

الجمهورية الجزائرية الديمقراطية الشعبية

Republique Algerienne Democratique Et Populaire

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة العربي التبسي - تبسة

Université Larbi Tébessi- Tébessa –

Faculté des Sciences et de la Technologie

Département de génie électrique

## MEMOIRE

Présenté pour l'obtention du diplôme de Master Académique

Spécialité : AUTOMATIQUE ET SYSTEMES

Présentées par :

BEYAZA Aya

NECEB Loubna

Sujet :

# AUTOMATISATION ET SUPERVISION PAR API D'UNE UNITE DE TRAITEMENT DES EAU

Présenté et soutenu publiquement, le ...../...../2022 , Devant le jury composés de:

Dr. DJARI Abdelhamid

Pr. SOUFI Youcef

Dr. THELAIDJIA Tawfik

MCA

Pr

MCA

Président

Rapporteur

Examineur 1

Promotion : 2021/2022

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

# Remerciements

*Louange à Allah le tout puissant qui m'a accordé la foi, le courage et la patience pour mener à bien ce travail.*

*Je tiens à remercier infiniment mon encadreur, Pr. Youcef SOUFI, pour son aide, sans réserve, sa patience et ses conseils précieux qui m'ont été très utiles.*

*Je remercie également les membres du jury qui m'ont honoré en acceptant de juger et d'enrichir ce travail.*

*Ces remerciements vont aussi à toute ma famille, et tous mes amis pour leur aide et leur soutien.*

*Je tiens à remercier également tous mes enseignants.*

*Aya BEYAZA  
Loubna NECIB*

# *Dédicace*

*A ma chère et adorable père 'Abd allah' en témoignage de ma  
grande affection.*

*A ma chère et adorable mère "Nassima" en témoignage de ma  
grande affection.*

*A ma sœur Malak, pour leurs soutiens moraux.*

*A mon frère : baha eddine*

*A tous les membres de ma famille surtout Ala, mima, Naima,  
noussaiba, sadjida, saïf.*

*A tous mes amis surtout : wissal, fulla, hadjer, chadia, ihssen,  
habiba, maïssa,*

*A mon amie est aussi binôme « loubna »*

*Mes camarades de la promotion 2021-2022*

*A tous les personnes que j'aime.*

*Je dédie ce travail.*

*AYA BAYAZA*

# *Dédicace*

*A l'âme de mon père adorable "Djemai" en témoignage de ma  
grande affection.*

*A ma chère et adorable mère "Dalila" en témoignage de ma  
grande affection.*

*A mon frère : Salah Eddine pour leur soutiens morales.*

*A mes sœurs Nesrine, Douaa et Asma pour leurs soutiens moraux.*

*A tous les membres de ma famille surtout Saleh, mon oncle antre  
et mon grand-père Hama.*

*A tous mes amis surtout : rima, Djamila, chaima, loubna, amna,  
wima, bouti*

*A mon amie est aussi binôme « Aya »*

*Mes camarades de la promotion 2021-2022.*

*A tous les personnes que j'aime.*

*Je dédiece travail.*

*Loubna*

# Sommaire

Remerciements	
Dédicace	
Notations et abréviations	I
Liste des figures	IV
Liste des Tableaux	VII
Résumé	IX
Introduction générale	1

## Chapitre 1

<b>I</b>	<b><i>Généralités sur les systèmes automatisés</i></b>	
<b>I.1</b>	Introduction	3
<b>I.2</b>	Les systèmes automatisés	3
<b>I.3</b>	Schéma d'un système automatisé	4
<b>I.4</b>	Chaîne d'information	4
<b>I.5</b>	Chaîne d'énergie	5
<b>I.6</b>	L'automatisme	5
<b>I.6.1</b>	Objectifs de l'automatisation	6
<b>I.6.2</b>	Les différents types d'automatisation	6
<b>I.6.2.1</b>	L'automatisation fixe ou rigide	6
<b>I.6.2.2</b>	L'automatisation programmable	7
<b>I.6.2.3</b>	L'automatisation flexible ou douce	7
<b>I.7</b>	Les principaux avantages de l'automatisation	7
<b>I.8</b>	Les principales parties d'un système automatisé	8
<b>I.9</b>	Structure d'un système automatisé	9

<b>I.9.1</b>	La partie opérative	10
<b>I.9.2</b>	La partie commande	11
<b>I.9.3</b>	La partie dialogue	11
<b>I.10</b>	Généralités sur les automates programmables	12
<b>I.10.1</b>	Historique	12
<b>I.10.2</b>	L'automate programmable	12
<b>I.10.3</b>	Critères de choix d'un automate	13
<b>I.10.4</b>	Avantages des automates programmables	14
<b>I.11</b>	Traitement du programme automate	14
<b>I.11.1</b>	Langage de programmation de l'API	15
<b>I.11.1.1</b>	Le langage LD (Ladder Diagram)	15
<b>I.11.1.2</b>	Le langage IL (Instruction List)	15
<b>I.11.1.3</b>	Le langage FBD (Function Block Diagram)	16
<b>I.11.1.4</b>	Le langage ST (Structured Text)	16
<b>I.11.1.5</b>	Le langage SFC (Sequential Function chart), ou GRAFCET	16
<b>I.12</b>	Architecture des automates programmables	17
<b>I.13</b>	Structure interne des automates programmables	18
<b>I.13.1</b>	Le processeur	19
<b>I.13.2</b>	Les mémoires	19
<b>I.13.3</b>	Les modules d'entrées/sorties	20
<b>I.13.4</b>	L'alimentation	20
<b>I.13.5</b>	Liaisons de communication	20
<b>I.14</b>	TIA Portal (Totally Integrated Automation)	21

<b>I.14.1</b>	Description du logiciel TIA Portal	21
<b>I.14.2</b>	Les avantages du logiciel TIA portal	21
<b>I.15</b>	Conclusion	22

## **Chapitre 2**

### **II**

#### *Environnement et outil de travail*

<b>II.1</b>	Introduction	23
<b>II.2</b>	L'automate S7-300 CPU 314-2DP	23
<b>II.3</b>	Programmation de l'API S7-300 de Siemens	24
<b>II.4</b>	TIA Portal	25
<b>II.5</b>	Vue du portail et vue du projet	26
<b>II.5.1</b>	Vue du portail	27
<b>II.5.2</b>	Vue du projet	27
<b>II.6</b>	Création d'un projet et configuration d'une station de travail	28
<b>II.6.1</b>	Création d'un projet	28
<b>II.6.2</b>	Configuration et paramétrage du matériel	29
<b>II.6.3</b>	Adressage des Entrées/Sorties	30
<b>II.6.4</b>	Compilation et chargement de la configuration matérielle	30
<b>II.6.5</b>	Connexion PC/ CPU	31
<b>II.7</b>	Les variables API	32
<b>II.7.1</b>	Adresses symbolique et absolue	32
<b>II.7.2</b>	Table des variables API	33
<b>II.7.3</b>	Signalisation des erreurs dans la table des variables	35
<b>II.7.4</b>	Renommer / réassigner des variables	35
<b>II.8</b>	Les opérations binaires	36



<b>II.8.1</b>	Interrogation à 1 et à 0	36
<b>II.8.2</b>	Affection / Mise à 1 et mise à 0	37
<b>II.8.3</b>	Fonctions logiques ET, OU, OU Exclusif	37
<b>II.9</b>	Conclusion	38

## **Chapitre 3**

### **III**

#### ***Problématique et solution proposée***

<b>III.1</b>	Introduction	39
<b>III.2</b>	Cahier de charges	40
<b>III.2.1</b>	Description de l'unité de traitement des eaux	40
<b>III.3</b>	Principe de fonctionnement de l'unité de traitement des eaux	41
<b>III.3.1</b>	Mode production	41
<b>III.3.1.1</b>	Démarrage de la procédure	41
<b>III.3.1.2</b>	Fin de production et lavage	41
<b>III.3.2</b>	Mode lavage	43
<b>III.3.3</b>	Mode CIP	43
<b>III.3.4</b>	Mode BYPASS	43
<b>III.3.5</b>	Mode manuel	44
<b>III.4</b>	le schéma de puissance et de commande	44
<b>III.4.1</b>	Le schéma de puissance	44
<b>III.4.2</b>	Le Schéma de commande	45
<b>III.5</b>	Conversion du schéma électrique établi en langage à contact	45
<b>III.6</b>	Conclusion	55

## **Chapitre 4**

### **IV**

#### ***programmation et supervision***

<b>IV.1</b>	Introduction	59
-------------	--------------	----

<b>IV.2</b>	La supervision	60
<b>IV.3</b>	WinCC sous TIA Portal	60
<b>IV.4</b>	Variables HMI	60
<b>IV.5</b>	Création de la supervision	60
<b>IV.5.1</b>	Insertion de l'automate	61
<b>IV.5.2</b>	Sélection du pupitre opérateur ou HMI de visualisation (Supervision)	61
<b>IV.5.3</b>	Présentation du pupitre opérateur utilisé	62
<b>IV.5.3.1</b>	Caractéristiques SIMATIC HMI TP900 Comfort	62
<b>IV.5.3.2</b>	Configuration de L'HMI	63
<b>IV.6</b>	Les variables HMI	66
<b>IV.7</b>	Vue Marche Manuel de l'unité de traitement des eaux	67
<b>IV.8</b>	Connexion HMI / API	67
<b>IV.9</b>	Conclusion	68
	<b>Conclusion Générale</b>	69
	<b>Bibliographie</b>	71

## *Notations et Abréviations*

<b>API</b>	: Automate <b>P</b> rogrammable <b>I</b> ndustriel
<b>IHM</b>	: Interface <b>H</b> omme <b>M</b> achine
<b>P.O</b>	: <b>P</b> artie <b>O</b> pérative
<b>P.C</b>	: <b>P</b> artie <b>C</b> ommande
<b>TOR</b>	: <b>T</b> out ou <b>R</b> ien
<b>CPU</b>	: <b>C</b> entral <b>P</b> rocessing <b>U</b> nit
<b>CP</b>	: <b>C</b> arte de <b>C</b> ommunication ( <b>C</b> ommunication <b>P</b> rocessors)
<b>E/S</b>	: <b>E</b> ntree/ <b>S</b> ortie
<b>PS</b>	: <b>P</b> ower <b>S</b> upply (Alimentation)
<b>DI</b>	: <b>D</b> igital <b>I</b> nput (Entrées TOR)
<b>DO</b>	: <b>D</b> igital <b>O</b> utput (sortie TOR)
<b>AI</b>	: <b>A</b> nalog <b>I</b> nput (Entrée Analogique)
<b>AO</b>	: <b>A</b> nalog <b>I</b> nput (Sortie analogique)
<b>PLC</b>	: <b>P</b> rogrammation <b>L</b> ogiciel <b>C</b> ontroller
<b>IM</b>	: <b>I</b> nterface <b>M</b> odule (Coupleur)
<b>GRAFCET</b>	: <b>G</b> raphe <b>F</b> onctionnel de <b>C</b> ommande <b>E</b> tape <b>T</b> ransition
<b>OB</b>	: <b>B</b> loc d'organisation
<b>FB</b>	: <b>B</b> loc <b>F</b> onctionnel
<b>FC</b>	: <b>F</b> onction
<b>DB</b>	: <b>B</b> locs de <b>D</b> onnées

<b>LD</b>	: Ladder Diagram
<b>ST</b>	: Structured Text
<b>CONT</b>	: langage à <b>contacts</b>
<b>LIST</b>	: Langage <b>Liste</b>
<b>LOG</b>	: portes <b>logiques</b>
<b>PPE</b>	: <b>Pompe</b>
<b>M</b>	: <b>Mémoire</b>
<b>IL</b>	: <b>Instruction List</b>
<b>FBD</b>	: <b>Fontion Bloc Diagram</b>
<b>SR\RS</b>	: <b>Set /Reset</b>
<b>BOOL</b>	: <b>Boolien</b>
<b>INT</b>	: <b>Integer (entier)</b>
<b>SM</b>	: <b>Modules De Simulation</b>
<b>FM</b>	: <b>Module de fonction</b>
<b>SFC</b>	: <b>Sequential Function Chart</b>
<b>MPI</b>	: <b>Message Passing Interface</b>
<b>TIA PORTAL</b>	: <b>Totally Integrated Automation</b>
<b>CIP</b>	: <b>Clean In Place</b>
<b>FL1</b>	: <b>Flotteur Niveau Bas Citerne Eau Brute</b>
<b>FL2</b>	: <b>Flotteur Niveau Haut Citerne Eau Traitee</b>
<b>FL3</b>	: <b>Flotteur Niveau Bas Bac De Lavage</b>
<b>CI</b>	: <b>Capteur De Pression Bp</b>

<b>C2</b>	: <i>Capteur De Pression Hp</i>
<b>SN1</b>	: <i>Sonde De Niveau Citerne Eau Brute (Ultrason)</i>
<b>SN2</b>	: <i>Sonde Niveau Citerne Eau Traitee (Ultrason)</i>
<b>PD1</b>	: <i>Pompe Doseuse Anti Scalant</i>
<b>PD2</b>	: <i>Pompe Doseuse Chlore</i>
<b>EV1</b>	: <i>Electrovanne D'entree Hp</i>
<b>EV2</b>	: <i>Electrovanne Flushing</i>
<b>EV3</b>	: <i>Electrovanne De Bypass Osmoseur</i>
<b>PPE HP</b>	: <i>Pompe <b>H</b>aut <b>P</b>ression</i>
<b>TEMPO</b>	: <i>Temporisation</i>
<b>NF</b>	: <i>Normalement <b>f</b>ermé</i>
<b>NO</b>	: <i>Normalement <b>o</b>uvert</i>

# Liste des figures

Figure	Titre	Page
<b>Chapitre 1</b>		
<b>Fig. I.1</b>	: Schéma d'un système automatisé	4
<b>Fig. I.2</b>	: Chaîne d'information	4
<b>Fig. I.3</b>	: Chaîne d'énergie	5
<b>Fig. I.4</b>	: Les principaux avantages des automatisations	8
<b>Fig. I.5</b>	: Le schéma fonctionnel PC-PO du système	9
<b>Fig. I.6</b>	: Structure d'un système automatisé	9
<b>Fig. I.7</b>	: Prés-actionneurs	10
<b>Fig. I.8</b>	: Actionneurs	10
<b>Fig. I.9</b>	: Capteurs	11
<b>Fig. I.10</b>	: Automate SIEMENS s7-300/2DP	13
<b>Fig. I.11</b>	: Fonctionnement cyclique d'un automate.	14
<b>Fig. I.12</b>	: Automate programmable industriel SIEMENS	18
<b>Fig. I.13</b>	Structure interne d'un API	18
<b>Chapitre 2</b>		
<b>Fig. II.1</b>	: Migration de Step 7 à TIA Portal	24
<b>Fig. II.2</b>	: La plateforme Totally Integrated Automation Portal	25
<b>Fig. II.3</b>	: Vue du portail	26
<b>Fig. II.4</b>	: Vue du projet	26
<b>Fig. II.5</b>	: Vue du portail éclaté	27
<b>Fig. II.6</b>	: Vue du projet éclaté	27
<b>Fig. II.7</b>	: Création du projet	28
<b>Fig. II.8</b>	: Choix de L'API	29
<b>Fig. II.9</b>	: Configuration et paramétrage du matériel	29
<b>Fig. II.10</b>	: Adressages des entrées et sorties	30
<b>Fig. II.11</b>	: Compilation et chargement du programme dans L'API	31
<b>Fig. II.12</b>	: Connexion PC/ CPU	31
<b>Fig. II.13</b>	: Aperçu sur le chargement	32
<b>Fig. II.14</b>	: Signalisation des dans la table erreurs des variables	35

