

**RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE**  
**MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE**  
**SCIENTIFIQUE**

## **UNIVERSITÉ IBN-KHALDOUN DE TIARET**

**FACULTÉ DES SCIENCES APPLIQUEES**  
**DÉPARTEMENT DE GENIE ELECTRIQUE**



# **MEMOIRE DE FIN D'ETUDES**

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

**Domaine : Sciences et Technologie**

**Filière : Automatique**

**Spécialité : Automatique et informatique Industrielle**

## **THÈME**

**Supervision d'un niveau d'eau via un automate programmable :  
Etude et Conception**

**Préparé par :**  
**Zerrouk Imad Eddine**  
**Yeddou Youcef**

**Devant le Jury :**

<b>Nom et prénoms</b>	<b>Grade</b>	<b>Qualité</b>
<b>M. LARBI</b>	<b>Pr</b>	<b>Président</b>
<b>T. ALLAOUI</b>	<b>Pr</b>	<b>Examineur 1</b>
<b>M. SEBAA</b>	<b>Pr</b>	<b>Examineur 2</b>
<b>H. BOUMEDIENE</b>	<b>MCB</b>	<b>Encadreur</b>
<b>S. HASSAINE</b>	<b>Pr</b>	<b>Co-Encadreur</b>

**PROMOTION 2021/2022**

## Dédicace

Nous dédions cet humble travail : A nos chers parents, pour leurs soutiens et leurs encouragements tout au long de nos années scolaires, aucun honneur ne peut égaler l'amour qu'ils continuent à donner. Que Dieu leur accorde bien-être et longue vie.

A nos chers frères et chères sœurs, que Dieu nous les garde.

A notre grand-mère, que Dieu prolonge sa vie.

À tous nos chers amis, tous nommés et spéciaux.

Et à **\*\*Ben Bella Walid\*\*** en particulier.

A tous nos enseignants et collègues.

Et à tous ceux qui placent nos succès dans leurs  
cœurs

Puissent-ils trouver ici l'expression de toutes nos  
gratitudes.

Imad Eddine & Youcef

# REMERCIEMENTS

Nous voulons d'abord remercier Dieu, pour avoir nous accorder la force et la patience pour accomplir cette humble tâche.

Deuxièmement, nous tenons à remercier nos chers parents qui ont toujours été avec nous et qui ont donné un excellent exemple de travail acharné et de persévérance.

Nous remercions également :

**Mr. Boumediene Hamid** et **Mr. Hassaine Saïd** qui nous ont gracieusement encadrés pour élaborer et suivre ce projet de fin d'études.

Nous tenons également à remercier les membres du jury qui nous ont honorés pour avoir examiné ce modeste travail.

Nous en profitons pour remercier tous les enseignants de l'Université de Tiaret et tous les personnels du laboratoire de génie énergétique et génie informatique en particulier « Mr. Tahri Ahmed » pour tout ce qu'ils nous ont apporté.

Imad Eddine & Youcef

# Sommaire

## INTRODUCTION GENERALE

### CHAPITRE I : LES AUTOMATES PROGRAMMABLES INDUSTRIELS ET REGULATION

<b>I.1. INTRODUCTION .....</b>	<b>4</b>
<b>I.2. SYSTEMES AUTOMATISES.....</b>	<b>4</b>
<b>I .2.1. Objectif de l'automatisation.....</b>	<b>4</b>
<b>I.2.2. Avantages et les Inconvénients de l'automatisation.....</b>	<b>4</b>
<b>I.2.3. Domaines d'application des systèmes automatisés.....</b>	<b>5</b>
<b>I .2.4. Structure d'un automatisme .....</b>	<b>5</b>
<b>I. 2.5. Mise en œuvre d'un automate.....</b>	<b>7</b>
<b>I .2.6. Caractéristiques d'un automatisme.....</b>	<b>7</b>
<b>I .3. Définition d'un API.....</b>	<b>7</b>
<b>I .3.1. Principe générale de fonctionnement d'un API.....</b>	<b>8</b>
<b>I .3.2. Architecture d'un API .....</b>	<b>8</b>
<b>I.3.2.1. Aspect extérieur .....</b>	<b>8</b>
<b>I.3.2.1.2. Modulaire .....</b>	<b>9</b>
<b>I.3.2.2. Structure interne.....</b>	<b>9</b>
<b>I.3.2.2.1. Un module d'alimentation.....</b>	<b>9</b>
<b>I.3.2.2.2. Unité de traitement ou processeur.....</b>	<b>9</b>
<b>I.3.2.2.3. Mémoire programme.....</b>	<b>9</b>
<b>I.3.2.2.4. Mémoire de données .....</b>	<b>9</b>
<b>I.3.2.2.5. Interface d'entrée .....</b>	<b>9</b>
<b>I.3.2.2.6. Une interface de sortie .....</b>	<b>10</b>
<b>I .3.3. Critères de choix de l'automate programmable industriel .....</b>	<b>10</b>
<b>I .4. PRESENTATION DE QUELQUES GAMMES SIMATIC .....</b>	<b>11</b>
<b>I .4.1. SIMATIC S7.....</b>	<b>11</b>
<b>I .4.1.1. SIMATIC S7-200 .....</b>	<b>11</b>
<b>I .4.1.2. SIMATIC S7-300 .....</b>	<b>11</b>
<b>I .4.1.3. SIMATIC S7-400 .....</b>	<b>12</b>
<b>I .4.1.4. SIMATIC S7-1200 .....</b>	<b>12</b>
<b>I .4.1.5. SIMATIC S7-1500 .....</b>	<b>13</b>
<b>I .4.2. Description de l'Automate S7-1200 .....</b>	<b>13</b>
<b>I -5. La régulation automatique .....</b>	<b>15</b>
<b>I .5.1. Définition de la régulation automatique .....</b>	<b>15</b>

## Sommaire

I.5.2. Objectif de la régulation automatique .....	16
I.5.3. Notion de Boucle ouverte/fermée.....	17
I.5.3.1. Système boucle ouverte.....	17
I.5.3.2. Système boucle fermée.....	17
I.6. DEFINITION DE LA REGULATION <i>PID</i> .....	17
I.6.1. Notion de correcteur <i>PID</i> .....	18
I.6.2. Structure des régulateurs électroniques .....	19
I.7. CONCLUSION.....	20

## CHAPITRE II : DESCRIPTION LOGICIELLE DE PROGRAMMATION : DU CONTROLE A LA SUPERVISION

II.1. INTRODUCTION.....	22
II.2. LOGICIEL DE PROGRAMMATION « <i>TIA PORTAL V16</i> » .....	22
II.2.1. Présentation du logiciel.....	22
II.3. <i>STEP 7</i> SUR <i>TIA PORTAL</i> .....	22
II.3.1. Vue du portail et vue du projet .....	22
II.3.2. Vue du portail .....	22
II.3.3. Vue du projet .....	23
II.3.4. Adressage des <i>E/S</i> .....	24
II.3.5. Les variables <i>API</i> .....	24
II.3.6. Types de variables utilisées en <i>STEP7</i> .....	25
II.3.7 Configuration de l' <i>API S7-1200</i> .....	26
II.4. <i>SIMATIC WinCC</i> COMFORT.....	32
II.5. CONFIGURER UNE VUE <i>IHM</i> .....	33
II.6. MODELISATION ET SIMULATION DU SYSTEME .....	37
II.7. CONCLUSION .....	40

## CHAPITRE III : COMPOSANTES MATERIELLES UTILISEES POUR LA CONCEPTION DU SYSTEME DE CONTROLE DU NIVEAU D'EAU

III.1. INTRODUCTION .....	42
III.2. PANNEAU DE FORMATION <i>API S7-1200 (CPU 1215C, KTP 700)</i> .....	42
III.3. LE <i>CPU 1215C DC/DC/DC</i> .....	43
III.4. ALIMENTATION RAIL <i>DIN</i> SIEMENS <i>SIMATIC PM 1207</i> .....	44
III.5. SIEMENS <i>SCALANCE XB005</i> .....	45
III.6. <i>IHM</i> SIEMENS <i>SIMATIC KTP700 PN</i> .....	48

## Sommaire

<b>III.7. Partie réalisation .....</b>	<b>49</b>
<b>III.8. Résultats obtenir.....</b>	<b>51</b>
<b>III.9. Conclusion .....</b>	<b>52</b>

## CONCLUSION GENERALE

# Liste Des Figures

## Liste Des Figures

### Chapitre I

Figure I.1: Représentation d'un automate programmable industriel.....	4
Figure I. 2: Structure d'un système automatisé.....	6
Figure I. 3 : Dialogue entre la partie commande et la partie opérative. ....	6
Figure I. 4: Fonctionnement d'un API. ....	8
Figure I. 5: Structure interne des automates.....	10
Figure I. 6: API SIMENS S7-200. ....	11
Figure I. 7: API SIMENS S7-300. ....	12
Figure I. 8: API SIMENS S7-400. ....	12
Figure I. 9: API SIMENS S7-1200. ....	13
Figure I. 10: API SIMENS S7-1500.....	13
Figure I. 11: Automate S7-1200- Siemens. ....	14
Figure I. 12: Possibilités d'extension de la CPU ....	15
Figure I. 13: Présentation d'un schéma fonctionnel d'un régulateur.....	16
Figure I. 14: Système en boucle ouverte. ....	17
Figure I. 15: Système en boucle fermée. ....	17
Figure I. 16: Schéma d'un régulateur PID mixte.....	19
Figure I. 17: Schéma d'un régulateur PID parallèle.....	19
Figure I. 18: Schéma d'un régulateur PID série.....	20

### Chapitre II

Figure II.1: Vue du portal.....	23
Figure II.2: Vue du projet.....	23
Figure II.3: Adressage des E/S.....	24
Figure II.4: Créer un projet.....	26
Figure II.5: Configurer un appareil. ....	26
Figure II.6: Ajouter un appareil.....	27
Figure II.7: Ajouter un appareil.....	27
Figure II.8: Bloc de donnée DB1. ....	28
Figure II.9: Commande d'une pompe à l'aide de la bascule SR.....	29
Figure II.10: La fonction FC1. ....	29
Figure II.11: Normalisation des signaux analogique.....	30
Figure II.12: La fonction FC2. ....	30

## Liste Des Figures

Figure II.13: Création d'un PID. ....	31
Figure II.14: Démarrage de simulation. ....	31
Figure II.15: Fenêtre du travail dans le WinCC. ....	32
Figure II.16: Configurer une vue IHM. ....	33
Figure II.17: Choisit le pupitre. ....	33
Figure II.18: Vue système en mode manuel et la pompe est éteint. ....	34
Figure II.19: Vue système en mode manuel et la pompe en marche. ....	34
Figure II.20: Vue système en mode automatique. ....	35
Figure II.21: Vue des paramètres. ....	36
Figure II.22: Vue des courbes. ....	36
Figure II.23: Système de réservoir d'eau.. ....	34
Figure II.24: Modèle Simulink du système de réservoir d'eau.. ....	35
Figure II.25: Modèle Simulink du contrôle PID appliqué au réservoir d'eau.....	36
Figure II.26: Résultats de simulation du contrôle PID appliqué au réservoir d'eau .....	36

## Chapitre III

Figure III.1: Panneau de formation API S7-1200. ....	42
Figure III.2: CPU 1215C DC/DC/DC. ....	43
Figure III.3: SIMATIC PM 1207. ....	44
Figure III.4: SCALANCE XB005. ....	46
Figure III.5: Caractéristiques techniques du SCALANCE XB005 . ....	48
Figure III.6: IHM SIEMENS SIMATIC KTP700 Basic PN. ....	48
Figure III.7: Vue du système de niveau d'eau.....	49
Figure III.8: Vue des composants utilisées.....	50
Figure III.9: Vue des courbes de PID.....	51

# Liste Des Tableaux

## Liste Des Tableaux

### Chapitre I

Tableau I.1: Notion de correcteur PID. ....	18
---	----

### Chapitre II

Tableau II.1: Types de variables utilisées en step7 . ....	25
Tableau II.2: Paramètres.....	38

# Liste Des Abréviations

## Liste Des Abréviations

API : Automate Programmable industriel

CPU : Unité centrale de l'automate (Central Processing Unit).

E/S: Entrée / Sortie.

RAM: Random Access Memory

PROFINET: Process Field Network

PROFIBUS: Process Field Bus

FM : Frequency Modulation

BO : Boucle Ouverte

BF : Boucle Fermée

GPRS: General Packet Radio Service

SB: Signal Board

CB : Communication Board

DB : Blocs de données

FB : Bloc fonctionnel

FC : Fonction

OB : Bloc d'organisation.

HMI : Human Machine Interface

IHM : Interface Homme/Machine

IP : Internet Protocol

PN/IE : Profinet/Industriel Ethernet

IP : Internet Protocol.

PC : Partie Commande.

PO : Partie Opérative.

DC : Direct Curant

## Liste Des Abréviations

DI: Digital Inputs

DO: Digital outputs

AI: Analog inputs

AO: Analog outputs

PID : Proportionnel-intégral-dérivée

PLC : Programmable Logic Controller

S7: Step7

SIMATIC: Siemens Automatic

SP: Sept Point

TOR : Tout ou Rien

TIA Portal : Totally Integrated Automation Portal

V16 : Version 16

WinCC : Windows Control Center

# **Introduction Générale**

# Introduction Générale

## Introduction générale

Les réservoirs sont inclus dans la composition de la plupart des usines, en particulier les réservoirs d'eau, car l'eau est incluse dans la composition de la plupart des matériaux fabriqués, nous avons donc travaillé pour contrôler le niveau d'eau pour garantir que le système fonctionne bien à travers une interface homme machine.

L'automatisation de notre système est cruciale car elle contribue à améliorer la productivité, la flexibilité, la qualité et les conditions de travail. Les systèmes automatisés sont rapides et précis car ils effectuent des tâches difficiles ou presque impossibles pour les humains. L'automatisation est synonyme de qualité de production, de sécurité et de précision.

Nous utilisons le programme *TIA Portal V16* pour l'automatisation et la surveillance du niveau d'eau. L'étude sera basée sur la précision absolue de l'analyse de tous les composants de notre système.

Un automate de type *S7-1200* a été utilisé et cette automatisation a été réalisée à l'aide du programme *TIA PORTAL V16* de *SIEMENS* pour refléter notre projet, considéré comme l'un des meilleurs programmes d'ingénierie développés par l'entreprise.

Les objectifs de notre projet sont de contrôler et de réguler le niveau d'eau, de l'améliorer et de l'automatiser, d'établir un système de surveillance et d'assurer la continuité de ces travaux.

Dans cette note, nous présentons trois chapitres principaux décrivant les éléments clés de notre projet de fin d'études :

Le premier chapitre est consacré aux généralités sur l'automatisation. Nous avons mentionné les objectifs et la structure du système automatisé, ainsi que le principe de fonctionnement et les différentes unités constituant l'*API*. Et la description de l'*API S7-1200*. Et la définition et l'objectif de la régulation automatique.

Dans le chapitre 2 une étude par simulation du système de contrôle du niveau d'eau incluant un algorithme *PID* est développée. Les principaux résultats du projet sont exposés. La simulation est présentée et démontrent l'efficacité de la mise en œuvre. Une partie importante du même chapitre est consacrée à la présentation du logiciel *TIA Portal V16* et comment choisir et modifier le matériel et mentionner les différentes étapes de simulation de notre programme et la modélisation du réservoir d'eau.

Le troisième et dernier chapitre, nous avons parlé du matériel que nous avons utilisé dans notre projet (Panneau de formation *API S7-1200(CPU1215C, KTP 700)*).

**Chapitre I :**

**Les Automates Programmables Industriels et  
régulation**

### I.1. Introduction

Les processus d'automatisation créent des opérations qui requièrent une intervention humaine pour automatiser les opérations. L'API vise à remplacer ce travail par des tâches simples, générales et pouvant être répétées nécessitant de la précision. Nous sommes passés d'un système manuel à un système automatisé utilisant des techniques électromécaniques, électroniques, pneumatiques et hydrauliques. L'automatisation est disponible dans tous les domaines (fabrication, alimentation, transport, automobile ...etc.).

### I.2. Systèmes automatisés

Un système automatisé est un ensemble d'éléments en interaction organisé dans un but précis : agir sur une matière d'œuvre pour lui donner une valeur ajoutée. Le système automatisé est soumis à des contraintes : énergétiques, de configuration, dérèglement et d'exploitation qui interviennent dans tous les modes de marche et d'arrêt du système [1].

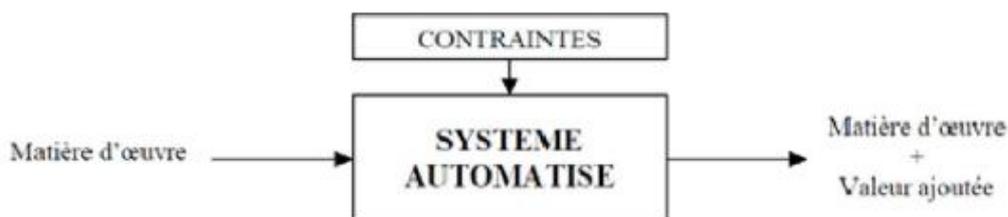


Figure I.1: Représentation d'un automate programmable industriel.

#### I.2.1. Objectif de l'automatisation

L'objectif de l'automatisation des systèmes est de garantir, en ayant recours le moins possible à l'homme, des produits de qualité et ce pour un coût le plus faible possible avec une augmentation de la sécurité tout en s'adaptant à des contextes particuliers. En effet l'automatisation des processus permet de :

- Accroître la productivité de l'entreprise en réduisant les coûts de production,
- Améliorer la flexibilité de la production,
- Perfectionner la qualité du produit,
- S'adapter à des contextes particuliers,
- Augmenter la sécurité.

#### I.2.2. Avantages et les Inconvénients de l'automatisation

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

- La capacité de production accélérée,
- L'aptitude à convenir à tous les milieux de production,
- La souplesse d'utilisation,
- La création de postes d'automaticiens,

Par contre, l'automatisation des processus possède certains inconvénients dont on citer entre autres :

- Le coût élevé du matériel, principalement avec les systèmes hydrauliques,
- La maintenance doit être structurée,
- La suppression d'emplois.

### I.2.3. Domaines d'application des systèmes automatisés

Aujourd'hui, il est difficile de concevoir un système de production sans avoir recours aux différentes technologies et composants qui forment les systèmes automatisés, à savoir dans les domaines d'automobile, aviation, industrie, médical et de transport.

### I.2.4. Structure d'un automatisme

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

- La partie commande ou automate qui élabore les ordres nécessaires à l'exécution du processus, en fonction des consignes qu'elle reçoit à l'entrée et des comptes rendus d'exécution qui lui sont fournis par la partie opérative.
- La partie opérative qui effectue les opérations en exécutant les ordres qui lui sont donnés par la partie commande.
- La partie relation ou dialogue qui représente l'échange des informations entre la partie commande et l'opérateur (pilote, usage, surveillant...) dont elle reçoit des consignes en lui donnant des comptes rendus visuels ou sonores [2].

