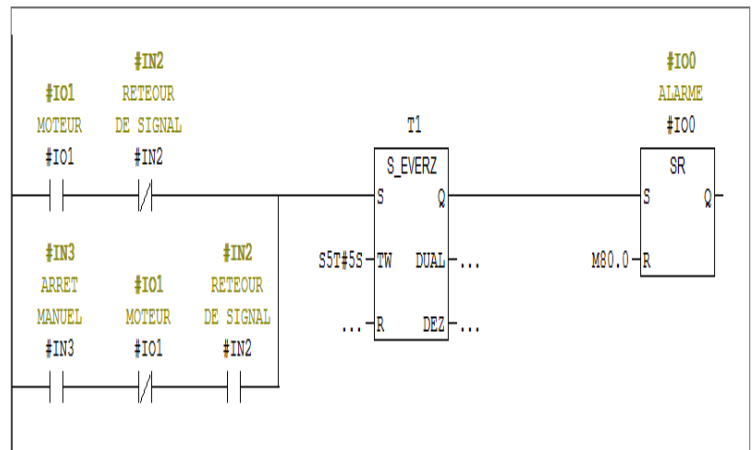
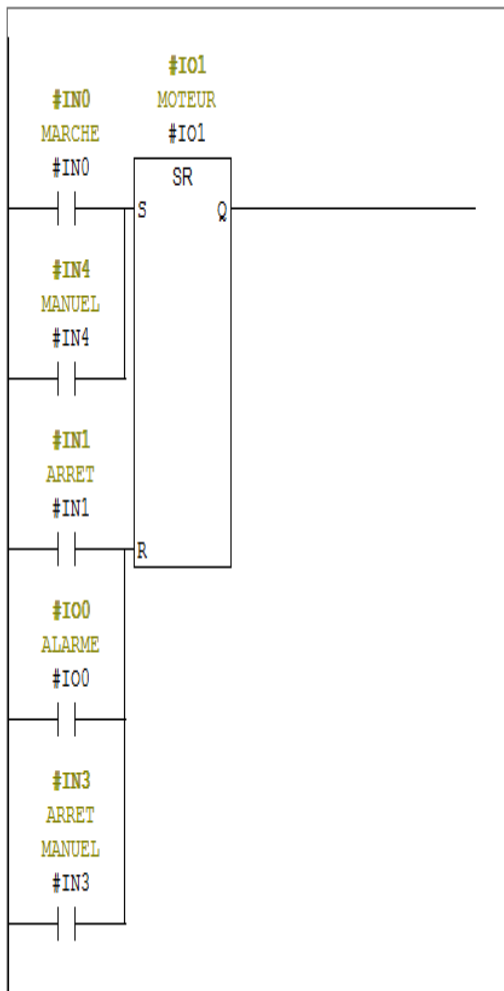
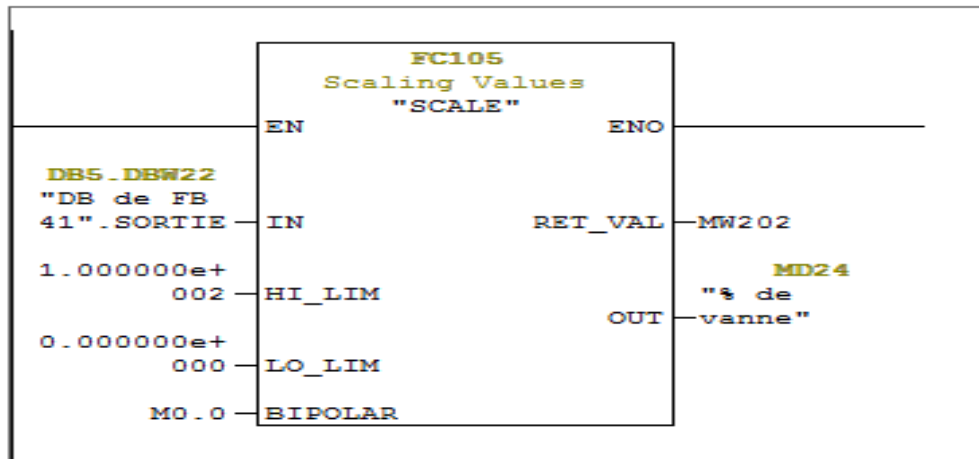


### Le fonctionnement de la pompe





### Le fonctionnement de la vanne régulatrice



## 4.10 Simulation

### 4.10.1 Simulation du programme sous STEP7 avec PLCSIM

Après programmation sous Step 7, on a intégré le programme dans le WinCC flexible pour réaliser le schéma de notre système.

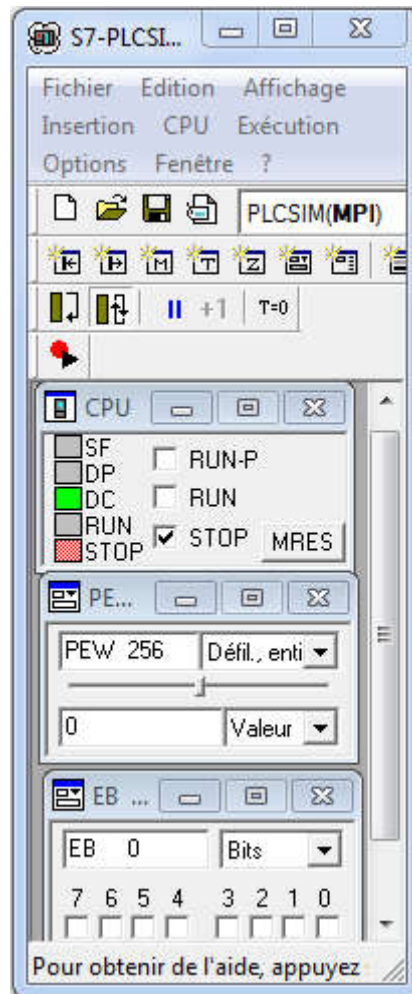


Figure 4.14 simulateur S7-PLC SIM

PEW 256 c'est la valeur de la température (la consigne).

EB 0 : c'est l'affichage des niveaux, haut et bas.