<b>Fig. II.15</b>	: Renommer / réassigner variables des	35
<b>Fig. II.16</b>	: Différents types de détecteur interrogé	36
<b>Fig. II.17</b>	: Set et Reset avec Bascule SR et RS	37
<b>Fig. II.18</b>	: Fonction logique OU	37
<b>Fig. II.19</b>	: Fonction logique ET	38

### Chapitre 3

<b>Fig. III.1</b>	: Synoptique de la station de traitement des eaux	40
<b>Fig. III.2</b>	: Schéma de puissance	44
<b>Fig. III.3</b>	: Circuit de commande d'unité de traitement des eaux	45
<b>Fig. III.4</b>	: les conditions initiales de démarrage production	45
<b>Fig. III.5</b>	: EV1 en marche	46
<b>Fig. III.6</b>	: Temporisation pour le fonctionnement d'EV1	46
<b>Fig. III.7</b>	: les pompes d'eau brute en marche	46
<b>Fig. III.8</b>	: Compteur de niveau d'eau dans ppe1	46
<b>Fig. III.9</b>	: Maintenance de ppe1	47
<b>Fig. III.10</b>	: Compteur de niveau d'eau dans ppe2	47
<b>Fig. III.11</b>	: Maintenance de ppe2	47
<b>Fig. III.12</b>	: Temporisation pour le fonctionnement des ppe1 et ppe2	47
<b>Fig. III.13</b>	: Les pompe HP, PD1 et PD2 en marche	48
<b>Fig. III.14</b>	: Fin de production	48
<b>Fig. III.15</b>	: Les conditions de fin de production et lavage	48
<b>Fig. III.16</b>	: Temporisation pour arrêt les pompe d'eau brute	48
<b>Fig. III.17</b>	: EV2 et phase 1 en marche	49
<b>Fig. III.18</b>	: Fin production en Phase 2	49
<b>Fig. III.19</b>	: Lavage de fin production	49
<b>Fig. III.20</b>	: Set de lavage de fin production	49
<b>Fig. III.21</b>	: Lavage de fin de production Phase 1	50
<b>Fig. III.22</b>	: Lavage de fin de production dans Phase 2	50
<b>Fig.III.23</b>	: Reset EV1 et set EV2	50
<b>Fig.III.24</b>	: Les pompe HP et PD1 en marche	50
<b>Fig.III.25</b>	: Reset d'ev1	51
<b>Fig.III.26</b>	: Les pompe HP et PD1 en marche	51
<b>Fig.III.27</b>	: EV3, pompe eau brute et PD2 en marche	51
<b>Fig.III.28</b>	: Marche et arrêt de ppe1	52
<b>Fig.III.29</b>	: Marche et arrêt de ppe2	52

<b>Fig.III.30</b>	: Marche et arrêt d'EV1	52
<b>Fig.III.31</b>	: Marche et arrêt d'EV2	53
<b>Fig.III.32</b>	: Marche et arrêt d'EV3	53
<b>Fig.III.33</b>	: Marche et arrêt de la pompe PD1	53
<b>Fig.III.34</b>	: Marche et arrête la pompe PD2	54
<b>Fig.III.35</b>	: Marche et arrêt de la pompe HP	54
<b>Fig.III.36</b>	: Block FC1 de mode production	54
<b>Fig.III.37</b>	: Block FC2 de mode lavage	55
<b>Fig.III.38</b>	: Block FC3 de mode CIP	55
<b>Fig.III.39</b>	: Block FC4 de mode By-pass	55
<b>Fig.III.40</b>	: Block FC5 de mode manuelle	55
<b>Fig.III.41</b>	: Remise à zéro du système	56
<b>Fig.III.42</b>	: Echèle analogique de capteur de pression C1	56
<b>Fig.III.43</b>	: Echèle analogique de capteur de pression C2	57
<b>Fig.III.44</b>	: Echèle analogique de capteur analogique SN1	57
<b>Fig.III.45</b>	: Echèle analogique de capteur analogique SN2	58

#### **Chapitre 4**

<b>Fig. IV.1</b>	: Insertion ou l'ajout de l'automate choisi-S7-314-2DP	61
<b>Fig. IV.2</b>	: Choix de pupitre de visualisation	61
<b>Fig. IV.3</b>	: SIMATIC HMI TP900 Comfort	62
<b>Fig. IV.4</b>	: Connexion HMI/API	63
<b>Fig. IV.5</b>	: Représentation vue	63
<b>Fig. IV.6</b>	: Configurations des paramètres des alarmes	63
<b>Fig. IV.7</b>	: Navigations intervues	64
<b>Fig. IV.8</b>	: Vue système	64
<b>Fig. IV.9</b>	: Les boutons	64
<b>Fig. IV.10</b>	: Vue racine du projet	65
<b>Fig.IV.11</b>	: HMI vue globale de l'unité de traitement des eaux réalisée	65
<b>Fig. IV.12</b>	: Les variables HMI	66
<b>Fig. IV.13</b>	: Vue Marche Manuel de l'unité de traitement des eaux	67
<b>Fig. IV.14</b>	: Connection HMI /API	67



# Liste des Tableaux

Tableau	Titre	Page
<b>Chapitre 2</b>		
<b>II.1</b>	: Variables d'entrées et sorties avec leurs adresses	53

## ملخص

يقدم هذا العمل الأتمتة والإشراف على وحدة معالجة المياه بناءً على استخدام وحدة التحكم المنطقية الصناعية القابلة للبرمجة S7-314 D-2P Siemens والواجهة التي تعمل باللمس. بعد دراسة معمقة، وتحليل المواصفات، تم تطوير برنامج يستند على استخدام برنامج TIA portal V15 والذي يتم نقله بعد ذلك إلى وحدة المبرمج الآلي الصناعي PLC لإدارة التشغيل التلقائي والإشراف على وحدة معالجة المياه قيد النظر وللإجابة والرد وفقاً للمواصفات المطلوبة التي من خلالها تم تقديم وصف مفصل لمختلف تقنيات الأتمتة المختلفة للأنظمة الصناعية وتحديداً التكنولوجيا القائمة على استخدام وحدات سيمنز. وتم تخصيص جزء كبير للبرمجة باستخدام برنامج TIA Portal V15، مع إبراز الخطوات التي يجب اتباعها لإنشاء مشروع أتمتة، وتكوين الأجهزة، وتطوير البرنامج ومحاكاته، وتم أيضاً تطوير وتصميم واجهة الإنسان والآلة بواسطة برنامج Wincc المرن المدمج في برنامج TIA Portal V15 للسماح للمشغل بمشاهدة وتحليل سلوك النظام في الوقت الفعلي.

الكلمات المفتاحية: الأتمتة، الإشراف، معالجة المياه، والواجهة بين الإنسان والآلة، وبرنامج TIA Portal V13، وبرنامج Wincc.

## Résumé

Ce travail présente l'automatisation et la supervision d'une unité de traitement des eaux basé sur l'utilisation de l'automate programmable industriel S7-314 D-2P Siemens et l'interface home machine. Après une étude profonde, l'analyse du cahier de charge, un programme a été élaboré à l'appui du logiciel TIA portal V15 qui est transféré par la suite à l'automate pour répondre conformément au cahier de charge et afin de gérer le fonctionnement automatique de l'unité de traitement des eaux considéré. Une description détaillée des différentes technologies d'automatisation des systèmes industriels et plus précisément la technologie basée sur l'utilisation des modules de la firme Siemens a été présentée. Une grande partie est consacrée à la programmation avec le logiciel TIA Portal, en mettant en évidence les étapes à suivre pour la création d'un projet d'automatisation, la configuration matérielle, l'élaboration du programme et sa simulation. Aussi, une l'interface Homme-Machine a été élaboré et conçu par le logiciel Wincc flexible intégrer dans le logiciel TIA portal V15 pour permettre à l'opérateur de visualiser et analyser le comportement du système en temps réel.

**Mots-clés :** Automatisation, supervision, traitement des eaux, interface homme-machine, le logiciel TIA Portal V15, le logiciel WinCC.

## Abstract

This work presents the automation and supervision of a water treatment unit based on the use of the industrial programmable logic controller S7-314 D-2P Siemens and a human machine interface. After an in-depth study, the analysis of considered problem, a program was developed based on TIA portal V15 software which is then transferred to PLC controller to respond in accordance with the specifications and in order to manage the operation of the considered automatic water treatment unit. A detailed description of the different automation technologies in industrial plant and specifically those using Siemens modules was presented. A large part is devoted to programming with the Tia Portal software, highlighting the steps to follow for the creation of an automation project, the hardware configuration, the development of the program and its simulation. Also, a Human-Machine interface has been developed and designed based on Wincc flexible software integrated into the TIA portal V15 software to allow the operator to view and analyze the behavior of the system in real time.

**Keywords:** Automation, supervision, water treatment station, human-machine interface, TIA Portal V13 software, Wincc software.

*INTRODUCTION*

*INTRODUCTION*

*GENERALE*

*GENERALE*

## Introduction Générale

Depuis que les progrès scientifiques et techniques ont commencé, le quotidien de l'homme a été nettement amélioré. Ce dernier se fait aider dans ses différentes tâches par des appareils et machines censés lui simplifier la vie. L'un des secteurs les plus affectés par ces changements est celui de l'industrie. L'automatisation des processus a complètement révolutionné la manière dont les ouvriers travaillent. Le segment de l'automatisation industrielle a pour but principal de développer des solutions et des produits destinés aux industries. De manière simplifiée, l'automatisation industrielle consiste en l'optimisation des processus industriels grâce à des systèmes automatisés. En d'autres termes, l'utilisation des technologies dans des processus spécifiques est requise. L'objectif visé par cette automatisation est l'augmentation de la productivité et de l'autonomie, l'amélioration des conditions de travail et la simplification de certaines opérations.

Avec l'automatisation des processus de fabrication et la réduction de l'effort humain, les entreprises industrielles sont plus productives. Les ouvriers exécutent les opérations plus rapidement et en toute sécurité, car l'intervention humaine n'est plus requise, ou beaucoup moins, dans les services considérés comme dangereux.

L'automatisation est un levier important dans la transformation numérique de toute entreprise. Avec le marché qui devient de plus en plus exigeant et concurrentiel, cet ensemble de technologies s'impose. Elle regroupe un ensemble de technologies permettant de réaliser un certain nombre de tâches tout en réduisant les interventions humaines. Utilisable dans tous les secteurs, l'automatisation se décline sous diverses formes.

La puissance de l'automatisation et l'autonomie vont définir l'objectif de la transformation numérique des entreprises dans un futur proche. L'automatisation et l'informatique autonome sont motivés par le besoin accru de rapidité, d'agilité et d'innovation.

Ce travail présente l'automatisation et la supervision d'une unité de traitement des eaux basé sur l'utilisation de l'automate programmable industriel S7-314 D-2P Siemens et l'interface home machine. Après une étude profonde, l'analyse du cahier de charge, un programme a été élaboré à l'appui du logiciel TIA portal V15 qui est transféré par la suite à l'automate pour répondre conformément au cahier de charge et afin de gérer le fonctionnement automatique de l'unité de traitement des eaux considéré. Une description détaillée des différentes technologies d'automatisation des systèmes industriels et plus précisément la technologie basée sur l'utilisation les modules de la société Siemens ont été présentés. Une grande partie est consacrée à la programmation avec le logiciel TIA Portal, en mettant en évidence les étapes à suivre pour la création

d'un projet d'automatisation, la configuration matérielle, l'élaboration du programme et sa simulation. Aussi, une l'interface Homme-Machine a été élaboré et conçu par le logiciel Wincc flexible intégrer dans le logiciel TIA portal V15 pour permettre à l'opérateur de visualiser et analyser le comportement du système en temps réel.

Pour ce faire et pour l'établissement de cette étude, outre une introduction générale et une conclusion générale, le contenu de ce mémoire est réparti en quatre chapitres :

Le premier chapitre est réservé à la présentation en générales des systèmes automatisés et leurs principaux constituant toute en passant en revue et les différents parties et modules d'automate programmable et les outils et méthodes nécessaires pour l'étude de ce type de système d'automatisation industriel. Ainsi, les principaux avantages et inconvénients des automates programmables ont été présentés à la fin de ce chapitre.

Le deuxième chapitre est consacré à la prestation, l'étude et à l'analyse du cahier de charge de l'unité de traitement des eaux considérée afin d'établir les schémas de commande et de signalisation nécessaires, conformément à la description dans le cahier de charger.

Le troisième chapitre décrit d'une manière générale les différentes étapes pour la programmation du l'automate programmable S7-300 D2P-Siemens : programmation matériel (hard) et logiciel (soft) basée sur l'utilisation du logiciel de programmation Tia Portal V15 de la solution proposée en utilisant le langage de programmation Ladder.

Le dernier chapitre est dédié au développement et la programmation d'un pupitre de signalisation pour la commande et à la supervision de la station de traitement des eaux envisagée conformément à description du cahier de charge.

A la fin de ce travail, une conclusion générale est donnée avec des perspectives à envisager pour la continuation du présent travail.

# *CHAPITRE I*

*Généralité sur les Systèmes Automatisés*

## I.1. Introduction

Dans l'industrie, les automatismes sont devenus indispensables : ils permettent d'effectuer quotidiennement les tâches les plus ingrates, répétitives et, dangereuses. Parfois, ces automatismes sont d'une telle rapidité et d'une telle précision, qu'ils réalisent des actions impossibles pour un être humain.

Les systèmes automatisés sont une source de transformation dans les milieux de travail principalement au plan de l'organisation du travail. Ils ont éliminé plusieurs tâches dangereuses en plus de diminuer la pénibilité du travail et ce, dans toutes les entreprises quelle que soit leur taille.

L'automatisation des opérations renforce la productivité, la fiabilité, la disponibilité et les performances. Elle peut également réduire les coûts d'exploitation. Le passage à des opérations en service réduit offre un bon retour sur investissement

Ce chapitre est consacré à la présentation en générales des systèmes automatisés et leurs principaux constituant toute en passant en revue et les différents parties et modules d'automate programmable et les outils et méthodes nécessaires pour l'étude de ce type de système d'automatisation industriel. Ainsi, les principaux avantages et inconvénients des automates programmables ont été présentés à la fin de ce chapitre.

## I.2. Les systèmes automatisés

Un système automatisé est composé de plusieurs éléments qui exécutent un ensemble de tâches programmées sans que l'intervention de l'homme ne soit nécessaire comme : le passage à niveau automatique, la porte de garage, etc. Il est composé de :

- Chaîne d'information / La partie commande (PC): elle donne les ordres et reçoit les informations de l'extérieur ou de la partie opérative. Elle peut se présenter sous trois manières différentes : un boîtier de commande, un microprocesseur (cerveau électronique), ou un ordinateur
- Chaîne d'énergie / La partie opérative (PO): c'est la partie d'un système automatisé qui effectue le travail. Autrement dit, c'est la machine. C'est la partie qui reçoit les ordres de la partie commande et qui les exécute. Elle comporte les capteurs et les actionneurs :
  - Un actionneur est un élément de la partie opérative qui est capable de produire une action physique tel qu'un déplacement, un dégagement de chaleur, une émission de lumière ou de son à partir de l'énergie qu'il a reçu.
  - Un capteur est un élément de la partie opérative qui permet de recueillir des informations et de les transmettre à la partie commande. Les capteurs sont choisis en

fonction des informations qui doivent être recueillies (température, son, lumière, déplacement, position).

