



Université Mohamed Khider de Biskra
Faculté des Sciences et de la Technologie
Département de génie électrique

MÉMOIRE DE MASTER

Sciences et Technologies
Automatique
Automatique et informatique industrielle

Réf. : Entrez la référence du document

Présenté et soutenu par :
BOUTALBI Ines
BOUAFIANE Dhikra Anfel

Le : jeudi 5 juin 2025

Système de Contrôle pour une Installation de Stockage de l'eau

Jury :

Mlle.	MIHI Assia	MCB	Université de Biskra	Présidente
M.	ZITOUNI Athmane	Pr	Université de Biskra	Examineur
M.	SAADOUNE Achour	Pr.	Université de Biskra	Encadreur

Année universitaire : 2024/2025



Université Mohamed Khider de Biskra
Faculté des Sciences et de la Technologie
Département de génie électrique

MÉMOIRE DE MASTER

Sciences et Technologies
Automatique
Automatique et informatique industrielle

Réf. : Entrez la référence du document

Systeme de Contrôle pour une Installation de Stockage de l'eau

Le :

Présenté par :
BOUALBI Ines
BOUAFIANE Dhikra Anfel

Avis favorable de l'encadreur :
Pr. SAADOUNE Achour

Signature Avis favorable du Président du Jury

Cachet et signature

Ministère de l'enseignement Supérieur et de la Recherche scientifique



Université Mohamed Khider Biskra
Faculté des Sciences et de la Technologie
Département de Génie Electrique
Filière : Automatique

Option : Automatique et Informatique Industrielle

Thème : ***Système de Contrôle pour une Installation de Stockage de l'eau***

Proposé par : Pr. SAADOUNE Achour

Dirigé par : Pr. SAADOUNE Achour

RESUMES

Ce projet a pour objectif la surveillance et le contrôle du niveau d'eau à l'aide d'un automate programmable SIMATIC S7-1200. Le système repose sur un capteur analogique pour mesurer en continu le niveau et une vanne proportionnelle pour réguler le débit d'eau. Le programme a été développé avec TIA Portal, la logique de commande a été réalisée dans STEP 7, et une interface de supervision a été conçue sur un écran IHM à l'aide de WinCC. Le système a été testé virtuellement avec l'outil de simulation PLCSIM.

Mots clés :

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

Abstract

This project aims to monitor and control the water level using a SIMATIC S7-1200 programmable logic controller. The system relies on an analog sensor to continuously measure the level and a proportional valve to regulate water flow. The program was developed using TIA Portal, with control logic created in STEP 7 and a monitoring interface designed on an HMI screen using WinCC. The system was tested virtually using the PLCSIM simulation tool.

Key words:

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

الملخص

يهدف هذا المشروع إلى مراقبة مستوى المياه والتحكم فيه باستخدام وحدة التحكم SIMATIC S7-1200. يعتمد النظام على مستشعر تناظري لقياس المستوى، وصمام تناسبى لتنظيم تدفق المياه. تم تطوير البرنامج باستخدام TIA Portal، مع البرمجة عبر STEP 7، وتصميم واجهة المراقبة على شاشة IHM باستخدام WinCC وقد تم اختبار النظام افتراضياً باستخدام أداة المحاكاة PLCSIM.

Dédicaces

*Je dédie ce modeste travail à : À mes chers
parents,
Votre amour, vos sacrifices et votre soutien
inestimable ont toujours été ma source
d'inspiration et de force. Merci pour tout ce que
vous avez fait pour moi. Qu'Allah vous accorde
santé, bonheur et longue vie.*

*À mes frères et sœurs : Djihane, Amani, Malak et
Abd ELraouf, votre affection et votre présence
constante ont été d'un grand réconfort tout au
long de ce parcours.*

*À mes enseignants, pour leur patience, leur
accompagnement et leur engagement envers
notre réussite.*

*À tous mes chers amis, pour leur soutien moral
et les souvenirs précieux partagés durant ces
années d'études.*

*À toutes les personnes qui ont contribué de près
ou de loin à la réalisation de ce travail.*

INES

Dédicaces

A TOUTE MA FAMILLE

A TOUS MES AMIS

Anfel

Sommaire

Introduction générale	01
Chapitre I : Contrôle du niveau d'eau dans un réservoir	
1.1 Introduction	04
1.2 Présentation du besoin de régulation du niveau d'eau dans un réservoir	04
1.2.1 Pourquoi réguler le niveau d'eau ?	04
1.2.2 Techniques de régulation du niveau d'eau	05
1.2.3 Applications Pratiques	05
1.3 Surveillance continue du niveau par capteur analogique	06
1.4 Importance d'une commande proportionnelle de la vanne pour maintenir le niveau	06
1.5 Objectifs de l'intégration d'une HMI pour la supervision en temps réel	08
I.6 Objectifs du projet	08
I.7 Outils et technologies utilisées	09
I.7.1 Automate programmable (PLC)	09
I .7.2 Critères de choix de l'automate programmable industriel	10
I.7.3 Présentation de quelques gammes SIMATIC	10
I .7.3.1 SIMATIC S7	10
I .7.3.2 SIMATIC S7-200	11
I .7.3.3 SIMATIC S7-400	11
I .7.3.4 SIMATIC S7-1200	12
I.7.3.5 SIMATIC S7-1500	13
I.7.4 Présentation l'automate utilise dans ce projet (S7-1200 Siemens)	13
I.7.4.1 Description	13
I.7.4.2 Modules d'extensions	14
I.7.5 Les type des module entrée / sortie	16
I .7.5.1 Les modules d'entrée / sortie TOR	16
I .7.5.2 Module d'entrée/sortie analogique	16

I .8 Logiciels de simulation	17
I.8.1 PLCSIM	17
I .9 Interface Homme-Machine (IHM)	18
I.9.1 SIMATIC WinCC	18
I.10 Vue du portail et vue du projet	20
I.11 Conclusion	23
Chapitre II : Conception du système de contrôle avec HMI	
II.1 Introduction	25
II.2 Description des composants	25
II.2.1 Capteur de niveau	25
II.2.2 Les types de capture de niveau	26
II.2.2.1 Capteurs de niveau analogiques	26
II.2.2.2 Capteurs de Niveau Numériques	26
II.2.3 Utilisé dans l'équation	29
II.2.4 Vanne proportionnelle	29
II.2.5 Types Vanne proportionnelle	29
II.2.5.1 Electrovanne proportionnelles	30
II.2.5.2 Vannes proportionnelles hydrauliques	30
II.2.5.3 Vannes proportionnelles pneumatiques	31
II.2.6 Commande d'une vanne proportionnelle en analogique (%)	31
II.2.7 Relation entre le signal de commande et l'ouverture de la vanne	32
II.3 Interaction avec L'HMI	32
II.3.1 Affichage des données du capteur	32
II.3.2 Contrôle manuel à travers les panneaux IHM.	33
II.3.3 Schéma fonctionnel	34
II.4 Principe de fonctionnement	34
II.4.1 Contrôle proportionnel de la vanne selon le niveau mesuré	34
II.4.2 Interface Homme-Machine (HMI)	35
II.4.2.1 Visualisation en temps réel	35
II.4.2.2 Alarmes et seuils de sécurité	35
II.4.2.3 Contrôle manuel	36

II.5 Programmation du système et de l'HMI	37
II.5.1 Lecture du signal du capteur analogique	37
II.5.2 Commande proportionnelle de la vanne	38
II.6 Développement de l'interface HMI	39
II.6.1 Conception de l'interface graphique WinCC ou équivalent	39
II.7 Conclusion	41

Chapitre III : Simulation et tests du système avec HMI

III.1 Introduction	43
III .2 Le cahier de charge	43
III .3 Configuration de l'appareil	44
III .4 Création de la table des variables	45
III .4.1 La table des variables	46
III .5 Création du programme	47
III .5.1 block_1[FC1]	48
III .5.2 Block Data [DB1]	50
III .5.3 Block OB1	51
III .6 Compilation et simulation du programme	54
III .7 Création de la supervision	56
III.7.1 Etablissement et liaison d'une HMI	56
III.7.2 Création d'interfaces HMI	56
III .7.3 Les différentes vues du projet	56
III .7.3.1 Vue de page principale (HOME)	56
III .8 Simulation pour WinCC	64
III .9 Conclusion	68

Liste des figures

Chapitre 1 : Contrôle du niveau d'eau dans un réservoir

Figure I.1 : Mesures de niveau ponctuelle et continue	6
Figure I.2 : Automate programmable industrie	10
Figure I.3 : API SIMENS S7-200	11
Figure I.4 : API SIMENS S7-400	12
Figure I.5 : API SIMENS S7-1200	12
Figure I.6 : API SIMENS S7-1500	13
Figure I.7 : Automate S7-1200- Siemens.	14
Figure I.8 : Possibilités d'extension de la CPU	15
Figure I.9 : Module d'entrée/sortie TOR	16
Figure I.10 : Module d'entrée/sortie analogique	17
Figure I.11 : Outil de simulation PLCSIM	17
Figure I.12 : Interface Homme-Machine	18
Figure I.13 : Fenêtre du travail dans le WinCC	19
Figure I.14 : Vue du TIA portal.	21
Figure I.15 : Vue du projet	22

Chapitre 2 : Conception du système de contrôle avec HMI

Figure II.1 : Capteur de niveau	25
Figure II.2 : Capteur de niveau analogique	26
Figure II.3 : Capteur de Niveau Numérique	27
Figure II.4 : Électrovannes proportionnelles	30
Figure II.5 : Vannes proportionnelles hydrauliques	30
Figure II.6 : Vannes proportionnelles pneumatiques	31
Figure II.7 : Schéma fonctionnel	34
Figure II.8 : Lecture et conversion d'un signal analogique	38
Figure II.9 : Commande proportionnelle de la vanne	39

Chapitre 3 : Simulation et tests du Système avec HMI

Figure III.1 : Configuration de CPU.	44
Figure III.2 : Configuration de HMI	45
Figure III.3 : Déclaration des variables	46
Figure III.4 : Tableau des variables	47
Figure III.5 : la programme dansFC1 (SCL)	49
Figure III.6 : la réseau dansFC1	50
Figure III.7 : Data block	50
Figure III.8 : réseau de part cycle	51
Figure III.9 : réseau scale niveau	51
Figure III.10 : réseau de la pransipe de l'ouverture de vanne	52
Figure III.11 : réseau de la valeur de l'ouverture de vanne	52
Figure III.12 : réseau de l'état normal	52
Figure III.13 : réseau d'alarme de fuite	53
Figure III.14 : réseau d'alarme de faut	53
Figure III.15 : réseau d'alarme de surveiller	53
Figure III.16 : Barre de la simulation de TIA portal.	54
Figure III.17 : Compilation de programme sans erreur.	54
Figure III.18 : Chargement de programme	55
Figure III.19 : Indication de la mise en ligne	55
Figure III.20 : Liaison PLC_HMI.	56
Figure III.21 : Vue de page principale (HOME)	57
Figure III.21 : Vue de cuve	57
Figure III.22 : Vue de courbe	58
Figure III.23 : Simulation de réseau de part cycle	58
Figure III.23 : Simulation de réseau de scale niveau	58
Figure III.24 : Simulation de réseau de la pransipe de l'ouverture de vanne	59
Figure III.25 : Simulation de réseau de la valeur de l'ouverture de vanne	60
Figure III.26 : Simulation de réseau d'état normal	60
Figure III.27 : Simulation de réseau d'alarme de fuite	60
Figure III.28 : Simulation de réseau d'alarme de défaut	61
Figure III.28 : Simulation de réseau d'alarme de surveiller	61
Figure III.29 : Simulation de Bock_1[FC1]	63
Figure III.30 : Simulation de Data_block_1[DB1]	63
Figure III.31 : représente cas 1	64
Figure III.32 : représente cas 2	65

Figure III.33 : représente cas 3	66
Figure III.34 : représente la courbe de la cas 1	66
Figure III.35 : représente la courbe de la cas 2	67
Figure III.36 : représente la courbe de la cas 3	67

Introduction générale

Introduction générale

Les réservoirs jouent un rôle essentiel dans la majorité des installations industrielles, notamment les réservoirs d'eau, car cette dernière entre dans la composition de nombreux processus de fabrication. Dans cette optique, nous avons développé un système de contrôle d'une installation de stockage d'eau, dont l'objectif principal est de surveiller et réguler le niveau d'eau de manière fiable. Pour cela, nous avons conçu une solution automatisée basée sur une interface homme-machine (IHM), permettant à l'opérateur de superviser et d'interagir efficacement avec le système.

Dans le cadre de ce projet nous avons conçu une solution complète permettant de surveiller, réguler et superviser en temps réel le niveau d'eau, en s'appuyant sur une interface homme-machine (IHM) conviviale et une automatisation avancée à l'aide d'un automate programmable Siemens S7-1200. L'objectif principal est de développer un système fiable, flexible et précis, capable de maintenir automatiquement le niveau d'eau à des valeurs souhaitées via une commande proportionnelle d'une vanne.

Ce mémoire est structuré autour de trois chapitres principaux qui décrivent les fondements techniques et pratiques de notre projet de fin d'études :

- Le premier chapitre, introduit les principes de base de la régulation du niveau d'eau, les besoins techniques et industriels qui en découlent, ainsi que les composants utilisés. Il présente également l'automate Siemens S7-1200, les technologies associées, et les outils logiciels comme TIA Portal, WinCC et PLCSIM.
- Le second chapitre, se concentre sur la réalisation concrète du système. Il aborde la description des composants matériels (capteur analogique, vanne proportionnelle, interface HMI), l'architecture fonctionnelle du système et la programmation du PLC et de l'IHM. L'accent est mis sur la lecture du signal analogique, la conversion en pourcentage, la régulation proportionnelle de la vanne et l'affichage dynamique sur l'IHM, avec possibilité d'ajustement des seuils et visualisation des alarmes.
- Le troisième et dernier chapitre se concentre sur la mise en œuvre pratique, avec le développement du programme automate, la création des interfaces HMI, et les simulations menées pour vérifier le bon fonctionnement du système. Elle présente également les différentes étapes de test, les résultats obtenus, et les vues principales de la supervision.

Chapitre 1

Contrôle du niveau d'eau dans un réservoir

1.1 Introduction

Dans un contexte d'industrialisation croissante et d'automatisation des procédés, la gestion intelligente des ressources, notamment l'eau, devient un enjeu majeur. Le contrôle du niveau d'eau dans un réservoir constitue une opération essentielle pour garantir la stabilité des processus, la protection des équipements et la qualité de la production.

Ce projet vise à développer un système automatique complet permettant de mesurer en continu le niveau d'eau à l'aide d'un capteur analogique, tout en régulant dynamiquement une vanne proportionnelle afin de maintenir ce niveau dans une plage prédéfinie. Pour assurer une supervision efficace et une interaction intuitive avec l'utilisateur, une interface homme-machine (HMI) a été intégrée. Celle-ci permet de visualiser en temps réel l'état du réservoir et d'agir sur les paramètres du système. Le développement s'appuie sur l'environnement TIA Portal, en complément d'outils de simulation PLCSIM, offrant ainsi une plateforme optimale pour le prototypage et la validation.

1.2 Présentation du besoin de régulation du niveau d'eau dans un réservoir

La régulation du niveau d'eau dans un réservoir est un enjeu fondamental dans de nombreux domaines industriels et domestiques. Elle permet de garantir un approvisionnement en eau constant, d'optimiser l'efficacité énergétique et de prévenir les risques liés aux variations de niveau.

1.2.1 Pourquoi réguler le niveau d'eau ?

- **Éviter le débordement** : Un niveau trop élevé peut entraîner des fuites, des dommages matériels ou une perte d'eau.
- **Prévenir les pénuries** : Un niveau trop bas peut causer des interruptions dans les processus industriels ou l'alimentation en eau.
- **Optimiser la consommation d'énergie** : Une régulation efficace permet d'éviter les cycles de pompage inutiles.
- **Garantir la stabilité des processus industriels** : Dans certaines applications (ex : traitement des eaux, refroidissement industriel), un niveau constant est essentiel pour assurer le bon fonctionnement des équipements.

1.2.2 Techniques de régulation du niveau d'eau

Il existe plusieurs approches pour contrôler le niveau d'eau dans un réservoir :

- **Régulation manuelle** : Intervention humaine pour ouvrir/fermer la vanne ou activer la pompe (peu efficace).
- **Régulation ON/OFF** : Utilisation d'un capteur pour activer ou désactiver une pompe ou une vanne (méthode simple mais avec risque d'oscillations).
- **Régulation proportionnelle (PID)** : Un contrôleur ajuste l'ouverture de la vanne ou la vitesse de la pompe en fonction de l'écart entre la consigne et le niveau mesuré (solution plus précise et stable).
- **Automatisation avec PLC (Automate Programmable Industriel)** : Utilisation de capteurs et d'un automate programmable pour assurer un contrôle précis et dynamique.

1.2.3 Applications Pratiques

a. Réservoirs Industriels

Les réservoirs industriels bénéficient d'une régulation automatique, permettant un contrôle précis des niveaux de liquides, optimisant ainsi la production et réduisant les risques de débordement. Ces systèmes sont essentiels dans les processus de fabrication impliquant des fluides sensibles à la température et à la pression.

b. Réservoirs pour l'Eau Potable

Pour l'approvisionnement en eau potable, la régulation automatique garantit un niveau constant, assurant ainsi la qualité et la disponibilité de l'eau. Cela permet également de réduire le gaspillage d'eau et d'assurer un service continu, même en période de forte demande.

c. Gestion des Eaux Pluviales

La gestion des eaux pluviales nécessite souvent des systèmes de régulation pour éviter les inondations. La régulation automatique permet l'évacuation contrôlée et efficace des eaux de pluie, protégeant ainsi les infrastructures et l'environnement.[1]

1.3 Surveillance continue du niveau par capteur analogique

L'utilisation d'un capteur de niveau analogique pour la surveillance continue permet de mesurer en temps réel le niveau d'un liquide dans un réservoir, offrant ainsi une gestion précise et efficace des processus industriels. Ces capteurs fournissent une sortie analogique proportionnelle au niveau mesuré, ce qui facilite l'intégration avec des systèmes de contrôle pour automatiser les actions en fonction des variations de niveau.