### 4.10.2 Simulation du programme sous WinCC

#### Systeme global

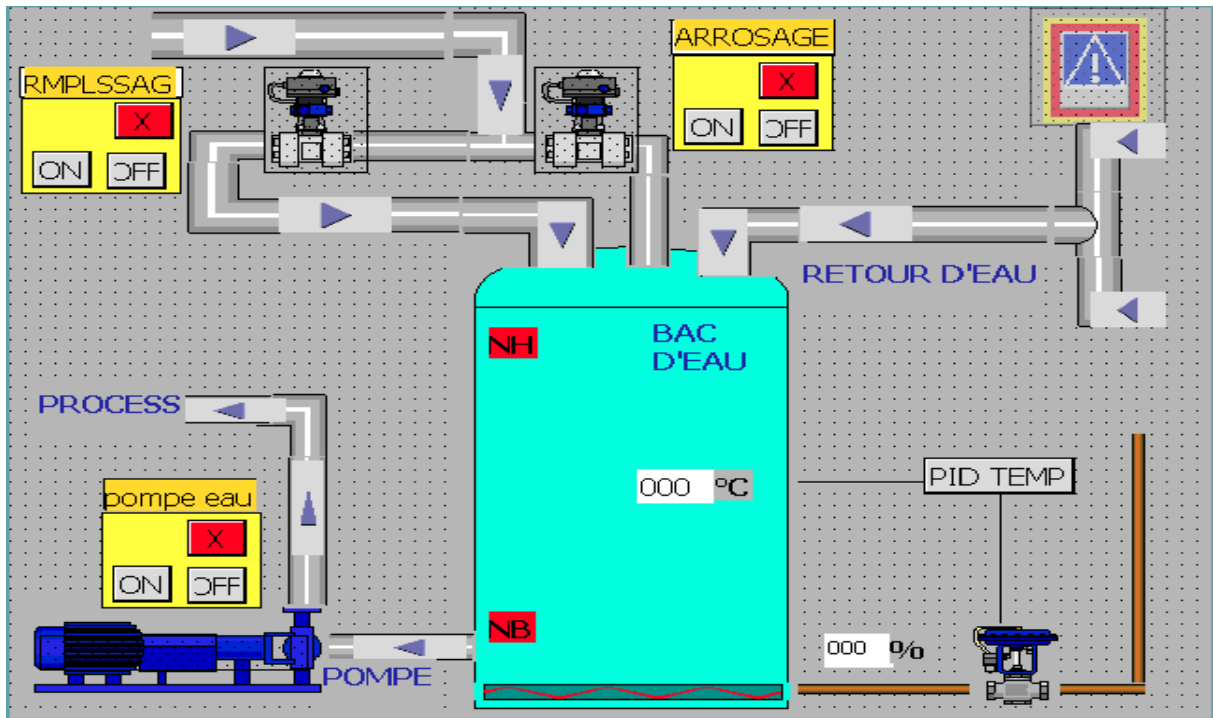


Figure 4.15 : Systeme global

### 4.10.3 Vue de régulation PID

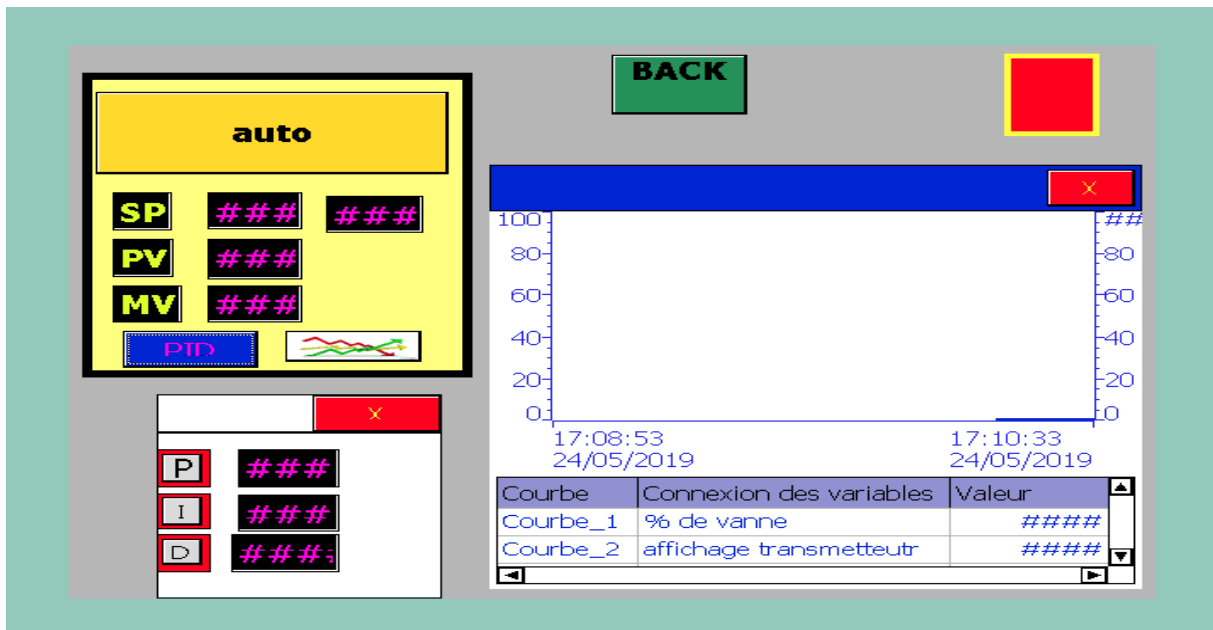


Figure 4.16 : Vue de régulateur PID

## 4.11 Conclusion

Dans ce chapitre, nous avons présenté une vue d'ensemble de l'automate programmable industriel que nous avons adopté pour le pilotage de notre système, le S7-300 ainsi que son logiciel de programmation SIMATIC Step 7, nous avons aussi, décrit la supervision en précisant sa place dans l'industrie.

La validation du programme que nous avons élaboré a été réalisée grâce au logiciel de simulation S7-PLCSIM. Cette procédure nous a permis d'apporter les modifications nécessaires pour la concrétisation de notre programme d'automatisation.

Nous avons développé une plateforme de supervision qui permet de suivre l'évolution de procédé en fonction du temps et qui simplifie la tâche de contrôle pour l'opérateur. Cette plateforme est développée sous le logiciel WinCC flexible, elle est composée de vues qui permettent une visualisation dynamique des entrées/sorties, afin de contrôler le bon fonctionnement de système.



# Conclusion générale

Au terme de ce travail, nous avons constaté que la réussite d'une réalisation d'un système de contrôle-commande repose essentiellement sur une meilleure analyse du procédé à commander, ainsi que le bon choix de l'équipement à utiliser.