## Chapitre I : les Automates Programmables Industriels et régulation

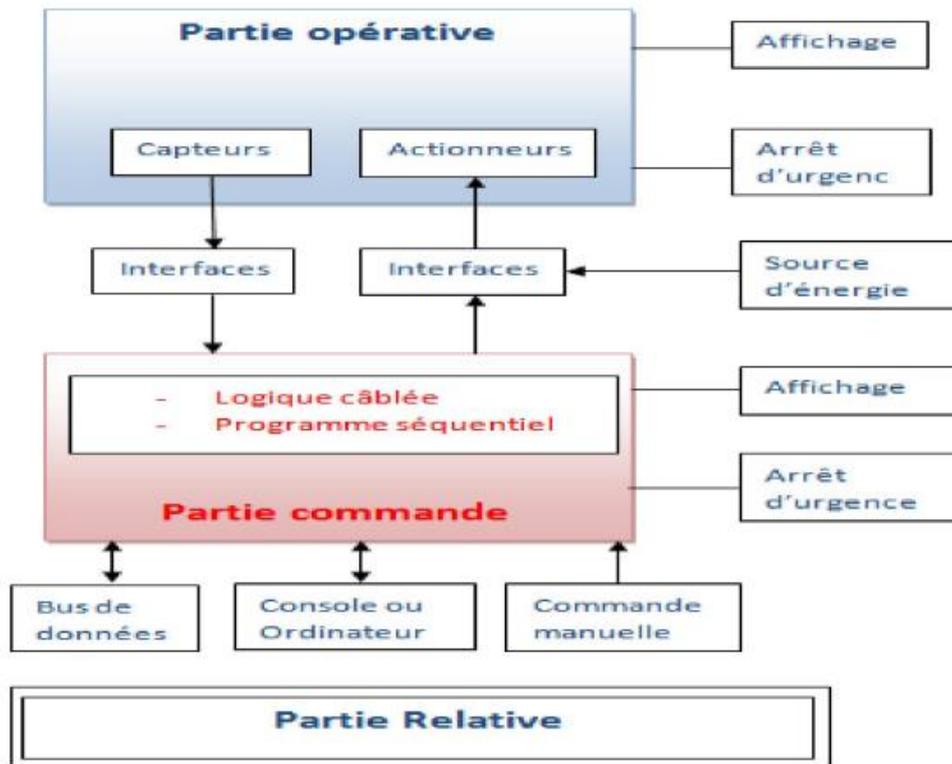


Figure I.2: Structure d'un système automatisé.

Cette structure permet un dialogue profitable entre le futur utilisateur du système et l'automaticien responsable de la partie commande.

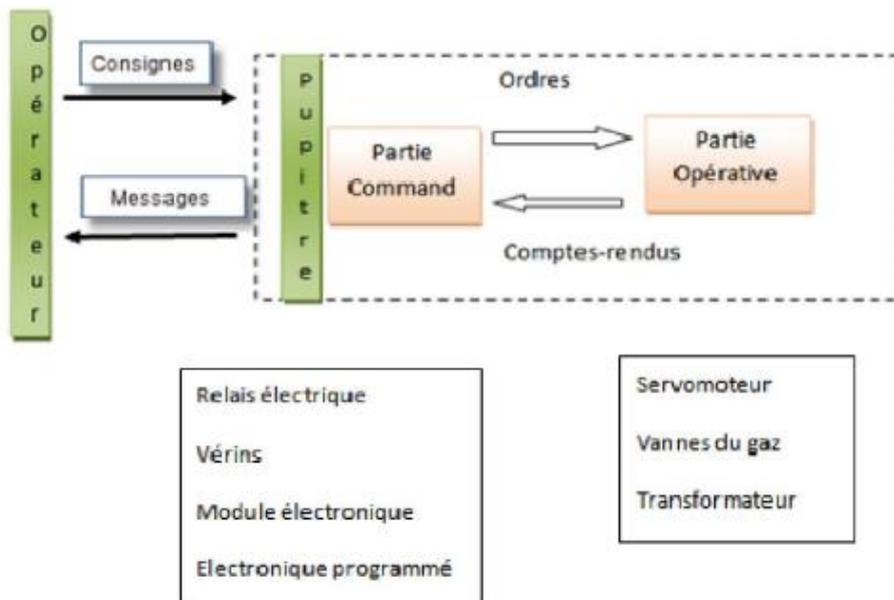


Figure I.3: Dialogue entre la partie commande et la partie opérative.

### I. 2.5. Mise en œuvre d'un automate

La mise en œuvre de tout système automatisé implique une série de tâches qui constituent autant d'étapes successives naturellement interdépendantes [2] :

- L'étude préalable,
- L'étude proprement dite et préparation,
- Fabrication et essais,
- Mise en route et exploitation.

### I. 2.6. Caractéristiques d'un automatisme

Les automatismes sont des dispositifs qui permettent à des machines ou des installations de fonctionner automatiquement [2].

Cependant, un automatisme bien conçu, simplifie considérablement le travail de l'homme qui, libéré vis-à-vis de la machine, peut se consacrer à des activités plus nobles permettant de :

- Réduire les tâches complexes, pénibles ou indésirables en les faisant exécuter par la machine,
- Faciliter les changements de fabrication en passant d'une quantité ou d'un type de production à un autre,
- Améliorer la qualité des produits en asservissant la machine à des critères de fabrication et à des tolérances qui seront respectées dans le temps,
- Accroître la production ainsi que la productivité,
- Réaliser des économies de matière et d'énergie,
- Augmenter la sécurité du personnel,
- Contrôler et protéger les installations et les machines.

### I. 3. Définition d'un API

Un automate programmable industriel (API) est un appareil électronique spécialisée dans la conduite et la surveillance en temps réel de processus industriels et tertiaires. Il exécute une suite d'instructions introduites dans sa mémoire sous forme de programme et s'apparente par conséquent aux machines de traitement d'information.

Trois caractéristiques fondamentales le distinguent totalement des outils d'informatiques tels que les ordinateurs utilisés dans les entreprises et tertiaires :

- Connexion directe aux différents capteurs et actionneurs grâce à ces entrées/sortie ;
- Fonctionnement dans des conditions industrielles sévères (température, vibrations, humidité, microcoupure de l'alimentation en énergie électrique... ) ;
- Son aspect pratique grâce à la possibilité de sa programmation en utilisant un langage spécialement développé pour le traitement de fonctions d'Automate (*Step7*) [3].

### I.3.1. Principe générale de fonctionnement d'un API

Tous les automates fonctionnent selon le même mode opératoire :

- **Traitement interne** : l'automate effectue des opérations de contrôles et met à jour certaines paramètres systèmes (détection des passages en *RUN/STOP*, mises à jour des valeurs de l'horodateur,);
- **Lecture des entrées** : l'automate lit les entrées (de façon synchrone) et les recopies dans la mémoire image des entrées ;
- **Exécution du programme** : l'automate exécute le programme instruction par instruction et écrit les sorties dans la mémoire image des sorties ;
- **Ecriture des sorties** : l'automate bascule les différentes sorties (de façon synchrone) aux positions définies dans la mémoire image des sorties.

Ces quatre opérations sont effectuées continuellement par l'automate (fonctionnement cyclique)[4].

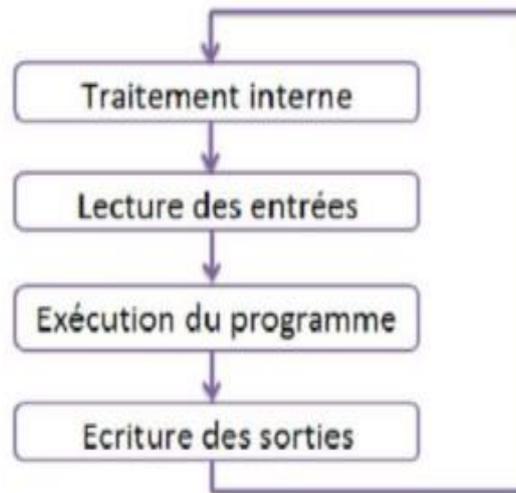


Figure I.4: Fonctionnement d'un API.

### I.3.2. Architecture d'un API

#### I.3.2.1. Aspect extérieur

Les automates peuvent être de type compact ou modulaire. Pour la configuration compacte, il intègre le processeur, l'alimentation et les entrées/sortie. Il peut réaliser certaines fonctions supplémentaires et recevoir des extensions limitées. Il est généralement destiné à la commande de petits automatismes.

### I.3.2.1.2. Modulaire

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

### I.3.2.2. Structure interne

Un automate programmable industriel est donc constitué des éléments suivants :

#### I.3.2.2.1. Un module d'alimentation

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

#### I.3.2.2.2. Unité de traitement ou processeur

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

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

#### I.3.2.2.3. Mémoire programme

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

#### I.3.2.2.4. Mémoire de données

La mémoire de donnée permet le stockage de :

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

#### I.3.2.2.5. Interface d'entrée

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

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

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

### I.3.2.2.6. Une interface de sortie

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

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

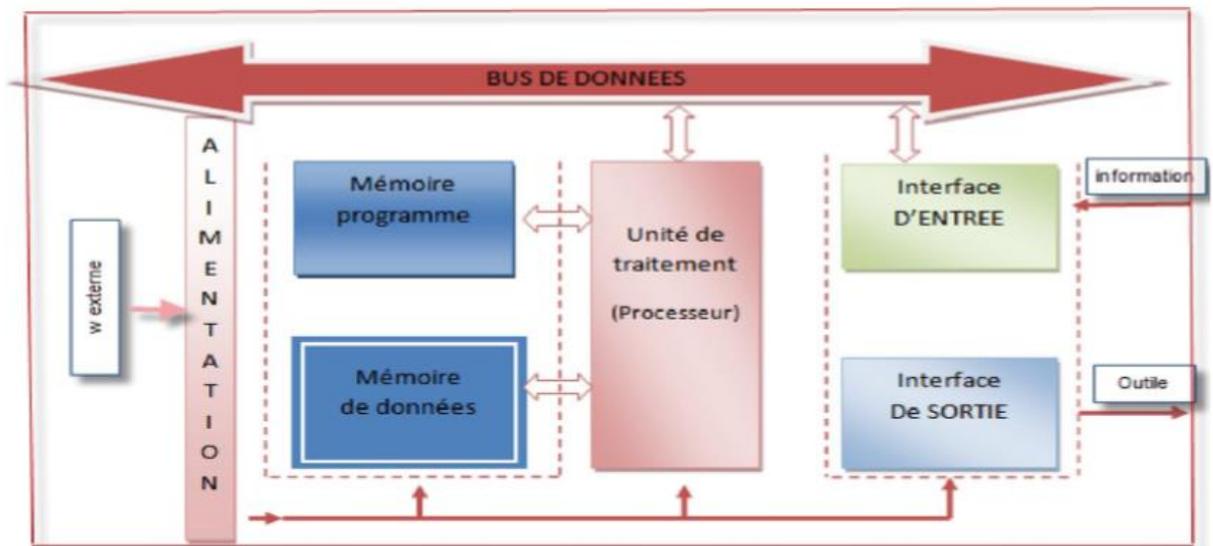


Figure I.5: Structure interne des automates.

### I.3.3. Critères de choix de l'automate programmable industriel

Après avoir étudié notre système dans les chapitres précédents, le choix *des API* repose sur des considérations tels que :

- Le nombre et la nature des entrées/sorties ;
- Le type du processeur, la taille de la mémoire, la vitesse de traitement et les fonctions spéciales offertes par le processeur ;
- Processeur en calcul afin de sécuriser le traitement et la communication avec le procédé ;
- Communication avec d'autre système ;
- La fiabilité et la robustesse ;
- Protection contre les parasites (champs électromagnétiques), baisse et pic de tension.
- Pour notre travail, nous avons choisi à l'Automate *SIEMENS S7-1200* et cela vue
- Le nombre d'entrées/sorties (14 /10) ;
- La nature Analogiques d'entrées/sorties [5].

## I .4. Présentation de quelques gammes *SIMATIC*

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

### I .4.1. *SIMATIC S7*

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

#### I .4.1.1. *SIMATIC S7-200*

La famille *S7-200* est constituée de micro-automates programmables utilisables dans des applications d'automatisations variées. La Figure ci-dessous présente un micro automate *S7-200*. Son dessin compact, ses possibilités d'expansion, son faible prix et son important jeu d'opérations en font une solution idéale pour la commande de petites applications. En outre, le large choix de tailles et de tensions de *CPU* offre la souplesse nécessaire pour résoudre un problème d'automatisation.

Un automate programmable *S7-200* consiste en un *CPU S7-200* seul ou complété de divers modules d'extension facultatifs connectés à cette dernière à l'aide d'un connecteur de bus fourni avec ce module d'extension [6].



Figure I.6: API SIMENS S7-200.

#### I .4.1.2. *SIMATIC S7-300*

La famille *S7-300* est constituée d'automates programmables de conception modulaire utilisés pour des automatismes de gamme, et peuvent être connectés entre eux au moyen d'un câble-bus *PROFIBUS*. Un automate *S7-300* consiste en un *CPU*, un module d'alimentation *PS*, un module de signaux *FM*, un module de signaux *SM* et un processeur de communication, comme indiqué dans la figure suivante [6].



Figure I.7: API SIMENS S7-300.

### I .4.1.3. SIMATIC S7-400

La famille *S7-400* est aussi constituée d'automates programmables de conception modulaire. Pratiquement chaque tâche d'automatisation peut être résolue par un choix approprié des constituants de *S7-400* et avec la possibilité d'expansion de plusieurs modules.

Les modules se présentent sous forme de boîtiers que l'on adapte sur un châssis [6].

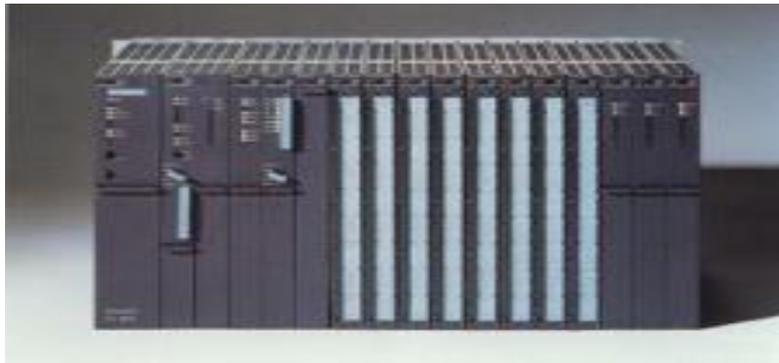


Figure I.8: API SIMENS S7-400.

### I .4.1.4. SIMATIC S7-1200

L'automate *SIMATIC S7-1200* est modulaire et compact, polyvalent et constitue une solution parfaitement adaptée à une grande variété d'applications. Une conception modulaire et flexible, une interface de communication répondant aux exigences les plus sévères dans l'industrie et une large gamme de fonctions technologiques performantes et intégrées faisant de lui une solution d'automatisation complète [7].



Figure I.9: API SIMENS S7-1200.

### I .4.1.5. SIMATIC S7-1500

Le *SIMATIC S7-1500* est le système de commande modulaire pour une multitude d'applications dans le secteur de l'automatisation discrète. Il dispose d'une vaste gamme de modules combinables individuellement [8].



Figure I.10: API SIMENS S7-1500.

### I .4.2. Description de l'Automate S7-1200

Le contrôleur *S7-1200* offre la souplesse et la puissance nécessaires pour commander une large gamme d'appareils afin de répondre aux besoins en matière d'automatisation. Sa forme compacte, sa configuration souple et son important jeu d'instructions en font une solution idéale pour la commande d'applications très variées [9].

Le *CPU* combine un microprocesseur, une alimentation intégrée, des circuits d'entrée et de sortie, un *PROFINET* intégré, des *E/S* rapides de commande de mouvement, ainsi que des entrées analogiques intégrées dans un boîtier compact en vue de créer un contrôleur puissant. Une fois le programme chargé, le *CPU* contient la logique nécessaire au contrôle et à la commande des appareils dans votre application.

Le *CPU* surveille les entrées et modifie les sorties conformément à la logique de votre programme utilisateur, qui peut contenir des instructions booléennes, des instructions de comptage, des

## Chapitre I : les Automates Programmables Industriels et régulation

instructions de temporisation, des instructions mathématiques complexes ainsi que des commandes pour communiquer avec d'autres appareils intelligents [10].

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

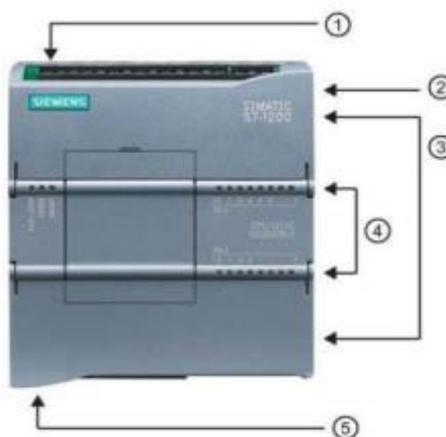


Figure I.11: Automate S7-1200- Siemens.

1. Connecteur d'alimentation.
2. Logement pour carte mémoire sous le volet supérieur.
3. Connecteurs amovibles pour le câblage utilisateur (derrière les volets).
4. *DEL* d'état pour les *E/S* intégrées.
5. Connecteur *PROFINET* (sur la face inférieure de la *CPU*).

### ▪ Choix de la *CPU*

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

Des modules *E/S* supplémentaires peuvent être ajoutés du côté droit de la *CPU* pour étendre la capacité *E/S*, *TOR* ou analogiques [11].

### ▪ Le choix des modules d'Entrées/Sorties

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

- Le type et la valeur de la tension d'entrée ou de sortie.

## Chapitre I : les Automates Programmables Industriels et régulation

- Le nombre de voies.
- Le type d'entrée ou de sortie (sonde, thermocouple, électrovanne...).
- **Possibilités d'extension du CPU**

La gamme *S7-1200* offre divers modules et cartes enfichables pour accroître les capacités du *CPU* avec des *E/S* supplémentaires ou d'autres protocoles de communication [11].

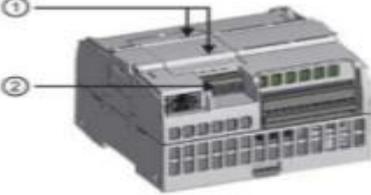
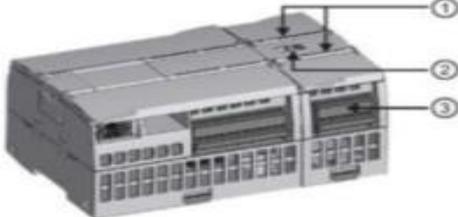
Type de module	Description
<p>La CPU prend en charge une carte d'extension enfichable :</p> <ul style="list-style-type: none"> <li>• Un Signal Board (SB) fournit des E/S supplémentaires pour votre CPU. Le SB se raccorde à l'avant de la CPU.</li> <li>• Un Communication Board (CB) vous permet d'ajouter un autre port de communication à votre CPU.</li> <li>• Un Battery Board (BB) permet une sauvegarde à long terme de l'horloge temps réel.</li> </ul>	 <p>① DEL d'état sur le Signal Board</p> <p>② Connecteur amovible pour le câblage utilisateur</p>
<p>Les modules d'entrées-sorties (SM) permettent d'ajouter des fonctionnalités à la CPU. Les SM se raccordent sur le côté droit de la CPU.</p> <ul style="list-style-type: none"> <li>• E/S TOR</li> <li>• E/S analogiques</li> <li>• RTD et Thermocouple</li> <li>• SM 1278 IO-Link maître</li> <li>• SM 1238 Energy Meter</li> </ul>	 <p>① DEL d'état</p> <p>② Languette coulissante du connecteur de bus</p> <p>③ Connecteur amovible pour le câblage utilisateur</p>
<p>Les modules de communication (CM) et les processeurs de communication (CP) ajoutent des options de communication à la CPU, telles que la connectivité PROFIBUS ou RS232/RS485 (pour PtP, Modbus ou USS) ou le maître AS-I.</p> <p>Un CP offre la possibilité d'autres types de communication, par exemple la connexion de la CPU par un réseau GPRS, LTE, IEC, DNP3 ou WDC.</p> <ul style="list-style-type: none"> <li>• La CPU accepte jusqu'à trois CM ou CP.</li> <li>• Chaque CM ou CP se raccorde sur le côté gauche de la CPU (ou sur le côté gauche d'un autre CM ou CP).</li> </ul>	 <p>① DEL d'état</p> <p>② Connecteur de communication</p>

Figure I.12: Possibilités d'extension de la CPU [12].