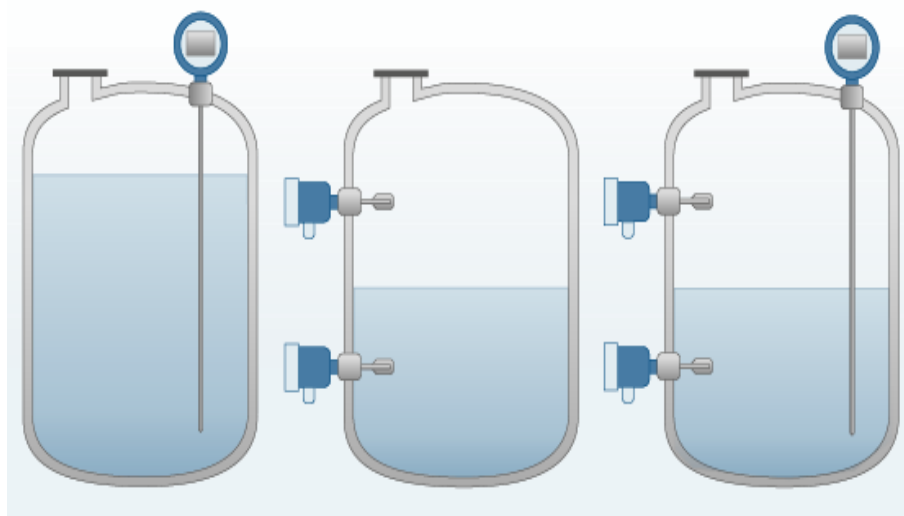


Figure I.1 : Mesures de niveau ponctuelle et continue.

1.4 Importance d'une commande proportionnelle de la vanne pour maintenir le niveau

Une commande proportionnelle de la vanne est essentielle pour maintenir un niveau de liquide stable dans un réservoir, en évitant les fluctuations excessives et en améliorant l'efficacité du système. Voici les principales raisons de son importance :

1. Réponse graduelle et précise

Contrairement aux vannes tout ou rien (ON/OFF), une vanne à commande proportionnelle permet d'ajuster finement le débit en fonction des besoins réels du système. Cela empêche les variations brusques du niveau et réduit les oscillations.

2. Réduction des phénomènes de dépassement et d'instabilité

Un contrôle trop agressif (tout ou rien) peut entraîner des phénomènes de dépassement où le niveau dépasse la consigne avant de redescendre. Avec une commande proportionnelle, l'ajustement est plus doux et progressif, ce qui assure une meilleure stabilité.

3. Amélioration de l'efficacité énergétique

Un contrôle proportionnel optimise l'ouverture de la vanne pour limiter la consommation d'énergie en évitant des cycles répétés d'ouverture et de fermeture qui pourraient solliciter inutilement les actionneurs et les pompes.

4. Réduction de l'usure des équipements

Moins de variations brutales signifie une sollicitation mécanique réduite, prolongeant ainsi la durée de vie des vannes, des actionneurs et des autres composants hydrauliques.

5. Meilleure adaptation aux variations de débit

Un système avec commande proportionnelle peut mieux gérer des variations imprévues du débit d'entrée ou de sortie, assurant un maintien plus précis du niveau souhaité.[2]

1.5 Objectifs de l'intégration d'une HMI pour la supervision en temps réel

L'intégration d'une Interface Homme-Machine (HMI) pour la supervision en temps réel vise plusieurs objectifs essentiels dans les systèmes industriels :

1. **Amélioration de la surveillance et du contrôle** : Une HMI fournit une visualisation en temps réel des processus, permettant aux opérateurs de surveiller et de contrôler efficacement les opérations.
2. **Réduction des temps d'arrêt** : En offrant des alertes immédiates en cas d'anomalies, l'HMI facilite une intervention rapide, minimisant ainsi les interruptions de production.
3. **Optimisation de la productivité** : Grâce à une interface intuitive, les opérateurs peuvent interagir plus efficacement avec le système, ce qui améliore les performances globales.
4. **Collecte et analyse des données** : L'HMI permet de recueillir des données en temps réel, facilitant l'analyse des tendances et la prise de décisions éclairées pour optimiser les processus.[3]

1.6 Objectifs du projet

- Mesurer et contrôler le niveau d'eau dans un réservoir à l'aide d'un capteur analogique, en traitant les signaux à l'aide d'un automate programmable industriel (API) pour afficher le niveau en pourcentage de manière précise et continue.
- Réguler automatiquement la position d'une vanne proportionnelle afin d'ajuster le débit d'eau entrant ou sortant, en tenant compte des seuils définis et des consignes programmées, dans une logique de régulation en boucle fermée.
- Développer une interface homme-machine (HMI) intuitive, permettant à l'utilisateur :
- de visualiser en temps réel l'état du système (niveau d'eau, ouverture de la vanne, alarmes éventuelles).
- de modifier les consignes de fonctionnement.

- de lancer ou arrêter le système manuellement en mode supervision.
- Assurer la sécurité et la fiabilité du système en intégrant des seuils de protection (niveau haut/bas critique), des temporisations, et un mode manuel de secours.
- Optimiser l'exploitation de l'installation en facilitant la maintenance, le diagnostic rapide des anomalies et la traçabilité des mesures (via un journal d'événements ou historique des valeurs).

1.7 Outils et technologies utilisées

1.7.1 Automate programmable (PLC)

Un automate programmable industriel (API) se présente comme un dispositif électronique offrant une mémoire programmable à l'utilisateur au moyen d'un langage spécifique. Son utilisation s'étend à diverses fonctions spécialisées, il est capable de stocker des instructions internes englobant une gamme variée de fonctions d'automatisation notamment :

- la logique séquentielle et combinatoire
- la temporisation
- le comptage, le décomptage et la comparaison
- le calcul arithmétique, le réglage, l'asservissement, la régulation, et bien d'autres.

L'objectif principal de l'automatisation est d'exécuter des tâches spécifiques et d'automatiser différents aspects des processus industriels. L'automate programmable est conçu pour être intégré dans un processus industriel et s'adapter à diverses situations, qu'il s'agisse de commandes humaines ou techniques. Son déploiement sur site et son utilisation continue nécessitent une grande adaptabilité. L'automate programmable est principalement chargé d'assurer la commande des procédés et de fournir des informations précieuses pour l'exploitation de la station.[4]



Figure I.2 : Automate programmable industrie

1.7.2 Critères de choix de l'automate programmable industriel

Après avoir étudié notre système dans les chapitres précédent, 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.
- La nature Analogiques d'entrées/sorties [5]

1.7.3 Présentation de quelques gammes SIMATIC

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

I.7.3.1 SIMATIC S7

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

I.7.3.2 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 [5].



Figure I.3: API SIMENS S7-200

I.7.3.3 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 [5].



Figure I.4: API SIMENS S7-400

I.7.3.4 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 [5].



Figure I.5: API SIMENS S7-1200

I.7.3.5 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 [5].



Figure I.6: API SIMENS S7-1500

I.7.4 Présentation l'automate utilise dans ce projet (S7-1200 Siemens)

I.7.4.1 Description

L'automate SIMATIC S7-1200 est fabriqué par SIEMENS et offre la flexibilité et la puissance nécessaires pour contrôler une large gamme d'appareils afin de répondre à vos besoins d'automatisation. Sa forme compacte, sa configuration flexible et son grand jeu d'instructions le rendent idéal pour contrôler une variété d'applications. Le processeur combine un microprocesseur, une alimentation électrique intégrée, des circuits d'E/S, un PROFINET intégré, un contrôle d'E/S rapide et des entrées analogiques intégrées dans un boîtier compact pour créer une console puissante. Une fois le programme chargé, la CPU contient la logique requise pour surveiller et contrôler le matériel de votre application.

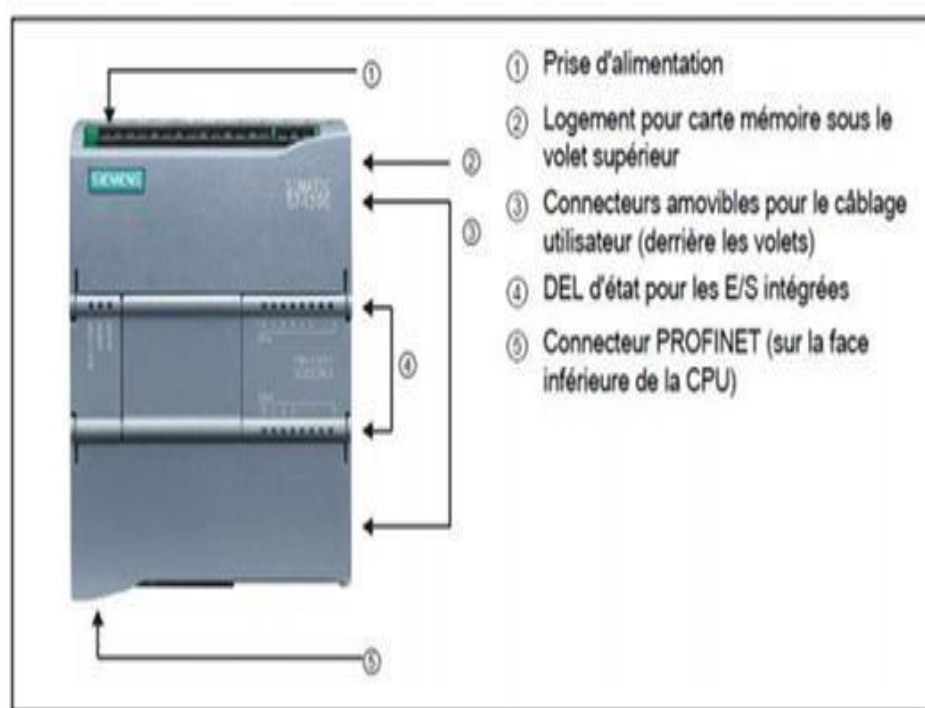


Figure I.7 : Automate S7-1200- Siemens. [6]

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).

I.7.4.2 Modules d'extensions

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

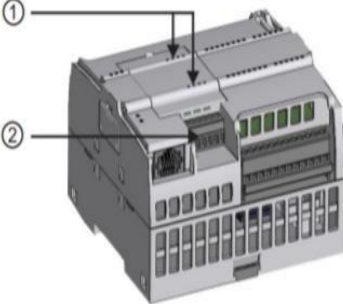
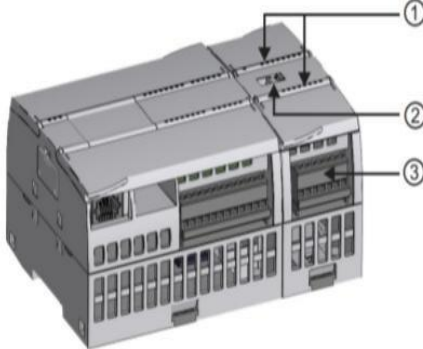
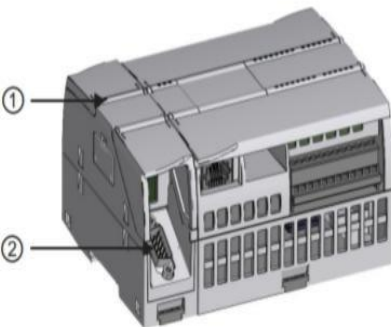
Type de module	Description		
La CPU prend en charge une carte d'extension enfichable : - Un Signal Board (SB) fournit des E/S supplémentaires pour votre CPU. Le SB se raccorde à l'avant de la CPU. - Un Communication Board (CB) vous permet d'ajouter un autre port de communication à votre CPU.		① DEL d'état sur le Signal Board ② Connecteur amovible pour le câblage utilisateur	
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. - E/S TOR - E/S analogiques - RTD et Thermocouple		① DEL d'état ② Connecteur de bus ③ Connecteur amovible pour le câblage utilisateur	
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. Un CP offre la possibilité d'autres types de communication, par exemple la connexion de la CPU par le biais d'un réseau GPRS. - La CPU accepte jusqu'à 3 CM ou CP. - Chaque CM ou CP se raccorde sur le côté gauche de la CPU (ou sur le côté gauche d'un autre CM ou CP).		① DEL d'état ② Connecteur de communication	

Figure I.8 : Possibilités d'extension de la CPU [6].

I.7.5 Les type des module entrée / sortie

I.7.5.1 Les modules d'entrée / sortie TOR

Le module d'entrées/sorties TOR constitue l'interface d'entrée et de sortie du signal TOR de l'automate. Ces modules permettent de connecter divers capteurs et actionneurs tout ou rien à l'automate S7-1200, en utilisant si nécessaire des dispositifs d'adaptation (régulation, conversion, etc.). [7]



Figure I.9 : Module d'entrée/sortie TOR

I.7.5.2 Module d'entrée/sortie analogique

Ces modules sont utilisés pour connecter des capteurs et des actionneurs analogiques à l'automate. Les modules d'entrées analogiques convertissent les signaux analogiques du processus en signaux numériques pour un traitement interne dans le S7-1200. Le module de sortie analogique convertit les signaux numériques internes en signaux analogiques pour simuler des actionneurs ou des pré-actionneurs. [7]



Figure I.10 : Module d'entrée/sortie analogique

I.8 Logiciels de simulation

I.8.1 PLCSIM

PLCSim est utilisé pour la simulation et les tests en temps réel dans le domaine de l'automatisation industrielle, améliorant la précision et la fiabilité des codes de contrôle, réduisant le temps de mise en service, permettant la formation dans un environnement sûr et améliorant l'efficacité des projets tout en minimisant les risques associés à la mise en œuvre du système.



Figure I.11: Outil de simulation PLCSIM

1.9 Interface Homme-Machine (IHM)

L'interface homme- machine (IHM) est l'interface utilisateur qui relie l'opérateur au dispositif de commande d'un système industriel. Dans notre projet, nous avons opté à partir des pupitres SIMATIC IHM pour la gamme TP1200 Confort, notre choix a été justifié selon les fonctionnalités homogènes de haute performance, la haute résolution et l'aptitude aux environnements difficiles de ces pupitres.[3]



Figure I.12 : Interface Homme-Machine

1.9.1 SIMATIC WinCC

WinCC (Windows Control Center), est le logiciel qui permet de créer une Interface Homme Machine (HMI) 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.[8]

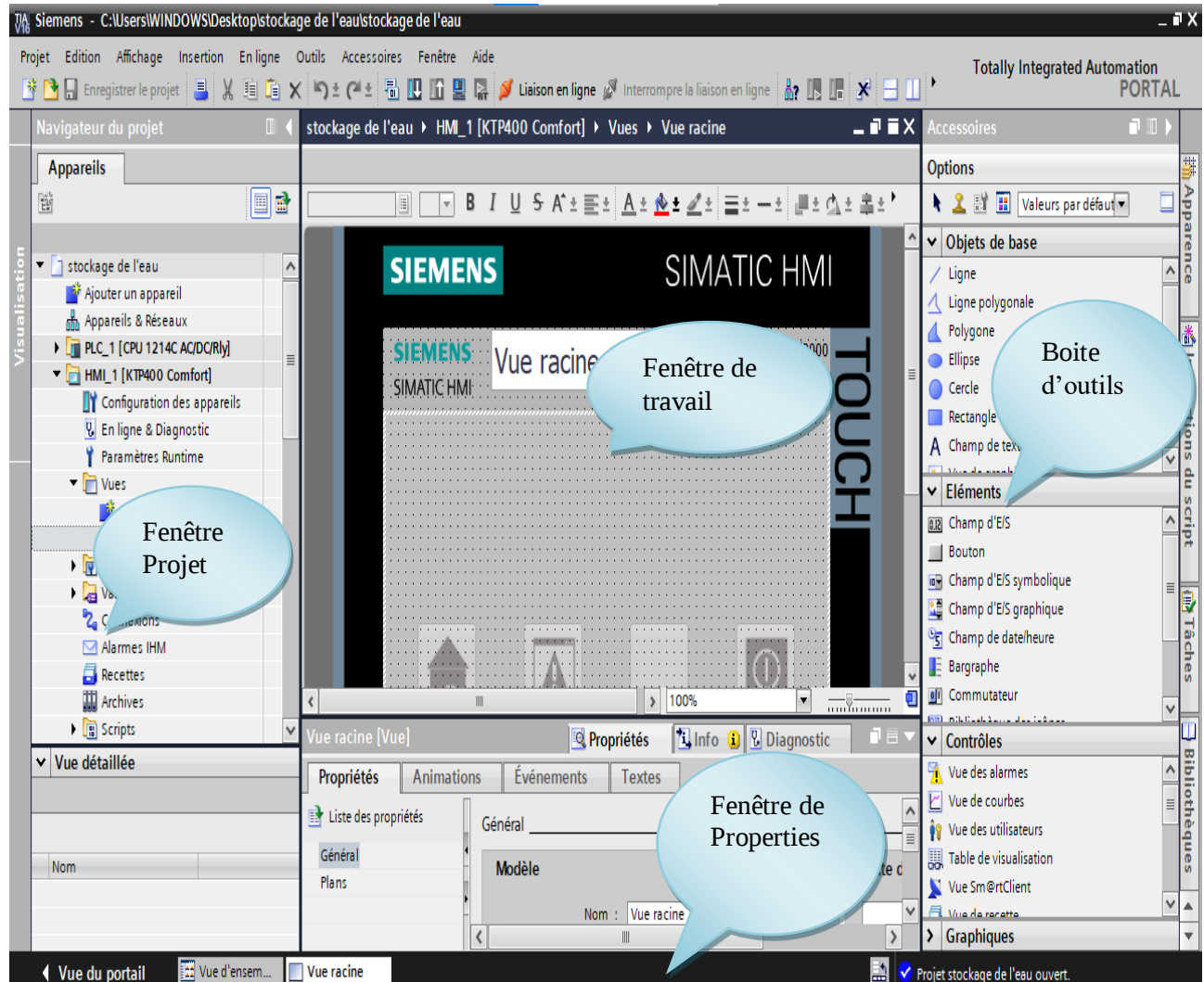


Figure I.13: Fenêtre du travail dans le WinCC.

Nous distinguons sur cette figure :

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.

WinCC gère les tâches suivantes :

- Représentation du processus : Le processus est représenté sur le pupitre opérateur. Si, par exemple, un changement intervient dans le processus, l'affichage est mis à jour sur le pupitre opérateur.
- Commande du processus : L'opérateur peut commander le processus via l'interface graphique. Par exemple, l'opérateur peut définir une consigne pour l'automate ou modifier des paramètres.
- Affichage d'alarmes : Si des états critiques surviennent dans le processus, une alarme se déclenche automatiquement. Par exemple, quand une limite fixée est dépassée.
- Archivage des valeurs de processus et des alarmes : Le système HMI peut archiver des alarmes et des valeurs de processus. Cela nous permet de documenter les caractéristiques du processus ou d'accéder ultérieurement à des données de production plus anciennes.
- Documentation des valeurs et des alarmes : Le système HMI affiche les alarmes et les valeurs de processus sous forme de protocole. Nous pouvons ainsi afficher les données de production à chaque changement d'équipe.
- Gestion des paramètres du processus et des machines : Le système HMI peut enregistrer les paramètres de processus et des machines dans des recettes. Cela nous permet de transférer ces paramètres en une seule fois à l'automate.[8]

I.10 Vue du portail et vue du projet

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

La vue du portail : elle est axée sur les tâches à exécuter et sa prise en main est très rapide.