Ce travail nous a permis de découvrir l'environnement industriel et de concrétiser nos connaissances théoriques dans le domaine pratique. Il nous a permis aussi de se familiariser avec des logiciels spécialisés dans les automatismes industriels comme le Step7, ainsi qu'avec un outil de supervision simple et puissant «WinCC flexible».

Notre travail présenté dans ce mémoire consiste à automatiser le futur système de recyclage d'eau osmose du complexe agro-alimentaire de Bejaia.

Dans ce contexte, nous avons étudié le fonctionnement du processus de système de recyclage et identifié les différents organes électriques et mécaniques. Ensuite, nous avons travaillé avec un automate programmable SIMATIC S7-300. La programmation a été réalisée à l'aide du logiciel Step7, cela afin d'arriver à la création d'une plateforme de supervision à l'aide du logiciel WinCC flexible, qui nous permet de réaliser des vues dans le but de contrôler l'état du système.

Cependant, la réalisation d'un bon système de supervision, nécessite la connaissance de certaines notions intégrées dans des technologies nouvelles de l'informatique.

En perspective, Pour mieux automatiser le système industriel, on propose de rajouter une vanne d'arrosage pour condenser la vapeur existante dans le bac.

## BIBLIOGRAPHIE

---

### BIBLIOGRAPHIE :

- [1] [www.google.dz/maps/place/Complexe+du+Groupe+Cevital](http://www.google.dz/maps/place/Complexe+du+Groupe+Cevital)
- [2] Christophe.B, « ENTRAINEMENT DES POMPES CENTRIFUGES A VITESSE VARIABLE POUR REDUIRE LA CONSOMMATION D'ENERGIE », Séminaire sur les entraînements électriques à haute école d'ingénierie et de Gestion du Canton de Vaud, Suisse, 12 novembre 2015.
- [3] ADJATI Arezki, « ETUDE DES MACHINES ASYNCHRONE DOUBLE ETOILE EN POMPAGE », Mémoire de Magister, Département Génie Électrique, Université de Bejaia 2012.
- [4] Ikhlef.M et Louibay.Y « AUTOMATISATION ET REGULATION D'UNE POMPE PAR UN DEBITMETRE », Mémoire de fin d'étude Master Automatique. Département Electronique, Université de Bejaia, 2016.
- [5] Documentation CEVITAL (Guide d'installation ; comment utiliser les électrovannes).
- [6] « JUMO Pressure Instruments », JUMO GmbH & Co. KG.
- [7] <http://www.endress.com/en/Tailor-made-field-instrumentation/level-measurement/Radar-measurement-Micropilot-FMR240>.
- [8] <https://www.wika.fr>
- [9] SEKHOK.S et LOUKILI.K « ETUDE D'UNE BOUCLE DE REGULATION DE NIVEAU : IMPLEMENTATION DU REGULATEUR ET REGULAGE DU PROCEDURE », projet de fin d'étude, 2011.
- [10] « Cours Automatique » Ecole polytechnique fédérale de Lausanne, 2008/2009.
- [11] S.MOUADH, « régulation industrielle de processus, système de régulation de niveau d'eau, interface à base de microprocesseur PIC 16F877, Commande & régulation avec LabVIEW », projet de fin d'étude, université de Tunisie.
- [12] GONZAKA Alian « Les automates programmable industriel » Novembre, 2004.
- [13] Andre Simon « Automate programmable niveau » Edition L'ELAN-LIEGE 1991.
- [14] JP VISKOVIc « Support Omron » ref : STA37-V3, 25/10/2007.
- [15] Philippe LE BRUN Automates programmables industriels ; Technologie, choix et mise en œuvre des automates programmables industriels; Lycée Louis ARMAND de Strasbourg ; Décembre 1999.
- [16] DEBBOU.S et GANI .L ; « automatisation de l'armoire SEG au sein de la cogénération du complexe de CEVITAL », mémoire d'ingénieur d'état en électrotechnique; 2011.

## BIBLIOGRAPHIE

---

- [17] Bssem.D « L'automate programmable industriel TS MAI l'automate programmable».
- [18] BERGOUGNOUX.L ; A.P.I. ; Automates Programmables Industriels ; POLYTECH' Marseille ; année 2004–2005.
- [19] F.HAMMOUCHI ; TP automatisme 2, université de Kasdi Merbah de Ouargla, année 2014/2015.
- [20] BTS CIRA, Automatismes, programmation des API Siemens S7-300, document de cours.
- [21] Louis SWEDI, « Commande des broyeurs secondaires par automate programmable industriel cas de l'entreprise minière RUASHI MINING », Université de LUBUMBASHI/école supérieure des ingénieurs industriels ; année 2009.
- [22] Walia and MANIK Jain, Vocational training at IATC in industrial automation (programmable logic controller).
- [23] Anchit.W et Manik.j « vocational training at IATC in industrial automation, programmable logic controllers ».
- [24] Bahloul.M et Toumi.A, « proposition d'automatisation d'un élévateur de préforme type A2 SIDEL par un automate programmable S7-300 (siemens) », mémoire de fin d'étude, 2012.
- [25] Programmer avec STEP 7; manuel siemens; édition 03/2006.
- [26] Configuration matérielle et communication dans STEP 7; manuel siemens; édition 03/2006.
- [27] Outils d'ingénierie S7-PLCSIM V5.4 ; manuel d'utilisation siemens ; édition 07/2011.
- [28] <http://www.cevital-agro-industrie.com/index.php/cevital-agr>
- [29] <http://www.cevital.com/fr/cevital-agro-industrie.html>
- [30] <http://www.sectoriel.fr>
- [31] Documentation sur cd Simatic « Wincc control center » ref : C79000-G7077-C516-01.

# Présentation de l'organisme d'accueil CEVITAL

## 1.1. Présentation de l'entreprise CEVITAL

Créée en 1998, CEVITAL Agro-industrie est le leader du secteur agroalimentaire en Algérie. Elle est composée de plusieurs unités de production telles que : raffinerie d'huile, raffinerie de sucre, margarinerie, unité de conditionnement d'eau minérale, unité de fabrication et de conditionnement de boisson rafraichissante, conserverie, silos portuaires ainsi qu'un terminal de déchargement portuaire [29].

Cette importante entreprise offre des produits de qualité supérieure à des prix compétitifs, grâce à son savoir-faire, ses unités de production ultramodernes, son contrôle strict de qualité et son réseau de distribution. Elle contribue largement au développement et à l'amélioration de l'industrie agroalimentaire nationale ainsi que de faire passer l'Algérie du stade d'importateur à celui d'exportateur pour les huiles, les margarines et le sucre [30].