### I -5. La régulation automatique

#### I .5.1. 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 physiques) 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 [1][13].

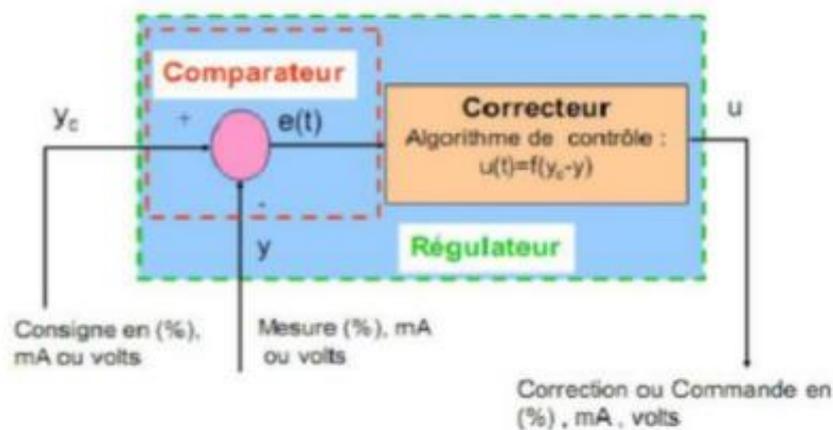


Figure I.13: Présentation d'un schéma fonctionnel d'un régulateur.

### I.5.2. Objectif de la régulation automatique

L'objectif d'une régulation automatique d'un procédé est de le maintenir le plus près possible de son fonctionnement optimum, qui a été prédéfini ultérieurement par un cahier des charges (conditions ou performances imposées).

Aussi il faut impérativement prendre en compte les aspects de protection du personnel, des installations techniques et le respect de l'environnement. Le cahier des charges définit les critères qualitatifs à imposer qui sont traduits le plus souvent par des critères quantitatifs, comme par exemple, la stabilité, la précision et la rapidité [14].

### I.5.3. Notion de Boucle ouverte/fermée

#### I.5.3.1. Système boucle ouverte

Pour une variation manuelle d'amplitude finie de la commande  $u$ , on a une variation de la grandeur à maîtriser ou à réguler.

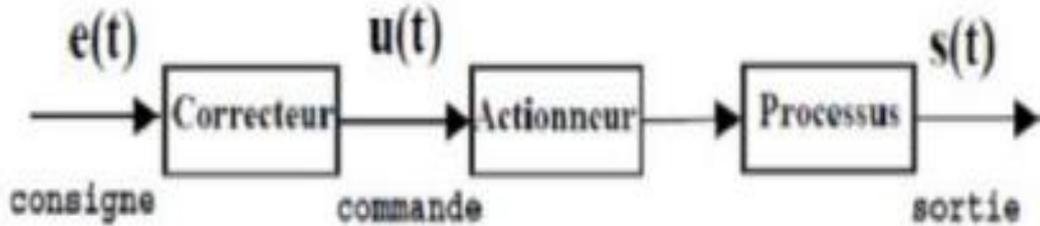


Figure I.14: Système en boucle ouverte.

#### I.5.3.2. Système boucle fermée

La grandeur réglant exerce une influence sur la grandeur réglée pour la maintenir dans des limites définies malgré les perturbations.

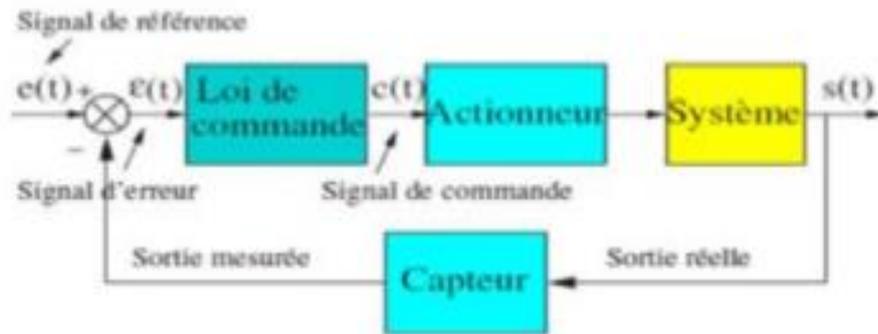


Figure I.15: Système en boucle fermée.

### I.6. Définition de la régulation *PID*

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

Si la dynamique dominante du système est supérieure à un deuxième ordre, ou si le système contient un retard important ou plusieurs modes oscillants, le régulateur *PID* n'est plus adéquat et un régulateur

## Chapitre I : les Automates Programmables Industriels et régulation

plus complexe (avec plus de paramètres) doit être utilisé, au dépend de la sensibilité aux variations des paramètres du procédé [1].

- La loi de commande du régulateur **PID** :

$$u(t) = K_p * \left( e(t) + \frac{1}{T_i} * \int_{-\infty}^t e(\tau). d\tau + T_d \cdot \frac{de}{dt} \right) \quad (I.1)$$

- Fonction de transfert du régulateur **PID** :

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

### I .6.1. 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	Points forts	Points faibles
<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 sollicitations de l'organe de commande

Tableau I.1: Notion de correcteur **PID**.

$P = K_R$  : est l'action proportionnelle. Sur la plupart des régulateurs, on règle la bande proportionnelle au lieu de régler le gain du régulateur.

$I=1/T_i$  : (min-1 en général) : est l'action intégrale.

$D=T_d$  : (s en général) : est l'action dérivée.

### I.6.2. Structure des régulateurs électroniques

Le régulateur proportionnel intégral dérivé *PID* combine les trois actions décrites précédemment ; il tire parti du présent (Proportionnel), du passé (intégrale) et du futur (dérivée).

$$u(t) = K_p \left( e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{d}{dt} e(t) \right) \quad (I.3)$$

La synthèse du régulateur *PID* se résume aux choix des constantes  $K_p > 0$ ,  $T_i > 0$  et  $T_d > 0$  afin de répondre aux exigences. Son schéma fonctionnel apparait dans la figure ci-après.

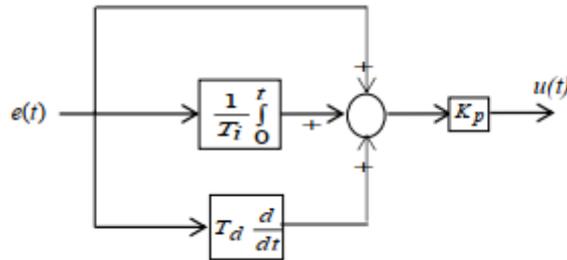


Figure I.16: Schéma d'un régulateur PID mixte.

Cette forme de régulateur *PID* est appelée structure mixte.

Avec  $K_i = K_p/T_i$  et  $K_d = K_p * T_d$ . Le régulateur *PID* présente la structure parallèle

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t) \quad (I.4)$$

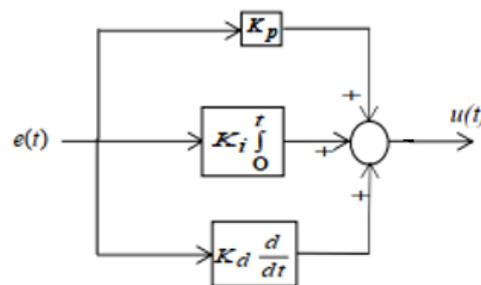


Figure I.17: Schéma d'un régulateur PID parallèle.

La loi de commande s'écrit ainsi :

$$\frac{S(s)}{\varepsilon(s)} = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right) \quad (I.5)$$

Il existe une troisième configuration du régulateur *PID* dite structure série (Figure I.17)

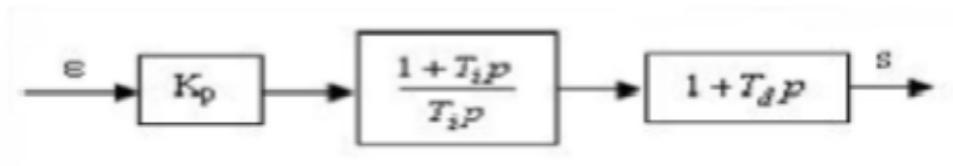


Figure I.18: Schéma d'un régulateur PID série.

Loi de commande s'écrit de cette configuration s'écrit de la manière suivante :

$$\frac{S(s)}{\varepsilon(s)} = K_p * \left(1 + \frac{1}{T_i s}\right) * (1 + T_d s) \quad (I.6)$$

### I.7.Conclusion

Nous avons présenté dans ce chapitre, l'étude des différentes composantes de l'automate programmable industriel de la gamme *S7-1200* ainsi que leurs caractéristiques. Ceci nous a permis d'évaluer leurs importances et de constater leur rôle primordial dans la gestion des processus industriels. Nous avons également mis l'accent sur l'importance de la régulation par utilisation des régulateurs *PID*.

## **Chapitre II :**

# **Description Logicielle de programmation : du contrôle à la supervision**

### II.1. Introduction

Pour identifier notre système, le programme réalise des solutions d'automatisation en concevant et en automatisant *SIEMENS TIA PORTAL*, qui comprend le système intégré de *STEP7*, *SIMATIC WINCC* et *PLC SIM*.

Dans ce chapitre, nous allons montrer les résultats de la programmation automatisée de notre système et les simulations utilisées dans ce projet à travers l'analyse fonctionnelle et la supervision.

### II.2. Logiciel de programmation « *TIA Portal V16* »

#### II.2.1. Présentation du logiciel

La plate-forme *TIA (Totally Integrated Automation) Portal* est la dernière évolution des logiciels de travail Siemens qui permet de mettre en œuvre des solutions d'automatisation avec un système d'ingénierie intégré, dans un seul logiciel cette plateforme regroupe la programmation des différents dispositifs d'une installation. On peut donc programmer et configurer, en plus de l'automate, les dispositifs *HMI* les variateurs... etc [15].

#### II.3. *STEP 7* sur *TIA portal*

*SIMATIC STEP 7 Basic (TIA Portal)* est une version économique et allégée du logiciel pour contrôleur *STEP 7 Professional Controller Software* dans le *TIA Portal*, pouvant être utilisé à la fois pour l'ingénierie des microcontrôleurs *SIMATIC S7-1200* et la configuration des *SIMATIC HMI Basic Panels*, étant donné que *WinCC Basic* fait partie intégrante de l'ensemble des logiciels.

##### II.3.1. Vue du portail et vue du projet

Lorsque l'on lance *TIA Portal*, l'environnement de travail se décompose en deux types de vue [16]:

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

##### II.3.2. Vue du portail

Chaque portail permet de traiter une catégorie de tâche (actions).

La fenêtre affiche la liste des actions pouvant être réalisées pour la tâche sélectionnée

## Chapitre II : Description Logicielle de programmation : du contrôle à la supervision

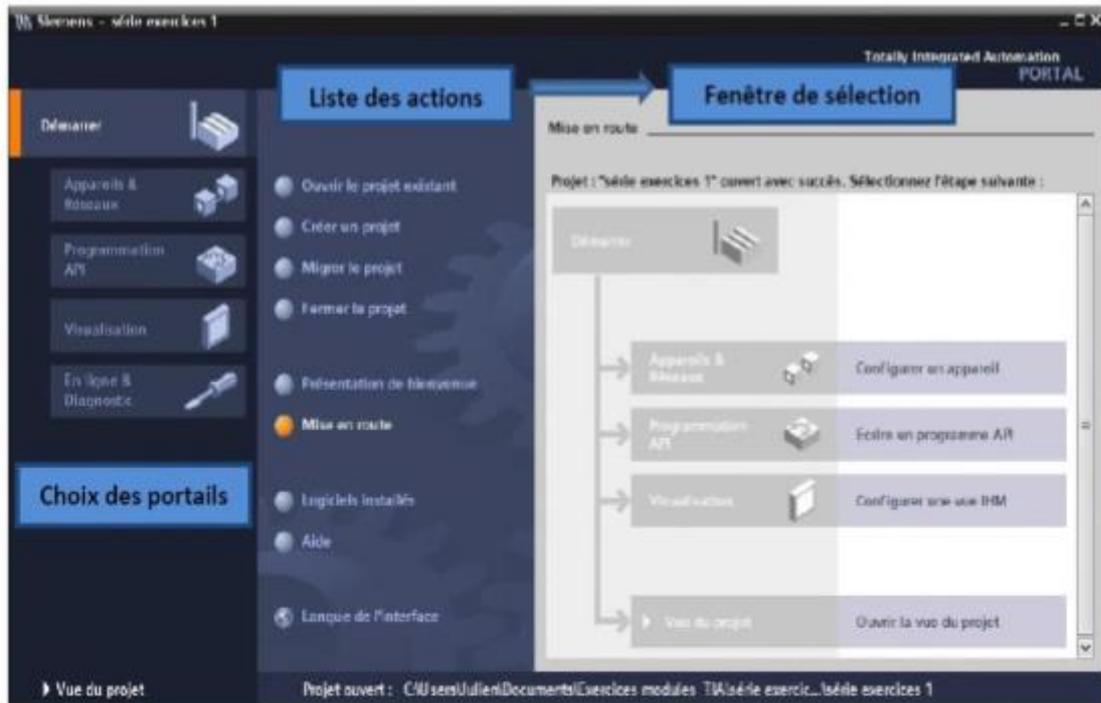


Figure II.1: Vue du portail.

### II.3.3. Vue du projet

L'élément « Projet » contient l'ensemble des éléments et des données nécessaires pour mettre en œuvre la solution d'automatisation souhaitée.

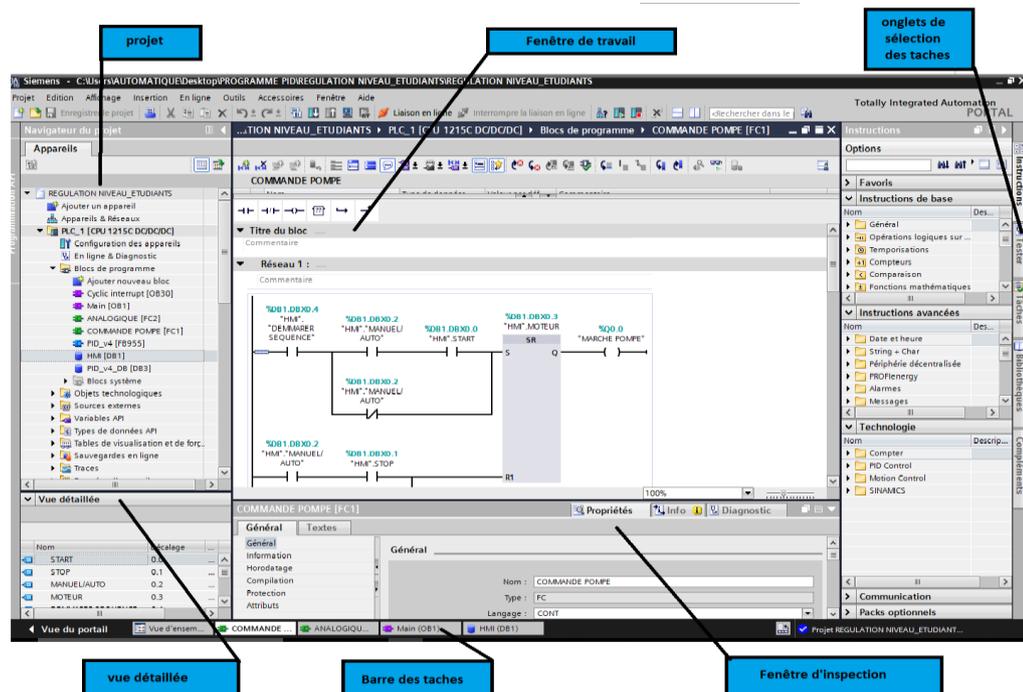


Figure II.2: Vue du projet.

## Chapitre II : Description Logicielle de programmation : du contrôle à la supervision

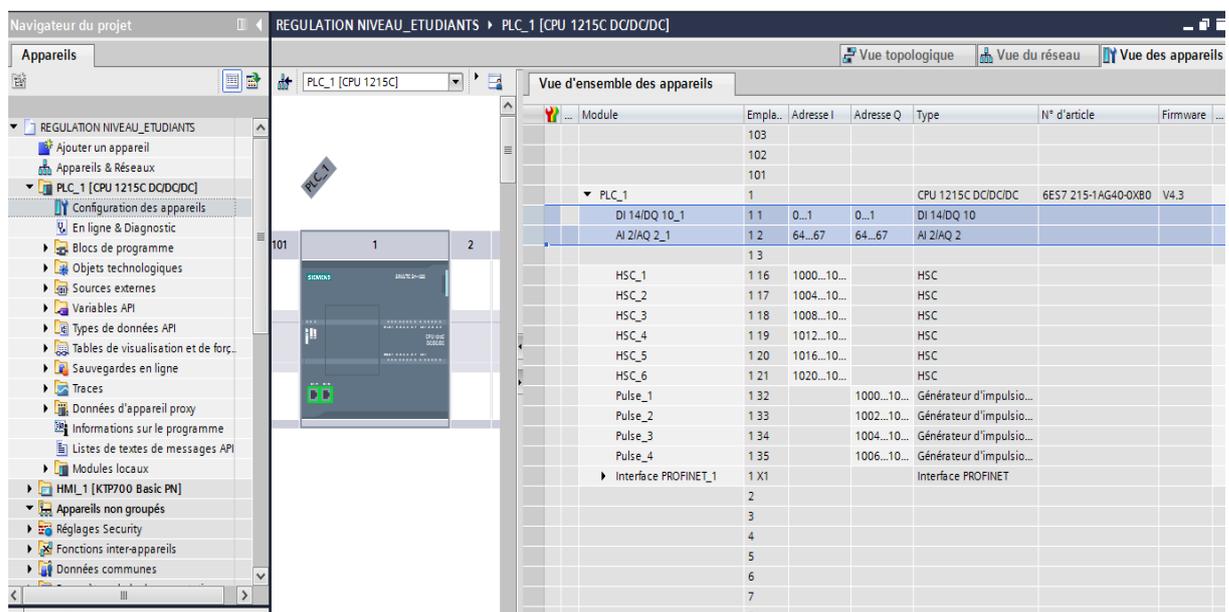
- **La fenêtre de travail** permet de visualiser les objets sélectionnés dans le projet pour être traités. Il peut s'agir des composants matériels, des blocs de programme, des tables des variables, des IHM, ...
- **La fenêtre d'inspection** permet de visualiser des informations complémentaires sur un objet sélectionné ou sur les actions en cours d'exécution (propriété du matériel sélectionné, messages d'erreurs lors de la compilation des blocs de programme, ...).
- **Les onglets de sélection de tâches** sont un contenu qui varie en fonction de l'objet sélectionné (configuration matérielle, bibliothèques des composants, bloc de programme, instructions de programmation).

Cet environnement de travail contient énormément de données. Il est possible de masquer ou réduire certaines de ces fenêtres lorsque l'on ne les utilise pas.

Il est également possible de redimensionner, réorganiser, désancrer les différentes fenêtres.

### II.3.4. Adressage des E/S

Pour connaître l'adressage des entrées et sorties présentes dans la configuration matérielle, il faut aller dans « appareil et réseau » dans le navigateur du projet. Dans la fenêtre de travail, on doit s'assurer d'être dans l'onglet « Vue des appareils » et de sélectionner l'appareil voulu [17].



Module	Empla..	Adresse I	Adresse Q	Type	N° d'article	Firmware
	103					
	102					
	101					
PLC_1	1			CPU 1215C DC/DC/DC	6ES7 215-1AG40-0XB0	V4.3
DI 14/DQ 10_1	1 1	0..1	0..1	DI 14/DQ 10		
AI 2/AQ 2_1	1 2	64..67	64..67	AI 2/AQ 2		
	1 3					
HSC_1	1 16	1000..10...		HSC		
HSC_2	1 17	1004..10...		HSC		
HSC_3	1 18	1008..10...		HSC		
HSC_4	1 19	1012..10...		HSC		
HSC_5	1 20	1016..10...		HSC		
HSC_6	1 21	1020..10...		HSC		
Pulse_1	1 32	1000..10...		Générateur d'impulsio...		
Pulse_2	1 33	1002..10...		Générateur d'impulsio...		
Pulse_3	1 34	1004..10...		Générateur d'impulsio...		
Pulse_4	1 35	1006..10...		Générateur d'impulsio...		
Interface PROFINET_1	1 X1			Interface PROFINET		
	2					
	3					
	4					
	5					
	6					
	7					
	8					

Figure II.3: Adressage des E/S.