La vue du projet : elle comporte une arborescence avec les différents éléments du projet, les éditeurs requis s'ouvrent en fonction des tâches à réaliser : données, paramètres et éditeurs, Ils peuvent être visualisés dans une seule et même vue.

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, la figure ci-dessous représente une vue du portail.

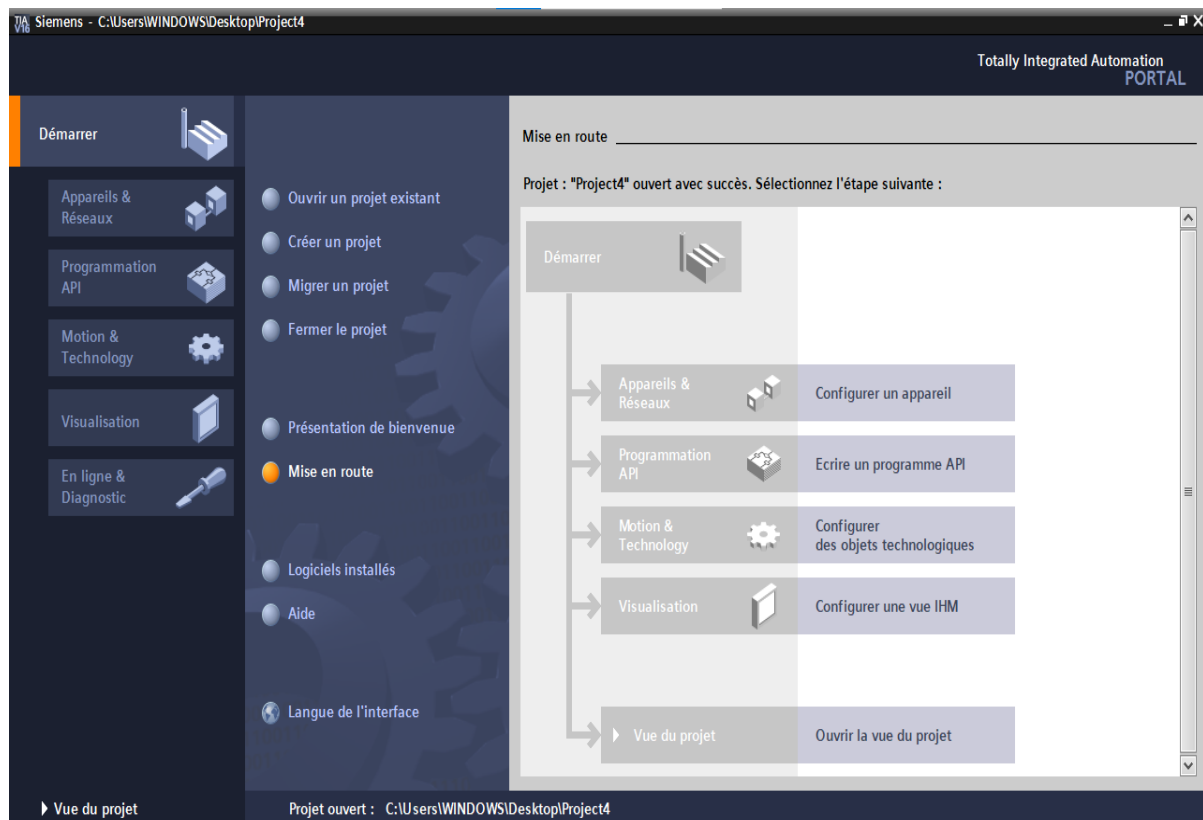


Figure I.14 : Vue du TIA portal.

Vue du projet :

L'élément « Projet » contient l'ensemble des éléments et des données nécessaires pour mettre en œuvre la solution d'automatisation souhaitée, la figure ci-dessous représente la vue du projet.

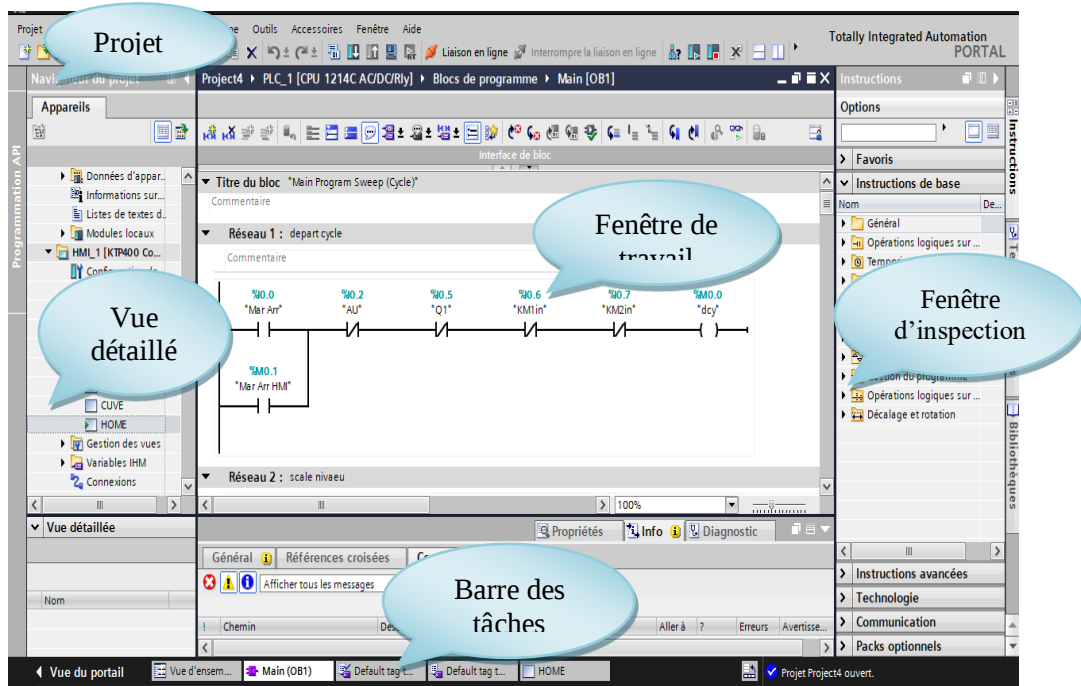


Figure I.15 : Vue du projet

- **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 interfaces homme machine (IHM)
- **La fenêtre d'inspection permet** de visualiser des informations complémentaires sur un objet sélectionné où sur les actions en cours d'exécution (propriété du matériel sélectionné, message d'erreur lors de la compilation des blocs de programme...).
- **Les onglets de sélection** de tâches ont un contenu qui varie en fonction de l'objet sélectionné (configuration matérielle → bibliothèques des composants, bloc de programme → instructions de programmation). Cet environnement de travail contient énormément de données. Il est possible de masquer ou réduire certaines de ces fenêtres lorsque l'on ne les utilise pas. Il est également possible de redimensionner, réorganiser, désancrer les différentes fenêtres.

I.11 Conclusion

Ce chapitre a permis de poser les fondations du projet de régulation du niveau d'eau, en identifiant les besoins fonctionnels, les objectifs techniques ainsi que les outils utilisés. Le système proposé repose sur une combinaison efficace entre mesure analogique, régulation proportionnelle et interface de supervision, assurant un contrôle fiable et adaptable du niveau d'eau.

Ce projet illustre concrètement l'intérêt de l'intégration entre les technologies d'automatisation et les interfaces utilisateurs dans le développement de solutions industrielles performantes. Le chapitre suivant détaillera la mise en œuvre technique du système, depuis la programmation de l'automate (PLC), en passant par la configuration du capteur et de la vanne, jusqu'à la conception et le déploiement de l'HMI.

Chapitre 2

Conception du système de contrôle avec HMI

II.1 Introduction

Dans le cadre de la mise en œuvre d'un système automatisé de gestion du niveau d'eau, la conception d'une interface de contrôle efficace constitue une étape essentielle. Ce chapitre présente en détail les composants principaux du système, notamment le capteur de niveau analogique, la vanne proportionnelle et l'interface HMI (Human Machine Interface). Il explique le fonctionnement du système dans son ensemble, en mettant l'accent sur le flux d'informations entre les différents éléments, le rôle du PLC dans le traitement des données, ainsi que la manière dont l'utilisateur peut interagir avec l'installation via l'HMI. Une attention particulière est accordée à la représentation graphique des données, à la régulation proportionnelle de la vanne en fonction du niveau mesuré, et à la configuration des seuils et alarmes. La programmation du PLC et de l'IHM est également abordée pour illustrer la logique de contrôle et la visualisation en temps réel du processus.

II.2 Description des composants

II.2..2 Capteur de niveau

Dans l'industrie, les capteurs de niveau sont utilisés pour mesurer le niveau d'un solide, liquide ou gaz dans une cuve, silo ou réservoir. Les capteurs de niveau peuvent être analogiques (capteur de niveau ultrasonique, hydrostatique) ou digitaux (capteur de niveau capacitif). Ainsi, on rencontre plusieurs technologies de capteurs de niveau parmi lesquelles on peut citer : les capteurs de niveau ultrasoniques, les capteurs de niveau capacitifs, les capteurs de niveau hydrostatiques, les capteurs de niveau à flotteur. [9]



Figure II.1: Capteur de niveau

II.2.2 Les types de capture de niveau

II.2.2.1 Capteurs de niveau analogiques

- Ils génèrent un signal constant (4-20mA ou 0-10V) qui illustre le niveau de liquide dans le conteneur.
- Ils sont employés dans les systèmes de contrôle automatique pour mesurer de manière précise et mettre en œuvre une gestion graduelle des vannes ou des pompes.



Figure II.2: Capteur de niveau analogique

Les types de capture analogique :

- Capteur de pression hydrostatique (Capteur Hydrostatique de Pression).
- Capteur à ultrasons (Capteur à Ultrason).
- Capteur capacitif (Capteur de type capacitif).

II.2.2.2 Capteurs de Niveau Numériques

- Ils ne délivrent qu'un signal de type ON/OFF dès que le niveau atteint une limite déterminée.
- Ils ne servent pas à mesurer un niveau graduel, mais sont couramment employés comme avertisseur de niveau élevé ou faible.



Figure II.3: Capteur de Niveau Numérique

Le capteur utilisé dans ce système

D'après les spécifications requises, un capteur de niveau analogique (Analogique) est nécessaire pour fournir une mesure progressive du niveau d'eau, car la vanne doit être régulée en fonction d'un signal continu (0-10V ou 4-20mA).

Principe de fonctionnement

Équation utilisée :

$$\text{Ouverture de la vanne} = (100 - \text{Niveau d'eau}) \times \frac{100}{30}$$

Où :

- Niveau d'eau est le pourcentage du niveau d'eau (entre 70% et 100%).
- 30 est la différence entre 100% et 70%.

Valeurs demandées :

- **Lorsque le niveau d'eau = 0% :**

$$\text{Ouverture de la vanne} = (100 - 0) \times \frac{100}{30} = 1000\%$$

Cela implique que l'équation ne reflète pas directement les valeurs de 0% à 70% dans cette situation, étant donné que nos calculs ont démarré à partir de 70% dans le cahier des charges. Autrement dit, l'équation que nous avons fournie est valide uniquement pour les valeurs comprises entre 70% et 100%.

- **Lorsque le niveau d'eau = 70% :**

$$\text{Ouverture de la vanne} = (100 - 70) \times \frac{100}{30} = 100\%$$

Ainsi, la vanne est complètement ouverte lorsque le niveau d'eau est de 70%

- **Lorsque le niveau d'eau = 100% :**

$$\text{Ouverture de la vanne} = (100 - 100) \times \frac{100}{30} = 0\%$$

Lorsque le niveau d'eau est de 100%, la vanne est complètement fermée.

Résultat :

- **À 70% :** la vanne est complètement ouverte (**100%**).
- **À 100% :** la vanne est complètement fermée (**0%**).

Dans ce contexte, la valeur de 0% n'est pas appropriée à l'équation, étant donné que nous entamons le calcul à partir de 70%. Donc, pour une plage de 0% à 70%, la vanne reste entièrement ouverte. Dans l'équation, le chiffre 30 représente la distinction entre 100% et 70%, en rapport avec la gestion de l'ouverture de la vanne. Interprétation du nombre 30%Le cahier des charges exige que la vérification de la vanne soit effectuée entre 70% et 100% du niveau d'eau. Autrement dit :

- Si le niveau de l'eau atteint 70%, la vanne doit être totalement ouverte (100%).
- Lorsque le niveau d'eau atteint 100%, la vanne doit être entièrement fermée (0%).

La différence entre 100% et 70% s'élève à 30%

II.2.3 Utilisé dans l'équation

Comment cela est appliqué dans l'équation

- L'objectif est de déterminer le taux d'ouverture de la vanne lorsque le niveau d'eau est compris entre 70 % et 100 %.
- 30 est utilisé comme dénominateur pour traduire la différence entre les niveaux souhaité et actuel en pourcentages d'ouverture de la vanne, car il représente la différence entre 70% et 100%.
- Le niveau d'eau de 100 montre la différence entre 100% et le niveau d'eau actuel.
- Le nombre 30 est utilisé comme dénominateur pour convertir la différence entre 100% et 70% en pourcentage d'ouverture de la vanne.

II.2.4 Vanne proportionnelle

Une vanne proportionnelle ou bien vanne de régulation, le passage peut être modulé progressivement. Elle permet une régulation précise des pressions sur différents points du système.[11]

II.2.5 Types Vanne proportionnelle

II.2.5.1 Électrovannes proportionnelles

Les électrovannes proportionnelles peuvent être ouvertes avec plus ou moins d'amplitude. Selon les types de vannes l'ouverture peut être proportionnelle au courant électrique de L'alimentation, ou à la tension électrique de l'alimentation. Ce type d'électrovanne est généralement piloté par l'intermédiaire d'une commande.[10]



Figure II.4: Électrovannes proportionnelles

II.2.5.2 Vannes proportionnelles hydrauliques

Sont des composants essentiels dans les systèmes hydrauliques modernes, permettant un contrôle précis et continu du débit, de la pression et de la direction du fluide. Elles sont largement utilisées dans divers secteurs industriels pour améliorer la performance, la sécurité et l'efficacité des machines.



Figure II.5: Vannes proportionnelles hydrauliques

II.2.5.3 Vannes proportionnelles pneumatiques

Sont des dispositifs essentiels dans les systèmes de contrôle automatisés, permettant une régulation précise et continue du débit ou de la pression de l'air comprimé. Elles sont largement utilisées dans diverses industries pour optimiser les performances des processus.



Figure II.6: Vannes proportionnelles pneumatiques

II.2.6 Commande d'une vanne proportionnelle en analogique (%)

- Un signal de commande analogique exprimé en pourcentage d'ouverture peut être utilisé pour contrôler progressivement un débit de fluide avec une vanne proportionnelle.

1. Principe de fonctionnement

- Le degré d'ouverture d'une vanne proportionnelle est contrôlé par un signal analogique (0-10V, 4-20mA, ou PWM). Selon le type de vanne, la relation entre l'ouverture et le débit peut être linéaire ou non linéaire.

2. Modification de la dette avec une directive comparable

- Le signal et l'ouverture sont directement corrélés dans une vanne linéaire.
- Une vanne logarithmique ou exponentielle donne un résultat différent (bonne pour les systèmes qui nécessitent un contrôle plus précis en bas du débit).

3. Application dans un système de remplissage d'eau

Si un réservoir est en train de se remplir avec la vanne :

- Un capteur de niveau mesure le niveau d'eau.
- La contrainte sur le réseau et l'ouverture de la vanne déterminent le débit entrant.

II.2.7 Relation entre le signal de commande et l'ouverture de la vanne

La vanne proportionnelle est contrôlée par un signal analogique 0-10V ou 4-20mA, représentant le pourcentage d'ouverture de la vanne :

- 0% du signal (0V ou 4mA) → valve intégralement fermée (pas de flux).
- 70% du signal (5V ou 12mA) → vanne ouverte à moitié (débit moyen).
- 100% du signal (10V ou 20mA) → valve entièrement ouverte (débit à son maximum).

Le lien entre le signal et l'ouverture peut être soit linéaire (si la soupape est prévue pour un contrôle direct), soit non linéaire (exponentiel ou logarithmique en fonction du genre de soupape).

II.3 Interaction avec L'HMI

II.3.1 Affichage des données du capteur

a) Présentation des informations du capteur

La présentation des informations provenant du capteur de niveau est une phase cruciale dans chaque système de suivi et de gestion. Elle offre non seulement aux superviseurs la possibilité d'observer en temps réel le niveau d'eau dans le réservoir, mais elle est également cruciale pour l'initiation manuelle ou automatique des opérations de régulation (remplissage ou vidage).

b) But de l'affichage

La mission première de l'affichage consiste à convertir les signaux analogiques bruts issus du capteur en des données intelligibles pour l'opérateur. Il est nécessaire que cette information soit :

Spécifiez : l'affichage du niveau d'eau doit se faire en unités physiques (litres, pourcentage ou hauteur en centimètres).

Mise à jour en continu : pour capturer toute fluctuation de niveau sans délai.

Accessible : via un écran local (HMI) ou à distance.

c) Modification et administration des données

En général, les capteurs analogiques délivrent une tension (0-10 V) ou un courant (4-20 mA) en relation directe avec la mesure réalisée. Pour que ces informations puissent être présentées, elles doivent être :

Effectuées par le Convertisseur Analogique-Numérique (ADC) intégré dans le module d'entrée analogique du PLC.

Normalisées en employant la fonction NORM_X afin d'obtenir un chiffre décimal compris entre 0 et 1. Dimensionnement réalisé via SCALE_X pour obtenir une mesure physique tangible (par exemple : de 0 à 100 cm).

Préparées (arrondies, ajout d'unité, et c.) pour une présentation conviviale.

d) Méthodes d'affichage

L'interface homme-machine est l'instrument majeur d'affichage. On la retrouve fréquemment dans l'armoire électrique en tant qu'écran tactile Siemens (KTP, TP). Elle présente :

L'état actuel (par exemple : 62 % ou 124 cm).

Les niveaux critiques (haut/bas).

Des signaux visuels ou acoustiques.

L'état opérationnel du système (remplissage, vidange, veille).

II.3.2 Contrôle manuel à travers les panneaux IHM.

En plus de la surveillance automatisée, les panneaux IHM donnent aux opérateurs l'opportunité d'intervenir manuellement sur les systèmes. Cette capacité est cruciale pour réaliser des modifications, des diagnostics ou des interventions particulières lorsque c'est requis. Les interfaces sont pensées pour être intuitives, facilitant une interaction aisée même dans des conditions de travail difficiles.

L'interaction manuelle grâce aux interfaces homme-machine procure une souplesse et un niveau de sûreté additionnels, en proposant aux opérateurs un moyen direct pour piloter et modifier les systèmes en temps réel.