### I.3. Schéma d'un système automatisé

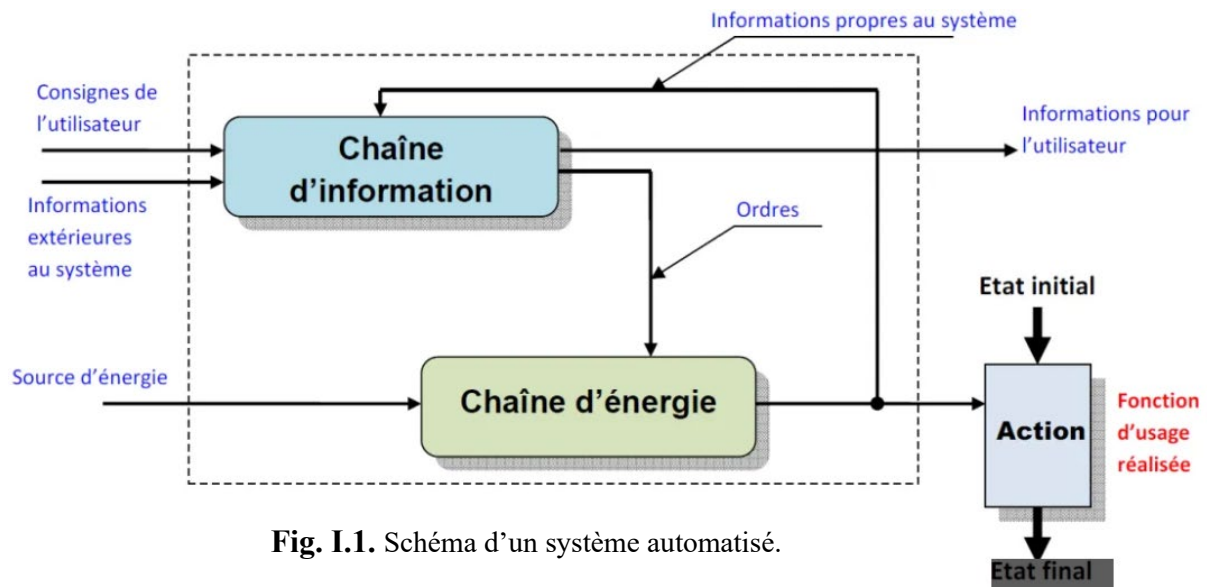


Fig. I.1. Schéma d'un système automatisé.

### I.4. Chaîne d'information

C'est la partie du système automatisé qui capte l'information et qui la traite. On peut découper cette chaîne en plusieurs blocs fonctionnels.

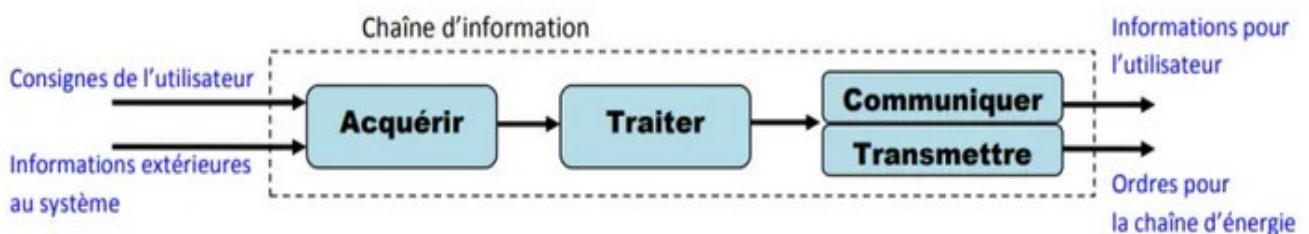


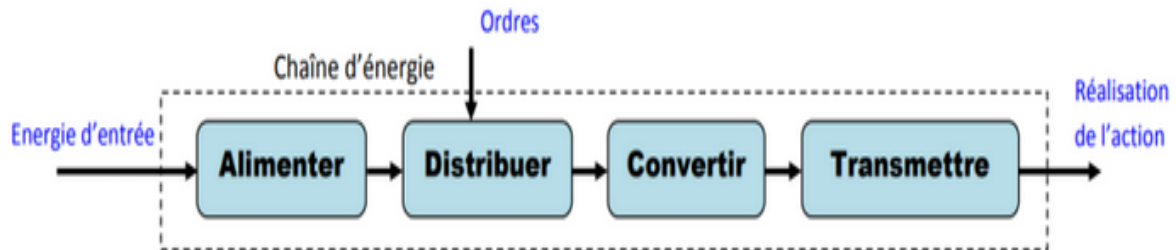
Fig. I.2. Chaîne d'information.

- **Acquérir** : Fonction qui permet de prélever des informations à l'aide de capteurs.
- **Traiter** : C'est la partie commande composée d'un automate ou d'un microcontrôleur.
- **Communiquer** : Cette fonction assure l'interface l'utilisateur et/ou d'autres systèmes.
- **Transmettre** : Cette fonction assure l'interface avec l'environnement de la partie commande.



## I.5. Chaîne d'énergie

Une chaîne d'énergie l'ensemble des procédés qui vont réaliser une action. On peut découper cette chaîne en plusieurs blocs fonctionnels.



**Fig. I.3.** Chaîne d'énergie.

- **Alimenter** : Mise en forme de l'énergie externe en énergie compatible pour créer une action.
- **Distribuer** : Distribution de l'énergie à l'actionneur réalisée par un distributeur ou un contacteur.
- **Convertir** : L'organe de conversion d'énergie appelé actionneur peut être un vérin, un moteur...
- **Transmettre** : Cette fonction est remplie par l'ensemble des organes mécaniques de transmission de mouvement et d'effort : engrenages, courroies, accouplement, embrayage...

## I.6. L'automatisme

Le besoin constant de réaliser des tâches complexes nécessitant beaucoup de main-d'œuvre a conduit les développeurs à se lancer dans le développement des logiciels d'opérations automatisées actuels. Le nombre et la diversité des produits ont considérablement augmenté pour inclure la planification, la gestion des messages des consoles, la sauvegarde, la restauration, l'optimisation des performances, etc. [1].

L'automatisation rend les services plus disponibles et réactifs. Elle évite aussi les risques d'accident liés aux erreurs humaines. La sécurité passe aussi au premier plan grâce à la mise en place de systèmes de sauvegarde.

Le but de la technique de l'automatisation est d'automatiser les machines et installations de telle sorte qu'elles travaillent de manière autonome avec efficacité et un faible taux d'erreur. Selon la complexité des installations, il est possible d'atteindre des degrés d'automatisation différents.

## I.6.1 Objectifs de l'automatisation

L'automatisation sert à remplacer un système à logique câblé par un appareil électronique programmable, adapté à l'environnement industriel, qui réalise des fonctions d'automatisme pour assurer la commande de pré actionneur et d'actionneur à partir d'informations logiques analogiques ou numériques, et la surveillance en temps réel de processus industriels. L'automatisation a pour objectif [2] :

- Accroître la productivité et augmentation de la qualité des produits ;
- Suppression de la pénibilité des tâches et simplifier le travail de maintenance ;
- Augmenter la sécurité des installations et du personnel ;
- Réaliser des économies de matière ;
- Eliminer les tâches répétitives ;
- Economiser les matières premières et l'énergie ;
- S'adapter à des contextes particuliers ;
- Maintenir la qualité.

## I.6.2 Les différents types d'automatisation

Les systèmes automatisés vont alors être classés en fonction de la nature des informations de commande ou de mesure, et également en fonction de la nature du traitement de ces informations. On distingue deux types d'informations : analogiques et discrètes (logiques).

L'automatisation industrielle permet le contrôle des machines et des processus utilisés dans l'industrie par des systèmes autonomes, grâce à l'utilisation de capteurs, d'actionneurs et de technologies telles que la robotique et les logiciels informatiques. Elle est actuellement utilisée dans une multitude de secteurs tels que l'industrie agroalimentaire ou la production de biens de grande consommation. Il existe 3 catégories d'automatisation industrielle dont le niveau de complexité et d'interaction avec l'humain varie.

### I.6.2.1. L'automatisation fixe ou rigide

Cette forme d'automatisation industrielle est la moins flexible. Elle est souvent utilisée pour exécuter des tâches répétitives avec des équipements dédiés afin d'améliorer l'efficacité de la production et les cadences. Une fois qu'une solution d'automatisation fixe est configurée, il est difficile de modifier le processus ou de reconfigurer l'équipement.

Cette forme d'automatisation se caractérise par un investissement initial élevé et convient aux produits fabriqués en grandes quantités.

### **I.6.2.2. L'automatisation programmable**

Elle est généralement utilisée pour la production de lots de produits où les instructions du système automatisé changent au fil du temps en fonction du produit en cours de fabrication. Le programme peut être reprogrammé pour chaque lot avec les spécifications données et les tâches de traitement et de séquençage nécessaires. Ce processus de changement prend du temps car dans la plupart des cas, l'équipement doit également être reconfiguré d'un lot à l'autre, conduisant ainsi à des périodes non productives. Les cadences de production dans l'automatisation programmable sont généralement plus faibles que dans l'automatisation fixe, car l'équipement est conçu pour faciliter le changement de produit plutôt que pour la spécialisation à un seul produit. L'automatisation programmable est souvent utilisée pour des volumes moyens.

### **I.6.2.3. L'automatisation flexible ou douce**

L'automatisation flexible ou douce : utilisée pour des volumes de production faibles, elle est similaire à l'automatisation programmable en ce sens qu'elle offre une flexibilité pour les changements de produits. Cependant, le grand avantage de l'automatisation flexible est que les changements de produits sont acheminés via le système de contrôle et se produisent rapidement et automatiquement, éliminant ainsi le temps nécessaire pour reconfigurer l'équipement entre les lots.

## **I.7. Les principaux avantages de l'automatisation**

Les systèmes automatisés offrent de nombreux avantages. Ils effectuent à la place des humains des travaux dangereux et physiques. Les tâches répétitives et fatigantes aussi sont effectuées par la machine. Ce n'est pas pour rien que le progrès de l'automatisation est en lien étroit avec la croissance démographique. Car avec l'augmentation de la population est né le besoin en biens de haute qualité et la production de masse est née.

Outre le soulagement offert aux humains, la technique de l'automatisation présente des avantages supplémentaires : les machines offrent une performance continue de haut niveau. Leur rythme de travail dépasse considérablement la vitesse des processus effectués manuellement. Par ailleurs, la technique de l'automatisation améliore la qualité des produits et réduit les coûts de personnel. Les avantages de l'automatisation des opérations les plus cités ont été les suivants : réduction des coûts, productivité, disponibilité, fiabilité et performances (figure ci-dessous).

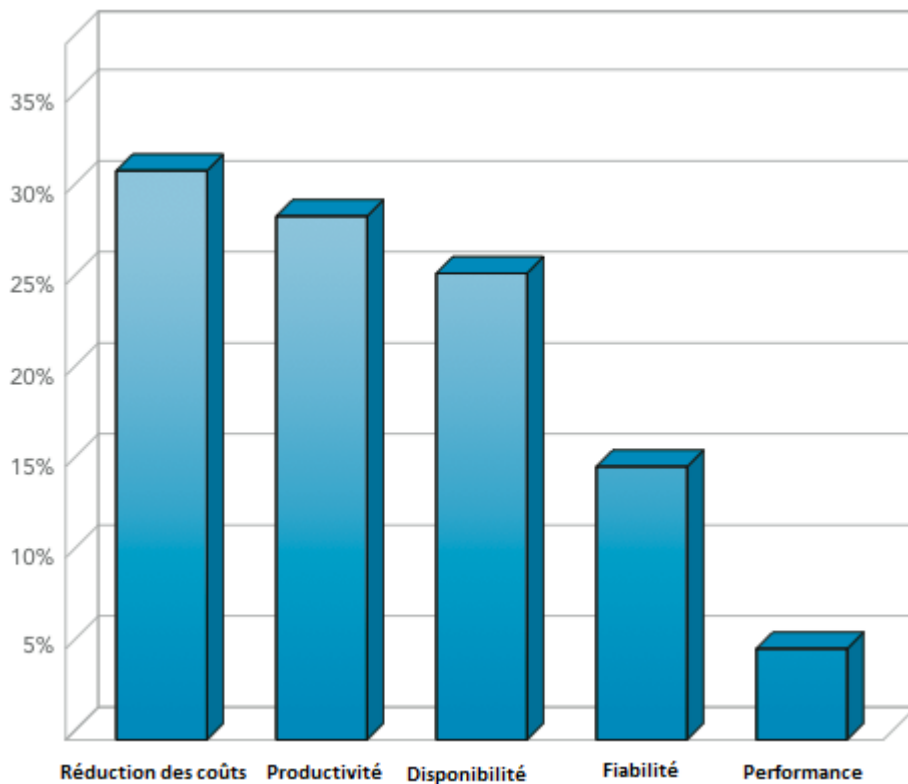


Fig. I.4. Les principaux avantages des automatisations.

## I.8. Les principales parties d'un système automatisé

Les systèmes automatisés comportent deux parties : La partie commande : elle donne des ordres et traite les informations qu'elle reçoit, elle est constituée par un ordinateur (*PC,  $\mu p$ ,  $\mu c$ , API ...*) ou un circuit électronique. La partie opérative : elle exécute les ordres qu'elle reçoit de la partie commande (c'est elle qui traite la matière d'œuvre pour lui apporter une valeur ajoutée), elle effectue toutes les actions que le système doit faire. La figure I.5 présente le schéma fonctionnel *PC-PO* d'un system automatisé [3].

Un système automatisé peut, pour faciliter l'analyse, se représenter sous la forme d'un schéma identifiant trois parties (P.O ; P.C ; P.P) du système et exprimant leurs interrelations (Informations, Ordres, Comptes rendus, Consignes).

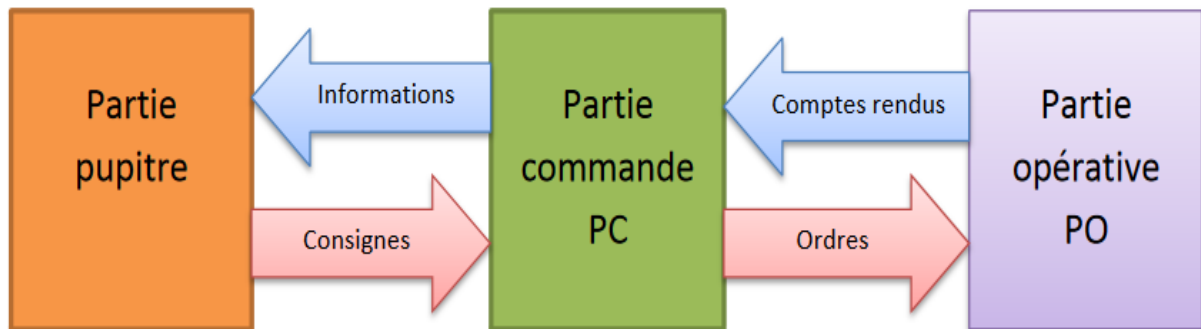


Fig. I.5. Le schéma fonctionnel PC-PO du système.

### I.9. Structure d'un système automatisé

Un système automatisé est toujours composé d'une partie commande (PC), et d'une partie opérative (PO), à lesquelles s'ajoute une troisième c'est la partie de dialogue ou une interface homme-machine. Tout système automatisé comporte les parties suivantes (Figure.I.6) [1].