### II.3.5. Les variables API

Adresses symbolique et absolue :

Dans *TIA Portal*, toutes les variables globales (entrées, sorties, mémentos...) possèdent une **Adresse symbolique** et une **adresse absolue**.

## Chapitre II : Description Logicielle de programmation : du contrôle à la supervision

1. **L'adresse absolue** représente l'identificateur d'opérande (I, Q, M, ...) et son adresse et numéro de bit.
2. **L'adresse symbolique** correspond au nom que l'utilisateur a donné à la variable (ex : Bouton Marche). Le lien entre les adresses symbolique et absolue se fait dans **la table des variables API**.

Lors de la programmation, on peut choisir d'afficher les adresses absolues, symboliques ou encore les deux simultanément.

### II.3.6. Types de variables utilisées en STEP7

Il existe plusieurs types de variables distinctes dans le *step7*, ces variables sont déclarées préalablement avant chaque début de programme. Le tableau suivant réunit ces différentes variables :

TYPE	TAILE
Bool	1 bit
Real	32 bits
Int	16 bits
Byte	8 bits
Word	16 bits
DWord	32 bits
DInt	32 bits
Char	8 bits
Time	32 bits
S5Time	16 bits
Date	16 bits
Time of Day	32 bits

Tableau II.1: Types de variables utilisées en step7 [18].

### II.3.7 Configuration de l'API S7-1200

Les étapes ci-dessous montrent comment créer un projet pour *SIMATIC S7-1200*

1. Les programmes pour *SIMATIC S7-1200* sont gérés sous forme de projets. Nous allons maintenant créer un nouveau projet via la vue portail (« Créer un projet > Nom : Régulation\_Niveau > Créer »).

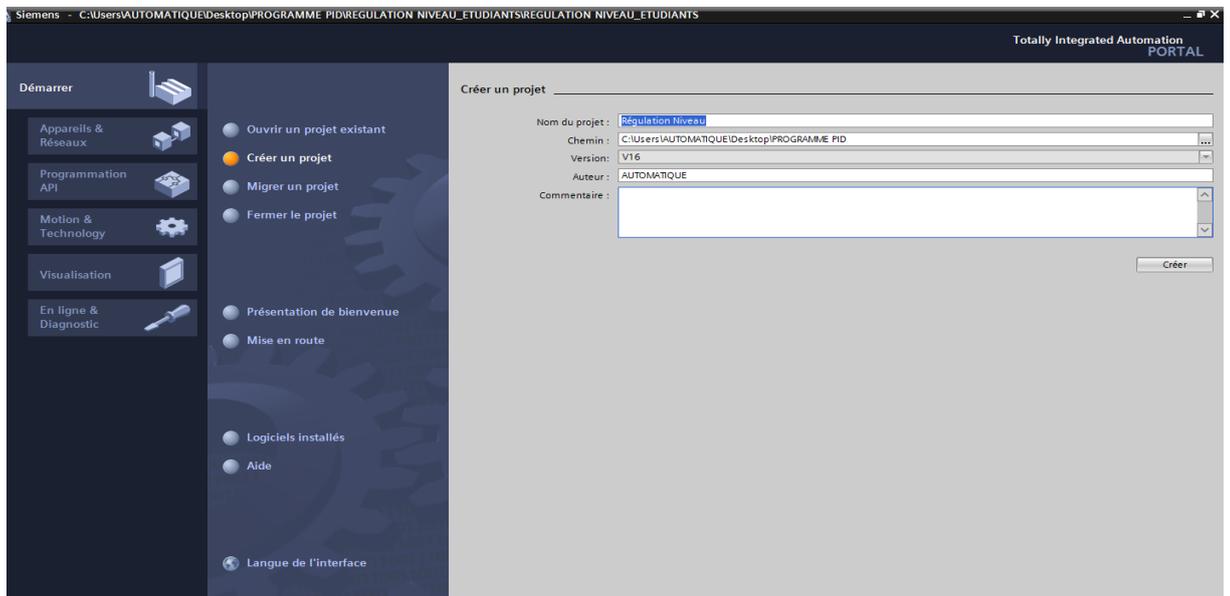


Figure II.4: Créer un projet.

2. « Mise en route » est recommandée pour le début de la création du projet.

Premièrement, nous voulons « Configurer un appareil » (« Mise en route > Configurer un appareil »).

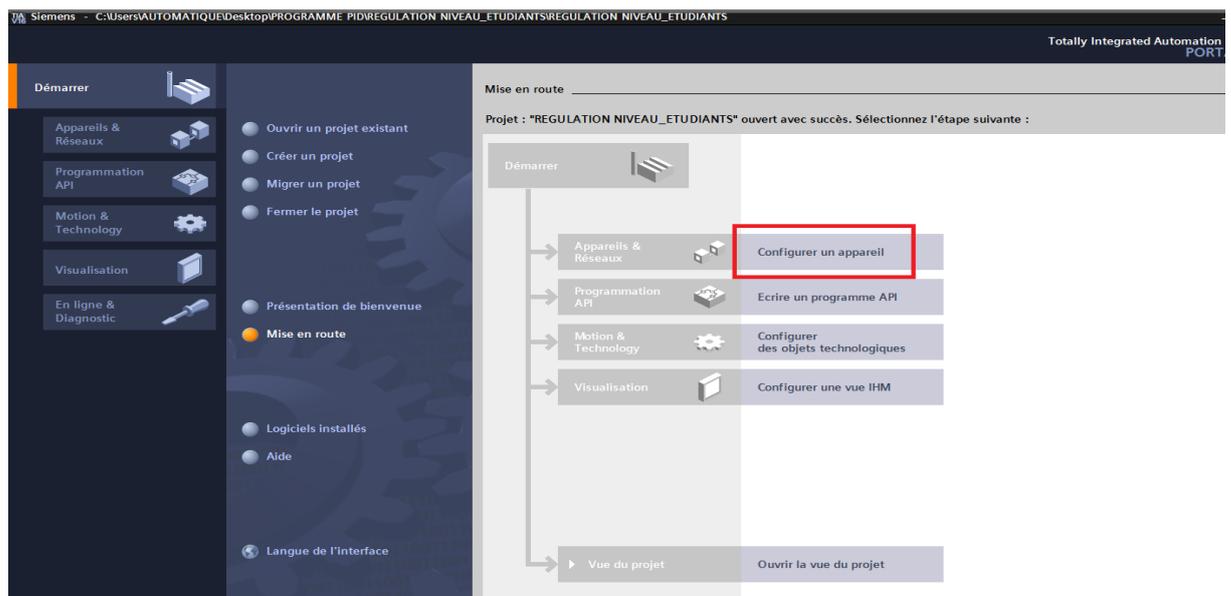


Figure II.5: Configurer un appareil.

## Chapitre II : Description Logicielle de programmation : du contrôle à la supervision

3. Puis « Ajouter un appareil » avec le nom d'appareil : Régulation\_ Niveau. Choisissez alors dans le catalogue la « CPU 1215C DC/DC/DC » avec la bonne combinaison de lettres derrière. « Ajouter un appareil > SIMATIC PLC > CPU 1215C DC/DC/DC > 6ES7 215-1AG40-0XB0 >Ajouter »).

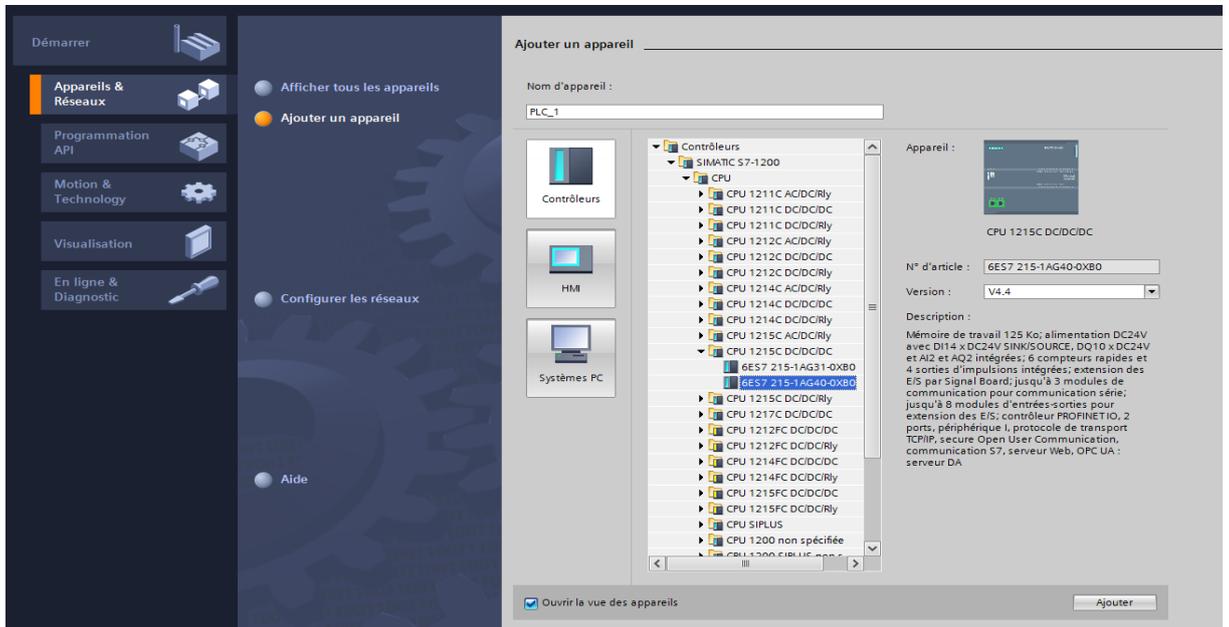


Figure II.6: Ajouter un appareil.

4. Afin que le logiciel puisse accéder dans la suite à la bonne CPU, son adresse IP et le masque de sous-réseau doivent être paramétrés (« Propriétés > Général > Interface PROFINET > Adresses Ethernet > Adresse IP : 192.168.0.1 et Masq. S/rés. : 255.255.255.0 »).

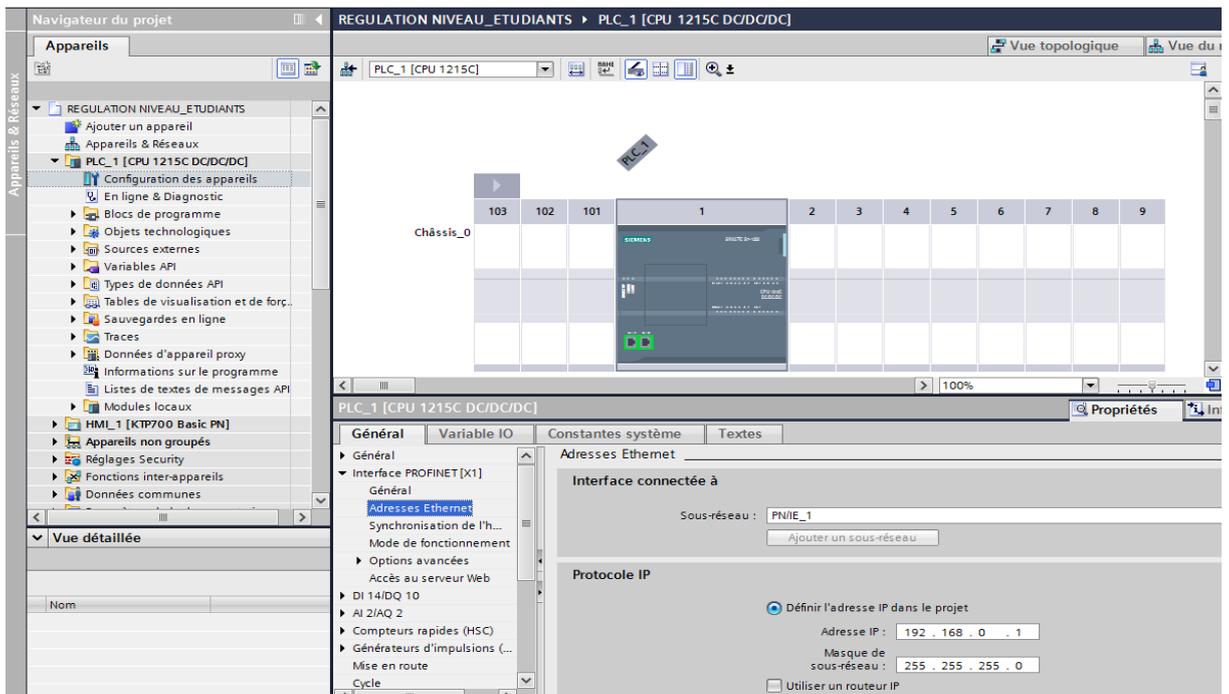


Figure II.7: Ajouter un appareil.

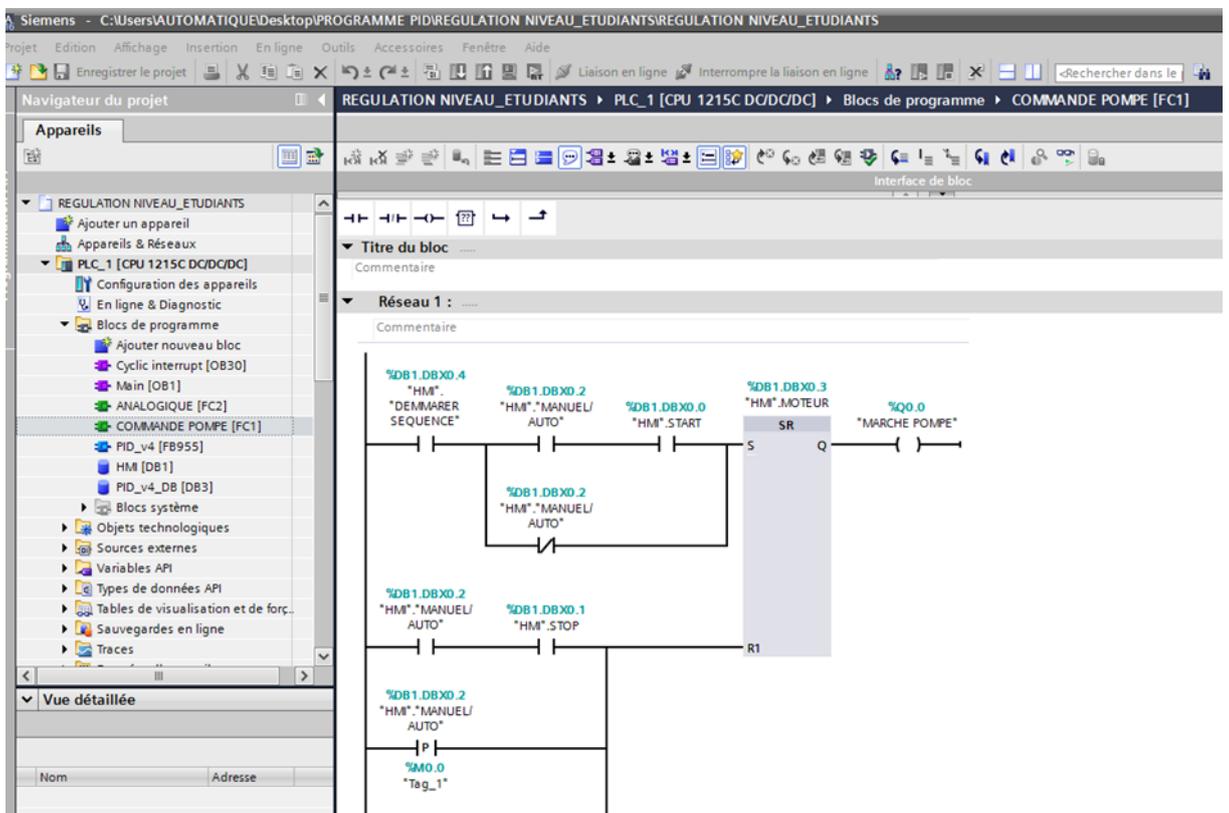
## Chapitre II : Description Logicielle de programmation : du contrôle à la supervision

5. Création d'un bloc de données *DB* contient les paramètres effectifs et les données statiques de notre programme. La figure suivante représente un bloc de données *DB1* :

Nom	Type de données	Décalage	Valeur de départ	Rémanence	Accessible...	Écritu...	Visible da...	Valeur de ...	Commentai
1	Static								
2	Bool	0.0	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
3	Bool	0.1	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
4	Bool	0.2	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
5	Bool	0.3	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
6	Bool	0.4	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
7	Bool	0.5	false		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
8	Int	2.0	0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
9	Real	4.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
10	Real	8.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
11	Real	12.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
12	Real	16.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
13	Real	20.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
14	Real	24.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
15	Real	28.0	0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		

Figure II.8: Bloc de donnée DB1.

6. Création de la fonction *FC1* appelée (commande pompe) ce qui montre comment la pompe marche et s'arrête, ainsi que le mode de contrôle est automatique ou manuel à l'aide de la bascule *SR*.



## Chapitre II : Description Logicielle de programmation : du contrôle à la supervision

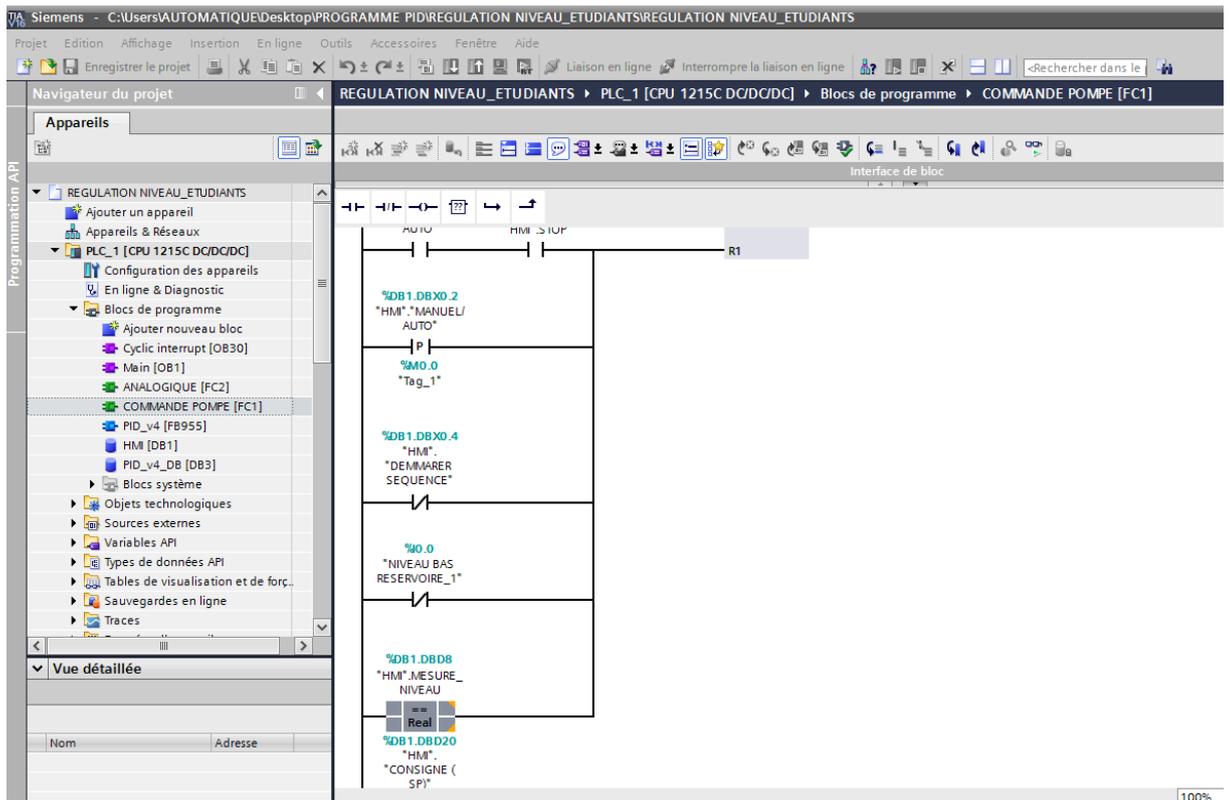


Figure II.9: Commande d'une pompe à l'aide de la bascule SR.