II.3.3 Schéma fonctionnel

Cette figure représente Diagramme du flux de données entre le capteur, le PLC, la vanne et l'HMI.

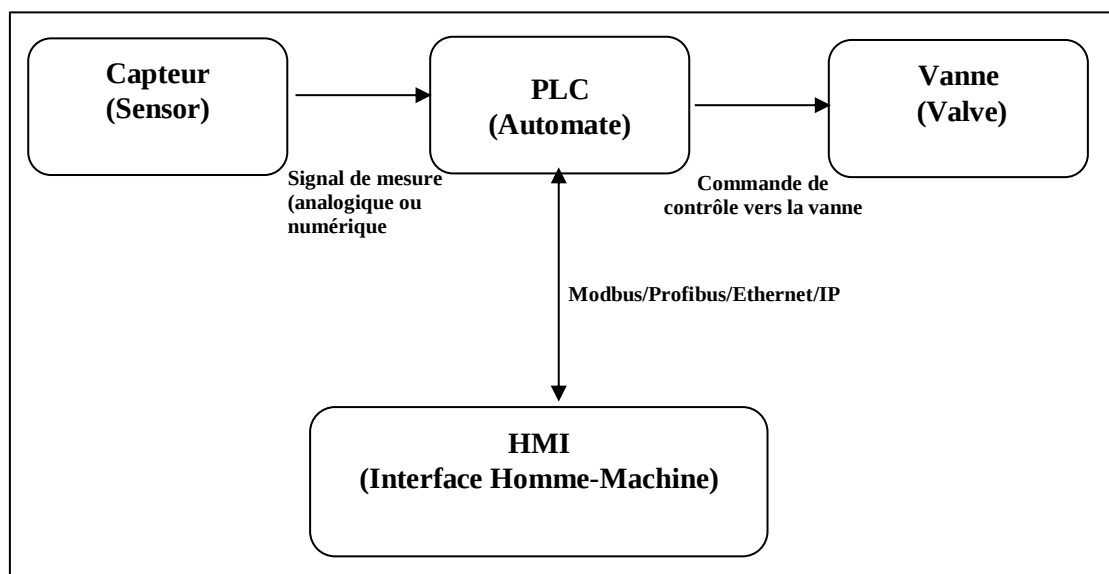


Figure II.7: Schéma fonctionnel

II.4 Principe de fonctionnement

II.4.1 Contrôle proportionnel de la vanne selon le niveau mesuré.

Dans un système de régulation proportionnelle, la vanne ne s'ouvre pas simplement en tout ou rien. Elle s'ajuste progressivement en fonction du niveau mesuré par le capteur. Plus l'écart entre le niveau réel et la consigne est important, plus l'ouverture de la vanne augmente, et inversement. Ce type de régulation permet une réponse souple et continue, ce qui évite les variations brusques du niveau et garantit une meilleure stabilité du système.[11]

II.4.2.1 Interface Homme-Machine (HMI)

L'Interface Homme-Machine (HMI) constitue un élément essentiel dans un système automatisé de gestion d'unité de stockage de l'eau. Elle assure la communication entre l'opérateur et le système de contrôle, en fournissant à la fois une visualisation claire des données et des moyens d'intervention directe. Dans notre projet, l'HMI est conçue pour être à la fois intuitive, réactive et fonctionnelle.

II.4.2.1 Visualisation en temps réel

L'HMI permet à l'utilisateur de surveiller en permanence l'état du système :

- Niveau d'eau : Affiché sous forme numérique (en %) et graphique (jauge ou courbe), il est mis à jour en temps réel à partir des données du capteur analogique.
- Position de la vanne : Exprimée en pourcentage (0 % = fermée, 100 % = ouverte), elle reflète l'état d'ouverture de l'actionneur, contrôlé soit automatiquement, soit manuellement par l'opérateur.
- Une interface graphique dynamique permet de visualiser clairement les variations du niveau d'eau et les réactions du système en fonction des consignes.

II.4.2.2 Alarmes et seuils de sécurité

Un système de contrôle de confiance devrait incorporer des dispositifs d'alerte et de sécurité afin de sauvegarder les équipements, prévenir les débordements ou l'opération à vide.

Seuils de sécurité courants

Niveau très haut (NH) : risque de débordement → la vanne se ferme automatiquement ou la pompe s'arrête.

Niveau très bas (NB) : risque de marche à vide → arrêt automatique du remplissage ou de la vidange.

Erreur de capteur : signal hors des limites (par exemple, 3.5 mA ou 21 mA).

Erreur de contrôle : différence persistante trop importante entre la valeur cible et la mesure réelle.

Dans le PLC, ces seuils sont établis comme des valeurs limites et sont mis en comparaison avec la valeur mesurée en temps réel.

Erreur de régulation : écart trop grand persistant entre la consigne et la valeur réelle.

Ces seuils sont définis comme valeurs limites dans le PLC, et comparés en temps réel à la valeur mesurée.

Réactions aux alarmes

- Activation d'un voyant lumineux sur l'IHM.
- Envoi d'un signal sonore (buzzer).
- Blocage de l'actionneur (en sécurité).
- Message d'alerte texte affiché (ex. : "ALERTE : niveau trop bas !").

- Journalisation de l'événement avec horodatage

II.4.2.3 Contrôle manuel

L'interface HMI offre également des options de commande directe pour permettre à l'opérateur d'intervenir à tout moment :

- **Boutons de réglage des consignes** : Permettent de modifier en temps réel les valeurs de consigne du niveau d'eau (par exemple, fixer un seuil à 60 % au lieu de 50 %).
- **Commande manuelle de la vanne** : L'opérateur peut ouvrir ou fermer partiellement la vanne à l'aide de boutons ou d'un curseur, ce qui est particulièrement utile en mode maintenance ou en cas d'urgence.
- **Activation/désactivation du mode automatique** : L'utilisateur peut basculer entre contrôle manuel et automatique via l'HMI.

II.5 Programmation du système et de l'HMI

II.5.1 Lecture du signal du capteur analogique

- Les valeurs analogiques sont lues et produites sous forme de mots d'informations dans l'automate.
- Chaque valeur analogique ("canal") occupe un mot d'entrée de périphérie (ex : **IWxxx**) et est codée sous format **INT** (entier 16 bits).

2. Conversion en valeur physique

- Pour notre application, nous lisons un signal de **4 mA à 20 mA** avec un module d'entrées analogiques.
- La valeur lue initialement (INT) doit être convertie en format **REAL** (réel).

3. Normalisation de la valeur

- L'équation utilisée pour normaliser la lecture est :

$$B + \left((A - B) \times \left(\frac{\text{Converti en reel}}{27648} \right) \right) = MDxxx$$

Avec :

A : valeur maximale du capteur

B : valeur minimale du capteur

- Cette valeur normalisée est stockée dans un mot double memento (**MDxxx**).

4. Adressage mémoire dans un API Siemens

- Bits internes M0.0 à M255.7 dépendants des mots suivants
- Octets internes : ensemble de 8 bits MBi
- Mots internes : ensemble de 16 bits MWi
- Mots doubles : ensemble de 32 bits MDi

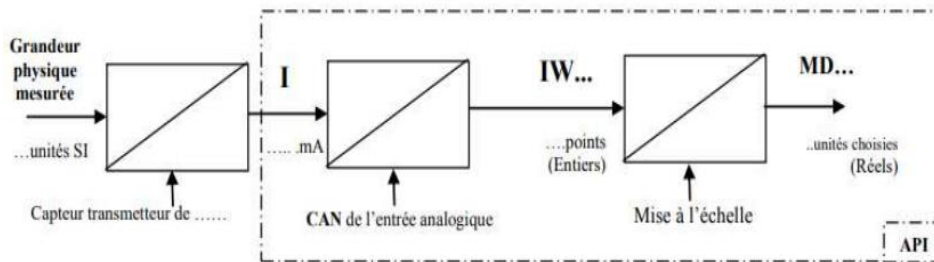


Figure II.8: Lecture et conversion d'un signal analogique

II.5.2 Commande proportionnelle de la vanne

- La valeur normalisée (stockée dans **MDxxx**) représente une grandeur physique réelle (niveau, pression, etc.).
- Pour commander une vanne proportionnelle, il faut transférer cette valeur à un module de **sortie analogique (QWxxx)**.

1. Conversion en signal analogique

- L'équation utilisée pour reconvertir la valeur normalisée est :

$$\left(27648 \times \frac{MDxxx - B}{A - B} \right) = QWxxx$$

- Cette opération permet de produire un signal compris entre **4 mA et 20 mA** ou entre **0 V et 10 V** selon la configuration.

2. Explication des blocs **NORM_X** et **SCALE_X**

- **NORM_X** :
Normalise une valeur d'entrée entre 0.0 et 1.0 sur une échelle linéaire basée sur les paramètres MIN et MAX.
- **SCALE_X** :
Met à l'échelle une valeur normalisée vers une plage spécifiée (exemple : 4–20 mA), résultat sous forme d'un entier.

3. Remarque importante

- Pour un module analogique Siemens standard : 0 – 10 V ↔ 0 – 27648 points
- Il est essentiel d'utiliser les bonnes plages de valeurs pour éviter les erreurs de commande.

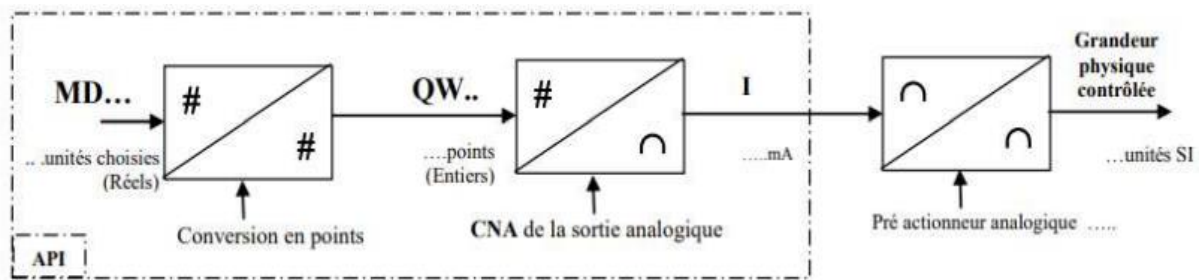


Figure II.9: Commande proportionnelle de la vanne

II.6 Développement de l'interface HMI

II.6.1 Conception de l'interface graphique WinCC ou équivalent

Étapes de conception avec WinCC Unified :

Création du projet : Dans TIA Portal, commencez par créer un nouveau projet. Ajoutez les dispositifs nécessaires, tels que les automates programmables (PLC) et les pupitres opérateurs (HMI), en configurant leurs paramètres matériels et de communication.

Définition des variables : Configurez les variables de processus (tags) qui serviront de lien entre les équipements physiques et l'interface graphique. Ces variables peuvent être importées directement depuis le PLC ou créées manuellement.

Conception des vues : Utilisez l'éditeur graphique pour concevoir les écrans de l'IHM. Intégrez des éléments tels que des boutons, des voyants, des jauges, des graphiques et des champs de texte. Chaque élément peut être lié à une variable spécifique pour refléter l'état du processus en temps réel.

Dynamisation des objets : Appliquez des animations et des comportements dynamiques aux objets graphiques. Par exemple, changez la couleur d'un voyant en fonction de l'état d'une machine ou affichez des alarmes lorsqu'une condition spécifique est remplie.

Gestion des alarmes et des messages : Configurez le système d'alarmes pour informer les opérateurs des anomalies ou des événements importants. Définissez les priorités, les conditions de déclenchement et les actions associées à chaque alarme.

Simulation et test : Avant le déploiement, utilisez les outils de simulation intégrés pour tester le comportement de l'IHM. Cela permet de vérifier la réactivité des éléments graphiques, la précision des données affichées et la gestion correcte des alarmes.

Déploiement : Une fois la conception validée, téléchargez le projet sur le pupitre opérateur ou le système cible. Assurez-vous que la communication entre le PLC et l'IHM est opérationnelle et que toutes les fonctionnalités répondent aux exigences du processus industriel.

II.7 Conclusion

Conception du système de contrôle avec interface HMI permet d'assurer une gestion précise et intuitive du niveau d'eau dans le réservoir. Grâce à la transmission continue des données entre le capteur, le PLC et la vanne, et à leur affichage clair sur l'IHM, l'opérateur bénéficie d'un suivi en temps réel et peut intervenir efficacement en cas d'anomalie. L'approche de commande proportionnelle assure une réponse souple et adaptée aux variations du niveau, tandis que la configuration des alarmes et seuils garantit une sécurité renforcée. Ce chapitre a ainsi mis en évidence l'importance de l'intégration entre matériel, automatisme et interface utilisateur dans un système industriel moderne.

Chapitre 3

Programmation, simulation et tests du Système avec HMI

III.1 Introduction

Dans le cadre de ce projet de fin d'études, nous avons conçu et développé un système de contrôle automatisé pour une installation de stockage de l'eau. Le but principal est d'assurer une régulation fiable et continue du niveau d'eau dans un réservoir, tout en optimisant le processus de remplissage grâce à l'utilisation d'une vanne proportionnelle et d'un capteur analogique de niveau. Pour réaliser ce système, nous avons utilisé un automate programmable Siemens S7-1200, programmé à l'aide du logiciel TIA Portal V16, avec une simulation du programme via PLCSIM Advanced. L'objectif est de créer une interface de supervision centralisée à l'aide de SIMATIC WinCC Professional, afin de faciliter la gestion du réservoir à distance et en temps réel. La mise en place de cette solution permet de :

- Réduire l'intervention manuelle des opérateurs,
- Améliorer la sécurité du personnel en minimisant le risque d'accidents,
- Surveiller et ajuster dynamiquement le débit d'eau via la vanne proportionnelle,
- Suivre l'état du système grâce à des alarmes, messages de défauts et visualisations graphiques sur l'interface HMI.

Enfin, une attention particulière a été portée sur l'ergonomie de l'interface HMI, qui permet non seulement le contrôle global du système depuis un seul point, mais aussi une visualisation claire des données en temps réel : niveau d'eau (en pourcentage), état de la vanne, historique des défauts, etc.

III.2 Le cahier de charge

L'objectif de ce système est de maintenir le réservoir toujours plein en utilisant un capteur analogique pour mesurer le niveau d'eau, et en contrôlant automatiquement l'ouverture de la vanne en fonction de la valeur mesurée. Pour le remplissage du réservoir :

- ✓ Le capteur du niveau mesure le niveau d'eau dans le réservoir.
- ✓ Comparer la valeur mesurée du niveau avec la valeur min pour le remplissage et la valeur max pour arrêter le remplissage du réservoir.
- ✓ Calculer le débit d'ouverture de la vanne.

III.3 Configuration de l'appareil

La configuration de l'appareil est une étape très importante pour un nouveau projet. Dans cette configuration on fait :

- ✓ Choix le CPU de l'API
- ✓ Choix les modules d'extensions (SB, CM, SM, ...)
- ✓ Choix le HMI
- ✓ Configuration les réseaux de communications

Nous avons choisi le CPU 1214 AC/DC/RLY, ajouté un module d'entrée et sorti numérique DI8/DQ8×24VDC_1 et ajouté un module d'entrée et sorti analogique AI4×13BIT/AQ2×14BIT

La (Figure III.1) représente la liste des entrées et sorti numériques et analogiques :

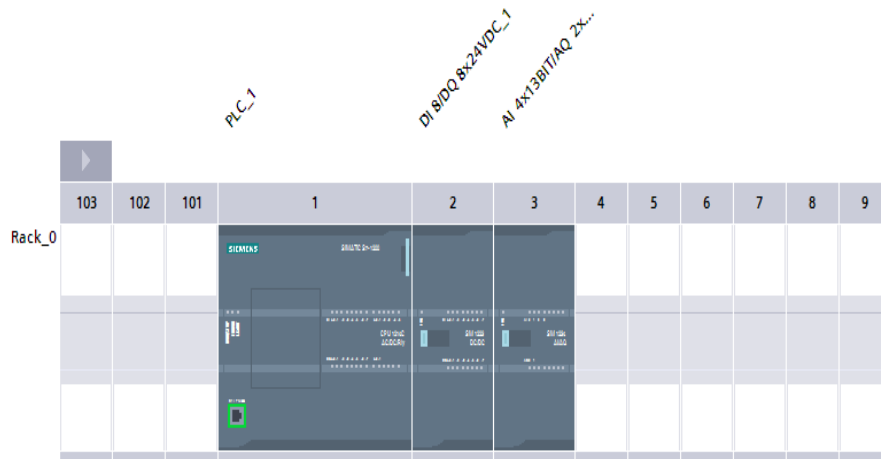


Figure III.1: Configuration de CPU

Nous avons choisi le HMI KTP400 confort (6AV2 124-2DC01.OAXO)

La (Figure III.2) représente le HMI :

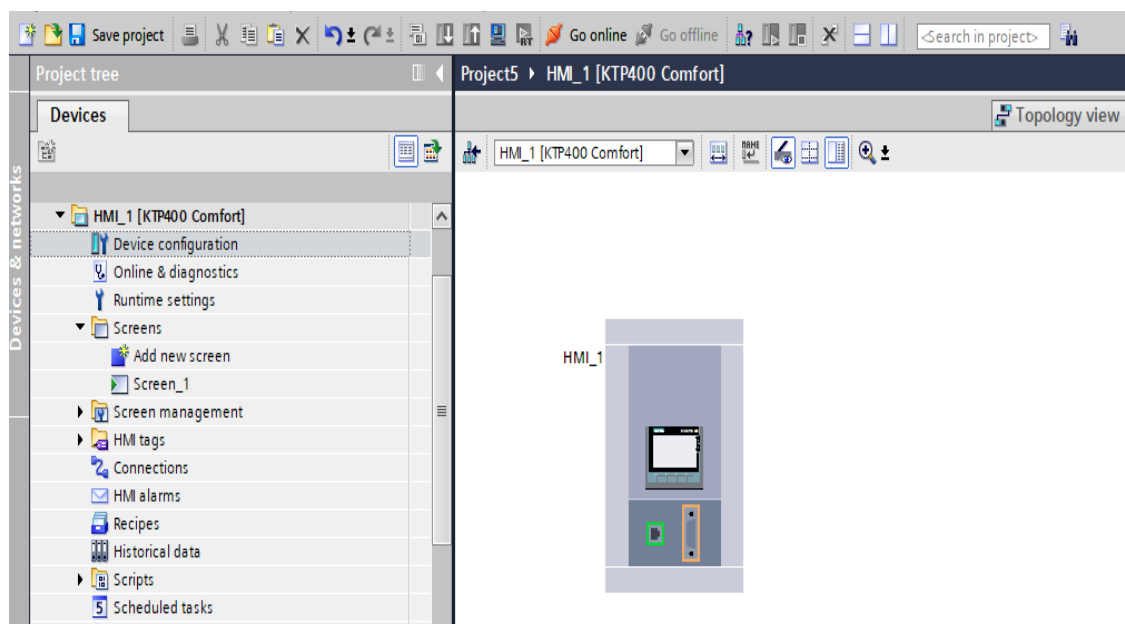


Figure III.2 : Configuration de HMI

III.4 Création de la table des variables

Après l'ouverture d'un nouveau projet comme montré dans le chapitre précédent on va créer la table des variables.