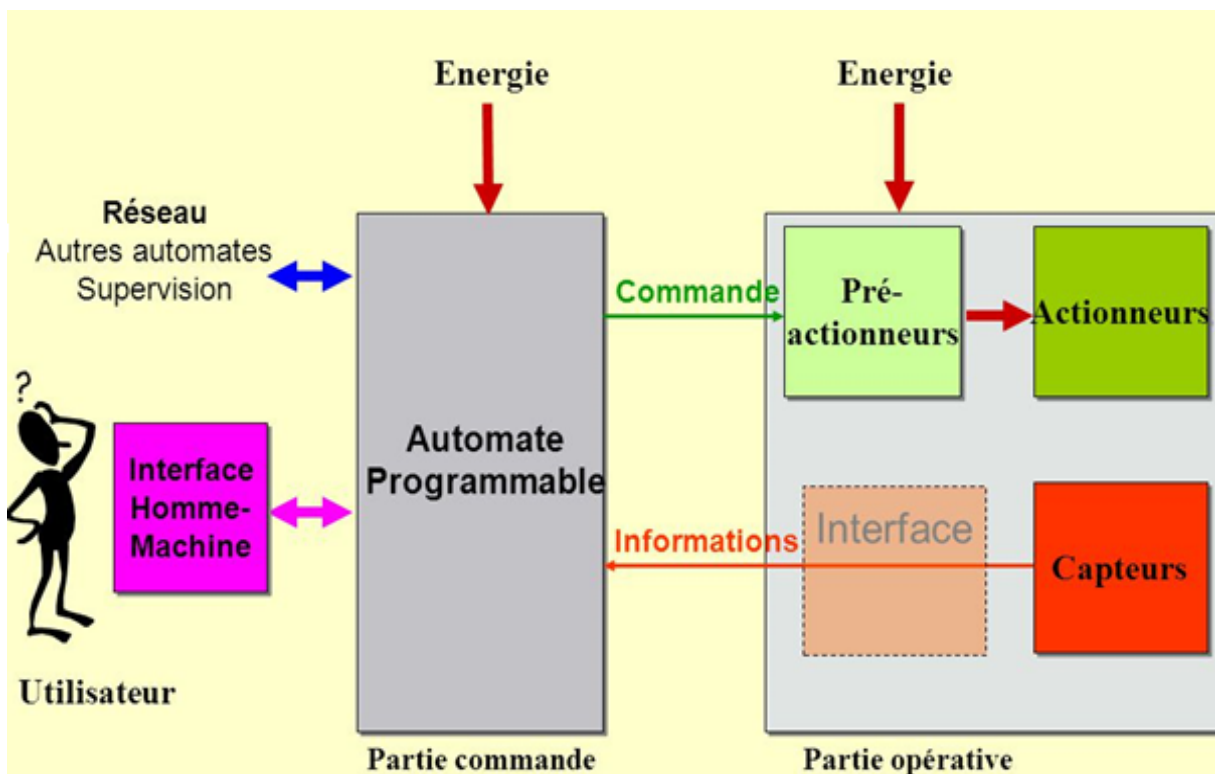


Fig. I.6. Structure d'un système automatisé.

## I.9. 1. La partie opérative

La partie opérative se compose des ensembles suivants :

- L'unité de production (effecteurs) dont la fonction est de réaliser la fabrication ou la transformation pour laquelle elle remplit un rôle dans le processus industriel.
- Les pré-actionneurs qui sont directement dépendants des actionneurs et sont nécessaires à leur fonctionnement (distributeur pour un vérin...etc)
- Les actionneurs qui apportent à l'unité de production l'énergie mécanique nécessaire à son fonctionnement à partir d'une source d'énergie extérieure (cas d'un moteur).
- Les capteurs qui créent, à partir de grandeurs physiques de natures divers (déplacement, température, pression...etc.), des informations utilisables par la partie de commande.

Elle comporte les éléments suivants :

- **Pré-actionneur** : est un constituant dont le rôle est de distribuer, sur ordre de la partie commande, l'énergie utile aux actionneurs. Les pré-actionneurs les plus utilisés sont les contacteurs (pour les moteurs électriques) et les distributeurs (pour les vérins pneumatiques) (Figure.I.7).



Distributeur pneumatique



Distributeur électro Pneumatique



Pré-actionneur électrique

**Fig. I.7.** Prés-actionneurs.

- **Actionneur** : transforme l'énergie d'entrée qui lui est appliquée en une autre type énergie de sortie (généralement mécanique) utilisable par un Effecteur pour fournir une action définie (Figure.I.8).



Moteur électrique



Moteur pneumatique



Vérin pneumatique

**Fig. I.8.** Actionneurs.

- **Capteur** : est un élément de prélèvement et de codage d'informations sur un processus ou sur l'environnement du système. Il convertit une grandeur physique (position, vitesse, ...) en une information appelée compte-rendu et compréhensible par la partie Commande (Figure.I.9).



Capteur de pression



Capteur de position



Détecteurs de fumée

**Fig. I .9.** Capteurs.

## I.9.2. La partie commande

La partie commande se compose des ensembles suivants :

- Les interfaces d'entrée qui transforment les informations issues des capteurs placés sur la partie opérative ou dans la partie dialogue en informations de nature et d'amplitude compatible avec les caractéristiques technologiques du système.
- Les interfaces de sortie qui transforment les informations élaborées par l'unité de traitement en informations de nature et d'amplitude compatibles avec les caractéristiques technologiques des pré-actionneurs d'une part, des visualisations et avertisseurs d'autre part ;
- L'unité de traitement (automates programmables industriels API, ordinateur, microprocesseurs) qui élabore les ordres destinés aux actionneurs en fonction des informations reçues des différents capteurs et du fonctionnement à réaliser.

## I.9.3. La partie dialogue

La partie dialogue se compose de deux ensembles :

- Les visualisations et avertisseurs qui transforment les informations fournies par l'automate en informations perceptibles par l'homme (informations optiques ou sonores) ;
- Les capteurs qui transforment les informations fournies par l'homme (action manuelle sur un bouton poussoir) et informations exploitables par l'automate.

## I.10. Généralités sur les automates programmables

### I.10.1 Historique

Les Automates Programmables Industriels (API) sont apparus aux Etats-Unis vers la fin des années soixante, à la demande de l'industrie automobile américaine (General Motors) qui réclamait plus d'adaptabilité de leurs systèmes de commande. Les ingénieurs américains ont résolu le problème en créant un nouveau type de produit nommée automates programmables. Ils n'étaient rentables que pour des installations d'une certaine complexité, mais la situation a très vite changé, ce qui a rendu les systèmes câblés obsolètes. De nombreux modèles d'automates sont aujourd'hui disponibles ; depuis les nano automates bien adaptés aux machines et aux installations simples avec un petit nombre d'entrées/sorties, jusqu'aux automates multifonctions capables de gérer plusieurs milliers d'entrées/sorties et destinés au pilotage de processus complexes.

### I.10.2 L'automate programmable

Un automate programmable industriel (API) est une machine électronique spécialisée dans la conduite et la surveillance en temps réel de processus industriels et tertiaires Il exécute une suite d'instructions introduites dans ses mémoires sous forme de programmes, et s'apparente par conséquent aux machines de traitement de l'information. [1]

Trois caractéristiques fondamentales le distinguent des outils informatiques tels que les ordinateurs utilisés dans les entreprises et le tertiaire :

- Il peut être directement connecté aux capteurs et pré-actionneurs grâce à ses entrées/sorties industrielles.
- Il est conçu pour fonctionner dans des ambiances industrielles sévères (Température, vibrations, microcoupures de la tension d'alimentation, parasites, etc...).
- Enfin, sa programmation à partir de langages spécialement développés pour le traitement de fonctions d'automatisme facilite son exploitation et sa mise en œuvre [8].

L'automate utilisé dans notre projet appartient à la gamme SIMATIC S7 de SIEMENS (Le S7-300-CPU314-2DP), qui est un automate modulaire pour les applications d'entrée et de milieu de gamme, avec possibilité d'extensions jusqu'à 32 modules, et une mise en réseau par l'interface multipoint (MPI), PROFIBUS et INDUSTRIAL Ethernet.





Fig. I. 10 : Automate Siemens S7-300-2DP.

### I.10.3 Critères de choix d'un automate

Le choix d'un automate programmable est en premier lieu le choix d'une société ou d'un groupe et les contacts commerciaux et expériences vécues sont déjà un point de départ. Les grandes sociétés privilégieront deux fabricants pour faire jouer la concurrence et pouvoir se retourner en cas de perte de vitesse de l'une d'entre elles. Le personnel de maintenance doit toutefois être formé sur ces matériels et une trop grande diversité des matériels peut avoir de graves répercussions. Un automate utilisant des langages de programmation de différents types pour assurer les mises au point et dépannages dans les meilleures conditions. La possession d'un logiciel de programmation est aussi source d'économies (achat du logiciel et formation du personnel). Des outils permettant une simulation des programmes sont également souhaitables. Il faut ensuite quantifier les besoins en termes de : [3]

- **Nombre d'entrées / sorties** : le nombre de cartes peut avoir une incidence sur le nombre de racks dès que le nombre d'entrées / sorties nécessaires devient élevé.
- **Type de processeur** : la taille mémoire, la vitesse de traitement et les fonctions spéciales offertes par le processeur permettront le choix dans la gamme souvent très étendue.
- **Fonctions ou modules spéciaux** : certaines cartes (commande d'axe, pesage ...) permettront de soulager le processeur et devront offrir les caractéristiques souhaitées (résolution, ...).
- **Fonctions de communication** : l'automate doit pouvoir communiquer avec les autres systèmes de commande (API, supervision ...) et offrir des possibilités de communication avec des standards normalisés (PROFIBUS ...).

### I.10.4 Avantages des automates programmables

- Evolutivité : très favorable à l'évolution et très utilisé en reconstruction d'armoire.
- Fonctions : assure les fonctions conduites, dialogue, communication et sûreté.
- Taille des applications : gamme importante d'automate.
- Vitesse : temps de cycle de quelque ms.
- Modularité : haute modularité. Présentation modularité en rack
- Architecture de commande : centralisée ou décentralisée avec l'apparition d'une offre importante en choix de réseaux, bus de terrain, blocs E/S déportés.
- Maintenance : échanges standards et aide au diagnostic intégré.

### I.11. Traitement du programme automate

Tous les automates fonctionnent selon le même mode opératoire

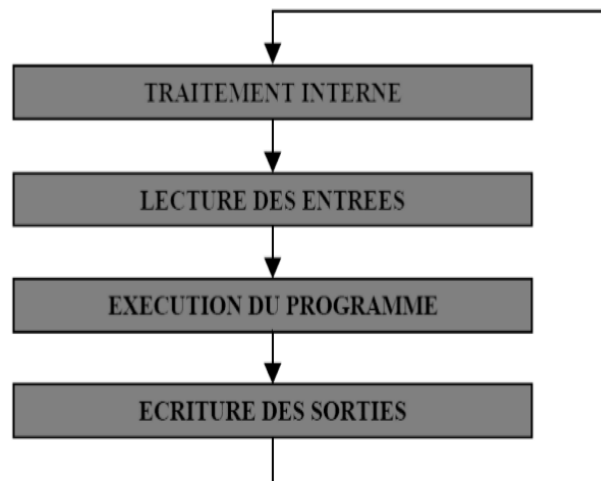


Fig. I.11. Fonctionnement cyclique d'un automate.

- **Traitement interne** : L'automate effectue des opérations de contrôle et met à jour certains paramètres systèmes (détection des passages en RUN / STOP, mises à jour des valeurs de l'horodateur, ...).
- **Lecture des entrées** : L'automate lit les entrées (de façon synchrone) et les recopie dans la mémoire image des entrées.
- **Exécution du programme** : L'automate exécute le programme instruction par instruction et écrit les sorties dans la mémoire image des sorties.

- **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) [3].

### I.11.1 Langage de programmation de l'API

Les programmes utilisés avec les API peuvent être écrits dans différents formats. Pour que les ingénieurs ayant peu de connaissances en programmation puissent élaborer des programmes pour les API, le langage à contacts a été conçu. La plupart des fabricants d'automates ont adopté cette méthode d'écriture des programmes.

Toutefois, puisque chacun a eu tendance à développer ses propres versions, une norme internationale a été établie pour le langage à contacts et, par voie de conséquence, pour toutes les méthodes de programmation employées avec les API [1]. La norme IEC 1131-3 définit cinq langages qui peuvent être utilisés pour la programmation des automates programmables industriels. Ces cinq langages sont :

#### I.11.1.1 Le langage LD (Ladder Diagram)

C'est une représentation graphique d'équations booléennes combinant des contacts (en entrée) et des relais (en sortie). Il permet la manipulation de données booléennes, à l'aide de symboles graphiques organisés dans un diagramme comme les éléments d'un schéma électrique à contacts. Les diagrammes LD sont limités à gauche et à droite par des barres d'alimentation. [5]

#### I.11.1.2 Le langage IL (Instruction List)

Un langage textuel de bas niveau. Il est particulièrement adapté aux applications de petite taille. Les instructions opèrent toujours sur un résultat courant (ou registre IL). L'opérateur indique le type d'opération à effectuer entre le résultat courant et l'opérande. Le résultat de l'opération est stocké à son tour dans le résultat courant.

Un programme est une liste d'instructions. Chaque instruction doit commencer par une nouvelle ligne, et doit contenir un opérateur, complété éventuellement par des modificateurs et si c'est nécessaire pour l'opération, un ou plusieurs opérandes, séparés par des virgules (','). Une étiquette suivie de deux points (':') peut précéder l'instruction.

Si un commentaire est attaché à l'instruction, il doit être le dernier élément de la ligne.

Des lignes vides peuvent être insérées entre des instructions. Un commentaire peut être posé sur une ligne sans instruction. [9]

### **I.11.1.3 Le langage FBD (Function Block Diagram)**

C'est un langage graphique. Il permet la construction d'équations complexes à partir des opérateurs standards, de fonctions ou de blocs fonctionnels. Les principales fonctions sont :

- L'énoncé RETURN (peut apparaître comme une sortie du diagramme, si liaison connectée prend l'état booléen TRUE, la fin du diagramme n'est pas interprétée.
- Les étiquettes et les sauts conditionnels sont utilisés pour contrôler l'exécution du diagramme.
- Saut à une étiquette (le nom de l'étiquette est « LAB »).

### **I.11.1.4 Le langage ST (Structured Text)**

Un langage textuel de haut niveau dédié aux applications d'automatisation. Ce langage est principalement utilisé pour décrire les procédures complexes, difficilement modélisables avec les langages graphiques. C'est le langage par défaut pour la programmation des actions dans les étapes et des conditions associées aux transitions du langage SFC. Un programme ST est une suite d'énoncés. Les noms utilisés dans le code source (identificateurs de variables, constantes, mots clés du langage...) sont délimités par des séparateurs passifs ou des séparateurs actifs, qui ont un rôle d'opérateur. Des commentaires peuvent être librement insérés dans la programmation.

### **I.11.1.5 Le langage SFC (Sequential Function chart), ou GRAFCET**

C'est un langage graphique utilisé pour décrire les opérations séquentielles. Le procédé est représenté comme une suite connue d'étapes (états stables), reliées entre elles par des transitions, une condition booléenne est attachée à chaque transition. Les actions dans les étapes sont décrites avec les langages ST, IL, LD ou FBD. Les principales règles graphiques sont :

- Un programme SFC doit contenir au moins une étape initiale.
- Une étape ne peut pas être suivie d'une autre étape.
- Une transition ne peut pas être suivie d'une autre transition.

L'automate utilisé dans notre projet appartient à la gamme SIMATIC S7 de SIEMENS ; Le S7-300 « CPU 315-2 PN/DP », qui est un automate modulaire pour les applications d'entrée et de milieu de gamme, avec possibilité d'extensions jusqu'à 32 modules, et une mise en réseau par l'interface multipoint (MPI), PROFIBUS et INDUSTRIAL Ethernet.