## 1.2. Situation géographique

Implantée au sein du port de Bejaia (Algérie) et s'étend sur une superficie de 45000 m<sup>2</sup> a 3 Km de cette ville, a proximité de la route national N° 9 et N° 12 et en effet cette situation lui a beaucoup profité étant donnée qu'elle lui donne l'avantage de proximité économique car elle est proche du port et de l'aéroport [29], [17], [31].

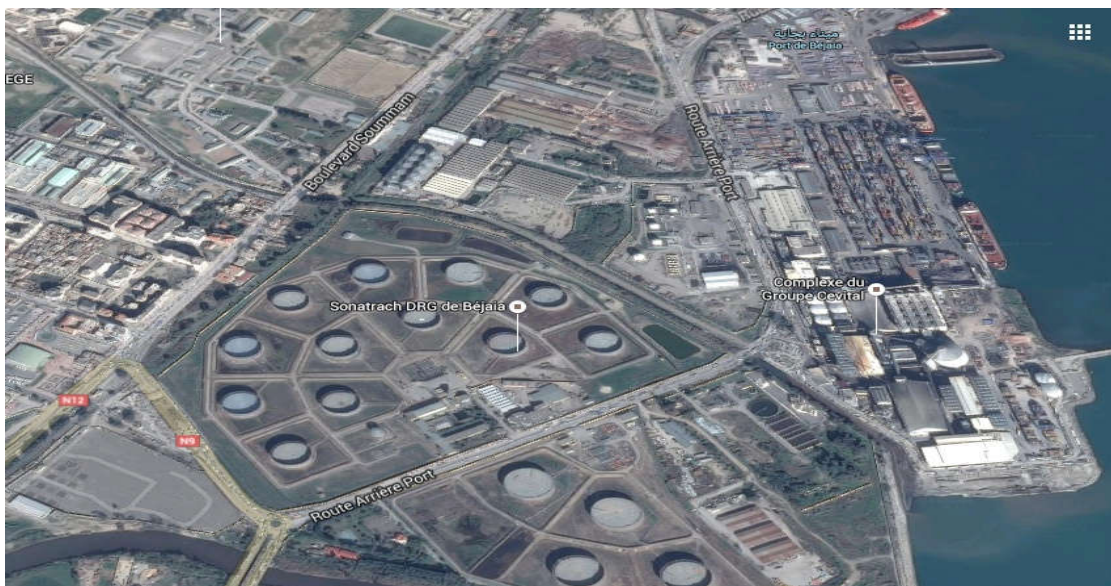


Figure 1 : Situation géographique du complexe CEVITAL [31].

## ANNEXE 1

### 1.3. Principal activité de l'entreprise

Avec un effectif de 2356 employée CEVITAL a débutée ses activités par :

- Le conditionnement en décembre 1998.
- Génie civile et de la raffinerie en février 1999 et devenue fonctionnel en août de la même année [24].
- CEVITAL concentre ses activités sur la production agroalimentaire qui sont représentées dans le tableau qui suit :
- **Tableau 1 : tableau récapitulatif de la capacité de production du complexe CEVITAL [29].**

Unité	Capacité
Une raffinerie d'huile	670 000 tonnes/ an
Une margarinerie	180 000 tonnes/an
Une raffinerie de sucre	2 000 000 tonnes/an
Une unité de sucre liquide	210 000 tonnes (matière sèche)/an
Une unité de conditionnement d'eau minérale lalla khedidja	3 000 000 bouteilles /jour
Une unité de production de boisson rafraichissante sans alcool Tchina	600 000 bouteilles/heure
Une conservation de tomates et de confiture de fruits	80 tonnes/jour
Des silos portuaires	182 000 tonnes