La (Figure III.3) représente la Déclaration des variables :

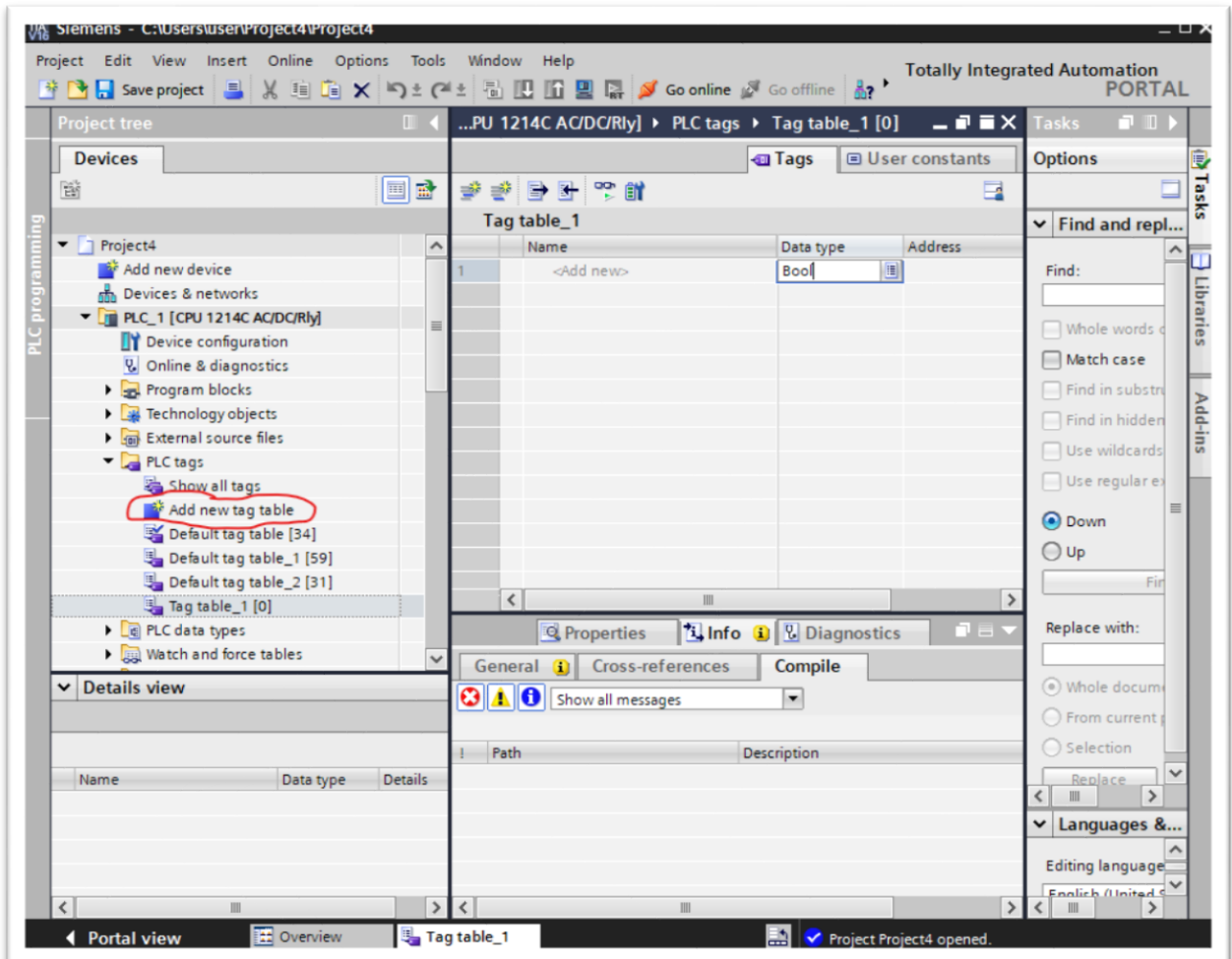


Figure III.3: Déclaration des variables

III.4.1 La table des variables

La (Figure III.4) suivante représente le tableau de variable utilisés dans notre programme.

	Name	Tag table	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	vanne_PRCsims	Default tag table	Int	%MW32	☐	☒	☒	☒	
2	vanne_PRCsim	Default tag table	Int	%MW30	☐	☒	☒	☒	
3	Q1	Default tag table	Bool	%I0.5	☐	☒	☒	☒	
4	niveau_PRC	Default tag table	Int	%MW22	☐	☒	☒	☒	
5	niveau_normalisé	Default tag table	Real	%MD80	☐	☒	☒	☒	
6	niveau	Default tag table	Int	%MW64	☐	☒	☒	☒	
7	Mar Arr HMI	Default tag table	Bool	%M0.1	☐	☒	☒	☒	
8	Mar Arr	Default tag table	Bool	%I0.0	☐	☒	☒	☒	
9	KM2in	Default tag table	Bool	%I0.7	☐	☒	☒	☒	
10	KM1in	Default tag table	Bool	%I0.6	☐	☒	☒	☒	
11	etat_normal	Default tag table	Bool	%M18.0	☐	☒	☒	☒	
12	dcy	Default tag table	Bool	%M0.0	☐	☒	☒	☒	
13	AU	Default tag table	Bool	%I0.2	☐	☒	☒	☒	
14	alarm_surveiller	Default tag table	Bool	%M36.0	☐	☒	☒	☒	
15	alarm_fuite	Default tag table	Bool	%M20.0	☐	☒	☒	☒	
16	alarm_default	Default tag table	Bool	%M34.0	☐	☒	☒	☒	
17	<Add new>				☐	☒	☒	☒	

Figure III.4: Tableau des variables

III.5 Création du programme

Le programme a été créé sous forme de blocs (FC, OB, DB)

OB Les OB cycliques sont traités de manière cyclique. Il s'agit de blocs de code de niveau supérieur dans le programme, dans lesquels vous pouvez programmer des instructions ou appeler d'autres blocs.

FC Les fonctions sont des blocs de code sans mémoire. [12]

DB Ces blocs de données servent uniquement à stocker des informations et des données mais pas d'instructions, ces données seront utilisées par d'autres blocs. [13]

III .5.1 block_1[FC1]

```
1
2 IF "niveau_PRC" <= 70 THEN
3
4     "vanne_PRCsims" := 100;
5
6 ELSIF ("niveau_PRC" > 70) AND ("niveau_PRC" <= 73) THEN
7     "vanne_PRCsims" := 100 - (("niveau_PRC" - 70) * (100 - 92) / (73 - 70));
8
9 ELSIF ("niveau_PRC" > 73) AND ("niveau_PRC" <= 76) THEN
10    "vanne_PRCsims" := 92 - (("niveau_PRC" - 73) * (92 - 82) / (76 - 73));
11
12 ELSIF ("niveau_PRC" > 76) AND ("niveau_PRC" <= 79) THEN
13    "vanne_PRCsims" := 82 - (("niveau_PRC" - 76) * (82 - 72) / (79 - 76));
14
15 ELSIF ("niveau_PRC" > 79) AND ("niveau_PRC" <= 82) THEN
16    "vanne_PRCsims" := 72 - (("niveau_PRC" - 79) * (72 - 62) / (82 - 79));
17
18 ELSIF ("niveau_PRC" > 82) AND ("niveau_PRC" <= 85) THEN
19    "vanne_PRCsims" := 62 - (("niveau_PRC" - 82) * (62 - 52) / (85 - 82));
20
21 ELSIF ("niveau_PRC" > 85) AND ("niveau_PRC" <= 88) THEN
22    "vanne_PRCsims" := 52 - (("niveau_PRC" - 85) * (52 - 42) / (88 - 85));
23
24 ELSIF ("niveau_PRC" > 88) AND ("niveau_PRC" <= 91) THEN
25    "vanne_PRCsims" := 42 - (("niveau_PRC" - 88) * (42 - 32) / (91 - 88));
26
27 ELSIF ("niveau_PRC" > 91) AND ("niveau_PRC" <= 94) THEN
28    "vanne_PRCsims" := 32 - (("niveau_PRC" - 91) * (32 - 22) / (94 - 91));
29
30 ELSIF ("niveau_PRC" > 94) AND ("niveau_PRC" <= 97) THEN
31    "vanne_PRCsims" := 22 - (("niveau_PRC" - 94) * (22 - 12) / (97 - 94));
32
33 ELSIF ("niveau_PRC" > 97) AND ("niveau_PRC" <= 100) THEN
34    "vanne_PRCsims" := 12 - (("niveau_PRC" - 97) * (12 - 0) / (100 - 97));
35
36 ELSE
```

```

31 "vanne_PRCsims" := 22 - (( "niveau_PRC" - 94) * (22 - 12) / (97 - 94));
32
33 ELSIF ("niveau_PRC" > 97) AND ("niveau_PRC" <= 100) THEN
34   "vanne_PRCsims" := 12 - (( "niveau_PRC" - 97) * (12 - 0) / (100 - 97));
35
36 ELSE
37
38   "vanne_PRCsims" := 0;
39 END_IF;
40
41 IF ("niveau_PRC" >= 70) AND ("niveau_PRC" <= 95) THEN
42   "etat_normal" := TRUE;
43 ELSE
44   "etat_normal" := FALSE;
45 END_IF;
46
47
48 IF "niveau_PRC" < 30 THEN
49   "alarm_fuite" := TRUE;
50 ELSE
51   "alarm_fuite" := FALSE;
52 END_IF;
53
54 IF ("niveau_PRC" > 100) OR ("niveau_PRC" < 0) THEN
55   "alarm_default" := TRUE;
56 ELSE
57   "alarm_default" := FALSE;
58 END_IF;
59
60
61 IF ("niveau_PRC" >= 96) AND ("niveau_PRC" <= 100) THEN
62   "alarm_surveiller" := TRUE;
63 ELSE
64   "alarm_surveiller" := FALSE;
65 END_IF;
66

```

Figure III.5 : Le programme dans FC1 (SCL)

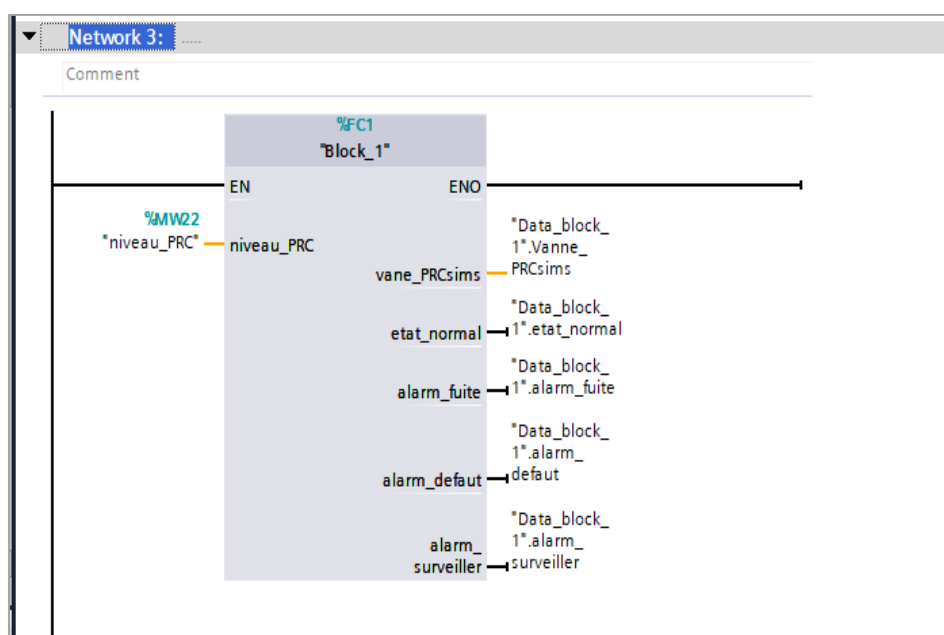


Figure III.6 : Le réseau dans FC1

III .5.2 Block Data [DB1]

Data_block_1									
	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	Comment
1	▼ Static			☐	☐	☐	☐	☐	
2	Vanne_PRCsims	Int	0	☐	☒	☒	☒	☐	
3	etat_normal	Bool	false	☐	☒	☒	☒	☐	
4	alarm_fuite	Bool	false	☐	☒	☒	☒	☐	
5	alarm_defaut	Bool	false	☐	☒	☒	☒	☐	
6	alarm_surveiller	Bool	false	☐	☒	☒	☒	☐	
7	<Add new>			☐	☐	☐	☐	☐	