## I.12. Architecture des automates programmables

De forme compacte ou modulaire, les automates sont organisés suivant l'architecture suivante :

- **Une unité centrale (CPU) :** assure le traitement de l'information et la gestion de l'ensemble des unités. Ce module comporte un microprocesseur, des circuits périphériques de gestion des entrées/sorties, des mémoires RAM et EEPROM nécessaires pour stocker les programmes, les données, et les paramètres de configuration du système.
- **Une alimentation (PS) :** à partir d'une tension 220V/50Hz ou dans certains cas de 24V fournit les tensions continues +/- 5V, +/-12V ou +/-15V.
- **Modules d'entrées et de sorties :**
  - Un ou plusieurs modules d'entrées 'Tout Ou Rien' (TOR) ou analogiques pour l'acquisition des informations provenant de la partie opérative (procédé à conduire).
  - Un ou plusieurs modules de sorties 'Tout Ou Rien' (TOR) ou analogiques pour transmettre à la partie opérative les signaux de commande. Il y a des modules qui intègrent en même temps des entrées et des sorties [4].
- **Module de fonction (FM) :** Ils réalisent les tâches de traitement des signaux de processus critique au niveau de temps et exigeant beaucoup de mémoire pour réduire la charge sur la CPU comme :
  - Le positionnement.
  - Régulation.
  - Comptage.
- **Processeur de communication(CP) :** Il s'en charge des tâches de communication par transmission série. Et aussi à établir des liaisons point à point avec : des autres automates SIMATIC ou d'autres constructeurs.
  - Interfaces série utilisant dans la plupart des cas comme support de communication, les liaisons RS-232 ou RS422/RS485.
  - Interfaces pour assurer l'accès à un bus de terrain.
  - Interface d'accès à un réseau Ethernet.
- **Modules de simulation (SM 374) :** c'est un module spécial qui offre à l'utilisateur la possibilité de tester son programme lors de la mise en service et en cours de fonctionnement. Dans le S7-300, ce module fonctionne telles que :

- La simulation des signaux de capteurs aux moyens d'interrupteurs.
  - La signalisation d'état des signaux de sortie par des LED.
- **Coupleurs** : Si l'application d'automatisation a besoins de plus de huit modules, il est possible de faire une extension de la configuration de la S7-300 en utilisant un châssis de base et trois châssis d'extension au maximum. Chaque châssis peut recevoir huit modules, alors le coupleur relie les différents châssis.



Fig. I. 11. Automate Programmable Industriel SIEMENS.

### I.13. Structure interne des automates programmables

La structure matérielle interne d'un API est donnée par la figure ci-dessous (Figure I.12).

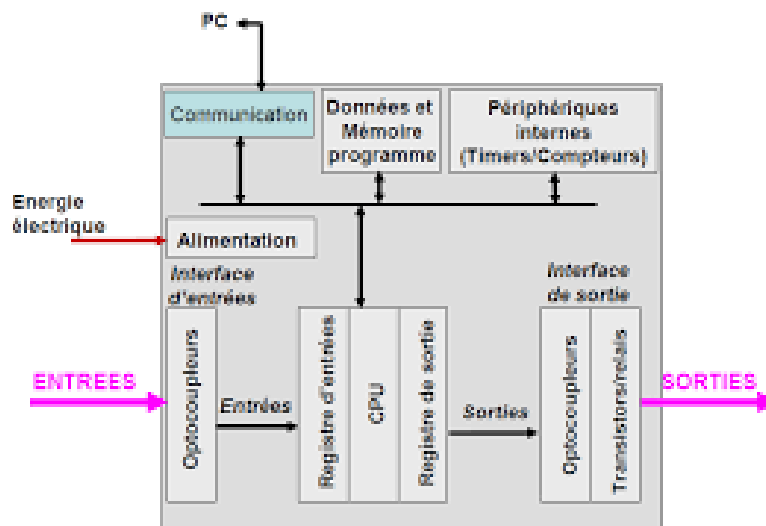


Fig. I. 12. Structure interne d'un API.

Les composants qui apparaissent sur ce schéma ci-dessus seront décrits en détails dans les paragraphes suivants :

### I.13.1 Le processeur

Le processeur a pour rôle principal le traitement des instructions qui constituent le programme de fonctionnement de l'application. Mais en dehors de cette tâche de base, il réalise également d'autres fonctions :

- Gestion des entrées/sorties.
- Surveillance et diagnostique de l'automate par une série de tests lancés à la mise sous tension ou cycliquement en cours de fonctionnement.
- Dialogue avec le terminal de programmation aussi bien pour l'écriture et la mise au point du programme qu'en cours d'exploitation pour des réglages ou des vérifications de données.

Le processeur est organisé autour d'un certain nombre de registres, ce sont des mémoires rapides permettant la manipulation des informations qu'elles retiennent, ou leur combinaison avec des informations extérieures.

Les principaux registres existants dans un processeur sont :

- **L'accumulateur** : C'est le registre où s'effectuent les opérations du jeu d'instruction, les résultats sont contenus dans ce registre spécial.
- **Le registre d'instruction** : Il reçoit l'instruction à exécuter et décode le code opération. Cette instruction est désignée par le pointeur.
- **Le registre d'adresse** : Ce registre reçoit, parallèlement au registre d'instruction, la partie opérande de l'instruction. Il désigne le chemin par lequel circulera l'information lorsque le registre d'instruction validera le sens et ordonnera le transfert.
- **Le registre d'état** : C'est un ensemble de positions binaires décrivant, à chaque instant, la situation dans laquelle se trouve précisément la machine.
- **Les piles** : Une organisation spéciale de registres constitue une pile, ces mémoires sont utilisées pour contenir le résultat de chaque instruction après son exécution. Ce résultat sera utilisé ensuite par d'autres instructions, et cela pour faire place à la nouvelle information dans l'accumulateur [5].

### I.13.2 Les mémoires

Un système de processeur est accompagné par un ou plusieurs types de mémoires.

Elles permettent :

- De stocker le système d'exploitation dans des ROM ou PROM,
- Le programme dans des EEPROM,

- Les données système lors du fonctionnement dans des RAM. Cette dernière est généralement secourue par pile ou batterie. On peut, en règle générale, augmenter la capacité mémoire par adjonction de barrettes mémoires.

### **I.13.3 Les modules d'entrées/sorties**

Ils assurent le rôle d'interface entre la CPU et le processus, en récupérant les informations sur l'état de ce dernier et en coordonnant les actions. Plusieurs types de modules sont disponibles sur le marché selon l'utilisation souhaitée :

- Modules TOR (Tout Ou Rien): l'information traitée ne peut prendre que deux états (vrai/faux, 0 ou 1 ...). C'est le type d'information délivrée par une cellule photoélectrique, un bouton poussoir ...etc.
- Modules analogiques : l'information traitée est continue et prend une valeur qui évolue dans une plage bien déterminée. C'est le type d'information délivrée par un capteur (débitmètre, capteur de niveau, thermomètre...etc.).
- Modules spécialisés : l'information traitée est contenue dans des mots codes sous forme binaire ou bien hexadécimale. C'est le type d'information délivrée par un ordinateur ou un module intelligent.

### **I.13.4 L'alimentation**

Elle a pour rôle de fournir les tensions continues nécessaires aux composants avec de bonnes performances, notamment face aux microcoupures du réseau électrique qui constitue la source d'énergie principale. La tension d'alimentation peut être de 5V, 12V ou 24V. D'autres alimentations peuvent être nécessaires pour les châssis d'extension et pour les modules entrées/sorties. Un onduleur est nécessaire pour éviter les risques de coupures non tolérées.

### **I.13.5 Liaisons de communication**

Elles Permettent la communication de l'ensemble des blocs de l'automate et des éventuelles extensions. Les liaisons s'effectuent :

- Avec l'extérieur par des borniers sur lesquels arrivent des câbles transportant les signaux électriques.
- Avec l'intérieur par des bus reliant divers éléments, afin d'échanger des données, des états et des adresses.



## I.14 TIA Portal (Totally Integrated Automation)

En réponse à la pression internationale croissante de la concurrence, il est aujourd'hui plus que jamais important d'exploiter à fond tous les potentiels d'optimisation sur l'ensemble du cycle de vie d'une machine ou d'une installation.

Des processus optimisés permettent de réduire le coût total de possession, de réduire le temps entre la conception et la commercialisation et d'améliorer la qualité. Cet équilibre parfait entre qualité, temps et coûts et plus que jamais le facteur décisif de la réussite industrielle.

Totally Integrated Automation apporte une réponse optimale à toutes les exigences et offre un concept ouvert vis à vis des normes internationales et de systèmes tiers. Avec ses six principaux caractéristiques systèmes et robustesse, Le TIA Portal accompagne l'ensemble du cycle de vie d'une machine ou d'une installation. L'architecture système complète offre des solutions complètes pour chaque segment d'automatisation sur la base d'une gamme de produits complète [6].

### I.14.1 Description du logiciel TIA Portal

La plateforme « TOTALLY INTERGRATED Automation Portal » est le nouvel environnement de travail Siemens qui permet de mettre en œuvre des solutions d'automatisation avec un système d'ingénierie intègre comprenant les logiciels Simatic Step7 et Winncc.

### I.14.2 Les avantages du logiciel TIA portal

- Programmation intuitive et rapide : avec des éditeurs de programmation nouvellement développés SCL, CONT, LOG, LIST et GRAPH.

- Efficacité accrue grâce aux innovations linguistiques de STEP 7 : programmation symbolique uniforme, ajout de blocs durant le fonctionnement, et bien plus encore.

- Performance augmentée grâce à des fonctions intégrées : simulation avec PLCSIM, télémaintenance avec TELESERVICE et diagnostic système cohérent.

- Technologie flexible : Fonctionnalité motion control évolutive et efficace pour les automates S7-1500 et S7-1200.

- Sécurité accrue avec security integrated : Protection du savoir-faire, protection contre la copie, protection d'accès et protection contre la falsification. – Environnement de configuration commun avec pupitres IHM et entraînements dans l'environnement d'ingénierie TIA Portal.

## **I.15. Conclusion**

Dans ce chapitre, on a présenté en générales des systèmes automatisés et leurs principaux constituants toute en passant en revue et les différents parties et modules d'automate programmable et les outils et méthodes nécessaires pour l'étude de ce type de système d'automatisation industriel. Ainsi, les principaux avantages et inconvénients des automates programmables ont été présentés à la fin de ce chapitre.

# *CHAPITRE II*

*Environnement Et Outil De Travail*

## II.1.Introduction

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

Ce chapitre est réservé à la présentation de l'automate SIMATIC S7-300 CPU 314-2DP et à la plateforme de programmation TIA Portal utilisés lors de la réalisation de ce travail toute en décrivant les différentes étapes de configuration matériel et de programmation nécessaires et susceptible d'être utilisées dans une telle applications.

## II.2. L'automate S7-300 CPU 314-2DP

Le système d'automatisation SIMATIC S7-300 est un automate modulaire de milieu de gamme. Il existe une gamme étendue de modules S7-300 pour répondre de manière optimale à différente tâche d'automatisation comme le SIMATIC S7-300 CPU 314-2DP.

L'automate S7 est constitué d'une alimentation (Modules PS), d'une CPU ainsi que des modules d'entrées / sorties avec :

- Modules d'extension IM pour configuration multirangées du S7-300 ;
- Modules de signaux SM pour entrées et sorties TOR et analogiques ;
- Modules de fonction FM pour fonctions spéciales ;
- Processeurs de communication CP pour la connexion au réseau

La CPU 314C-2DP est livrée avec une interface PROFIBUS DP et des entrées/sorties intégrées. Pour la CPU 314C-2DP, les protocoles du PROFIBUS sont disponibles :

- Interface DP en tant que maître
- Interface DP en tant qu'esclave

Le PROFIBUS-DP (Périphérique décentralisé) est le protocole pour la liaison avec des périphériques à temps de réaction très brefs. Une autre particularité est que, dans cette CPU, les adresses d'entrée/sortie des modules peuvent être paramétrées. La qualité de fonctionnement avec les données suivantes est particulièrement bien adaptée [9].

- Mémoire vive : 48Ko, Mémoire de chargement Micro Memory CARD enfichable 64Ko ;
- 4 Mo - 8192 octets DE/DA dont 992 octets centraux
- 512 octets AE/AA dont 248 octets centraux - 0,1 ms / 1K commandes
- 256 Compteurs - 256 temporisations - 256 octets de mémoire interne
- 24 DI dont 16 utiles pour des fonctions intégrées, et tous utilisables en entrées d'alarme
- 16 DO intégrés dont 4 sorties rapides

- 4 AI Courant/tension, 1 AI résistance intégrés
- 2 AO courant/tension intégrés
- 4 sorties d'impulsion (2,5kHz)
- Compteur 4 canaux et mesure avec capteur incrémentale 24V (60kHz)
- Fonction de position intégrée [9]

### II.3. Programmation de l'API S7-300 de Siemens

Le S7-300 est un automate modulaire de la famille SIMATIC pour les applications d'entrée et de milieu de gamme modulaire ou compacte et sans ventilateur, l'architecture décentralisée est facile à mettre en œuvre, ce qui en fait une solution économique pour les tâches les plus diverses dans les petites et moyennes applications. Il fournit des produits et des ressources complètes pour résoudre des tâches techniques telles que le comptage, le chronométrage, la mesure, la régulation, etc. En cas de développement ultérieur de l'application, l'installation de modules supplémentaires peut amener l'automate au niveau de performance requis.

L'API S7-300 est programmable à l'aide d'un PC avec le logiciel STEP 7 (ou à l'aide La plate-forme Siemens TIA Portal (Totally Integrated Automation) sous Windows qui offre les fonctions suivantes pour l'automatisation d'une installation :

- Configuration et paramétrage du matériel ;
- Paramétrage de la communication ;
- Test, mise en service et maintenance ;
- Documentation, archivage ;
- Fonctions de diagnostic et d'exploitation.



Fig. II.1. Migration de Step 7 à TIA Portal.

## II.4. TIA Portal

La plateforme Totally Integrated Automation Portal est le nouvel environnement de travail Siemens qui permet de mettre en œuvre des solutions d'automatisation avec un système d'ingénierie intégré comprenant les logiciels SIMATIC STEP 7 V15 et SIMATIC WinCC V15 (la version du programme utilisée dans notre travail) [10].



Fig. II.2. La plateforme Totally Integrated Automation Portal.

## II.5. Vue du portail et vue du projet

Lorsque l'on lance TIA Portal, l'environnement de travail se décompose en deux types de vue [10]. :

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

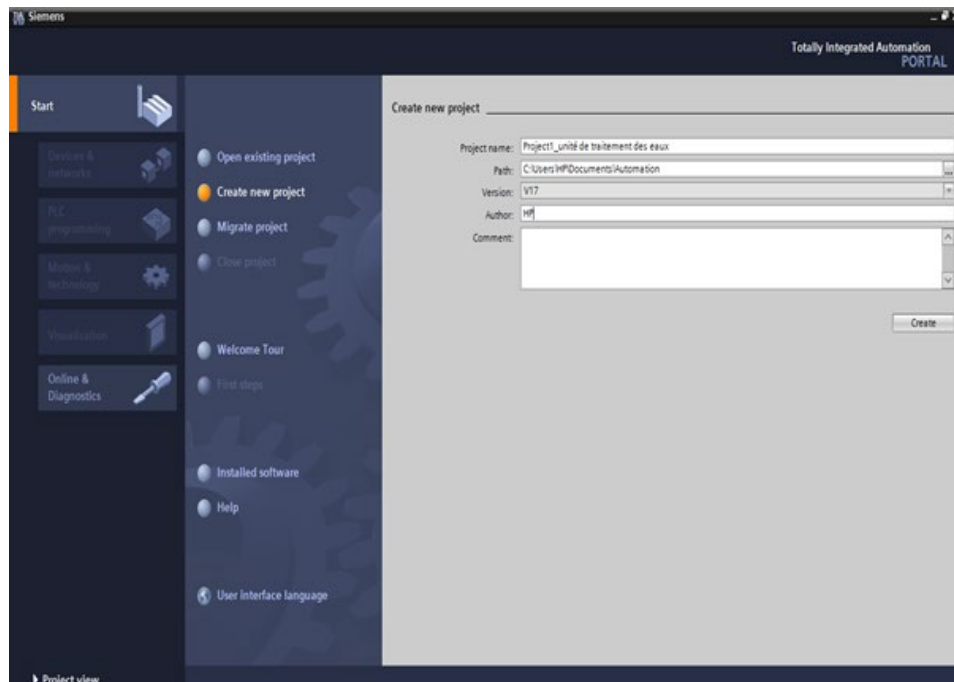


Fig. II.3. Vue du portail.