# ANNEXE 1

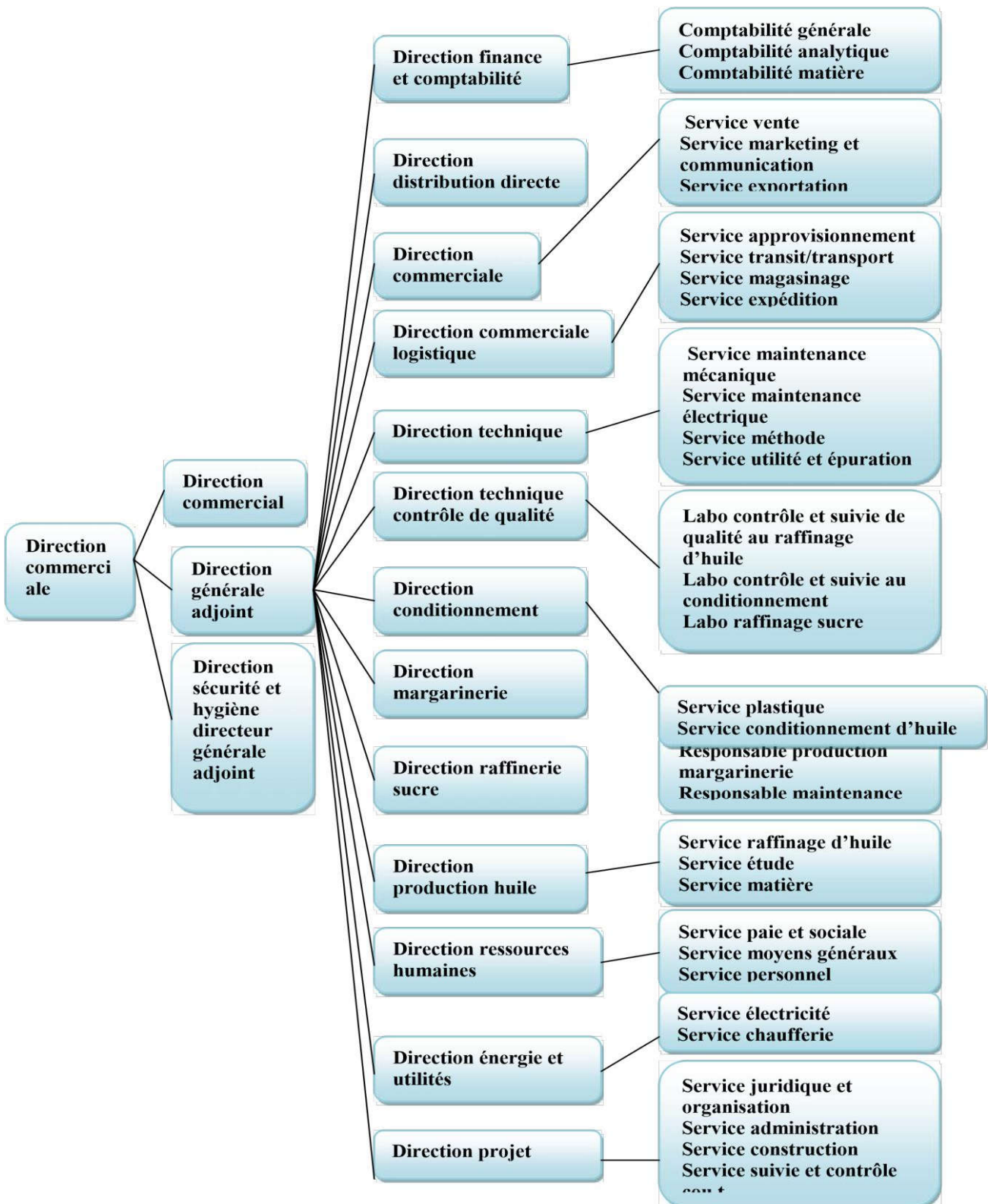


Figure 1: Organigramme de l'entreprise [29].

## ANNEXE 1

---

## ANNEXE 2

Le OB 35 :

DB35			
FB41			
Continuous Control			
"CONT_C"			
EN	ENO		
<b>DB5.DEX0.0</b> "DB de FB 41".cmprst	COM_RST	LMN	...
<b>DB5.DEX0.1</b> "DB de FB 41".maon	MAN_ON	LMN_PER	<b>DB5.DEX22</b> "DB de FB 41".SORTIE
<b>DB5.DEX0.2</b> "DB de FB 41". pvperon	FVPER_ON	QLMN_HLM	...
		QLMN_LLM	...
		LMN_P	...
<b>DB5.DEX0.3</b> "DB de FB 41".psel	P_SEL	LMN_I	...
		LMN_D	...
<b>DB5.DEX0.4</b> "DB de FB 41".isel	I_SEL	FV	...
		ER	...
	INT_HOLD		
	I_ITL_ON		
<b>DB5.DEX0.5</b> "DB de FB 41".dsel	D_SEL		
	T#2MS	CYCLE	
<b>DB5.DEX2</b> "DB de FB 41".SP	SP_INT		
<b>MD10</b> "affichage transmett eutr"	FV_IN		
PEW256	FV_PER		

<b>DB5.DEX6</b> "DB de FB 41".MAN	MAN
<b>DB5.DEX10</b> "DB de FB 41".GAIN	GAIN
<b>DB5.DEX14</b> "DB de FB 41".TI	TI
<b>DB5.DEX18</b> "DB de FB 41".TD	TD
	... TM_LAG
	... DEADB_W
	... LMN_HLM
	... LMN_LLM
	... FV_FAC
	... FV_OFF
	... LMN_FAC
	... LMN_OFF
	... I_ITLVAL
	... DISV



## ANNEXE 2

### Le contenu du bloc de donné d'instance pour le FB41

Adresse	Nom	Type	Valeur initiale	Commentaire
0.0		STRUCT		
+0.0	cmprst	BOOL	FALSE	
+0.1	maon	BOOL	FALSE	
+0.2	pvperon	BOOL	FALSE	
+0.3	psel	BOOL	FALSE	
+0.4	isel	BOOL	FALSE	
+0.5	dsel	BOOL	FALSE	
+2.0	SP	REAL	0.000000e+000	
+6.0	MAN	REAL	0.000000e+000	
+10.0	GAIN	REAL	0.000000e+000	
+14.0	TI	TIME	T#0MS	
+18.0	TD	TIME	T#0MS	
+22.0	SORTIE	INT	0	
=24.0		END_STRUCT		

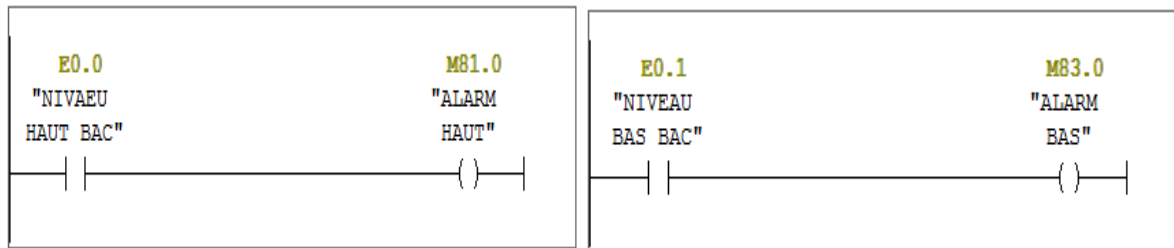
### Le bloc de donné en temps réel de la fonction FB41

	Adresse	Décl.	Nom	Type	Valeur initiale	Valeur en cou	Commentaire
1	0.0	in	COM_RST	BOOL	FALSE	FALSE	complete restart
2	0.1	in	MAN_ON	BOOL	TRUE	TRUE	manual value on
3	0.2	in	PVPER_...	BOOL	FALSE	FALSE	process variable peripherie on
4	0.3	in	P_SEL	BOOL	TRUE	TRUE	proportional action on
5	0.4	in	I_SEL	BOOL	TRUE	TRUE	integral action on
6	0.5	in	INT_HOLD	BOOL	FALSE	FALSE	integral action hold
7	0.6	in	I_ITL_ON	BOOL	FALSE	FALSE	initialization of the integral action
8	0.7	in	D_SEL	BOOL	FALSE	FALSE	derivative action on
9	2.0	in	CYCLE	TIME	T#1S	T#1S	sample time
10	6.0	in	SP_INT	REAL	0.000000e+...	0.000000e+...	internal setpoint
11	10.0	in	PV_IN	REAL	0.000000e+...	0.000000e+...	process variable in
12	14.0	in	PV_PER	WORD	W#16#0	W#16#0	process variable peripherie
13	16.0	in	MAN	REAL	0.000000e+...	0.000000e+...	manual value
14	20.0	in	GAIN	REAL	2.000000e+...	2.000000e+...	proportional gain
15	24.0	in	TI	TIME	T#20S	T#20S	reset time
16	28.0	in	TD	TIME	T#10S	T#10S	derivative time
17	32.0	in	TM_LAG	TIME	T#2S	T#2S	time lag of the derivative action
18	36.0	in	DEADB_W	REAL	0.000000e+...	0.000000e+...	dead band width
19	40.0	in	LMN_HLM	REAL	1.000000e+...	1.000000e+...	manipulated value high limit
20	44.0	in	LMN_LLM	REAL	0.000000e+...	0.000000e+...	manipulated value low limit
21	48.0	in	PV_FAC	REAL	1.000000e+...	1.000000e+...	process variable factor
22	52.0	in	PV_OFF	REAL	0.000000e+...	0.000000e+...	process variable offset
23	56.0	in	LMN_FAC	REAL	1.000000e+...	1.000000e+...	manipulated value factor
24	60.0	in	LMN_OFF	REAL	0.000000e+...	0.000000e+...	manipulated value offset
25	64.0	in	I_ITLVAL	REAL	0.000000e+...	0.000000e+...	initialization value of the integral acti
26	68.0	in	DISV	REAL	0.000000e+...	0.000000e+...	disturbance variable
27	72.0	out	LMN	REAL	0.000000e+...	0.000000e+...	manipulated value
28	76.0	out	LMN_PER	WORD	W#16#0	W#16#0	manipulated value peripherie