Figure III.7: Data block

III.5.3 Block OB1

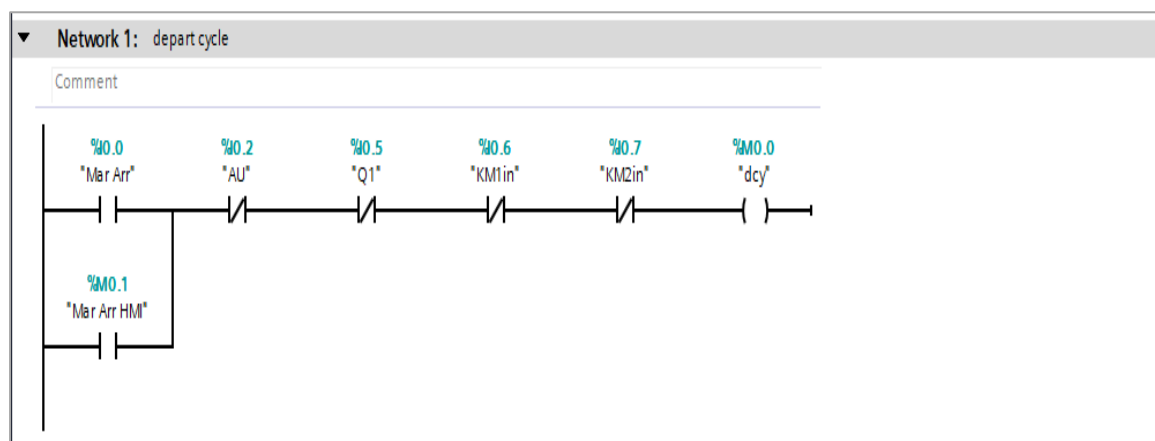


Figure III.8 : Réseau de part cycle

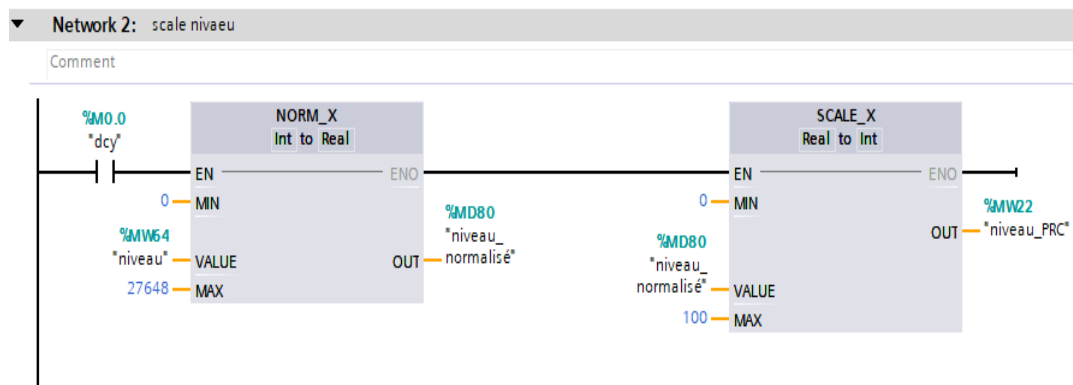


Figure III.9 : Réseau scale niveau

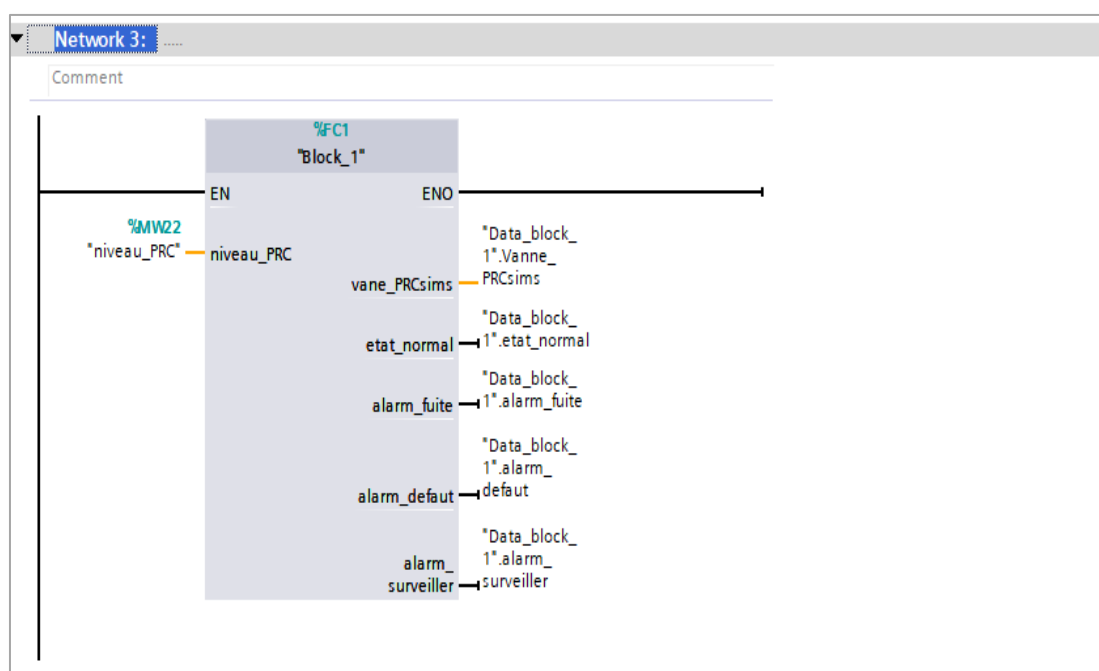


Figure III.10 : Réseau de la pransipe de l'ouverture de la vanne



Figure III.11: Réseau de la valeur de l'ouverture de la vanne

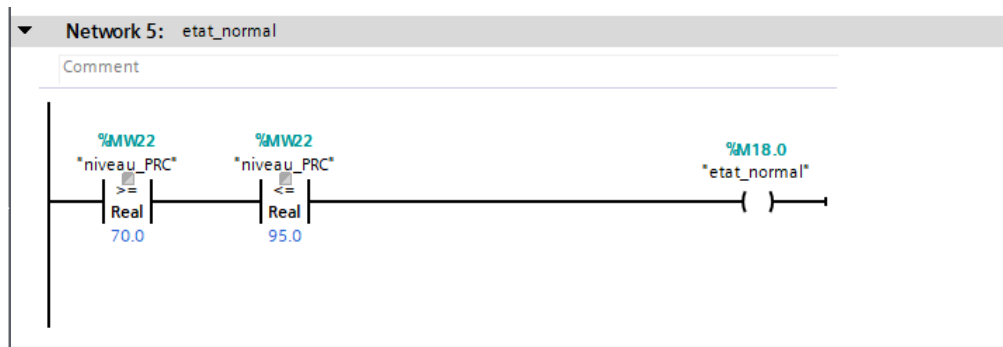


Figure III.12: Réseau de l'état normal

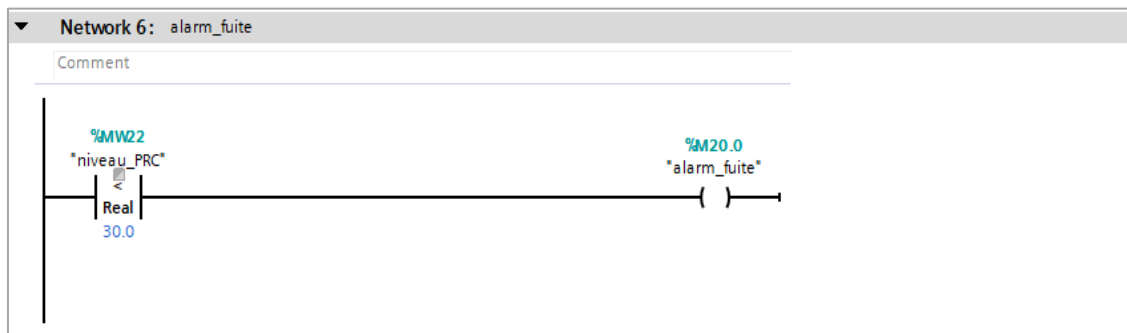


Figure III.13: réseau de l'alarme de fuite

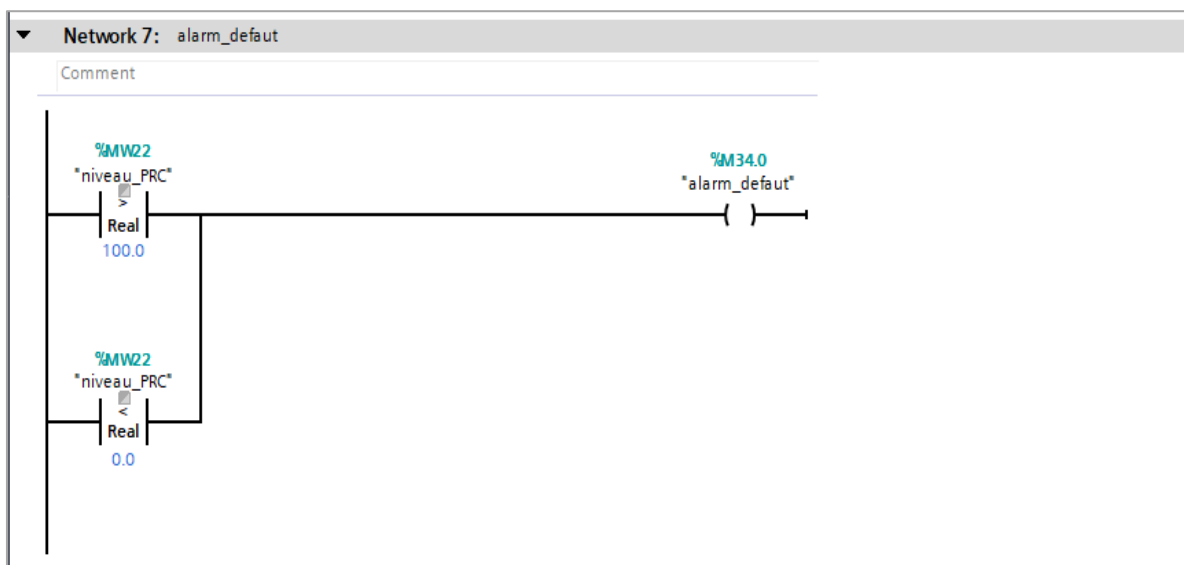


Figure III.14: Réseau de l'alarme de faut

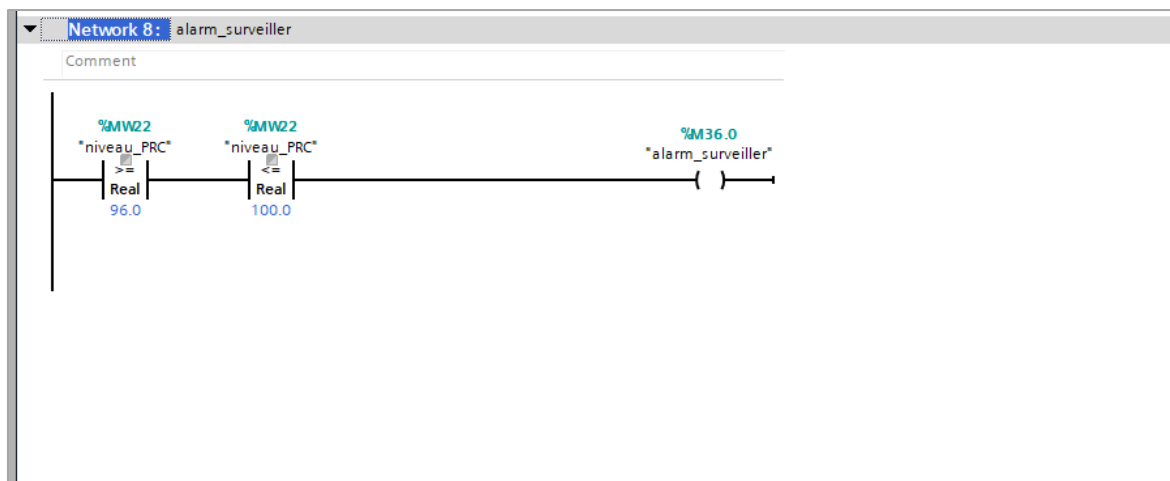


Figure III.15: réseau de l'alarme de surveiller

III.6 Compilation et simulation du programme

Une fois le système programmé, TIA PORTAL permet de le simuler à l'aide de simulateur PLC SIM, en commençant par la compilation, puis le chargement du programme dans l'automate. La figure suivante présente la barre de simulation.



Figure III.16: Barre de la simulation de TIA portal.

General ⓘ Cross-references Compile Syntax							
Show all messages							
Compiling finished (errors: 0; warnings: 0)							
!	Path	Description	Go to	?	Errors	Warnings	Time
✓	PLC_1				0	0	8:17:33 PM
✓	Program blocks				0	0	8:17:51 PM
✓	Main (OB1)	Block was successfully compiled.					8:17:51 PM
✓		Compiling finished (errors: 0; warnings: 0)					8:17:56 PM

Figure III.17: Compilation de programme sans des erreurs.

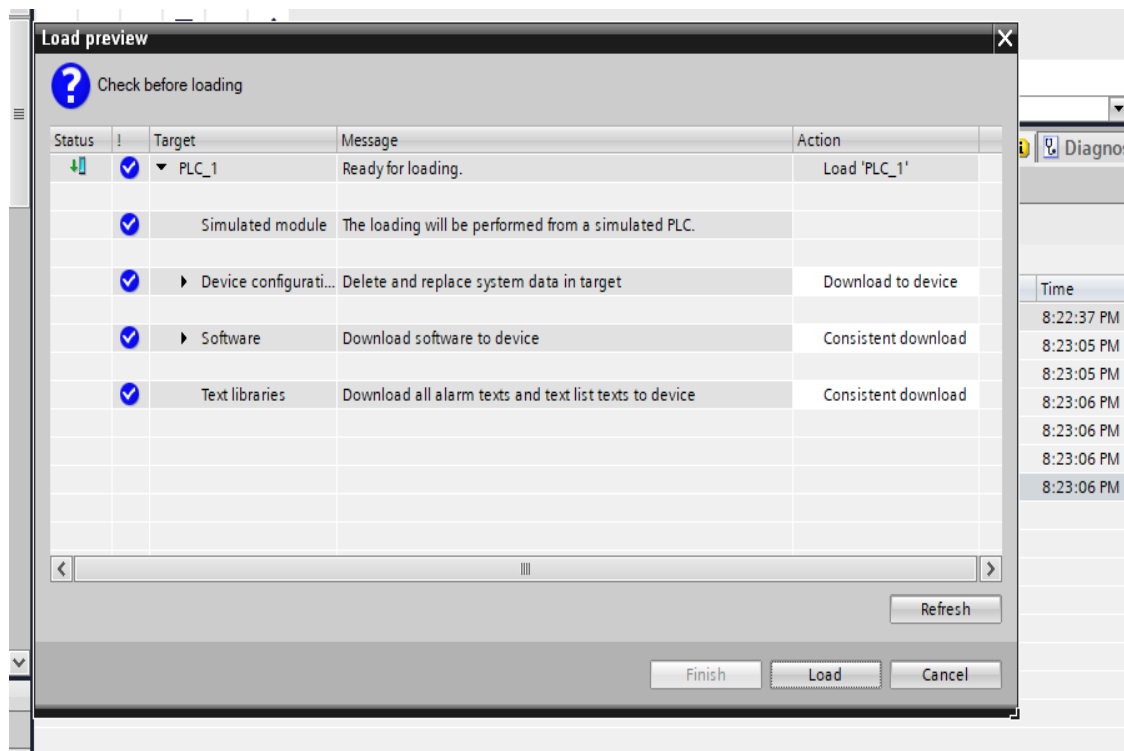


Figure III.18: Chargement de programme

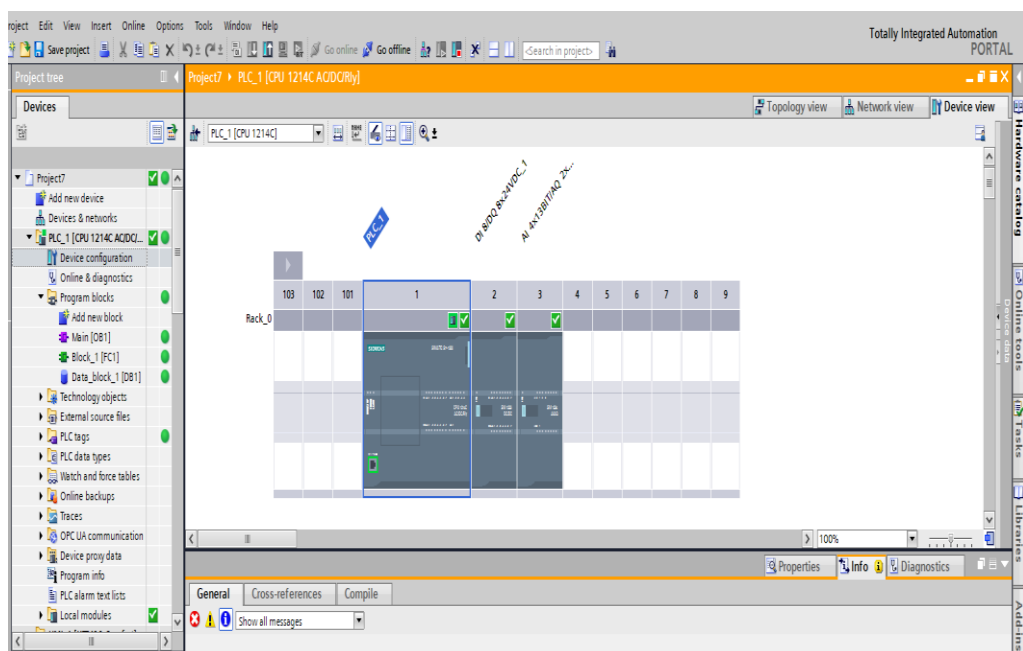


Figure III.19: Indication de la mise en ligne.

III .7 Création de la supervision

III.7.1 Etablissement et liaison d'une HMI

L'établissement de l'HMI se fait par l'ajout d'un nouvel écran (KTP400comfort), et après une liaison entre l'HMI et l'API, afin de pouvoir lire les données dans l'automate.

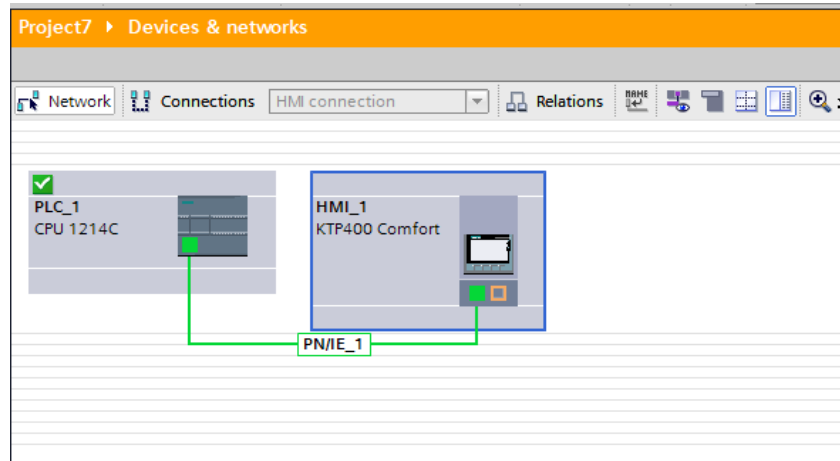


Figure III.20 : Liaison PLC_HMI.

III.7.2 Création d'interfaces HMI

Pour le suivi et le contrôle de la station de stockage d'eau, nous devons créer une interface permettant d'afficher et de donner des ordres (contrôle) aux automates programmables (PLC) pour assurer un contrôle efficace

III .7.3 Les différentes vues du projet

Pour le suivi et le contrôle de la station de stockage d'eau, nous devons créer une interface permettant la visualisation. L'interface graphique de notre station se compose de plusieurs scènes:

III .7.3.1 Vue de page principale (HOME)

La (figure III.21) suivante montre la page d'accueil en HOME :



Figure III.21 : Vue de page principale (HOME)

✓ La vue de cuve :

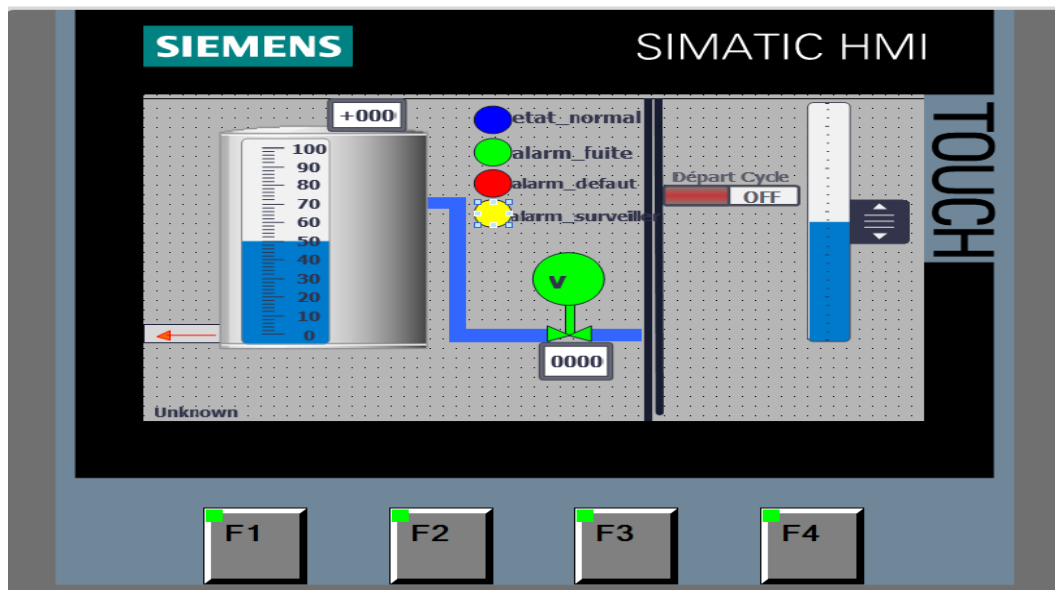


Figure III.21 : Vue de cuve

✓ La vue de courbe :