- **La vue du projet** : Elle comporte une arborescence avec les différents éléments du projet. Les éditeurs requis s'ouvrent en fonction des tâches à réaliser. Données, paramètres et éditeurs peuvent être visualisés dans une seule et même vue [10].

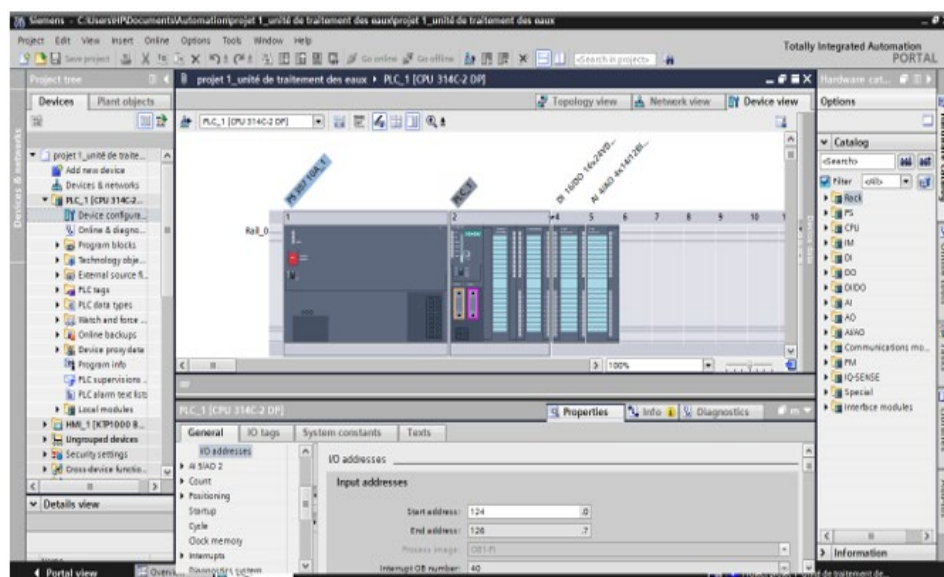


Fig. II.4. Vue du projet.

## II.5.1. 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.

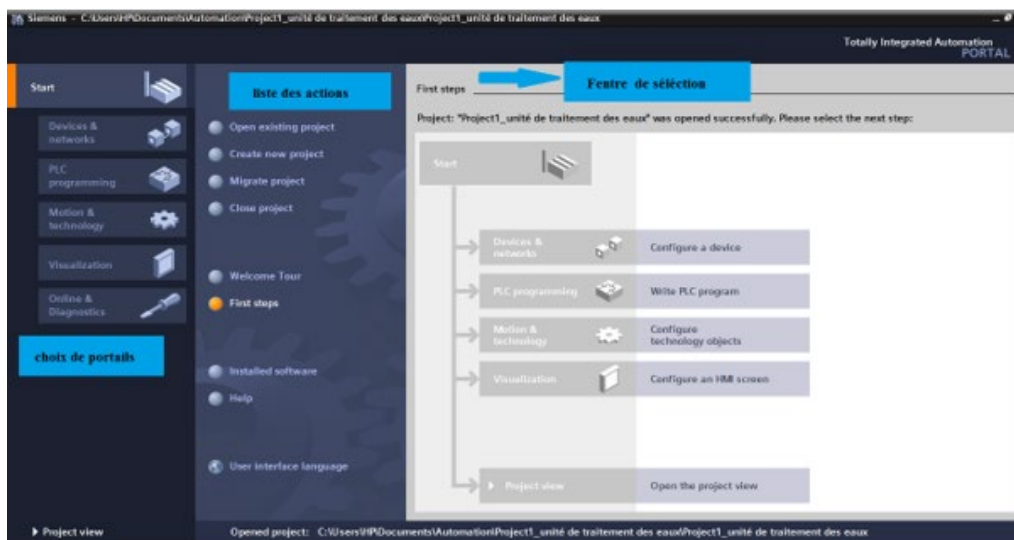


Fig. II.5. Vue du portail éclaté.

## II.5.2. Vue du projet

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

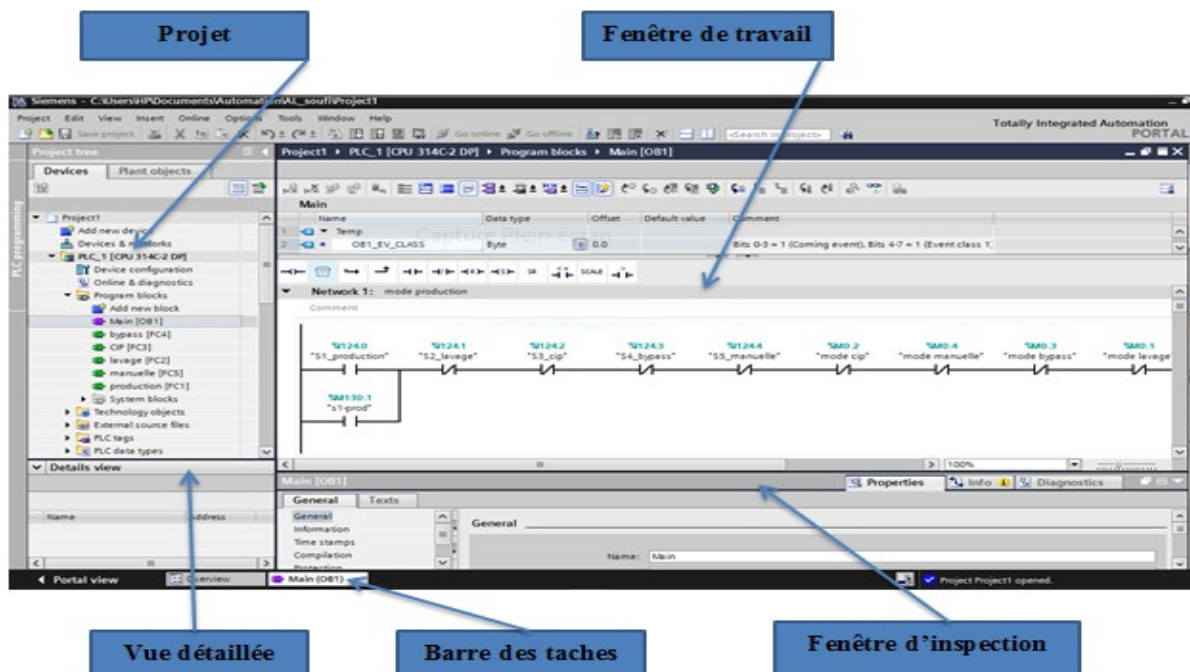


Fig. II.6. Vue du projet éclaté.



- La **fenêtre de travail** permet de visualiser les objets sélectionnés dans le projet pour être traités. Il peut s'agir des composants matériels, des blocs de programme, des tables des variables, des HMI,...
- La **fenêtre d'inspection** permet de visualiser des informations complémentaires sur un objet sélectionné ou sur les actions en cours d'exécution (propriété du matériel sélectionné, messages d'erreurs lors de la compilation des blocs de programme...).
- Les onglets de sélection de tâches ont un contenu qui varie en fonction de l'objet sélectionné (configuration matérielle → bibliothèques des composants, bloc de programme → instructions de programmation).
- Cet environnement de travail contient énormément de données. Il est possible de masquer ou réduire certaines de ces fenêtres lorsque l'on ne les utilise pas.
- Il est également possible de redimensionner, réorganiser, désancrer les différentes fenêtres.

## II.6. Création d'un projet et configuration d'une station de travail

### II.6.1. Création d'un projet

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

- On peut donner un nom au projet, choisir un chemin où il sera enregistré, indiquer un commentaire ou encore définir l'auteur du projet.
- Une fois que ces informations sont entrées, il suffit de cliquer sur le bouton « créer »

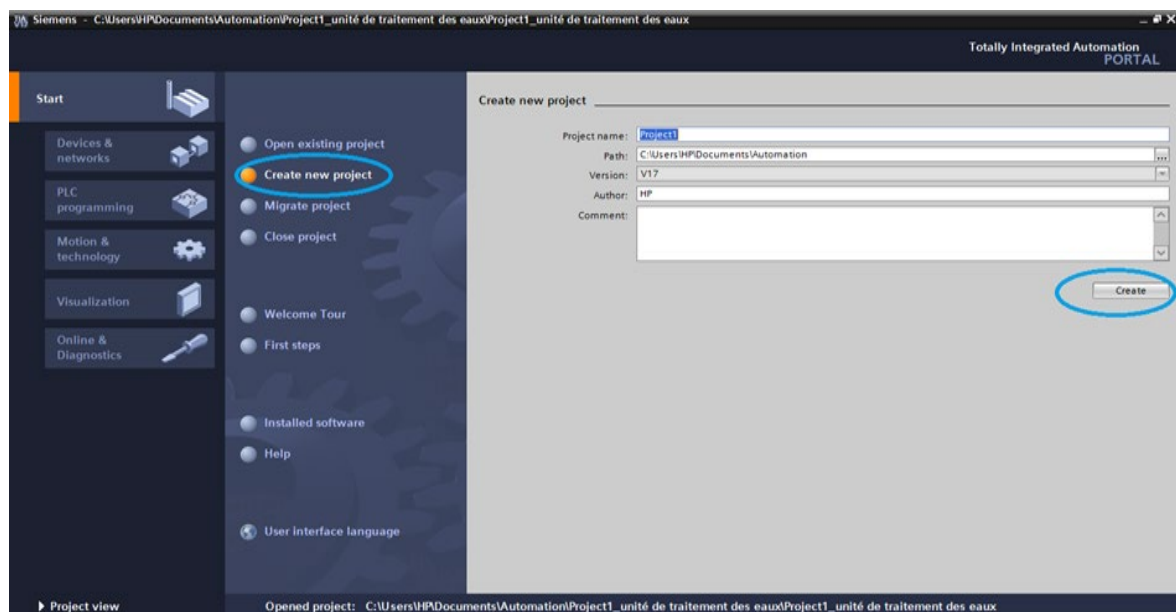


Fig. II.7. Création du projet.

## II.6.2. Configuration et paramétrage du matériel

Une fois notre projet est créé, on peut configurer la station de travail. La première étape consiste à définir le matériel existant. Pour cela, on peut passer par la vue du projet et cliquer sur « ajouter un appareil » dans le navigateur du projet.

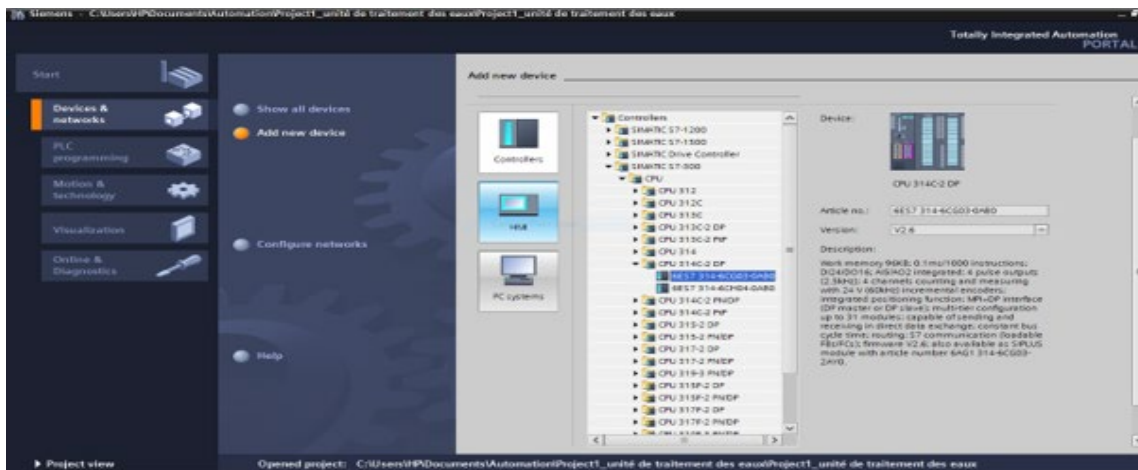


Fig. II.8. Choix de L'API.

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

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

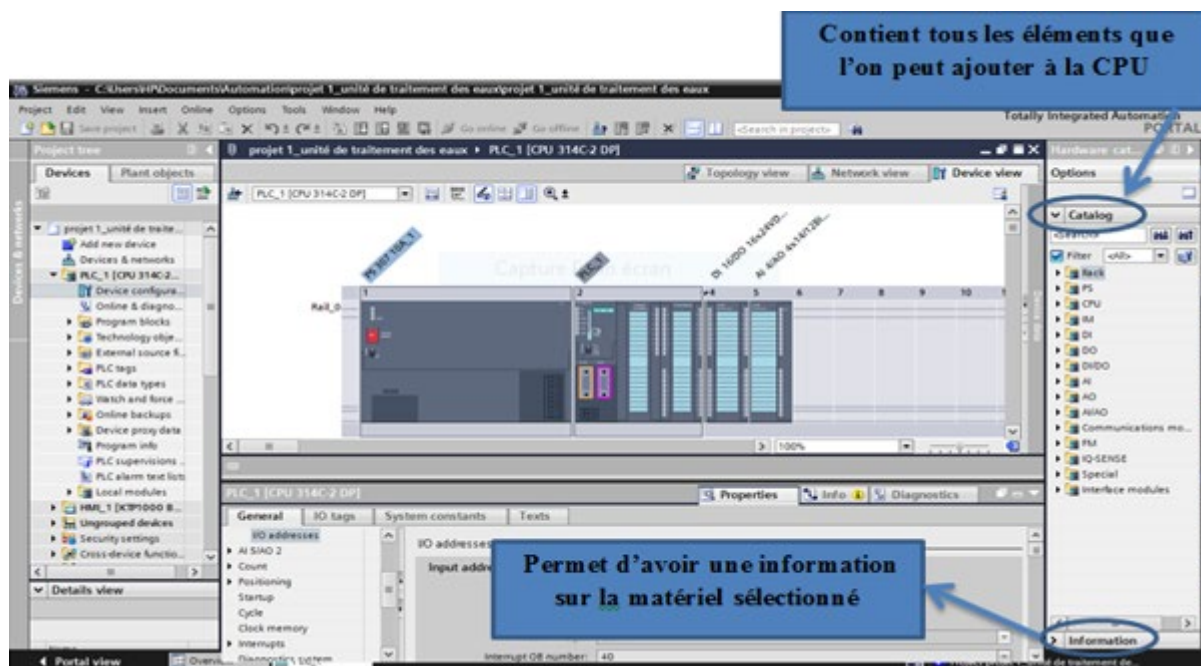


Fig. II.9. Configuration et paramétrage du matériel.

### II.6.3. Adressage des Entrées/Sorties

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 [10].