### 7. Glisser cette fonction *FC1* dans le programme principal (la main)

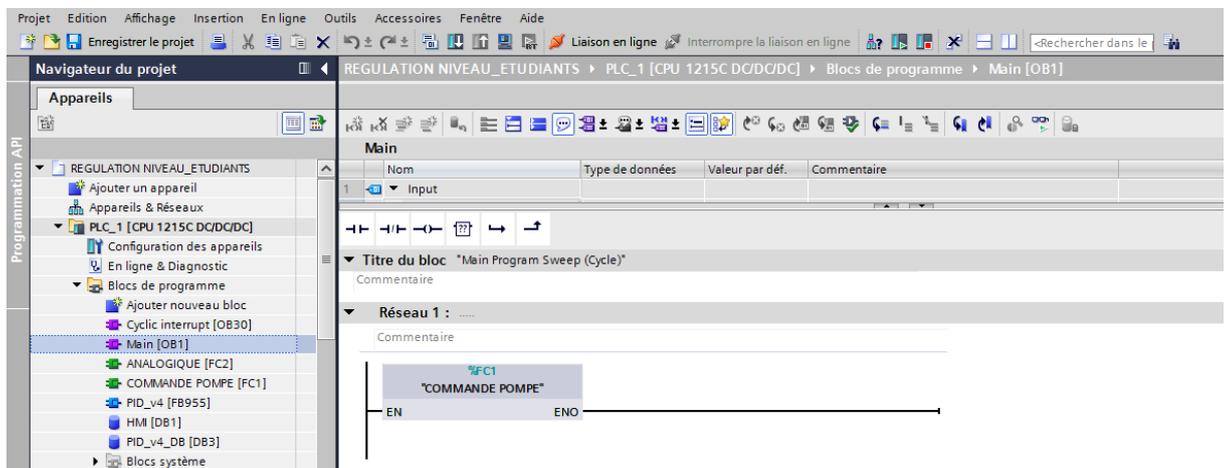


Figure II.10: La fonction FC1.

### 8. Création de la fonction *FC2* appelée (analogique)

- Le réseau1 pour la normalisation des signaux analogiques (consigne et mesure), conversion analogique Int (4-20mA) to Real.
- Le réseau2 pour convertir la valeur de *PID* de Real to Int.

## Chapitre II : Description Logicielle de programmation : du contrôle à la supervision

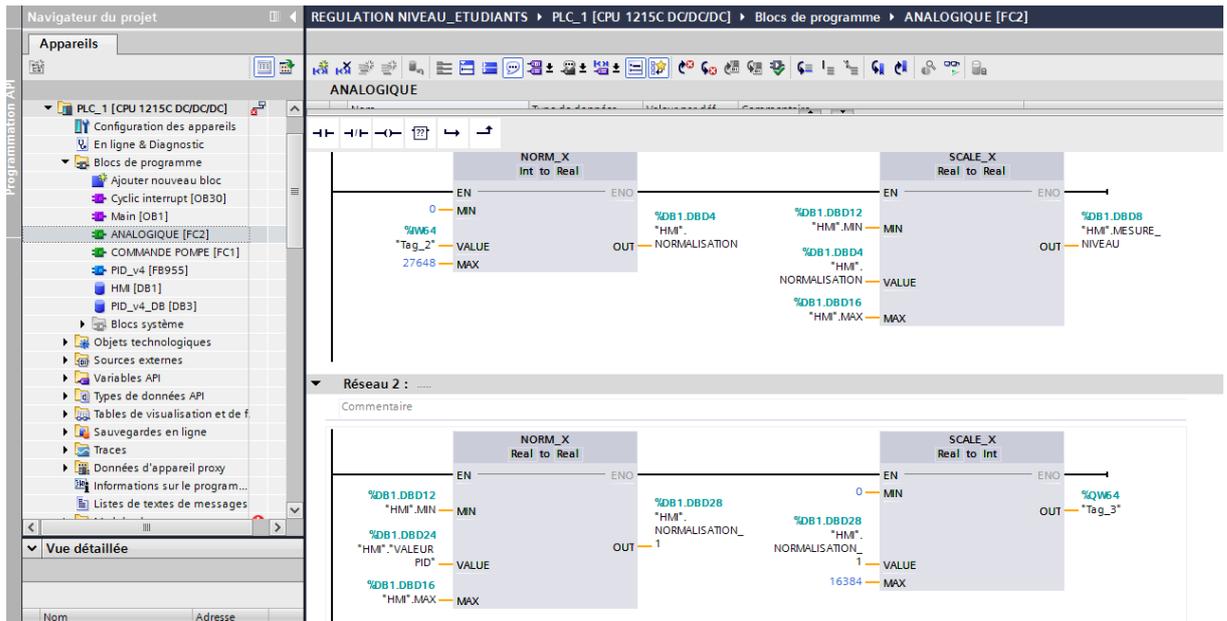


Figure II.11: Normalisation des signaux analogique.

### 9. Glisser cette fonction *FC2* dans le programme principal (la main)

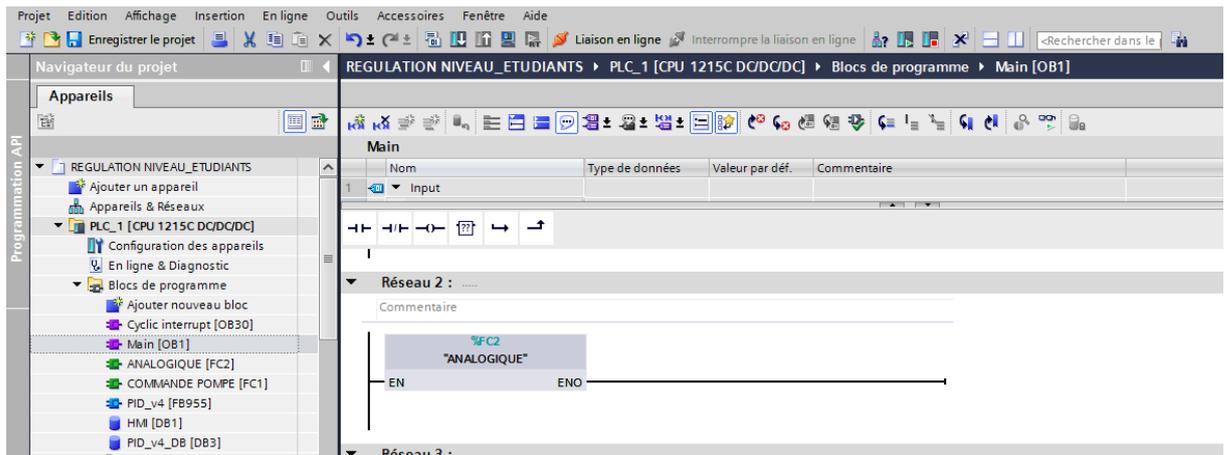


Figure II.12: La fonction FC2.

### 10. Création d'un régulateur *PID* qui comporte deux entrées (consigne et mesure niveau) et une sortie (valeur de *PID*).

## Chapitre II : Description Logicielle de programmation : du contrôle à la supervision

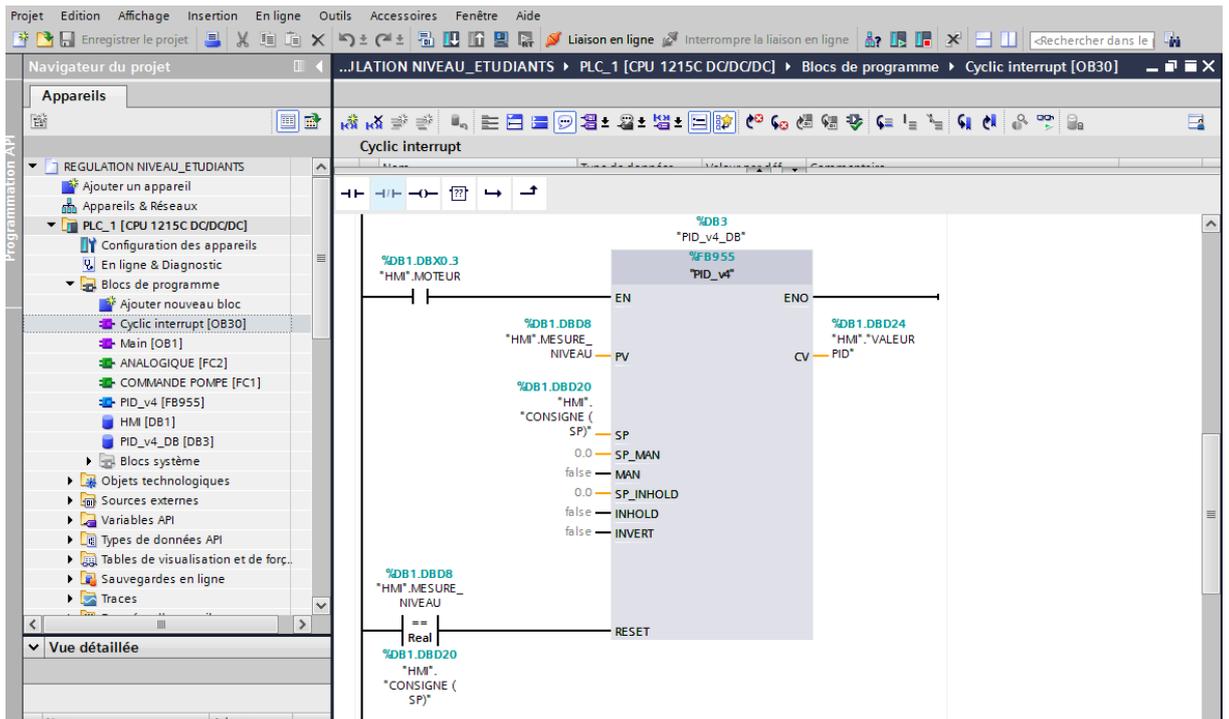


Figure II.13: Création d'un PID.

11. Le démarrage de la simulation et faire la liaison entre le *PC* et *PLC*, charge le programme dans *PLC*.

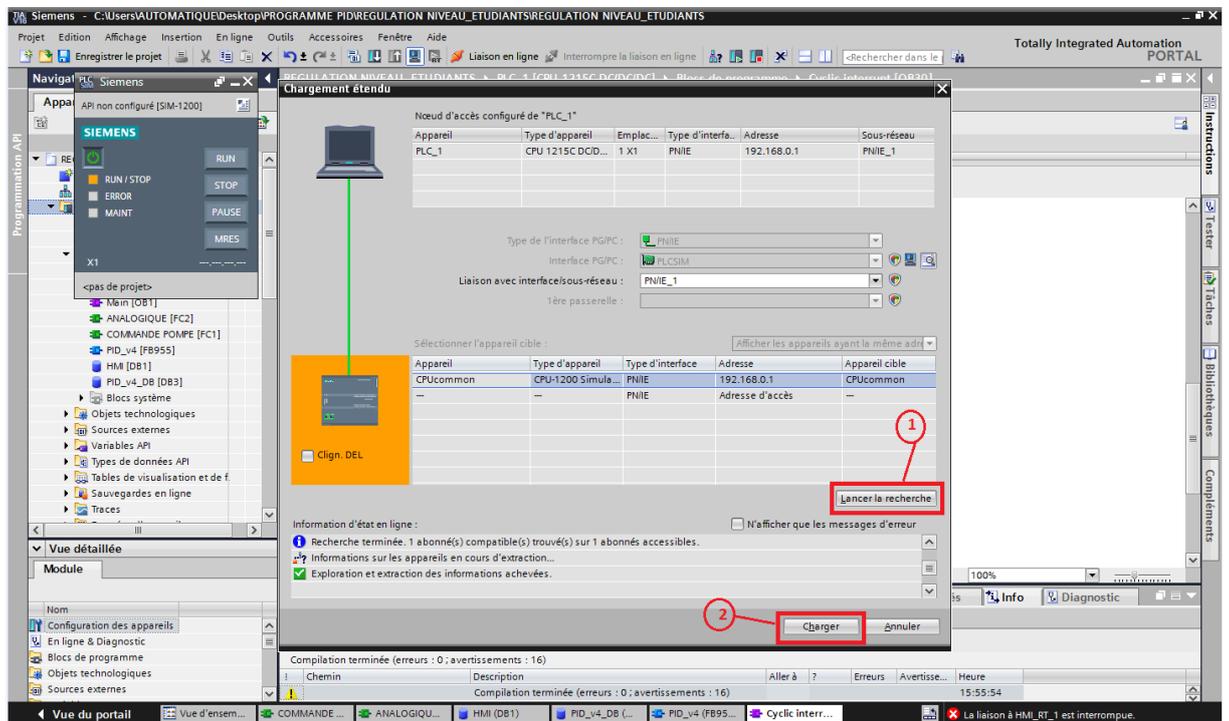


Figure II.14: Démarrage de simulation.

### II.4. SIMATIC WinCC Comfort

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

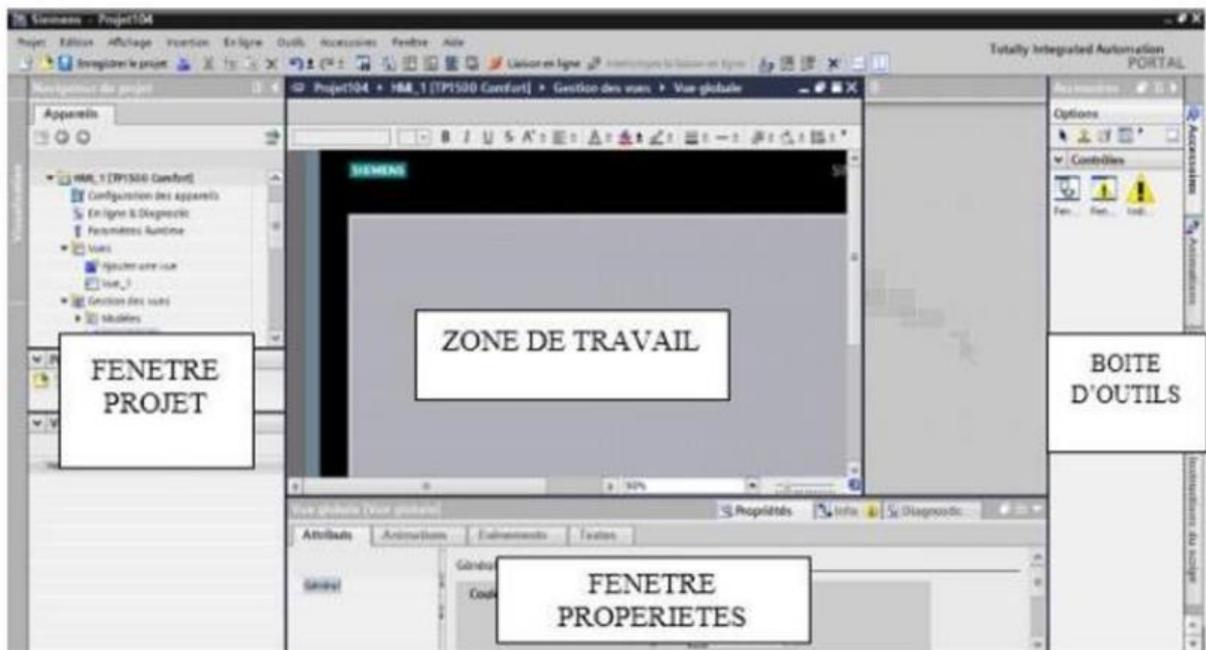


Figure II.15: Fenêtre du travail dans le WinCC.

Nous distinguons sur cette figure :

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

### II.5. Configurer une vue IHM

Les étapes ci-dessous montrent comment configurer une vue IHM

1. Sur la page d'accueil de *TIA Portal*, on choisit la visualisation en cliquant sur configurer une vue IHM.

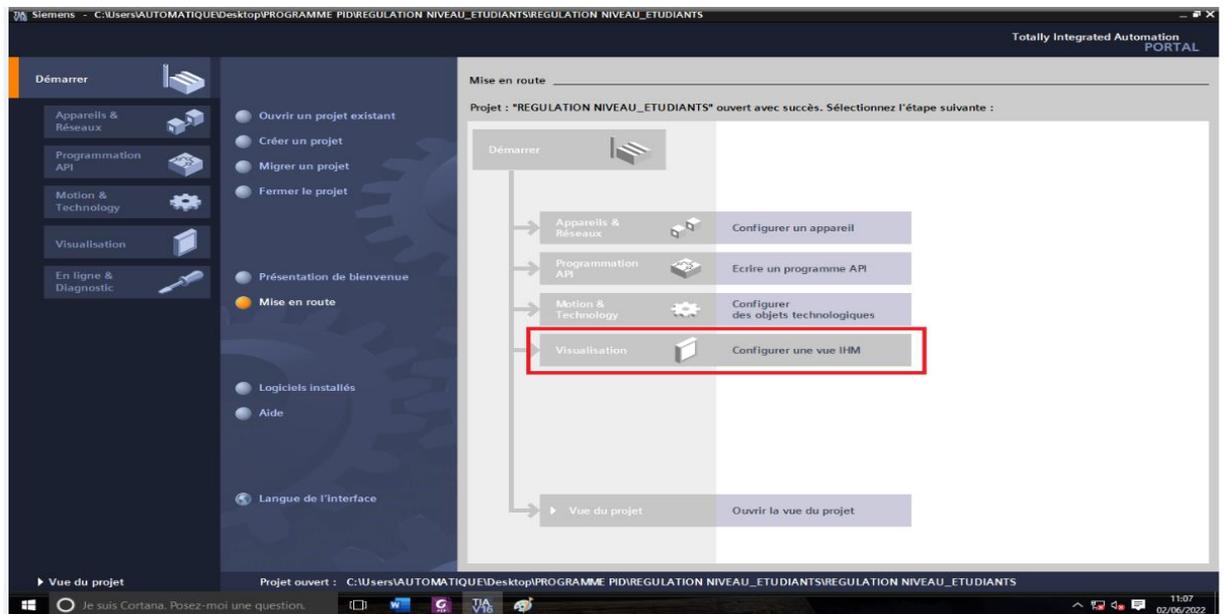


Figure II.16: Configurer une vue IHM.