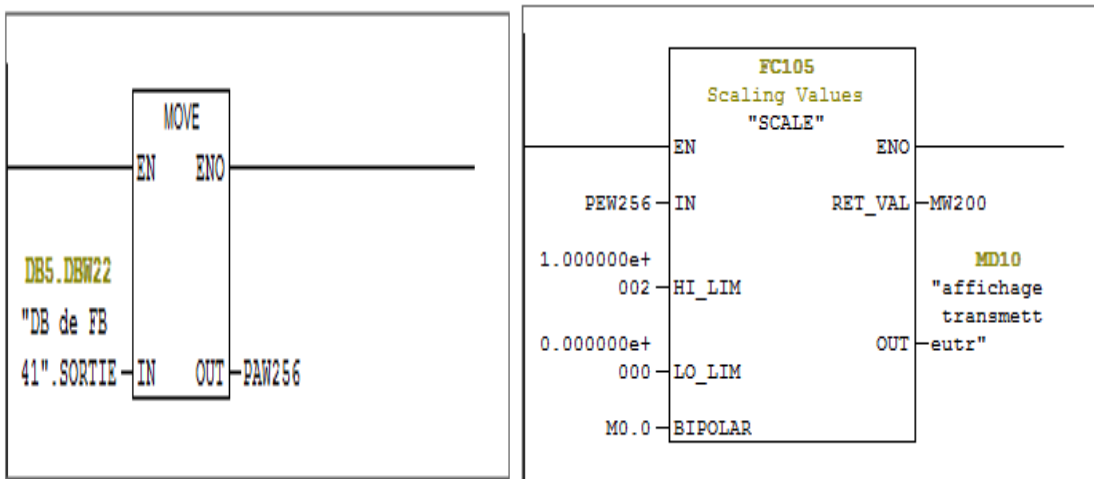
## ANNEXE 2

29	78.0	out	QLMN_...	BOOL	FALSE	FALSE	high limit of manipulated value reach
30	78.1	out	QLMN_L...	BOOL	FALSE	FALSE	low limit of manipulated value reach
31	80.0	out	LMN_P	REAL	0.000000e+...	0.000000e+...	proportionality component
32	84.0	out	LMN_I	REAL	0.000000e+...	0.000000e+...	integral component
33	88.0	out	LMN_D	REAL	0.000000e+...	0.000000e+...	derivative component
34	92.0	out	PV	REAL	0.000000e+...	0.000000e+...	process variable
35	96.0	out	ER	REAL	0.000000e+...	0.000000e+...	error signal
36	100.0	stat	sInvAlt	REAL	0.000000e+...	0.000000e+...	
37	104.0	stat	sIanteilAlt	REAL	0.000000e+...	0.000000e+...	
38	108.0	stat	sRestInt	REAL	0.000000e+...	0.000000e+...	
39	112.0	stat	sRestDif	REAL	0.000000e+...	0.000000e+...	
40	116.0	stat	sRueck	REAL	0.000000e+...	0.000000e+...	
41	120.0	stat	sLmn	REAL	0.000000e+...	0.000000e+...	
42	124.0	stat	sbArwH...	BOOL	FALSE	FALSE	
43	124.1	stat	sbArwL...	BOOL	FALSE	FALSE	
44	124.2	stat	sbILimOn	BOOL	TRUE	TRUE	