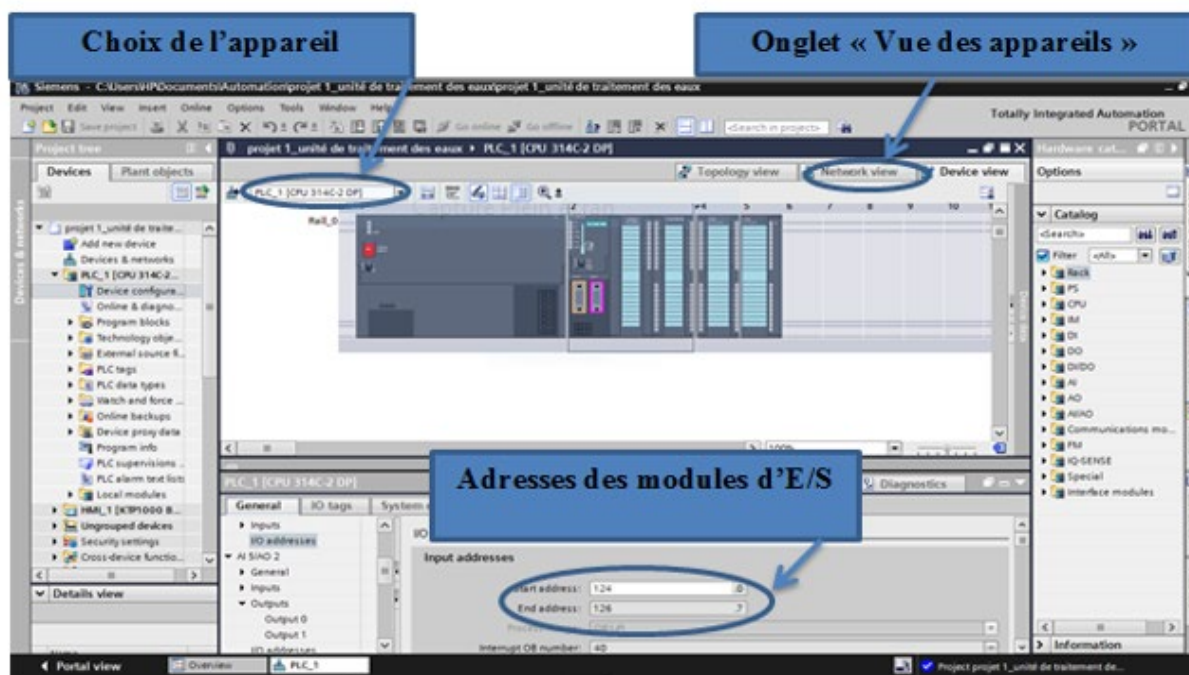


Fig. II.10. Adressages des entrées et sorties.

On sélectionne la CPU puis à l'aide des deux petites flèches (voir figure), on fait apparaître l'onglet « Vue d'ensemble des appareils ». Les adresses des entrées et sorties apparaissent. On fait la modification en entrant une nouvelle valeur dans la case correspondante.

### II.6.4. Compilation et chargement de la configuration matérielle

Une fois la configuration matérielle réalisée, il faut la compiler et la charger dans l'automate. La compilation se fait à l'aide de l'icône « compiler » de la barre de tâche. On sélectionne l'API dans le projet puis cliquer sur l'icône « compiler ». En utilisant cette manière, on effectue une compilation matérielle et logicielle. Une autre solution pour compiler est de faire un clic droit sur l'API dans la fenêtre du projet et de choisir l'option « Compiler → Configuration matérielle ».

Pour charger la configuration dans l'automate, on effectue un clic sur l'icône « charger dans l'appareil ». La fenêtre ci-dessous s'ouvre et vous devez faire le choix du mode de connexion (Profibus, MPI).

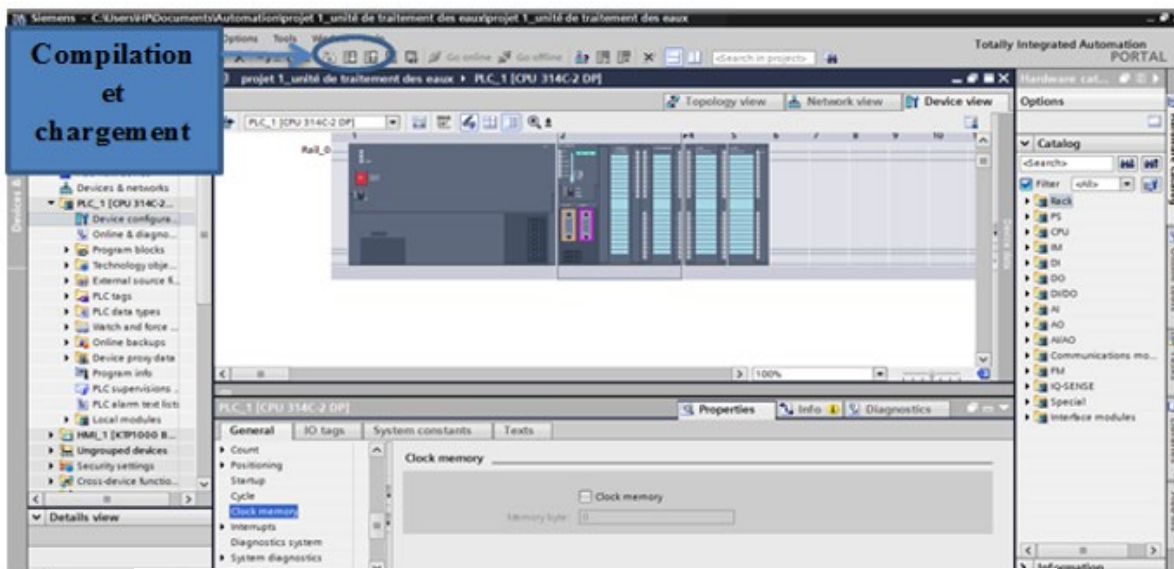


Fig. II.11. Compilation et chargement du programme dans L'API.

### II.6.5. Connexion PC/ CPU

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

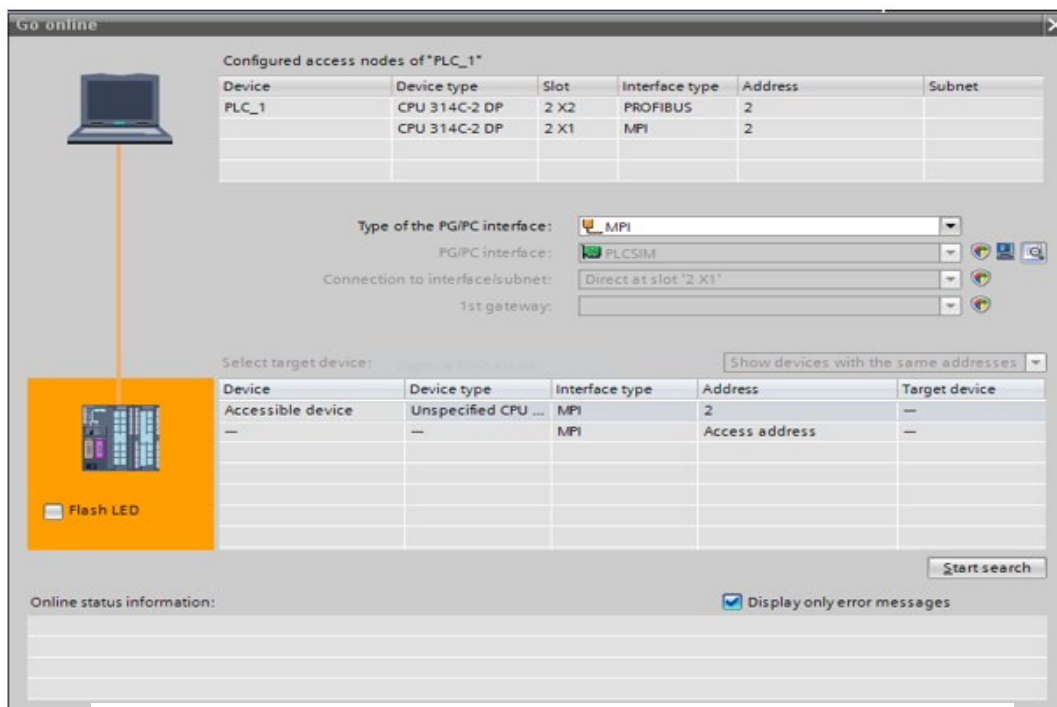


Fig. II.12. Connexion PC/ CPU.

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

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

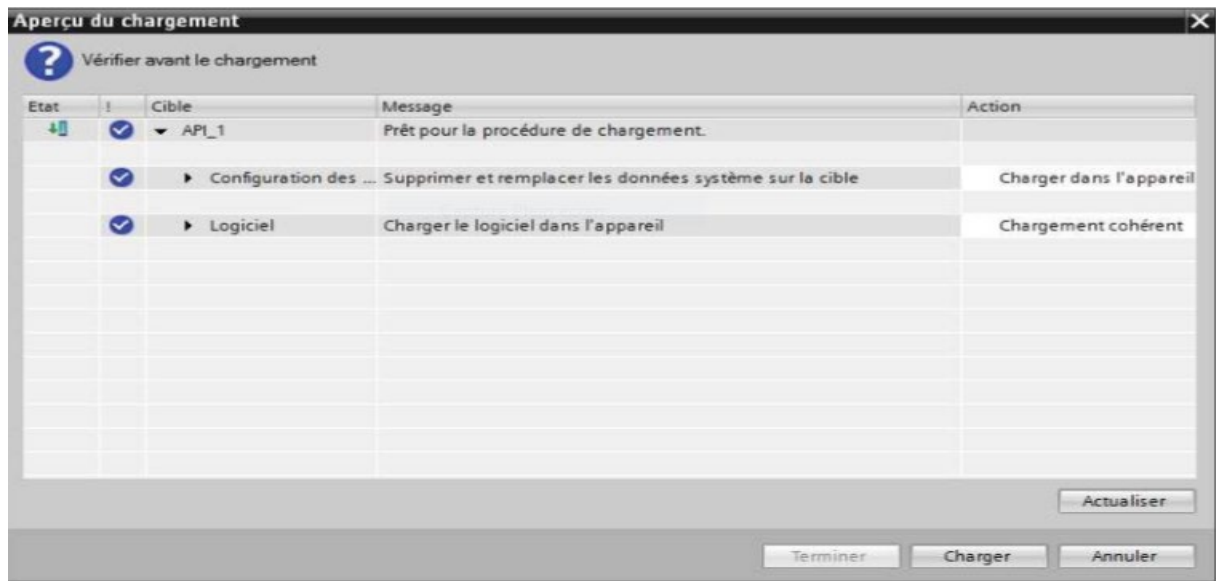


Fig. II.13. Aperçu sur le chargement.

## II.7. Les variables API

### II.7.1. Adresses symboliques et absolue

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

- **L'adresse absolue** représente l'identificateur d'opérande (I, Q, M,...) et son adresse et numéro de bit.
- **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.7.2 Table des variables API

C'est dans la table des variables API que l'on va pouvoir déclarer toutes les variables et les constantes utilisées le programme.

Lorsque l'on définit une variable API, il faut définir :

- Un nom : c'est l'adressage symbolique de la variable.
- Le type de donnée : BOOL, INT,...
- L'adresse absolue comme l'adresse Q1.5

On peut également insérer un commentaire qui nous renseigne sur cette variable.

Le commentaire peut être visible dans chaque réseau utilisant cette variable.

	Name	Data Type	ADDRESS
1	SN1_sup10	BOOL	%M80.0
2	SN2_sup10	BOOL	%M80.1
3	Tag_7	BOOL	%M50.0
4	Tag_8	BOOL	%M50.1
5	Tag_9	BOOL	%M50.2
6	Tag_10	BOOL	%M50.3
7	Tag_11	BOOL	%M50.4
8	Tag_12	BOOL	%M50.5
9	Tag_13	BOOL	%M50.6
10	Tag_14	BOOL	%M50.7
11	Tag_15	BOOL	%M51.0
12	Tag_1	INT	%IW752
13	Tag_2	BOOL	%M100.0
14	Tag_3	WORD	%MW10
15	C1_BP	REAL	%MD11
16	C1_INF	BOOL	%M0.5
17	Tag_4	INT	%IW754
18	Tag_5	BOOL	%M100.1
19	Tag_6	WORD	%MW16
20	C2_HP	REAL	%MD18
21	C2_SUP1	BOOL	%M0.6
22	Tag_16	INT	%IW758
23	Tag_17	WORD	%MW90
24	SN1_Analog	REAL	%MD92
25	Tag_18	INT	%IW760
26	Tag_19	WORD	%MW96
27	mode production	BOOL	%M0.0
28	CONDITION_INITIAL_PROD	BOOL	%M0.7
29	FL1_flotteur niveau bas eau brute	BOOL	%I124.5
30	FL2_flotteur niveau haut CITERNE EAU TRAITE	BOOL	%I124.6
31	EV1	BOOL	%Q124.2
32	phase1_lavage_prod	BOOL	%M1.6
33	FIN_PRODUCTION	BOOL	%M1.2
34	ppe2_eau brute	BOOL	%Q124.1
35	maintenance_ppe1	BOOL	%M1.0

36	ppe1_eau brute	BOOL	%Q124.0
37	compteur_ppe1	INT	%MW22
38	compteur_ppe2	INT	%MW24
39	fin_prod_phase1	BOOL	%M1.3
40	PPE_HP	BOOL	%Q124.5
41	phase2_lavage_fin_prod	BOOL	%M1.7
42	ppe_PD1	BOOL	%Q124.6
43	ppe_PD2	BOOL	%Q124.7
44	fin_prod_phase2	BOOL	%M1.4
45	EV2	BOOL	%Q124.3
46	LAVAGE_FIN_PROD	BOOL	%M1.5
47	FL3_niveau bas bac de lavage	BOOL	%I124.7
48	EV3	BOOL	%Q124.449
49	start_ppe1	BOOL	%M2.7
50	stop_ppe1	BOOL	%M3.3
51	RTH_ppe1	BOOL	%I125.0
52	start_ppe2	BOOL	%M2.3
53	stop_ppe2	BOOL	%M3.4
54	RTH_ppe2	BOOL	%I125.1
55	start_EV1	BOOL	%M2.0
56	stop_EV1	BOOL	%M3.0
57	start_EV2	BOOL	%M2.1
58	stop_EV2	BOOL	%M3.1
59	start_EV3	BOOL	%M2.2
60	stop_EV3	BOOL	%M3.2
61	start_PD1	BOOL	%M2.4
62	stop_PD1	BOOL	%M3.5
63	RTH_PD1	BOOL	%I125.2
64	start_PD2	BOOL	%M2.5
65	stop_PD2	BOOL	%M3.6
66	RTH_PD2	BOOL	%I125.3
67	START_PPE_HP	BOOL	%M2.6
68	STOP_PPE_HP	BOOL	%M3.7
69	RTH_PPE_HP	BOOL	%I125.4
70	S1_production	BOOL	%I124.0
71	S2_lavage	BOOL	%I124.1
72	S3_cip	BOOL	%I124.2
73	S4_bypass	BOOL	%I124.3
74	S5_manuelle	BOOL	%I124.4
75	mode lavage	BOOL	%M0.1
76	mode CIP	BOOL	%M0.2
77	mode BYPASS	BOOL	%M0.3
78	mode manuelle	BOOL	%M0.4
79	s1-prod	BOOL	%M130.1
80	s2-lavg	BOOL	%M130.2
81	s3-ciip	BOOL	%M130.3
82	s4-byypass	BOOL	%M130.4
83	s5-manueellee	BOOL	%M130.5

Tableau II.1 : Variables d'entrées et sorties avec leurs adresses.

### II.7.3 Signalisation des erreurs dans la table des variables

Lorsqu'il y a une erreur de syntaxe dans la table des variables API, celle-ci est signalée en rouge ou en orange. Lorsque l'on sélectionne la case colorée, un message signalant le type d'erreur apparaît. Une table des variables contenant des erreurs peut être enregistrée mais ne pourra pas être compilée et chargée dans l'automate.