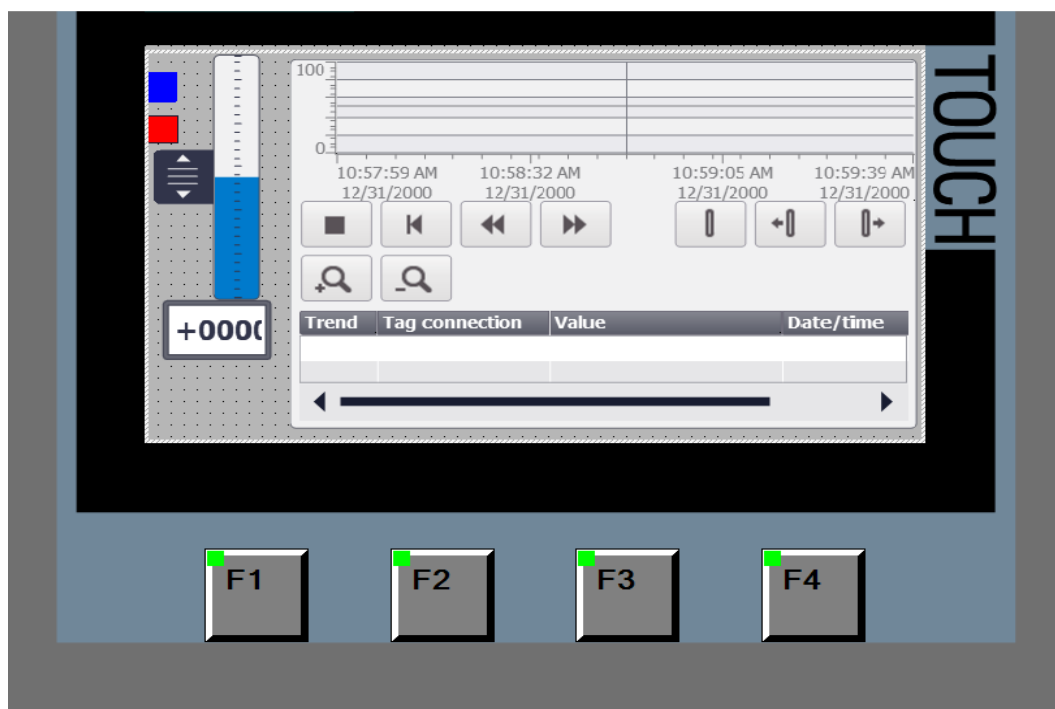


Figure III.22 : Vue de courbe

Simulation par PLCSIM

- OB1

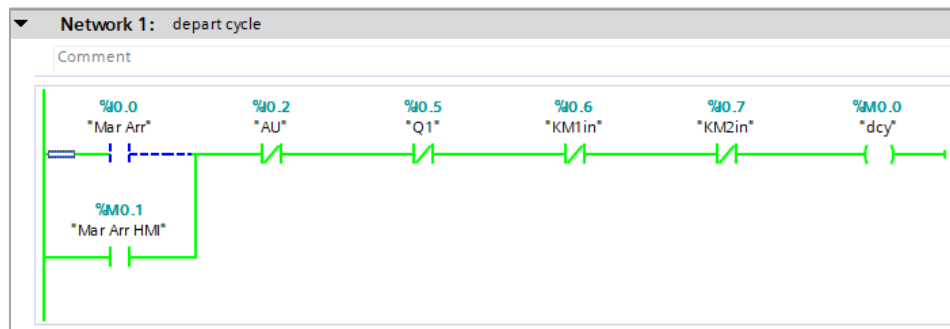


Figure III.23 : Simulation de réseau de départ de cycle

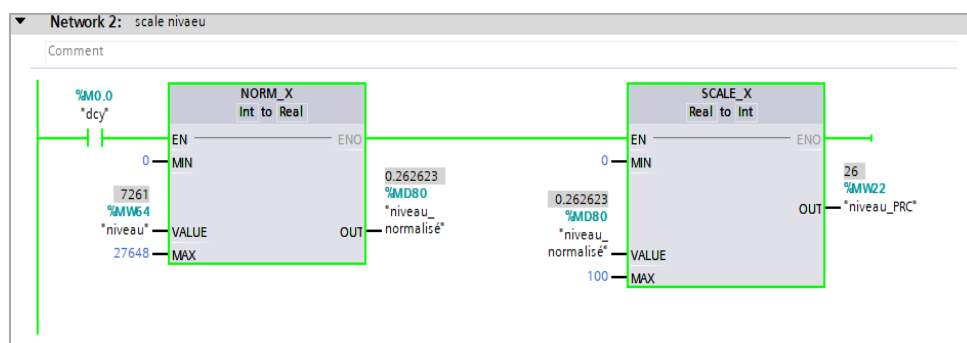


Figure III.24 Simulation de réseau de scale niveau

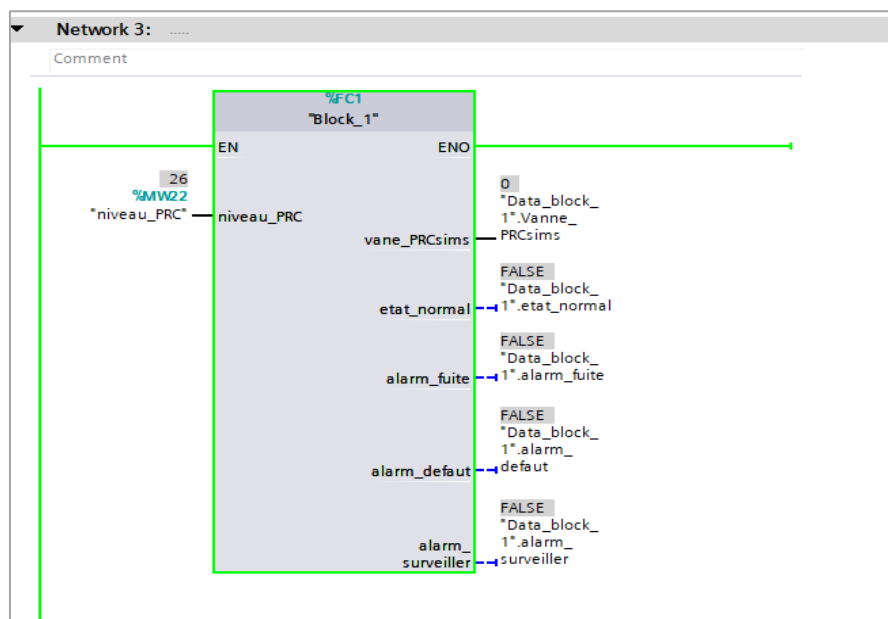


Figure III.24 : Simulation de réseau de la pransipe de l'overteur de vanne

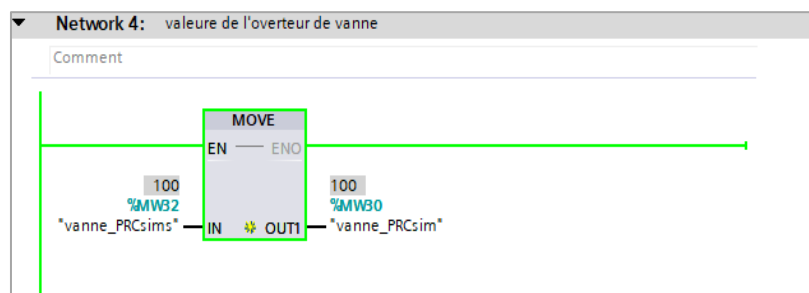


Figure III.25: Simulation de réseau de la valeur de l'ouverture de la vanne

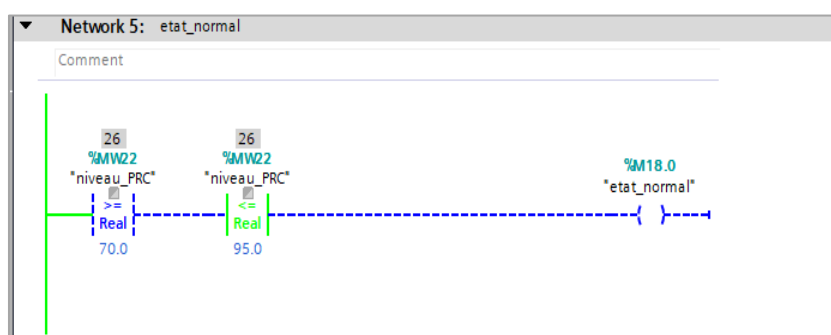


Figure III.26: Simulation de réseau de l'état normal

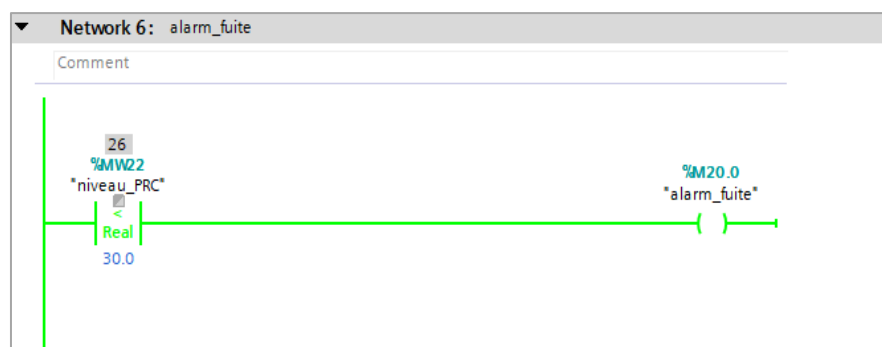


Figure III.27: Simulation de réseau d'alarme de fuite

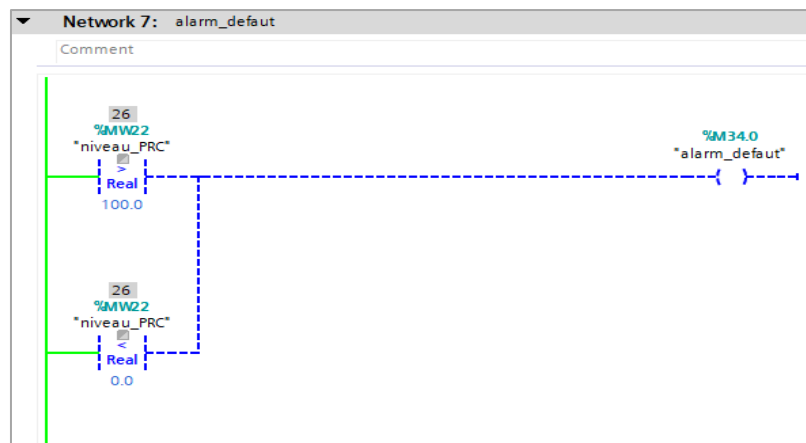


Figure III.28: Simulation de réseau de l'alarme de défaut

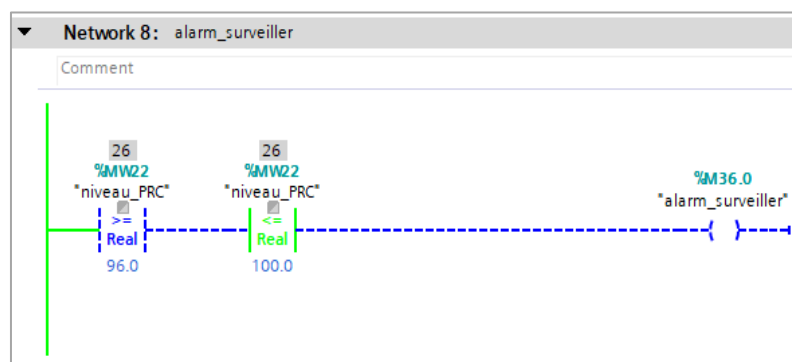


Figure III.28: Simulation de réseau de l'alarme de surveiller

- Bock_1[FC1] :

Project7 > PLC_1 [CPU 1214C A/C/D/Rly] > Program blocks > Block_1 [FC1]

```

No condition defined.

1
2 IF "niveau_FRC" <= 70 THEN
3
4     "vanne_FRCsims" := 100;
5
6 ELSIF ("niveau_FRC" > 70) AND ("niveau_FRC" <= 73) THEN
7     "vanne_FRCsims" := 100 - ((("niveau_FRC" - 70) * (100 - 92)) / (73 - 70));
8
9 ELSIF ("niveau_FRC" > 73) AND ("niveau_FRC" <= 76) THEN
10    "vanne_FRCsims" := 92 - (((niveau_FRC") - 73) * (92 - 82)) / (76 - 73));
11
12 ELSIF ("niveau_FRC" > 76) AND ("niveau_FRC" <= 79) THEN
13    "vanne_FRCsims" := 82 - (((niveau_FRC") - 76) * (82 - 72)) / (79 - 76));
14
15 ELSIF ("niveau_FRC" > 79) AND ("niveau_FRC" <= 82) THEN
16    "vanne_FRCsims" := 72 - (((niveau_FRC") - 79) * (72 - 62)) / (82 - 79));
17
18 ELSIF ("niveau_FRC" > 82) AND ("niveau_FRC" <= 85) THEN
19    "vanne_FRCsims" := 62 - (((niveau_FRC") - 82) * (62 - 52)) / (85 - 82));
20
21 ELSIF ("niveau_FRC" > 85) AND ("niveau_FRC" <= 88) THEN
22    "vanne_FRCsims" := 52 - (((niveau_FRC") - 85) * (52 - 42)) / (88 - 85));
23
24 ELSIF ("niveau_FRC" > 88) AND ("niveau_FRC" <= 91) THEN
25    "vanne_FRCsims" := 42 - (((niveau_FRC") - 88) * (42 - 32)) / (91 - 88));
26
27 ELSIF ("niveau_FRC" > 91) AND ("niveau_FRC" <= 94) THEN
28    "vanne_FRCsims" := 32 - (((niveau_FRC") - 91) * (32 - 22)) / (94 - 91));
29
30 ELSIF ("niveau_FRC" > 94) AND ("niveau_FRC" <= 97) THEN
31    "vanne_FRCsims" := 22 - (((niveau_FRC") - 94) * (22 - 12)) / (97 - 94));
32
33 ELSIF ("niveau_FRC" > 97) AND ("niveau_FRC" <= 100) THEN

```

Result	TRUE
"niveau_FRC"	26
"vanne_FR..."	100

Result	
"vanne_PR..."	

Result	
"vanne_PR..."	

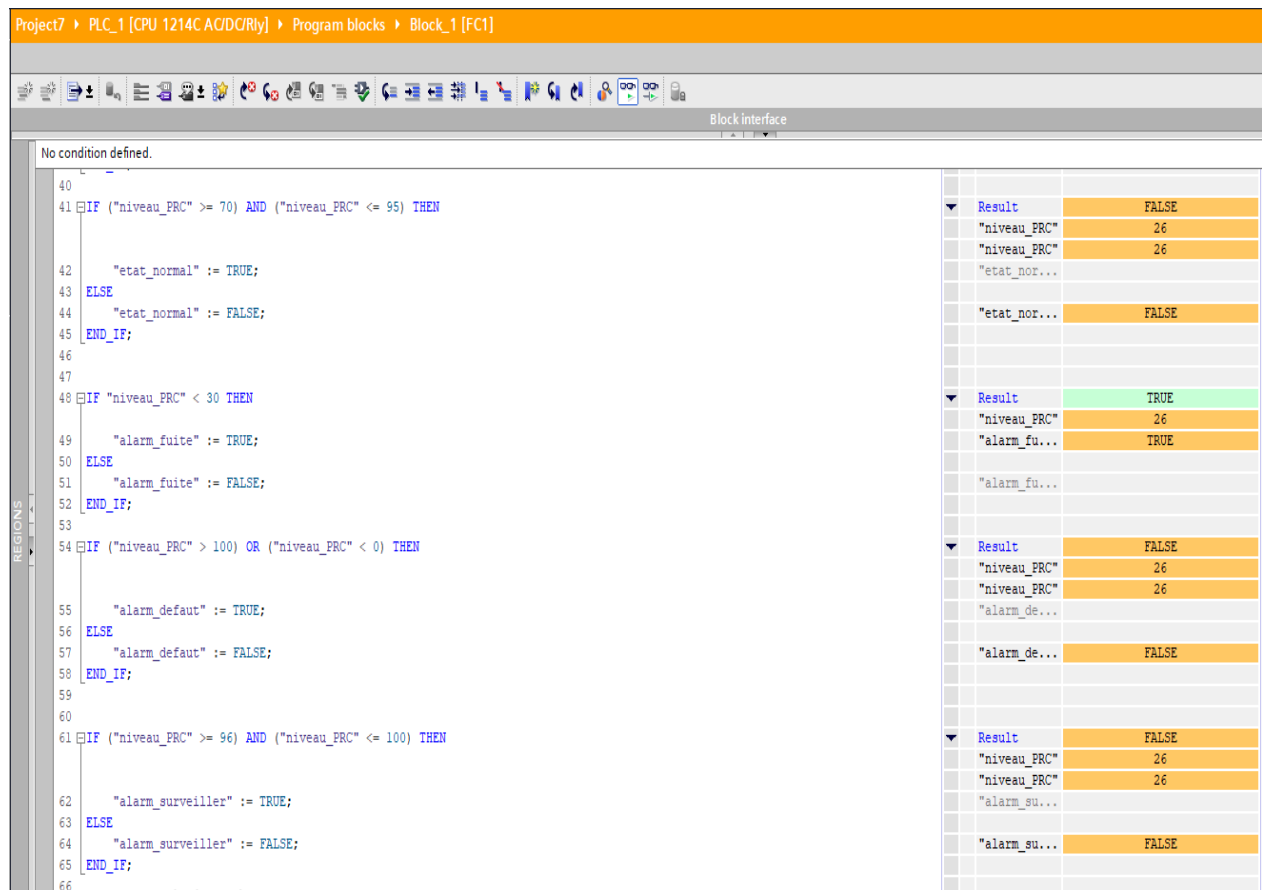
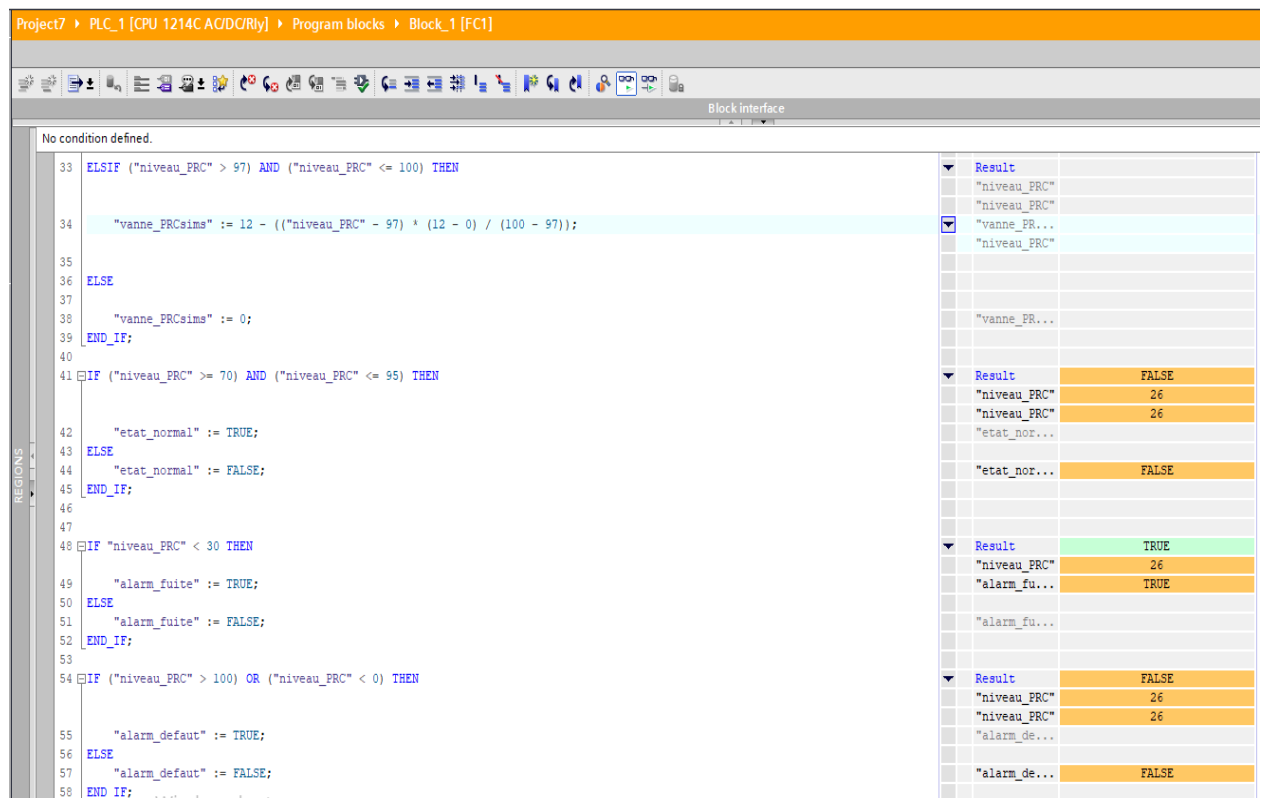


Figure III.29: Simulation de Bock_1[FC1]

- Data_block_1[DB1]