### Le programme des alarmes :

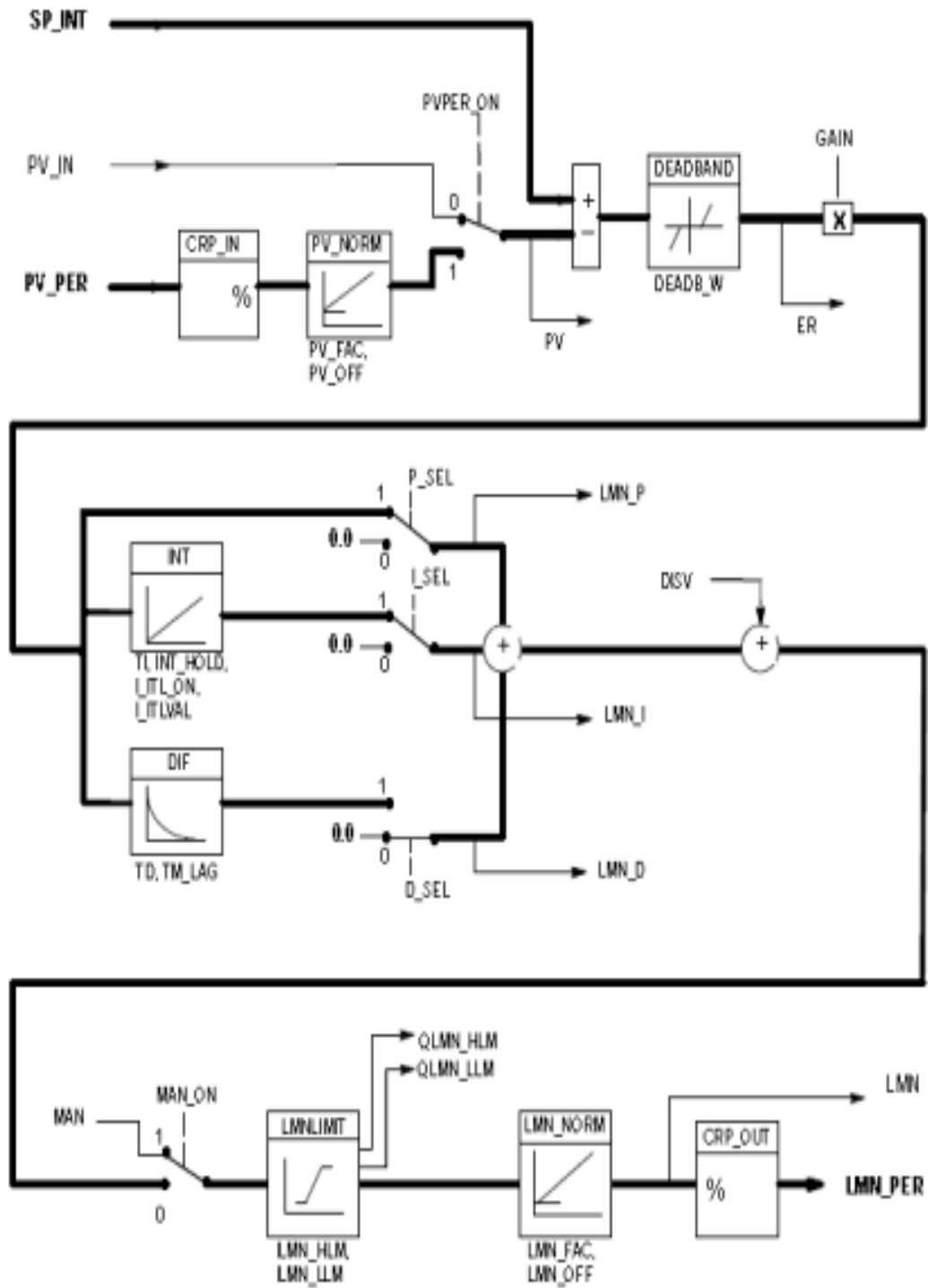


### Capteur de température :



# ANNEXE 3

Le schéma fonctionnel d'un PID [31] :



## ANNEXE 3

Paramètres d'entrée (INPUT) du bloc FB 41 [31]

Paramètre	Type de données	Valeurs admises	Par défaut	Description
COM_RST	BOOL		FALSE	COMPLETE RESTART / Démarrage Le bloc renferme un sous-programme de démarrage 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. La valeur deréglage manuelle est sortie comme grandeur de réglage.
PVPER_ON	BOOL		FALSE	PROCESS VARIABLE PERIPHERY ON / Activation de la mesure de périphérie Pour que la mesure soit lue en 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 positionnée sur 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 est active quand cette entrée est à 1.
CYCLE	TIME	$\geq 1$ ms	T#1s	SAMPLE TIME / Période d'échantillonnage Le temps s'écoulant entre les appels de bloc doit être constant. Il est indiqué au niveau de cette entrée.
SP_INT	REAL	-100,0 à 100,0 (%) ou grandeur physique <sup>1</sup>	0,0	INTERNAL SETPOINT / Consigne interne Cette entrée sert à introduire une valeur de consigne.
PV_IN	REAL	-100,0 à 100,0 (%) ou grandeur physique <sup>1</sup>	0,0	PROCESS VARIABLE IN / Mesure d'entrée Cette entrée permet de paramétrer une valeur de mise en service ou d'appliquer une mesure externe en virgule flottante.

## ANNEXE 3

Paramètres d'entrée (INPUT) du bloc FB41 (suite) [31]

Paramètre	Type de données	Valeurs admises	Par défaut	Description
PV_PER	WORD		W#16#0000	PROCESS VARIABLE PERIPHERIE / Mesure de périphérie La mesure en format de périphérie est appliquée au régulateur par cette entrée.
MAN	REAL	-100,0 à 100,0 (%) ou grandeur physique <sup>2</sup>	0,0	MANUAL VALUE / Valeur de réglage manuelle Cette entrée sert à introduire une valeur de réglage manuelle moyennant 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 la réponse temporelle de l'intégrateur.
TD	TIME	>= CYCLE	T#10 s	DERIVATIVE TIME / Temps de dérivation Cette entrée détermine la réponse temporelle de l'unité de dérivation.
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 <sup>1</sup>	0,0	DEAD BAND WIDTH / Largeur de zone morte Le signal d'erreur traverse une zone morte. Cette entrée détermine la taille de la zone morte.
LMN_HLM	REAL	LMN_LLM à 100,0 (%) ou grandeur physique <sup>2</sup>	100,0	MANIPULATED VALUE HIGH LIMIT / Limite supérieure de la valeur de réglage La valeur de réglage est toujours bornée à une limite supérieure et une limite inférieure. Cette entrée indique la limite supérieure.
LMN_LLM	REAL	-100,0 à LMN_HLM (%) ou grandeur physique <sup>2</sup>	0,0	MANIPULATED VALUE LOW LIMIT / Limite inférieure de la valeur de réglage La valeur de réglage est toujours bornée à une limite supérieure et une limite inférieure. Cette entrée indique la limite inférieure.
PV_FAC	REAL		1,0	PROCESS VARIABLE FACTOR / Facteur de mesure Cette entrée est multipliée par la mesure. Elle sert à adapter l'étendue de valeur de mesure.
PV_OFF	REAL		0,0	PROCESS VARIABLE OFFSET / Décalage de mesure Cette entrée est ajoutée à la mesure. Elle sert à adapter l'étendue de valeur de 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 l'étendue de réglage.

## ANNEXE 3

Paramètres d'entrée (INPUT) du bloc FB41 (suite) [31]

Paramètre	Type de données	Valeurs admises	Par défaut	Description
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 l'étendue de réglage.
I_ITLVAL	REAL	-100,0 à 100,0 (%) ou grandeur physique <sup>2</sup>	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_ITL_ON. La valeur d'initialisation est appliquée à l'entrée I_ITLVAL.
DISV	REAL	-100,0 à 100,0 (%) ou grandeur physique <sup>2</sup>	0,0	DISTURBANCE VARIABLE / Grandeur perturbatrice La grandeur perturbatrice est appliquée à cette entrée pour l'action anticipatrice.

Paramètres de sortie (OUTPUT) du bloc FB 41 [31]

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 Cette sortie fournit la valeur de réglage en format de périphérie.
QLMN_HLM	BOOL		FALSE	HIGH LIMIT OF MANIPULATED VALUE REACHED / Grandeur de réglage à la limite supérieure La valeur de réglage est toujours bornée à une limite supérieure et à une limite inférieure. Cette sortie signale le dépassement de la limite supérieure.
QLMN_LLM	BOOL		FALSE	LOW LIMIT OF MANIPULATED VALUE REACHED / Grandeur de réglage à la limite inférieure La valeur de réglage est toujours bornée à une limite supérieure et à une limite inférieure. Cette sortie signale le dépassement de la limite inférieure.
LMN_P	REAL		0,0	PROPORTIONALITY COMPONENT / Composante P Cette sortie contient la composante proportionnelle de la grandeur de réglage.