2. On choisit le pupitre l'IHM SIMATIC Panel KTP700 Basic PN.

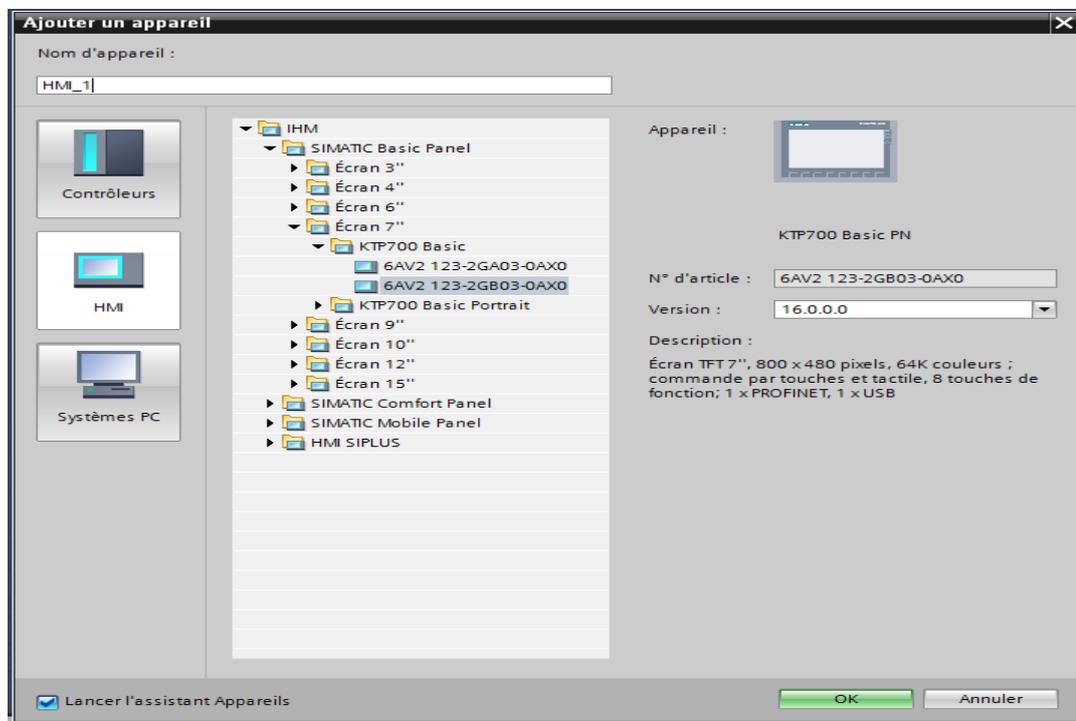


Figure II.17: Choisit le pupitre.

## Chapitre II : Description Logicielle de programmation : du contrôle à la supervision

Notre projet on a configuré 3 vues :

- 1<sup>ère</sup> vue : vue système
- 2<sup>ème</sup> vue : vue paramètres
- 3<sup>ème</sup> vue : vue courbes

La vue système est la vue principale, qui a deux modes, manuel et automatique.

- a) La figure suivante représente la vue de notre système en mode manuel et la pompe est éteint (stop)

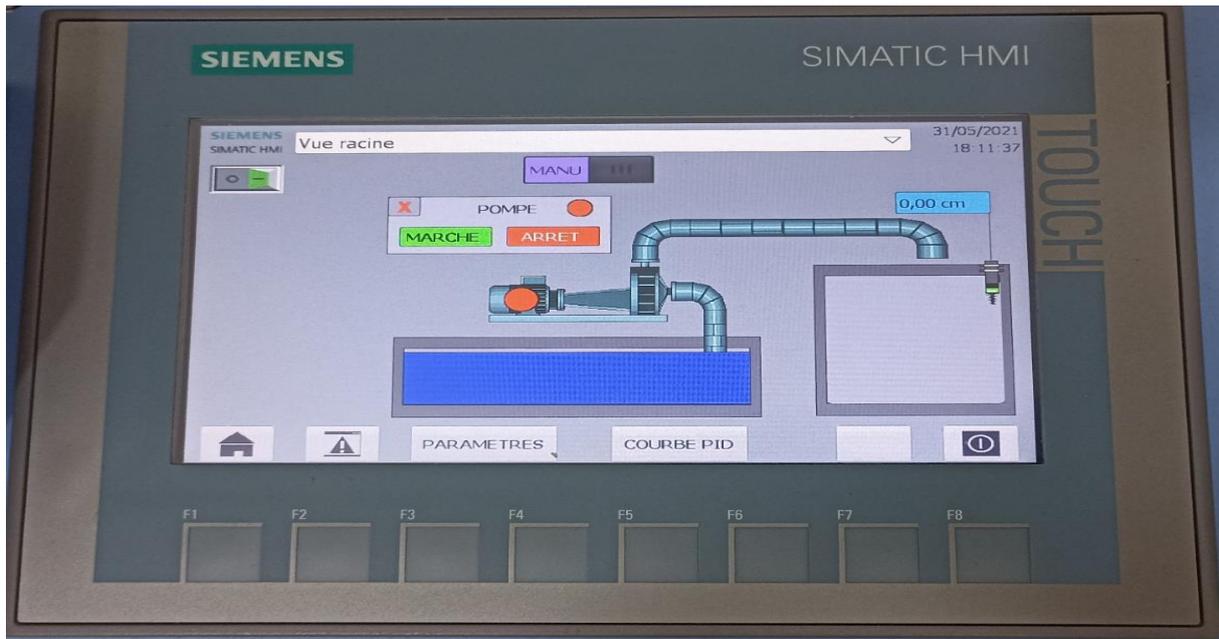


Figure II.18: Vue système en mode manuel et la pompe est éteint.

- b) La figure suivante représente la vue de notre système en mode manuel et la pompe en marche

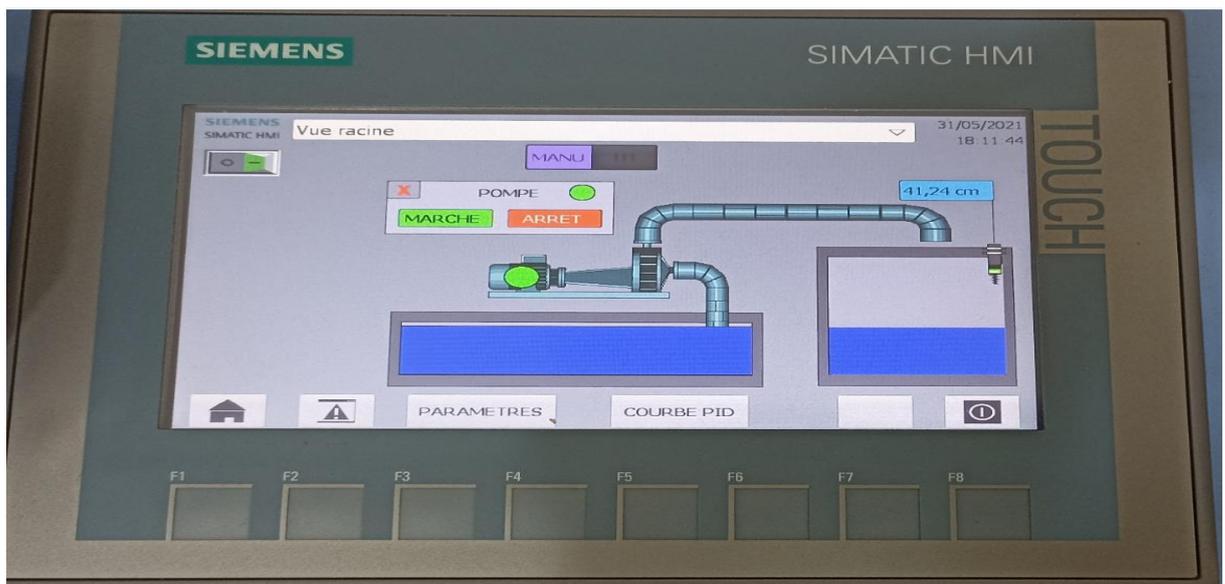


Figure II.19: Vue système en mode manuel et la pompe en marche.

## Chapitre II : Description Logicielle de programmation : du contrôle à la supervision

c) La figure suivante représente la vue de notre système en mode automatique

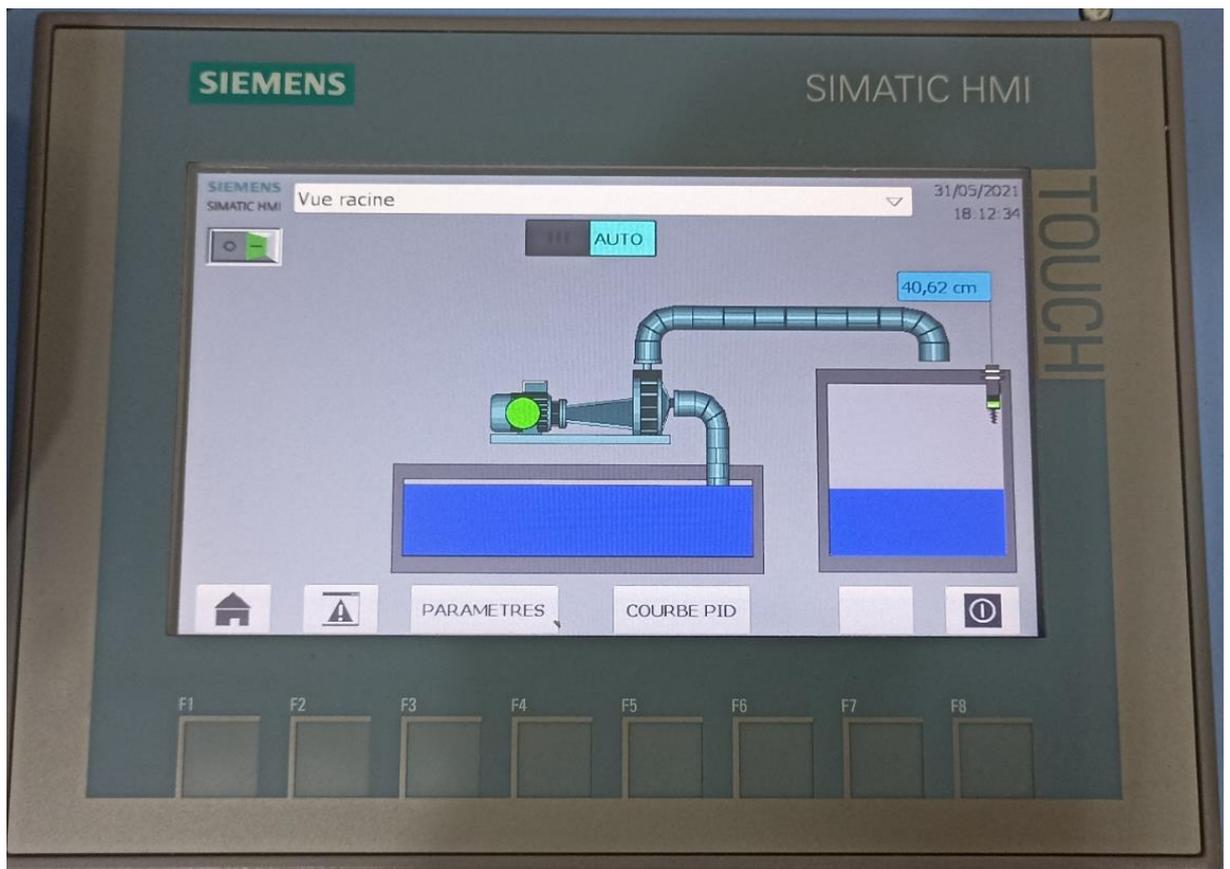


Figure II.20: Vue système en mode automatique.

La figure suivante représente la vue des paramètres du système :

### 1. Paramètres de réservoir :

- Echelle minimale
- Echelle maximale

### 2. Paramètres de *PID* :

- La consigne (*SP*)
- La mesure (*PV*)
- La correction de *PID* (*CV*)
- Proportionnel (*Kp*)
- L'intégrateur (*Ki*)
- Dérivateur (*Kd*)

## Chapitre II : Description Logicielle de programmation : du contrôle à la supervision

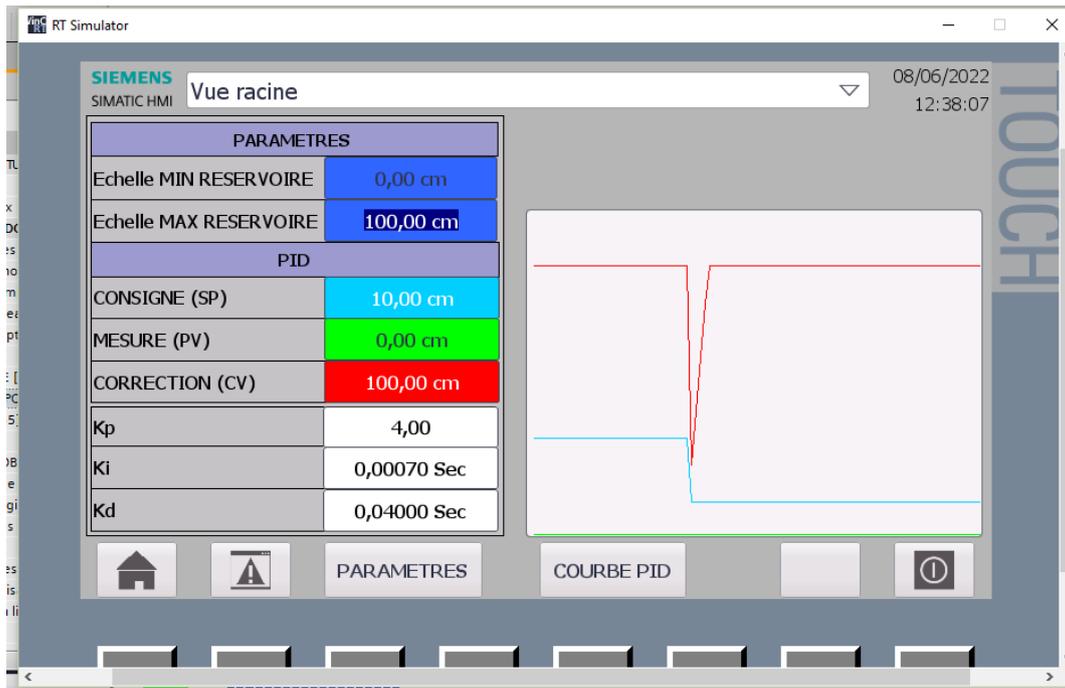


Figure II.21: Vue des paramètres.

La figure suivante représente la vue des courbes du système :

- Courbe de la commande (*CV*)
- Courbe de la consigne (*SP*)
- Courbe de la mesure (*PV*)

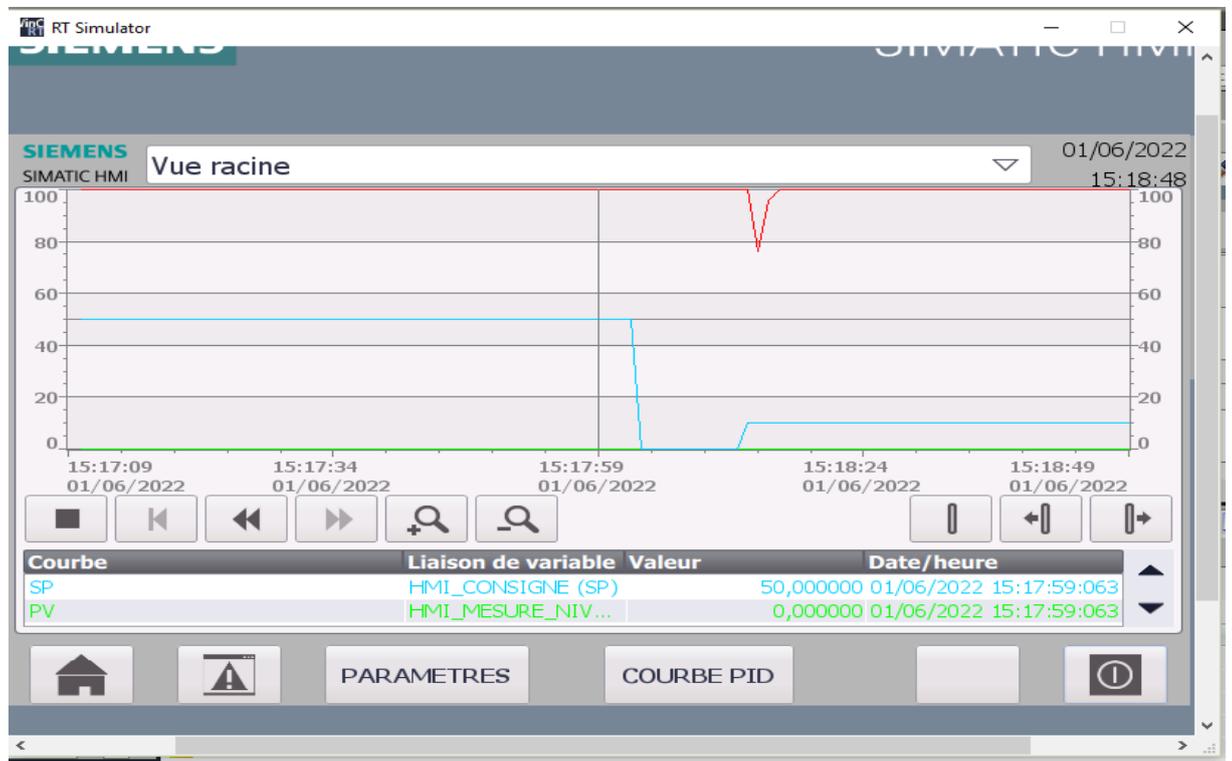


Figure II.22: Vue des courbes.

## II.6. Modélisation et simulation du système

Dans cette section, nous discuterons de la modélisation du système composé du réservoir d'eau. Nous pouvons effectuer une simulation sous Matlab/Simulink pour vérifier le modèle. Ce modèle nous aidera à choisir les paramètres de contrôle *PID* dans l'expérience.

**Modélisation du système** Les symboles suivants sont définis :

- $H$  : hauteur réelle
- $H_0$  : hauteur souhaitée
- $A$  : Zone transversale du réservoir d'eau
- $a$  : Zone transversale du tuyau
- $q_{in}$  : Dans débit du réservoir d'eau, supposé constant
- $q_{out}$  : Sortie du débit du réservoir d'eau
- $g$  : Accélération de la gravité
- $k_1$  : Vanne coefficient proportionnel
- $k_2$  : Coefficient proportionnel de flux à la hauteur désirée ( $H_0$ )

Les significations physiques des symboles du système de réservoir d'eau sont illustrées ci-avant. On note que  $A$ ,  $a$ ,  $q_{in}$  et  $g$  sont des paramètres invariants dans le temps, et  $q_{out}$  est lié à la valeur actuelle de la vanne de régulation.

Alors, selon la loi de conservation de la masse, la hauteur réelle est égale à la masse totale de flux d'informations en lui soustrait la masse totale en sortie.

Par conséquent, les équations qui viennent montrent la relation entre ces trois paramètres. La sortie peut être calculée comme suit :

$$A \frac{\partial}{\partial t} H = q_{in} - q_{out} \quad (\text{II.1})$$

$$q_{in} = k_1 a \sqrt{2gH} \quad (\text{II.2})$$

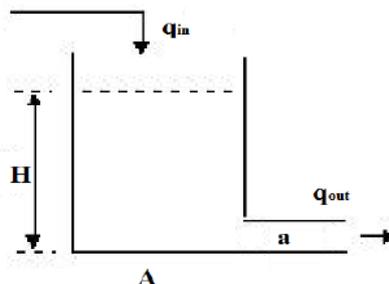


Figure II.23: Système de réservoir d'eau.