Project7 > PLC_1 [CPU 1214C AC/DC/Rly] > Program blocks > Data_block_1 [DB1]

Keep actual values Snapshot Copy snapshots to start values Load start values as actual values

Data_block_1

	Name	Data type	Start value	Monitor value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint
1	▼ Static								
2	Vanne_PRCsims	Int	0	0	☐	☒	☒	☒	☐
3	etat_normal	Bool	false	FALSE	☐	☒	☒	☒	☐
4	alarm_fuite	Bool	false	FALSE	☐	☒	☒	☒	☐
5	alarm_defaut	Bool	false	FALSE	☐	☒	☒	☒	☐
6	alarm_surveiller	Bool	false	FALSE	☐	☒	☒	☒	☐

Figure III.30: Simulation de Data_block_1[DB1]

III.8 Simulation par Wincc

Cas 1 :

Lorsque le niveau d'eau (niveau_PRC) est à 0, ouvrez la vanne(vanne_PRCsim) à 100 et allumez alarm_fuite. La (Figure III.31) représente cas 1

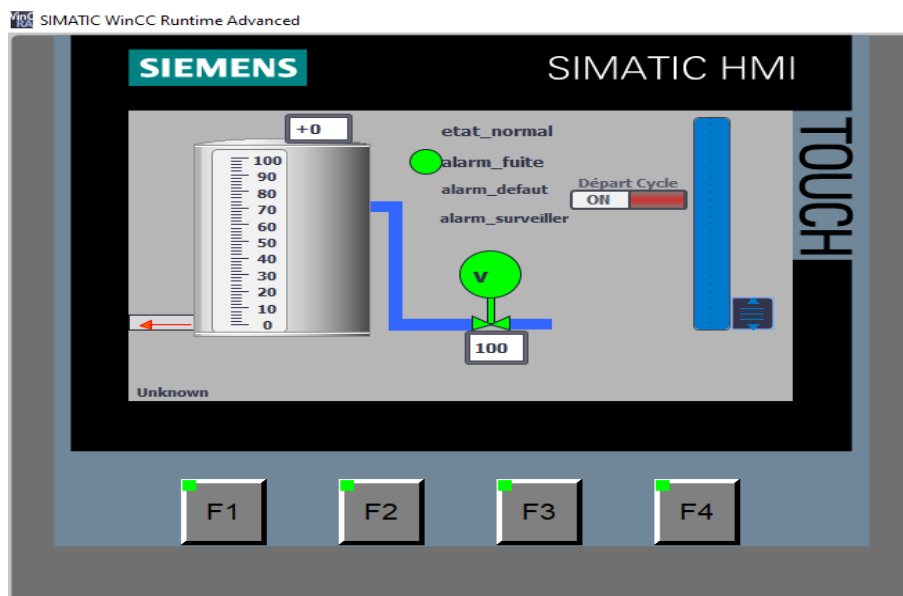


Figure III.31 : Représente cas1

Cas 2 :

Lorsque le niveau d'eau (niveau_PRC) est à 70, ouvrez la vanne(vanne_PRCsim) à 100 et allumez etat_normal. La (Figure III.32) représente cas 2

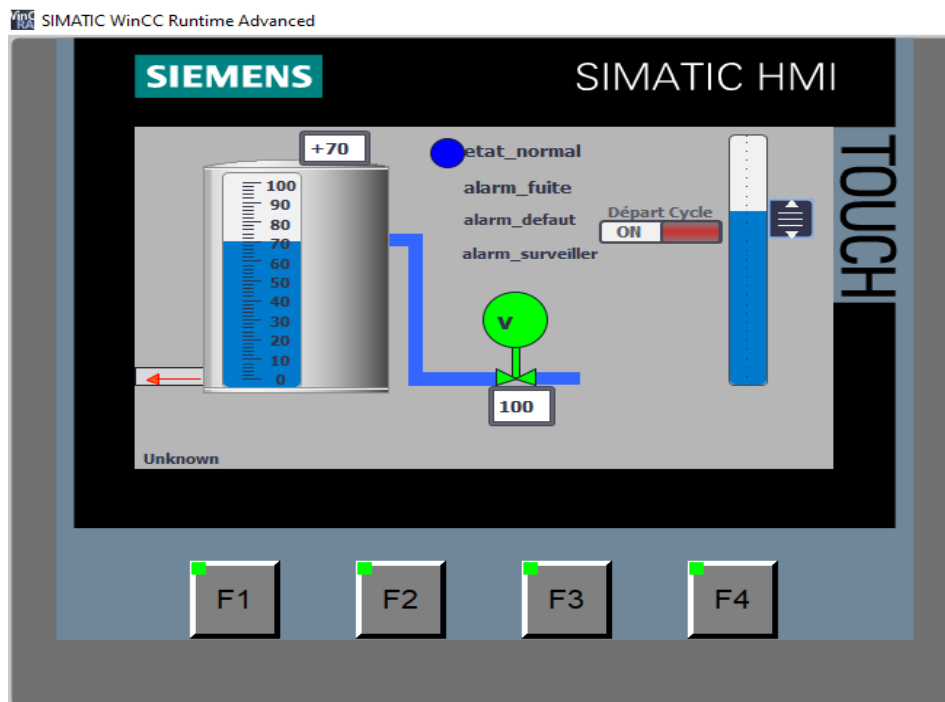


Figure III.32 : représente cas 2

Cas 3 :

Lorsque le niveau d'eau (niveau_PRC) est à 100, ouvrez la vanne (vanne_PRCsim) à 0 et allumez alarm_surveiller. La (Figure III.33) représente cas 3

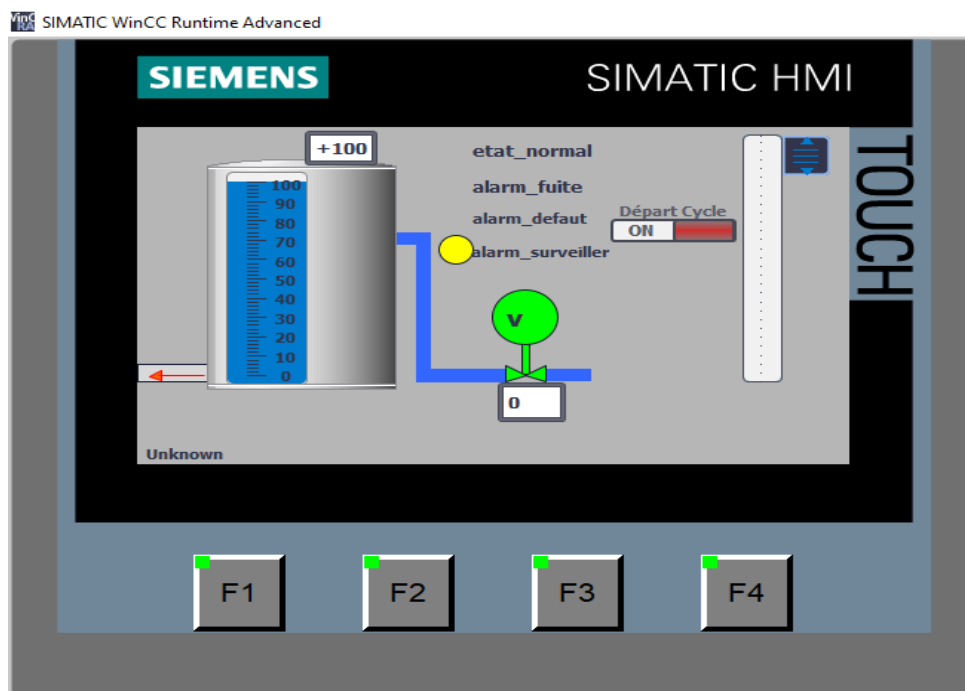


Figure III.33 : Représente cas 3

Courbe Cas 1

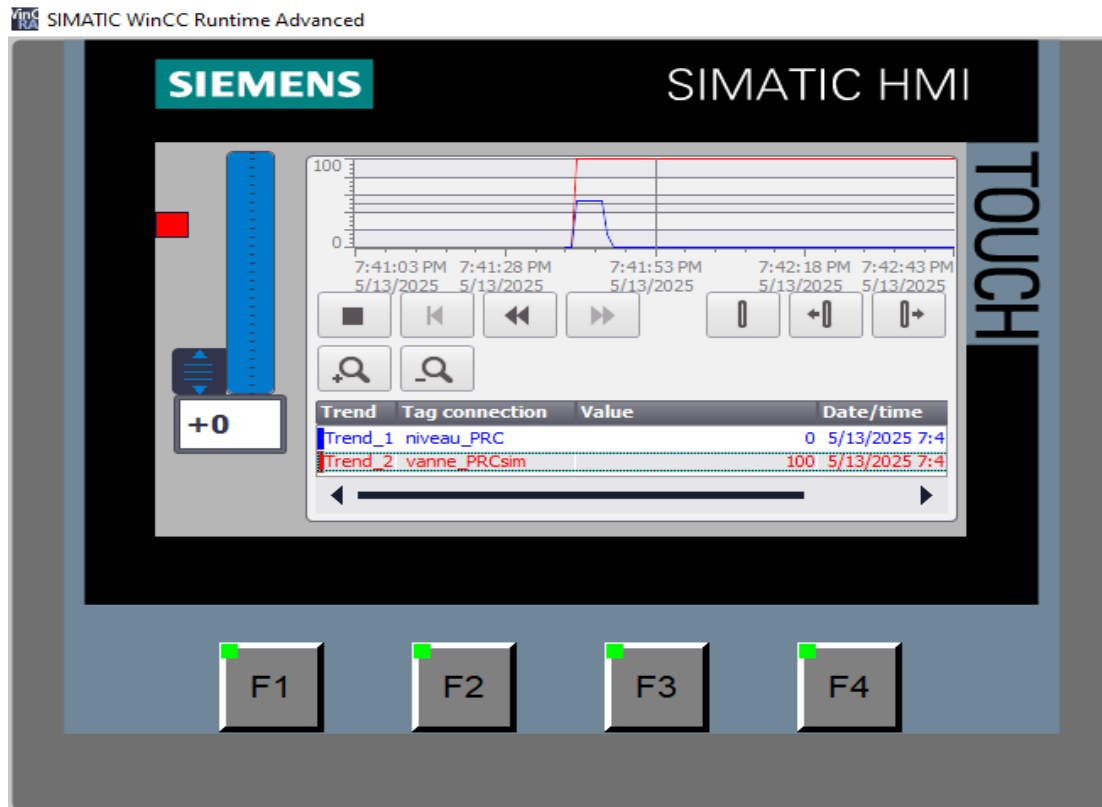


Figure III.34 : Représente la courbe de la cas 1

Cas 2 :

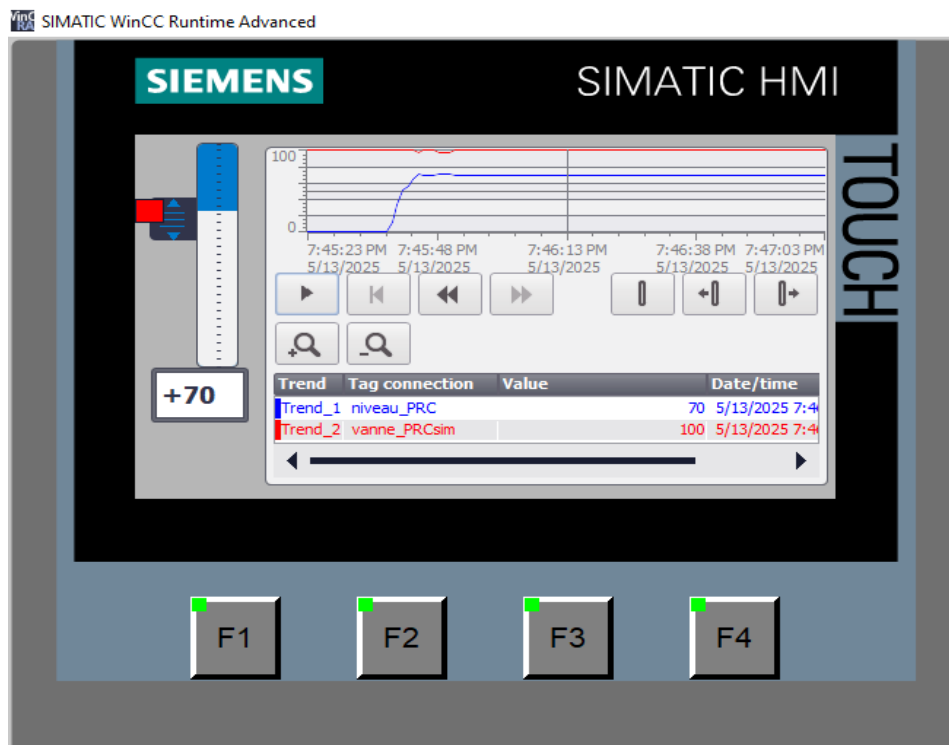


Figure III.35 : Représente la courbe de la cas 2

Cas3

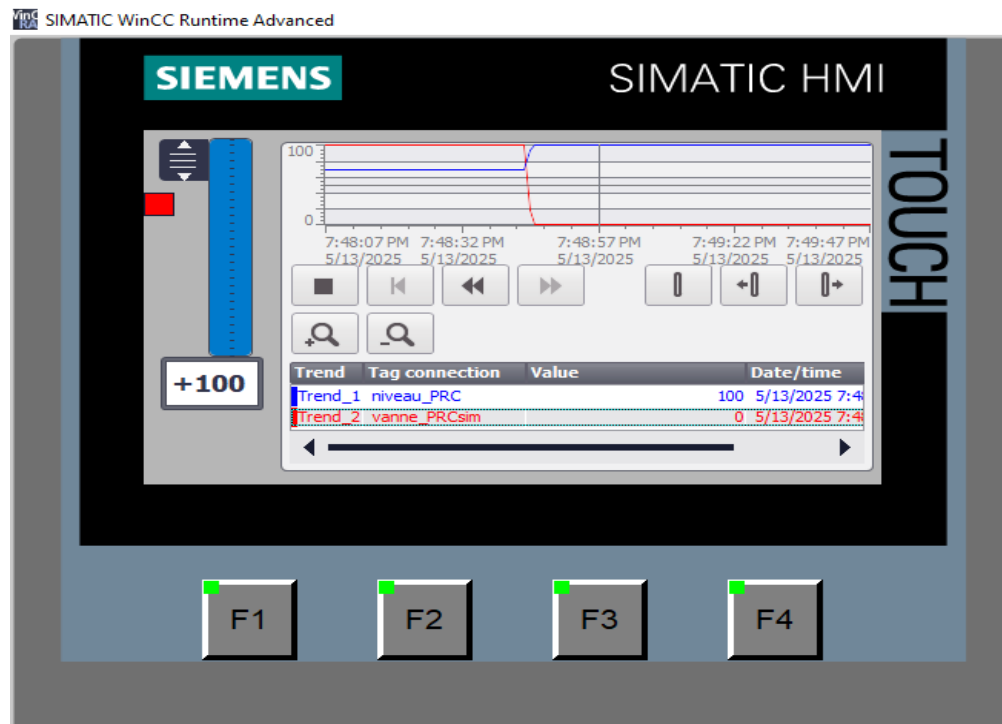


Figure III.36 : Représente la courbe de la cas 3

III .9 Conclusion

Dans ce chapitre, nous avons réalisé une simulation complète du système de contrôle du niveau d'eau à l'aide du logiciel TIA Portal V16. Nous avons commencé par modéliser le réservoir, le capteur analogique et la vanne proportionnelle, puis nous avons simulé le comportement du système à l'aide de l'outil S7-PLCSIM. Ensuite, nous avons conçu une interface Homme-Machine (HMI) avec WinCC, permettant d'afficher en temps réel les données du système, de configurer les seuils de sécurité et de déclencher des alarmes. Enfin, différents scénarios de test ont été exécutés pour analyser la réaction du système et valider son bon fonctionnement.

Conclusion générale

Conclusion générale

Au terme de ce projet de fin d'études, portant sur Système de Contrôle pour une Installation de Stockage de l'eau, nous avons pu mettre en œuvre une solution complète s'appuyant sur un automate programmable industriel SIEMENS SIMATIC S7-1200, couplé à une interface HMI tactile via SIMATIC WinCC, le tout développé dans l'environnement TIA Portal V16.

Nous avons ainsi approfondi nos connaissances dans plusieurs domaines techniques essentiels :

- La lecture de signaux analogiques issus d'un capteur de niveau.
- La commande proportionnelle d'une vanne.
- Et la supervision en temps réel à l'aide d'une interface HMI intuitive et interactive.

Les tests réalisés montrent que notre système réagit correctement aux variations du niveau d'eau. La vanne est commandée avec précision selon les consignes, et les alertes sont bien déclenchées en cas de dépassement de seuils. L'opérateur peut surveiller et ajuster le fonctionnement en temps réel depuis l'HMI.

Bien que nous ayons rencontré plusieurs difficultés techniques au cours du développement (programmation, configuration matérielle, gestion des signaux analogiques), nous avons su persévérer et atteindre nos objectifs. Ce projet nous a permis de développer une réelle autonomie dans la programmation d'API, la régulation analogique, et l'interface homme-machine.

Bibliographique

Bibliographiques

- [1] https://prezi.com/p/zya701r6_hwz/la-regulation-automatique-du-niveau-dun-reservoir/
- [2] <https://tameson.fr/pages/electrovannes-proportionnelles-leur-fonctionnement>
- [3] <https://mall.industry.siemens.com/mall/nl/nl/Catalog/Products/9109999>
- [4] http://thesis.essa-tlemcen.dz:8080/bitstream/handle/STDB_UNAM/523/Master-Conception%20d%27un%20syst%C3%A8me%20%284%29.pdf?sequence=1&isAllowed=y
- [5] <http://dspace.univ-tiaret.dz/bitstream/123456789/2577/1/TH.M.GE.FR.2022.13.pdf>
- [6] <https://dspace.univ-ouargla.dz/jspui/bitstream/123456789/35133/1/LEMLEM-ELMAGBED.pdf>
- [7] <http://archives.univbiskra.dz/bitstream/123456789/21825/1/Bouzinaoui%20Khalil%20Djabir.pdf>
- [8] <https://dspace.univ-ouargla.dz/jspui/bitstream/123456789/35133/1/LEMLEM-ELMAGBED.pdf>
- [9] http://archives.univbiskra.dz/bitstream/123456789/17963/1/BENLEMBAREK_Salah_Eddine.pdf
- [10] <https://dspace.univ-temouchent.edu.dz/handle/123456789/1454>
- [11] Université de Guelma – Régulation Industrielle, page 32 (PDF)
- [12]. SIEMENS, SIMATIC S7-1200 Easy Book Manuel 2015
- [13] <https://support.industry.siemens.com/cs/mdm/109759862?c=89390166923&lc=fr-lu>