## ANNEXE 3

---

Paramètres de sortie (OUTPUT) du bloc FB41 (suite) [31]

Paramètre	Type de données	Valeurs admises	Par défaut	Description
LMN_I	REAL		0,0	INTEGRAL COMPONENT / Composante I Cette sortie contient la composante intégrale de la grandeur de réglage.
LMN_D	REAL		0,0	DERIVATIVE COMPONENT / Composante D Cette sortie contient la composante différentielle de la grandeur de réglage.
PV	REAL		0,0	PROCESS VARIABLE / Mesure Cette sortie donne la mesure effective.
ER	REAL		0,0	ERROR SIGNAL / Signal d'erreur Cette sortie donne le signal d'erreur effectif.

## ANNEXE 4

La plaque signalétique de la pompe centrifuge

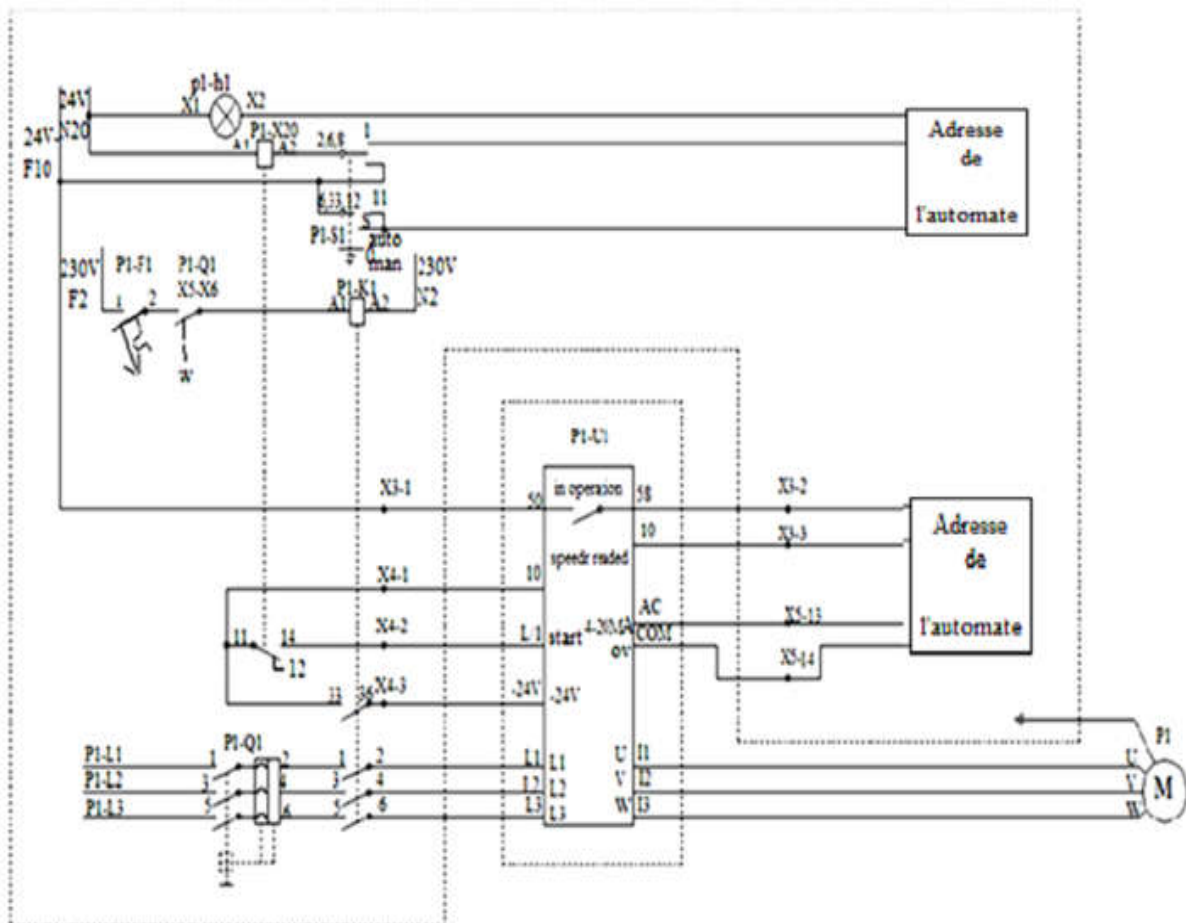
KSB Aktiengesellschaft									
V	Hz	A	kW	cos $\phi$	l/min	Certif.			
400 $\Delta$	50	65	37	0,89	2955				
690 Y		37,5							
460 $\Delta$	60	63	41,5	0,89	3555				

IEC/EN 60034

80-420 VA, 67-63 A; 660-725 V, 39-36,5 A; 400 Hz

440-480 VA, 66-61 A, 60 Hz

Le schéma fonctionnel de la pompe





## Résumé

---

Les automates programmables industriels représentent l'élément important de la chaîne automatisée, car ils assurent une meilleure flexibilité et facilitent la maintenance.

Ce présent travail est réalisé au sein de l'entreprise CEVITAL, avec l'équipe maintenance raffinerie d'huile.

Le but de notre travail consiste à automatiser une partie de l'usine en mettant au point une régulation de température avec un système de fonctionnement commandé par un automate programmable industriel API Siémons S7-300 et nous allons intégrer une nouvelle technique de surveillance qui est la supervision.

La réalisation de ce projet nous a permis de mettre en évidence l'aspect pratique de nos connaissances théoriques acquises, ainsi enrichir nos connaissances et notre savoir-faire dans le domaine de la conception et l'automatisation des systèmes industriels.

### **Abstract**

The industrial programmable automats represent the important component of the automated chain, because they ensure a better flexibility and facilitate maintenance.

The goal of our work consists in automating part of the factory by developing a regulation of tempertur with a system of operation ordered by an industrial programmable automat API Siemens S7-300 and we will integrate a novel method of monitoring which is the supervision.

This present work is completed within company CEVITAL, with the team maintenance refinery of oil.

The realization of this project enabled us to highlight the practical aspect of our acquired theoretical knowledge, thus enrich our knowledge and our know-how in the field by the design and the automation of the industrial systems.