## Chapitre II : Description Logicielle de programmation : du contrôle à la supervision

Cependant, autour de la hauteur désirée, l'équation de sortie peut être linéarisée sous forme de modèle linéaire, qui peut être réécrit approximativement comme suit :

$$\frac{q_{out}}{2gH} \approx \frac{K_1 a}{\sqrt{2gH_0}} = K_2 \quad (\text{II.3})$$

$$q_{out} = K_2 2gH \quad (\text{II.4})$$

Ensuite, en substituant les équations données auparavant, le modèle du système est généré comme suit :

$$A \frac{\partial}{\partial t} H = q_{in} - K_2 2gH \quad (\text{II.5})$$

Le coefficient de valve est proportionnel à la position de la vanne et égal au courant réel de la vanne sur la valeur maximale du courant de la vanne. En prenant  $K_1$  et  $K_2$  dans les équations précédentes, le modèle du système peut être représenté par le courant réel  $I_{act}$  et la hauteur réelle  $H$ . L'équation est écrite sous forme suivante. Les valeurs de chaque paramètre sont indiquées dans le tableau ci-après.

$$\frac{\partial}{\partial t} H = C_1 q_{in} - C_2 I_{act} H \quad (\text{II.6})$$

Où, les constantes  $C_1$  et  $C_2$  peuvent être calculées par  $C_1 = \frac{1}{A}$  et  $C_2 = \frac{2ga}{\sqrt{2gH_0} I_{max} A}$

**Résultat de la simulation.** Le modèle Simulink est construit comme indiqué dans la Figure II.17 et la Figure II.18. Le modèle Simulink se compose de deux parties : le programme principal et un sous-système qui est le modèle du réservoir d'eau

Paramètres	Valeur	Unités
$A$	78.5	$\text{Cm}^2$
$a$	1.25	$\text{Cm}^2$
$H_0$	40	$\text{Cm}$
$q_{in}$	40	$\text{Cm}^3/\text{sec}$
$G$	1000	$\text{Cm}/\text{sec}^2$

Tableau II.2: Paramètres

La hauteur désirée est représentée par un signal échelon. L'erreur entre la hauteur désirée et celle réelle, qui est représentée par  $e(t)$ . Le jeu des gains et le paramétrage du contrôleur  $PID$  est effectué d'une manière empirique. Cependant, le contrôleur  $PID$  calcule un signal de commande  $u(t)$  sur la base de  $e(t)$ . Ensuite, le signal de commande  $u(t)$  est appliqué au système défini par un réservoir d'eau montré sur la figure suivante. Dans ce système, le bloc de gain représente le paramètre  $C_2$ . Étant donné que l'entrée  $q_{in}$  et la surface transversale du réservoir d'eau  $A$  sont constants, le rapport  $q_{in}/A$  est défini comme un nombre dans un marquage de bloc de fonction constante. Le résultat de la vue d'ensemble de la simulation est illustré à la Figure ci-après.

Sur la Figure II.19, on peut voir que la hauteur réelle augmente avec une pente constante parce que le

## Chapitre II : Description Logicielle de programmation : du contrôle à la supervision

Le système est sous le processus de remplissage sans action de canalisation. Pour le jeu de paramètres du contrôleur *PID*, la durée du remplissage du réservoir est de 20 sec environ. D'après le résultat de la simulation, nous pouvons voir que le contrôleur *PID* fonctionne très bien, et le but final est atteint.

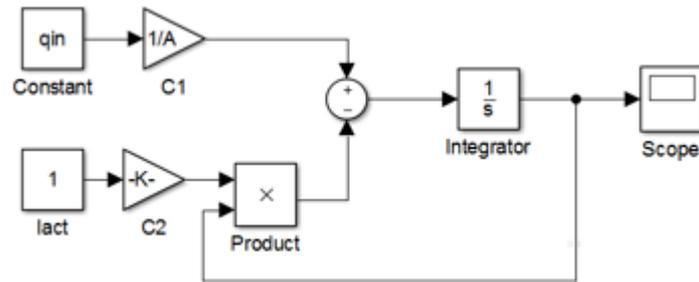


Figure II.24: Modèle Simulink du système de réservoir d'eau.

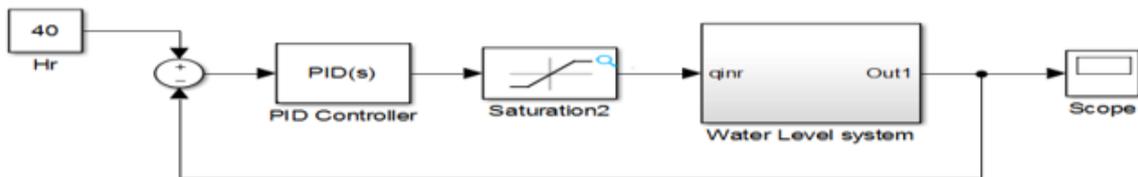


Figure II.25: Modèle Simulink du contrôle PID appliqué au réservoir d'eau.

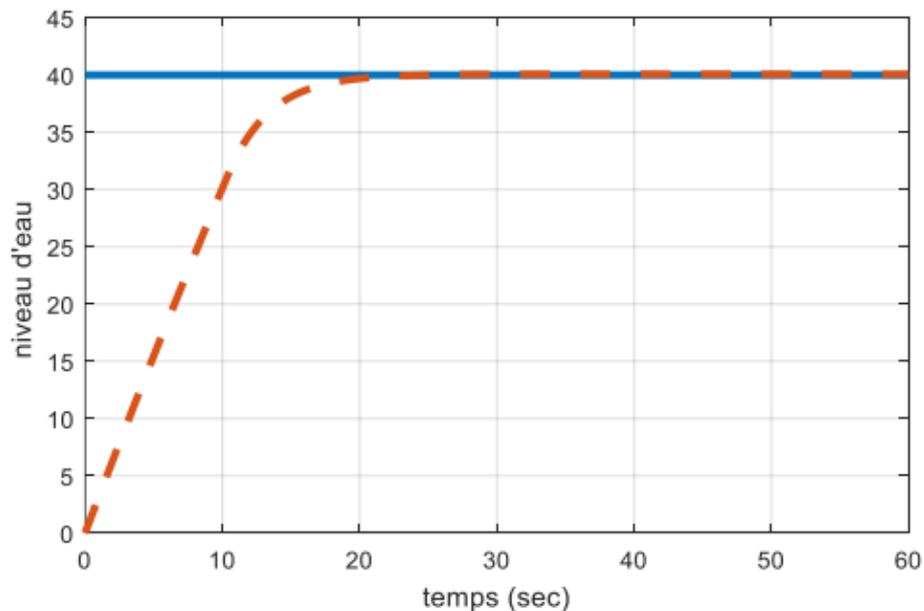


Figure II.26: Résultats de simulation du contrôle PID appliqué au réservoir d'eau.

### II.7. Conclusion

Dans ce chapitre, nous avons décrit le processus de création d'un programme *TIA PORTAL V16*. Les blocs *STEP 7 SIM*, *SIMATIC WINCC* et *API SIM*, disponibles et sont implémentés dans notre console *S7-1200*.

La partie *IHM* nous permettra de contrôler notre unité et de bien comprendre les capacités de notre système et le langage auquel la programmation de l'automate est connectée.

## **Chapitre III :**

# **Composantes matérielles utilisées pour la conception du système de contrôle du niveau d'eau**

## Chapitre III : Composantes matérielles utilisées pour la conception du système de contrôle du niveau d'eau

### III.1. Introduction

En tant qu'ingénieur en automatisation, on a tendance à avoir une vue systématique des systèmes et des processus de niveau d'eau. Notre objectif est d'automatiser et de standardiser ces systèmes pour augmenter la productivité, améliorer la qualité et la sécurité.

Dans ce chapitre, nous définirons les différents types de matériels à utiliser ainsi que les composants qui composent notre système et leurs caractéristiques.

### III.2. Panneau de formation API S7-1200 (CPU 1215C, KTP 700)

La figure suivante représente le panneau de formation API S7-1200 sur laquelle nous avons travaillé au laboratoire.



Figure III.1: Panneau de formation API S7-1200.

Ce dernier est composé de :

- API SIEMENS SIMATIC S7-1200 (CPU 1215C PN) : 14DI, 24V DC/10DO, 24V DC, 0,5A/2AI, 0-10V DC/2AO, 0-20mA
- MODULE COMMUTATEUR COMPACT SIEMENS : 5 X RJ 45
- MODULE D'ALIMENTATION SIEMENS SIMATIC S7-1200 : 120/230 V CA, 24 V CC, 2,5 A.
- IHM SIEMENS SIMATIC KTP700 avec connexion PROFINET

Le panneau comprend :

- 8 interrupteurs (rémanents et impulsions) pour simulation DI et 8 douilles de sécurité 4 mm pour DI externe
- 8 douilles de sécurité de 4 mm et LED d'état pour DO

### Chapitre III : Composantes matérielles utilisées pour la conception du système de contrôle du niveau d'eau

- 2 potentiomètres pour simuler les entrées des canaux analogiques dans la plage 0V DC...+10V DC
- 2 interrupteurs ON/ON/ON pour la sélection de la source AI (simulé via potentiomètre/ connecteur analogique SYSLINK / douilles de sécurité 4mm)
- 2 paires de douilles de sécurité de 4 mm pour les signaux AI externes
- 2 paires de douilles de sécurité 4 mm pour les signaux AO
- 1 paire de douilles de sécurité de 4 mm pour l'alimentation d'équipements externes via une sortie 24VDC
- 1 x connecteur numérique SYSLINK pour DI/DO (A)
- 1 x connecteur analogique SYSLINK (DB15) pour AI/AO (C)
- Prise d'alimentation avec interrupteur général et indicateur LED
- 1 paire de douilles de secours 4mm
- Prise Ethernet
- DIMENSIONS (L x H x P) : 532 x 297 x 170 mm
- Poids net : 3,35 kg
- LE PACKAGE COMPREND ÉGALEMENT : Câble d'alimentation, câble Ethernet, manuel d'utilisation.
- RECOMMANDÉ : Interrupteur, Câbles de connexion de sécurité 4 mm, TIA Portal – SIEMENS [19].

#### III.3. Le CPU 1215C DC/DC/DC

La figure suivante représente le CPU 1215C DC/DC/DC



Figure III.2: CPU 1215C DC/DC/DC.

### Chapitre III : Composantes matérielles utilisées pour la conception du système de contrôle du niveau d'eau

Description de notre *CPU* :

Série SIMATIC S7-1200 avec CPU 1215C, alimentation 24VDC, 14 entrées 24VDC, 10 sorties transistor, 2 entrées analogiques et 2 sorties analogiques 0...10V et la possibilité d'installer 8 extensions supplémentaires.

N° de catalogue : 6ES7 215-1AG40-0XB0 [20].

Informations techniques :

- Alimentation : 24 VDC
- Entrées numériques : 14x 24VDC (3 peuvent fonctionner comme compteurs rapides)
- Sorties numériques : 10x sorties transistor
- Entrées analogiques : 2x 0...10VDC (résolution 10 bits)
- Sorties analogiques : 2x 0...20mA (résolution 10 bits)
- Mémoire : 100 Ko
- Programme utilitaire : Step 7 Basic (CONT, LOG)
- Compteurs rapides : max 3 (100kHz)
- Régulateur PID avec réglage automatique
- Ethernet intégré - PROFINET
- Extensions de communication : RS-232, RS-485[20].

#### III.4. Alimentation rail *DIN* Siemens *SIMATIC PM 1207*

La figure suivante représente l'alimentation *SIMATIC PM 1207*



Figure III.3: SIMATIC PM 1207.

### Chapitre III : Composantes matérielles utilisées pour la conception du système de contrôle du niveau d'eau

Informations techniques :

- Type : Alimentation de coupure
- Tension d'entrée : 85 - 132 ou 176 - 264 V/DC
- Courant de sortie (max) : 2.5 A
- Courant de sortie (max) – arrondi : 2.5 A
- Phases : 1
- Nombre de sorties : 2 x
- Dim: (l x H x P) 70 x 100 x 75 mm
- Poids : 0.3 kg
- Profondeur : 75 mm
- Hauteur : 100 mm
- Largeur : 70 mm
- Connexion : borne à vis
- Référence fabricant : 6EP1332-1SH71
- Type: SIMATIC PM 1207 24 V/2,5 A
- Tension d'entrée nominale : 115 V/AC, 230 V/AC
- Tension d'entrée (max) : 264 V/AC
- Tension d'entrée (min) : 85 V/AC
- Tension de sortie maximale : 24 V/DC
- Tension de sortie minimale : 24 V/DC
- Tension de sortie (nom) : 24 V/DC
- Puissance : 60 W
- Connexions : Borniers à vis
- Tension de sortie : 24 V/DC
- Courant de sortie (détails) : 2,5 A
- Température max : +60 °C
- Température min : 0 °C
- Type de produit : Alimentation rail DIN [21].

#### III.5. Siemens *SCALANCE XB005*

Le *SCALANCE XB005* possède cinq connecteurs femelles RJ45 pour la connexion d'équipements terminaux ou d'autres segments de réseau [22].

### Chapitre III : Composantes matérielles utilisées pour la conception du système de contrôle du niveau d'eau



Figure III.4: SCALANCE XB005.

La figure suivante représente Caractéristiques techniques du *SCALANCE XB005*

### Chapitre III : Composantes matérielles utilisées pour la conception du système de contrôle du niveau d'eau

<b>Caractéristiques techniques</b>		
<b>Numéro de référence</b>		
SCALANCE XB005	6GK5 005-0BA00-1AB2	
<b>Connexion au réseau Industrial Ethernet</b>		
Nombre	5	
Exécution	Connecteurs femelles RJ45 avec brochage MDI-X	
Propriétés	semi-duplex / duplex	
Vitesse de transmission	10/100 Mbit/s	
<b>Longueurs de câble admissibles (Industrial Ethernet)</b>	<b>Combinaisons alternatives par plage de longueurs</b>	
0 ... 55 m	<ul style="list-style-type: none"> <li>max. 55 m IE TP Torsion Cable avec IE FC RJ45 Plug 180</li> <li>max. 45 m IE TP Torsion Cable avec IE FC RJ45 + 10 m TP Cord via IE FC RJ45 Outlet</li> </ul>	
0 ... 85 m	<ul style="list-style-type: none"> <li>max. 85 m IE FC TP Marine/Trailing Cable avec IE FC RJ45 Plug 180</li> <li>max. 75 m IE FC TP Marine/Trailing Cable + 10 m TP Cord via IE FC RJ45 Outlet</li> </ul>	
0 ... 100 m	<ul style="list-style-type: none"> <li>max. 100 m IE FC TP Standard Cable avec IE FC RJ45 Plug 180</li> <li>max. 90 m IE FC TP Standard Cable + 10 m TP Cord via IE FC RJ45 Outlet</li> </ul>	
<b>Caractéristiques électriques</b>		
Alimentation électrique	Plage de tensions	DC 19,2 ... 28,8 V très basse tension de sécurité (TBTS)
	Tension nominale	DC 24 V
	Exécution	Bornier enfichable à 3 bornes
Consommation	Typique	100 mA
Puissance active dissipée sous DC 24 V	Typique	2,30 W
Surintensité à l'entrée	Fusible PTC réarmable (0,6 A / 60 V)	
<b>Conditions ambiantes admissibles</b>		
Température ambiante	En fonctionnement	-10 °C ... +60 °C
	Pendant le stockage	-40 °C ... +80 °C
	Pendant le transport	-40 °C ... +80 °C
Humidité relative de l'air	En fonctionnement	≤ 95 %, sans condensation

### Chapitre III : Composantes matérielles utilisées pour la conception du système de contrôle du niveau d'eau

Caractéristiques techniques		
Altitude de service	En fonctionnement	≤ 2 000 m au-dessus du niveau de la mer à une température ambiante de 56 °C max. ≤ 3.000 m au-dessus du niveau de la mer à une température ambiante de 50 °C max.
Forme, dimensions et poids		
Immunité aux perturbations	EN 61000-6-2	
Emission de perturbations	EN 61000-6-4	
Degré de protection	IP 20	
MTBF (EN/CEI 61709; 40 °C)	241 ans	
Matériau du boîtier	Polycarbonate (matière plastique)	
Poids	165 g	
Dimensions (L x H x P)	45 x 100 x 87 mm	
Possibilités de montage	<ul style="list-style-type: none"><li>• Montage sur rail DIN symétrique</li><li>• Fixation murale</li></ul>	
Propriétés de commutation		
Aging Time	280 secondes	
Nombre max. d'adresses MAC assimilables	2048	
Comportement en présence de télégrammes LLDP	Blocking	
Comportement en présence de télégrammes BPDU Spanning Tree	Forwarding	
QoS Priority Queues	2	

Figure III.5: Caractéristiques techniques du SCALANCE XB005 [22].

#### III.6. IHM SIEMENS SIMATIC KTP700 PN

La figure suivante représente l'écran *IHM KTP700 Basic PN*



Figure III.6: IHM SIEMENS SIMATIC KTP700 Basic PN.

### Chapitre III : Composantes matérielles utilisées pour la conception du système de contrôle du niveau d'eau

Description de l'écran :

C'est un écran TFT, 800\*480 pixels, 64k couleurs, commande par touches et tactile, 8 touches de fonction, 1\*PROFINET, 1\*USB [23].

Fiche de données :

- Conception de l'affichage: TFT widescreen display, LED backlighting
- Diagonale de l'écran : 7 in
- Nombre de couleurs : 65 536
- Résolution d'image horizontale : 800 Pixel
- Résolution d'image verticale : 480 Pixel
- Les touches de fonction : 8
- USB interfaces: 1; Up to 16 GB
- Écran tactile: OUI
- Position de montage : Horizontal, vertical
- Ethernet interfaces : 1
- Tension d'entrée : 24 VDC [23].

#### III.7. Partie réalisation

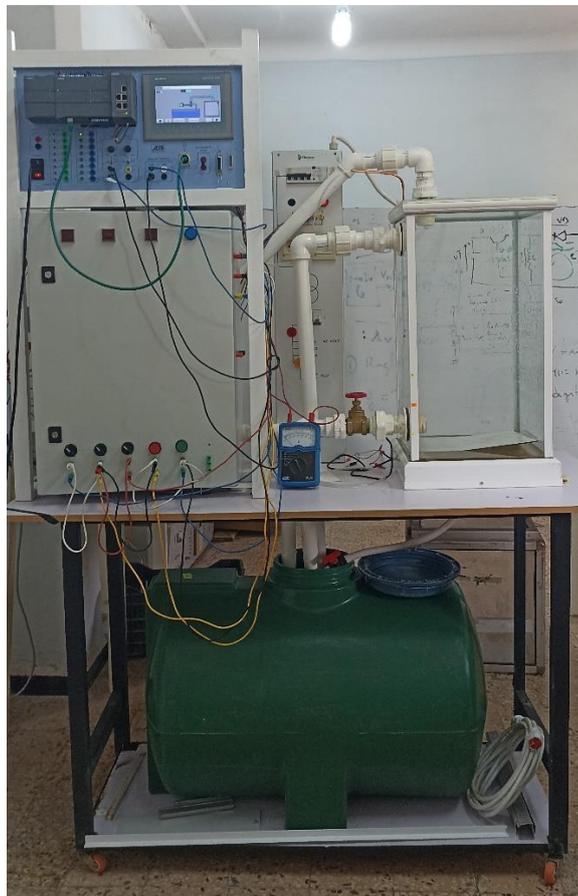


Figure III.7: Vue du système de niveau d'eau.

### Chapitre III : Composantes matérielles utilisées pour la conception du système de contrôle du niveau d'eau

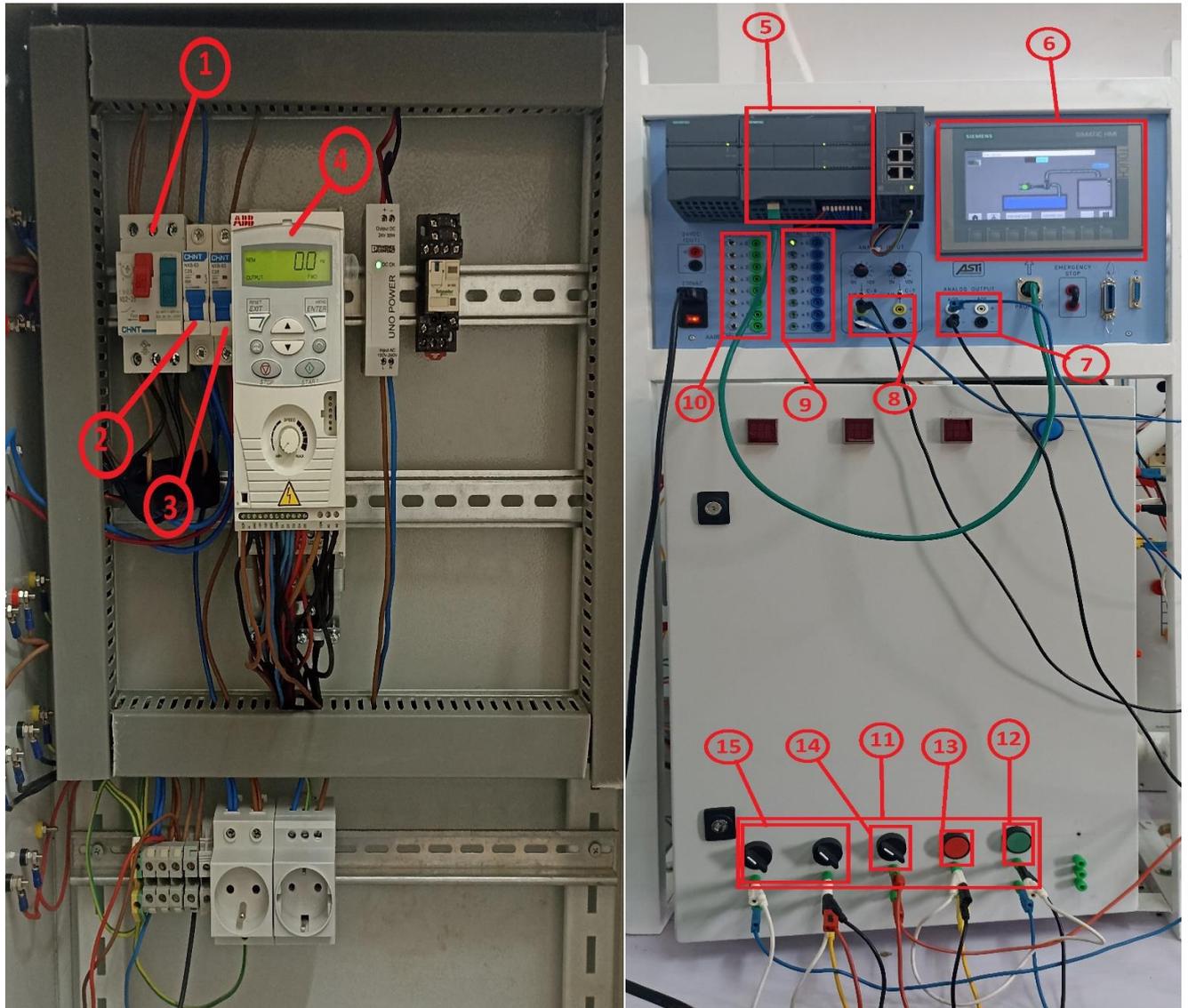


Figure III.8: Vue des composants utilisés.

1. Disjoncteur (protection de moteur)
2. Protection de phase
3. Protection de neutre
4. Variateur de vitesse ABB
5. CPU 1215C DC/DC/DC
6. IHM KTP700 Basic PN
7. Sorties analogiques
8. Entrées analogiques
9. Sorties numériques

### Chapitre III : Composantes matérielles utilisées pour la conception du système de contrôle du niveau d'eau

10. Entrées numériques
11. Commande RUM de variateur de vitesse
12. Bouton start
13. Bouton arrêt
14. Commutateur pour inverse le sens de rotation
15. Commutateurs pour varier la vitesse

#### III.8. Résultats obtenir

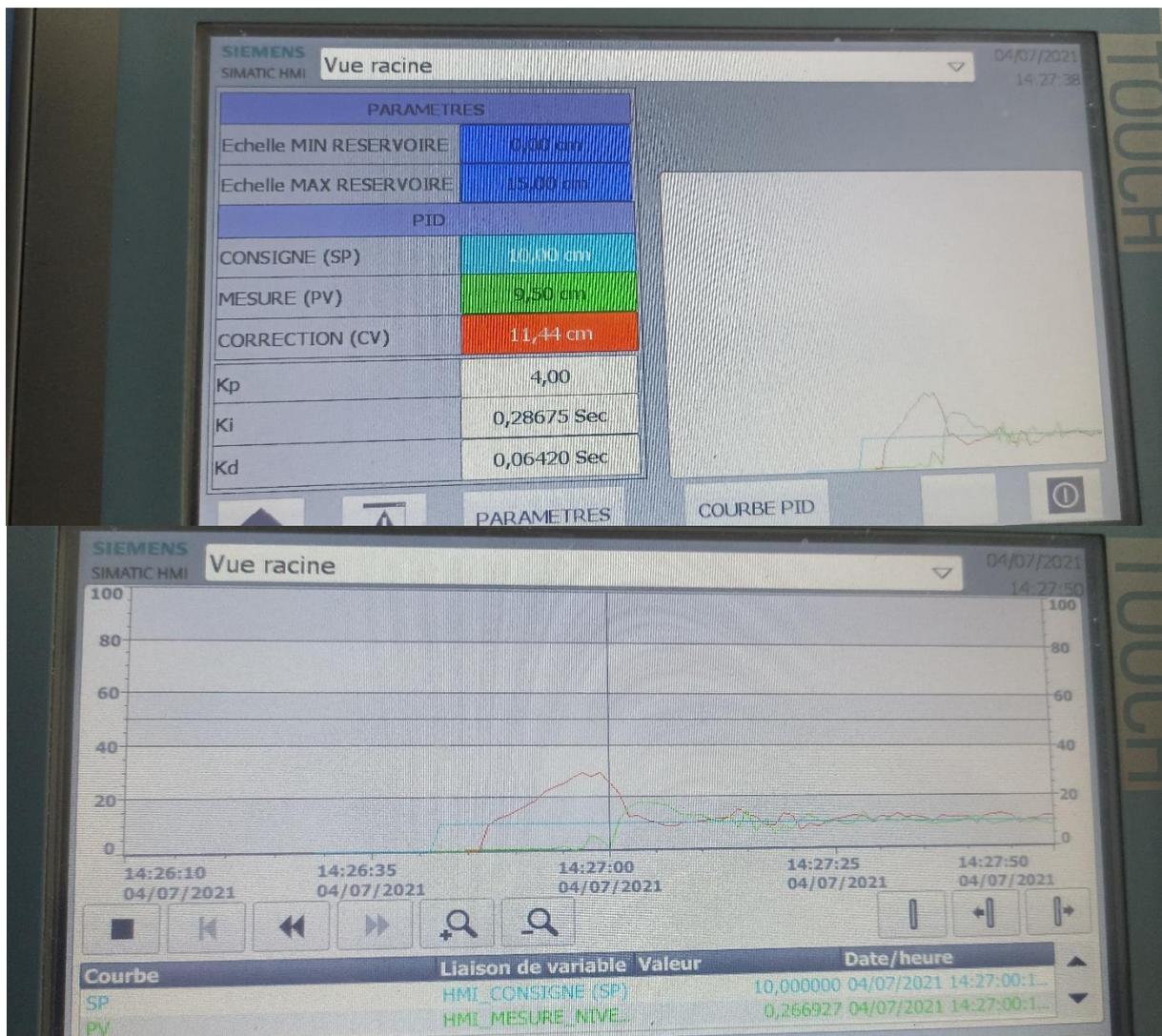


Figure III.9: Vue des courbes de PID.

## **Chapitre III : Composantes matérielles utilisées pour la conception du système de contrôle du niveau d'eau**

### **III.9. Conclusion**

Dans ce chapitre, nous avons effectué une description détaillée du panneau de formation *API S7-1200* sur lequel nous avons travaillé tout en balayant ses principaux composants (*CPU 1215C*, *PM 1207*, *SCALANCE XB005*, *KTP700*) et leurs propriétés et leurs caractéristiques...

# **Conclusion Générale**

# Conclusion Générale

## Conclusion Générale

Au cours de ce projet de fin d'études, ayant comme objectif le contrôle et supervision d'un système de niveau d'eau, avec un automate industriel programmable de type *SIEMENS SIMATIC HMI*, associé à un écran tactile. Nous avons eu l'opportunité d'enrichir nos connaissances en matière de gestion de niveau notamment avec l'emploi des contrôleurs *PID*, de l'actionneur et le variateur de vitesse qui commande la pompe à débit variable et qui régule à son tour le niveau. On a appris à programmer une machine. *API SIEMENS SIMATIC S7-1200*, de plus nous avons appris à travailler sur le logiciel *TIA Portal V16*, et à faire de la supervision et de la simulation à l'aide du logiciel *SIMATIC WINCC* intégré dans *TIA Portal*.

Ensuite, nous avons proposé la solution de contrôle numérique et de contrôle analogique par un correcteur *PID* intégré. L'efficacité de la solution proposée a été démontrée en simulation à l'aide du logiciel *TIA Portal V16* de Siemens.

Dans la dernière partie, nous avons développé une plateforme de monitoring basée sur *SIMATIC WINCC* intégré dans logiciel *TIA Portal*. Cette plateforme permet à l'opérateur de suivre le fonctionnement du procédé physique et d'intervenir en cas de panne voire la modification des conditions de fonctionnement.

Enfin nous pouvons dire que ce travail n'a pas été facile parce que nous avons rencontré beaucoup de problèmes durant sa réalisation, mais nous avons réussi à le réaliser grâce à notre volonté.

En perspective, nous espérons qu'il y aura plus de travail pratique sur ce sujet pour les mises à niveau dans les années à venir

# **Bibliographie**

# Bibliographie

## Bibliographie

- [1] SEKHSOKH. S, OUKILI. K « ÉTUDE D'UNE BOUCLE DE RÉGULATION DE NIVEAU : IMPLÉMENTATION DU RÉGULATEUR ET RÉGLAGE DU PROCÈDE » Projet de fin d'étude, 2010/2011.
- [2] ANDRE. S, Automates programmable, programmation, automatisation, et logique programmée, Edition L'ELAN, 1983.
- [3] "Sommaire des cours". Siemens, centre de formation industrie, édition 2000.
- [4] BERGOUGNOUX.L., "API automate programmable industriel". Poly Tech Marseille, Édition 2004-2005.
- [5] GRARE.P & KACEM.I., "AUTOMATISME Ce qu'il faut s'avoir sur les automatismes ». Edition ellipses, 2008.
- [6] MICHEL. G, Les API, Architecture Et Application Des Automates Programmables. Industriels. Dunod, Paris, [1987].
- [7]<https://www.usinenouvelle.com/expo/simatic-s7-1200-controleur-modulaire-p196958.html>, consulté le : 15/04/2022
- [8]<https://www.google.com/search?q=definition+d%27un+automate+s7-1500&oq=definition+d%27un+&aqs=chrome.2.69i57j35i39l2j0i512l7.13644j0j15&sourceid=chrome&ie=UTF-8>, consulté le : 15/04/2022
- [9] 01 1200 Initiation au programme du SIMATIC S7-1200.pdf
- [10] Automatisation et supervision de l'unité de traitement des eaux par osmose inverse du complexe CEVITAL via le logiciel TIA portal v13 de SIEMENS.pdf
- [11] Automatisation et supervision d'une station de Thermolaquage par un automate S7-1200 complet.pdf
- [12] Automate programmable S7-1200 Manuel système, V4.3.0 02/2019, A5E02486682-AM
- [13] « COURS AUTOMATIQUE » École polytechnique fédérale de Lausanne, 2008/2009.
- [14] PATRICK PROUVOST « AUTOMATIQUE CONTROLE ET REGULATION COURS ET EXERCICES CORRIGES » DUNOD (2004).
- [15] ALAIN GONZAGA « Les automates Programmables Industriels ». PDF

## Bibliographie

- [16] Automatisation et supervision d'une station de Thermolaquage par un automate S7-1200 complet.pdf
- [17] Commande\_d'un\_système\_pneumatique\_didactique\_par.pdf
- [18] SIEMENS, « S7-1200\_System\_Manual », Numéro de référence du document : A5E02486682-AG 03/2014
- [19] <https://astiautomation.com/product/plc-s7-1200-training-panel-cpu-1215c-ktp-700>, consulté le : 06/06/2022.
- [20] <https://consteel-electronics.com/S7-1200-6ES7215-1AG40-0XB0-en>, consulté le : 06/06/2022
- [21] <https://www.conrad.fr/p/alimentation-rail-din-siemens-simatic-pm-1207-24-v25-a-24-vdc-25-a-60-w-2-x-513085>, consulté le : 07/06/2022
- [22] SCALANCE XB-000 Instructions de service, 07/2014, A2B00077300-04
- [23] <https://www.plc-city.com/shop/fr/siemens-simatic-hmi-basic-panels-2nd-generation/6av2123-2gb03-0ax0.html>, consulté le : 07/06/2022

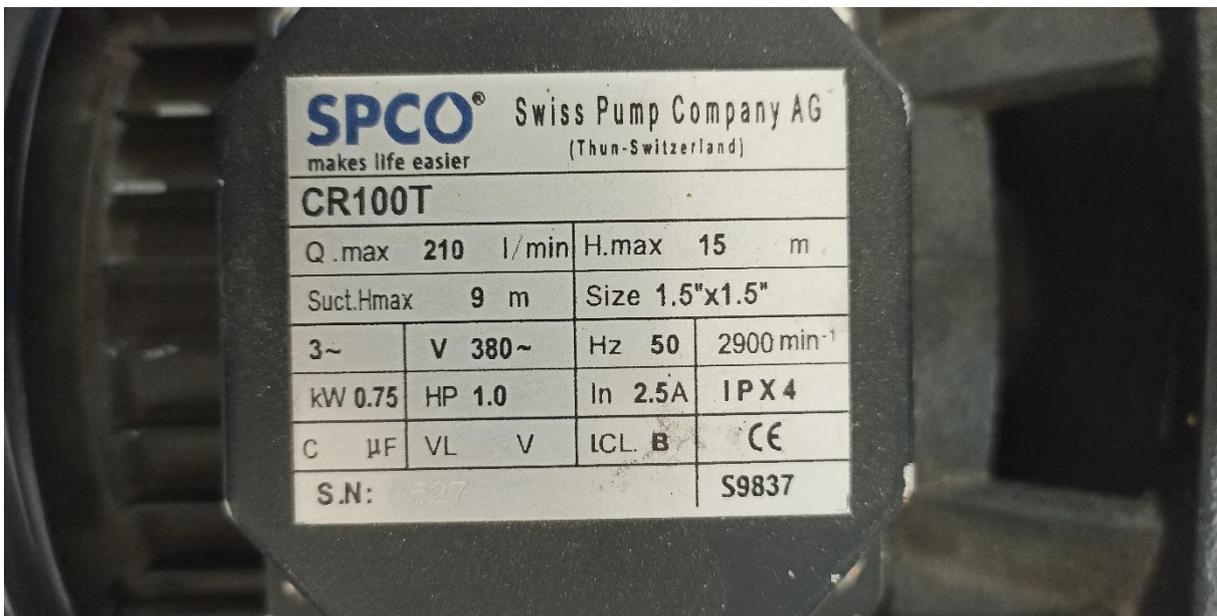
# **Annexe**

## Annexe

### 1- La plaque signalétique de variateur de vitesse ABB :



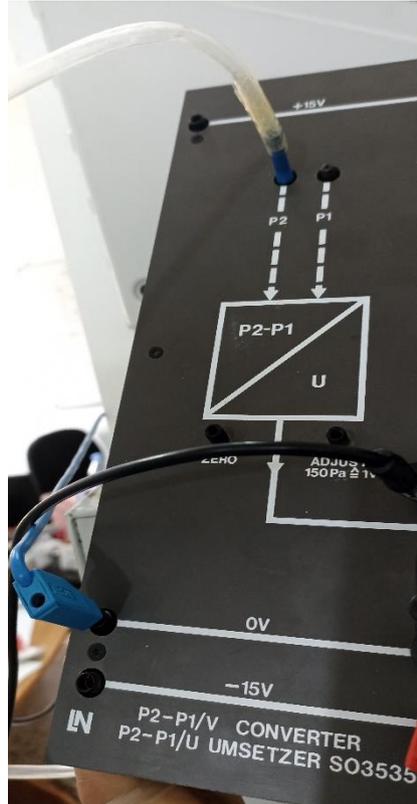
### 2- La plaque signalétique de la pompe:



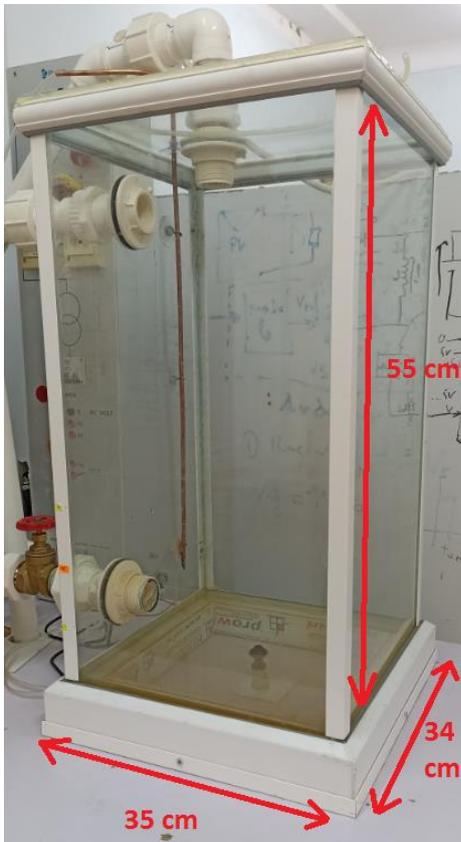
### 3- Le capteur de pression:

Chaque 1V nous donne 150Pa.

## Annexe



### 4- La dimension du réservoir:



$$P = \frac{F}{S}$$

Avec:  $F = V * g$

$$S = 35 * 34 = 11.9 m^2$$

$$V = 35 * 34 * 55 = 654.5 m^3$$

$$\text{Donc : } P = \frac{654.5 * 9.807}{11.9} \approx 540 Pa = 0.005 bar$$

Avec :

P : La Pression ; V : Le Volume

F : La Force ; g : la gravité

S : La Surface

## Résumé

Le but de ce projet de fin d'études est la surveillance et le contrôle du niveau d'eau au moyen d'un correcteur numérique de type PID, avec un automate programmable de SIEMENS SIMATIC S7-1200 disponible au Laboratoire d'Automatique du Département de Génie Electrique. Cependant la programmation d'automate, le contrôle et la supervision via l'écran de contrôle d'IHM. Dans ce système on emploie le TIA Portal v16 qui contient Step7 et SIMATIC WINCC.

### Mots clés :

Automatiser, contrôle de niveau d'eau, API, Siemens, S7-1200, step7, Simatic WINCC, IHM, TIA PORTAL V16.

## Abstract

The purpose of this end-of-study project is the monitoring and control of the water level by means of a PID digital corrector, with a SIEMENS SIMATIC S7-1200 programmable controller available at the Automation Laboratory of the Department of Electrical Engineering., we started PLC programming, supervision and control via HMI control screen in This system uses TIA Portal v16 which contains Step7 and SIMATIC WINCC.

### Key words:

Automate, regulate water level, PLC, Siemens, S7-1200, step7, Simatic WINCC, HMI, TIA PORTAL V16.

## الملخص

الهدف من مشروع نهاية الدراسة هذا هو الإشراف على مستوى المياه والتحكم فيه عن طريق المصحح الرقمي PID، مع وجود وحدة تحكم قابلة للبرمجة SIEMENS SIMATIC S7-1200 متوفرة في مختبر الأتمتة التابع لقسم الهندسة الكهربائية، وقد بدأنا برمجة PLC والإشراف والتحكم عن طريق شاشة تحكم IHM في هذا النظام باستخدام برنامج TIA Portal v16 الذي يحتوي على Step7 و SIMATIC WINCC.

### الكلمات المفتاحية:

أتمتة، تنظيم مستوى المياه، API، Siemens، S7-1200، Step7، Simatic WINCC، IHM، TIA Portal V16.