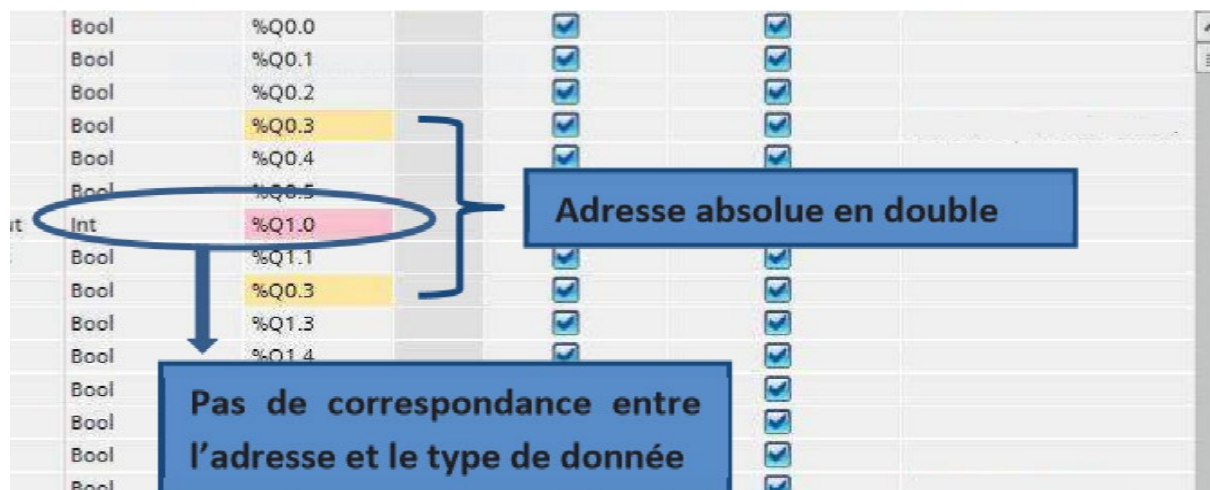


Fig. II.14. Signalisation des dans la table erreurs des variables.

### II.7.4 Renommer / réassigner des variables

Il est possible de renommer et réassigner les variables API dans la fenêtre de programmation. Pour cela, il faut faire un clic droit sur la variable en question et choisir l'option renommer ou réassigner.

- Renommer la variable donne l'accès aux paramètres Nom et Commentaire
- Réassigner la variable donne l'accès aux paramètres Section, Adresse, Type de données, Table des variables API et Commentaire.

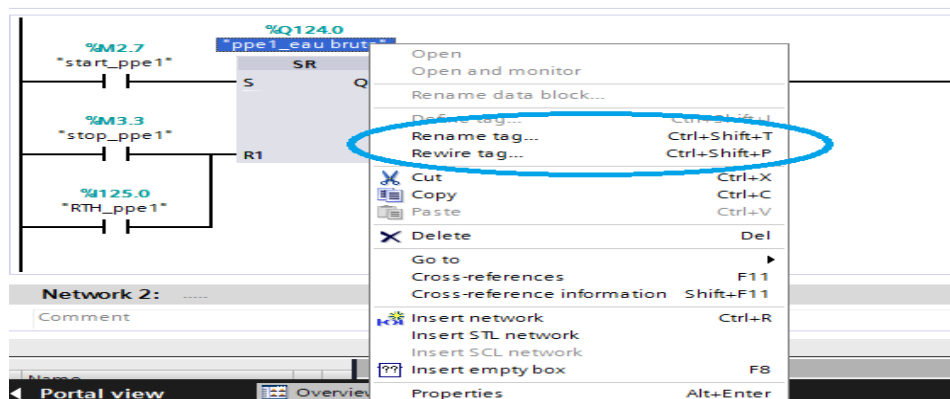


Fig. II.15. Renommer / réassigner des variables.




## II.8. Les opérations binaires


Les opérations binaires sont les opérations réalisées sur les variables de type « BOOL ». On retrouve dans ces opérations les fonctions logiques (ET, OU, OU Exclusif), les bascule RS, les détections de fronts, ...

Plusieurs langages de programmation sont disponibles : le schéma à contact (LADDER), les logigrammes (LOG) ou le langage LIST (instructions). Seul le langage LADDER sera utilisé dans ce cours.

### II.8.1 Interrogation à 1 et à 0

Les symboles d'interrogation à 1 et à 0 sont très utilisés en LADDER. C'est à partir de ces symboles que l'on va pouvoir réaliser des fonctions logiques, lancer des temporisations, des compteurs,...

 ⇒ Interrogation à 1 ou Contact NO (à fermeture). Il fournit le résultat d'interrogation 1 lorsque l'opérande interrogé est à l'état 1.

 ⇒ Interrogation à 0 ou Contact NF (à ouverture). Il fournit le résultat d'interrogation 1 lorsque l'opérande interrogé est à l'état 0.

Le résultat de l'interrogation va être différent en fonction du type de détecteur interrogé. Le tableau ci-dessous montre les différents cas qui peuvent être rencontrés.











Processus			Evaluation du programme dans l'automate				
Le capteur est un ...	Le capteur est ...	La tension à l'entrée est ...	Etat du signal à l'entrée	Symbole / Instruction	Résultat Interrogation	Symbole / Instruction	Résultat Interrogation
contact NO 	activé 	présente	1		« oui » 1		« non » 0
	inactif 	absente	0	CONT : 	« non » 0	CONT : 	« oui » 1
contact NF 	activé 	absente	0	contact dit « NO » 	« non » 0	contact dit « NF » 	« oui » 1
	inactif 	présente	1		« oui » 1		« non » 0

Tableau II.16. Différents types de détecteur interrogé.

## II.8.2 Affection / Mise à 1 et mise à 0

← )

⇒ L'affectation transmet le résultat logique à l'opérande indiquée. Ce résultat affecté reste disponible après l'affectation et peut être affecté à un autre opérande ou relié à d'autres opérations

← ( S )

⇒ Mise à 1 → Si le résultat logique = « 1 », l'opérande indiqué est mis à « 1 » (SET). Si le résultat = 0, l'opérande reste inchangé.

← ( R )

⇒ Mise à 0 → Si le résultat logique = « 1 », l'opérande indiqué est mis à « 0 » (RESET). Si le résultat = 0, l'opérande reste inchangé.

Les instructions SET et RESET peuvent être combinées en utilisant les bascules SR et RS.

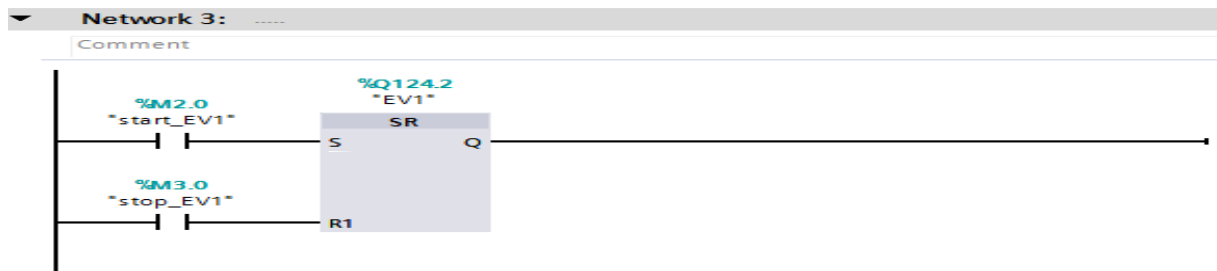


Fig. II.17. Set et Reset avec Bascule SR et RS.

## II.8.3 Fonctions logiques ET, OU, OU Exclusif

- **Fonction ET** : le résultat logique est égal à « 1 » lorsque toutes les entrées sont à l'état « 1 ». En langage LADDER, cela revient à mettre les contacts en série.
- **Fonction OU** : le résultat logique est égal à « 1 » lorsqu'au moins une des entrées est à l'état « 1 ». En langage LADDER, cela revient à mettre les contacts en parallèle.

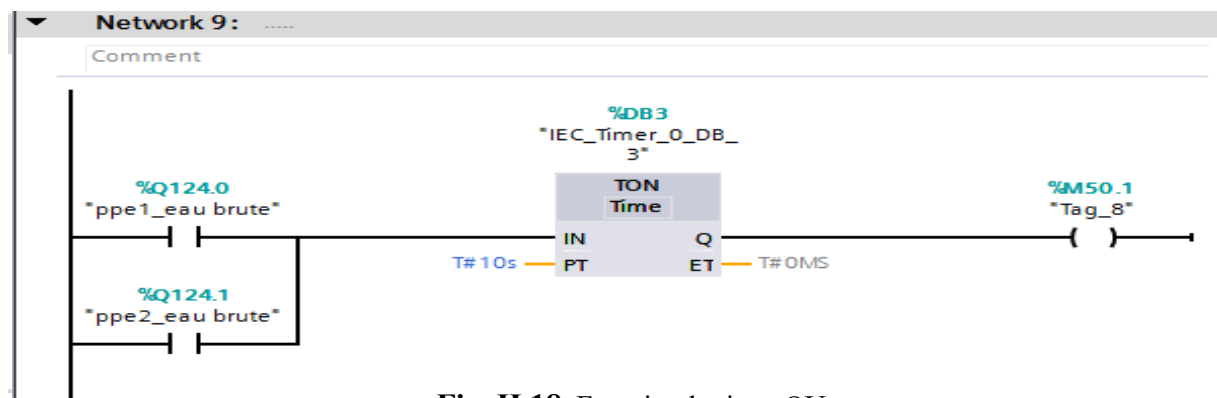
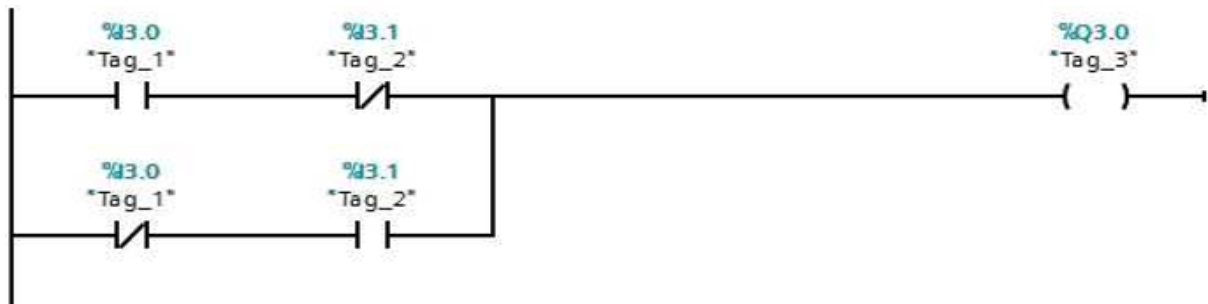


Fig. II.18. Fonction logique OU.

- **Fonction OU Exclusif** : le résultat logique est égal à « 1 » lorsqu'une seule des deux entrées est à l'état 1 (cas de deux entrées). En langage LADDER, l'instruction se programme Comme indiqué dans la figure ci-dessous.



**Fig. II.19.** Fonction logique ET.

## II.9. Conclusion

Dans ce chapitre, on a présenté, l'automate S7- CPU 314-2DP et à la plateforme de programmation TIA Portal utilisés lors de la réalisation de ce travail toute en décrivant les différentes étapes de configuration matériel, Compilation et chargement, la connexion entre le PC et le CPU et la programmation et les fonctions logiques susceptible d'être utilisé dans telle applications toute en donnant les variables d'entrées et de sorties et leurs adresses.

# *CHAPITRE III*

*Problématique Et Solution Proposé*

### **III.1 .Introduction**

Pour la résolution de nombreux problèmes de commande, le choix s'oriente de plus en plus vers les automates industriels (API). Les API assurent un gain de temps de développement considérable, de souplesse accrue dans la manipulation de haute fiabilité, de localisation et d'élimination rapide des erreurs. Ils possèdent des techniques de régulation et de contrôle complexes satisfaisant aux exigences de flexibilité, transparence et disponibilité imposées aux installations industrielles, en prenant en considération la garantie d'investissement, l'économie et le plus grand confort de client.

L'objectif dans ce chapitre est d'acquiescer les outils et méthodes afin de gérer de façon optimale un projet d'automatisme industriel, ainsi que, réaliser le cahier des charges du système à automatiser, et concevoir un système automatisé de l'unité du traitement des eaux à base de l'automate S7-314-2DP, nous avons utilisé le logiciel TIA PORTAL V15 à programmation et la supervision pour visualiser et d'analyser le comportement du système en temps réel.

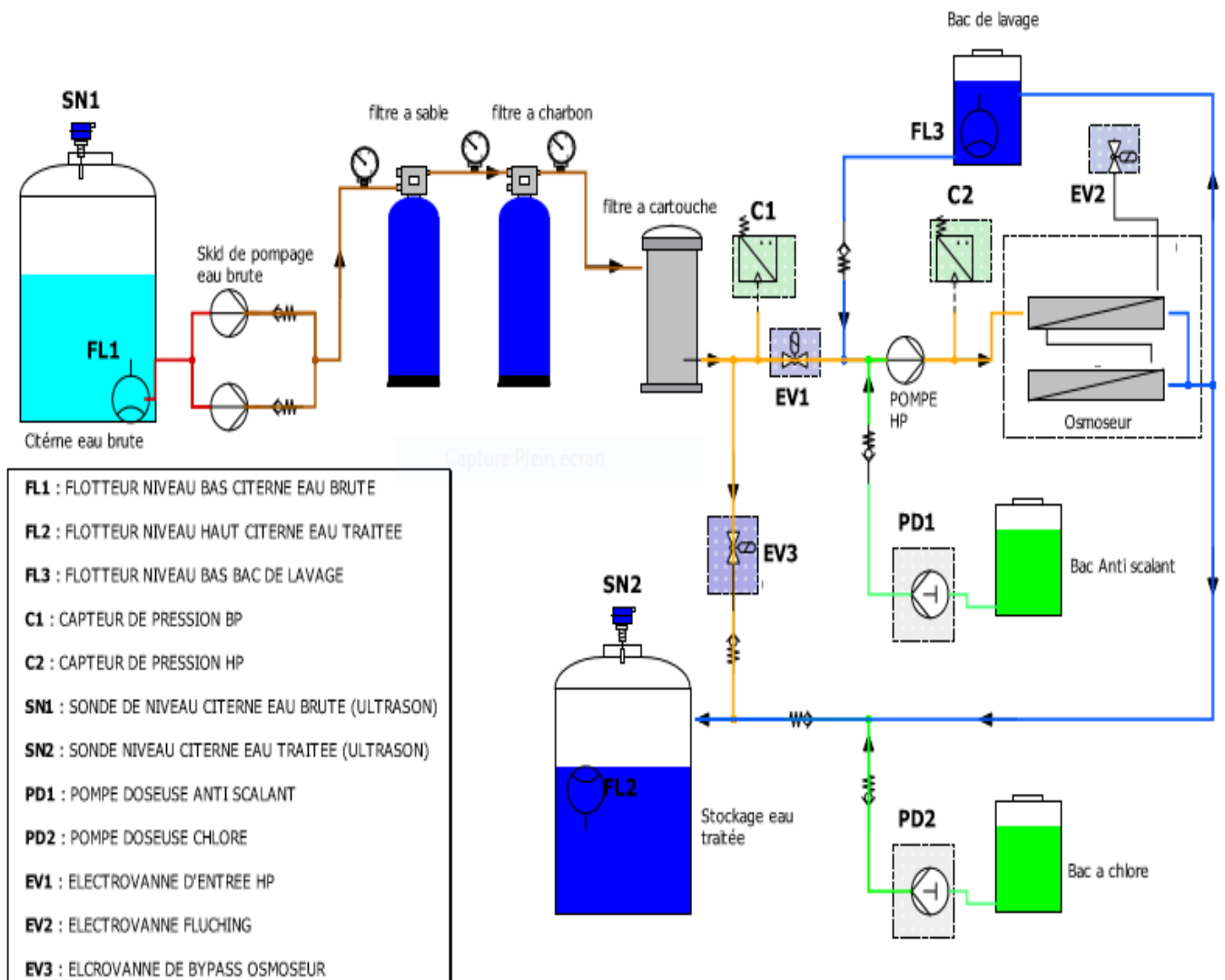
Ce chapitre décrit en premier lieu l'unité de traitement des eaux considéré toute en donnant en détails les différentes étapes de fonctionnement et développement d'un programme à base de TIA portal V15 pour répondre conformément à la description et à la demande du cahier des charges. Ensuite, les réseaux de programme sont présentés pour chaque étape de fonctionnement.

## III.2. Cahier de charges

### III.2. 1. Description de l'unité de traitement des eaux

L'installation comporte (figure III.1):

- ✚ Une citerne eau brute ;
- ✚ Une citerne eau traité ;
- ✚ Deux pompes eaux brute ;
- ✚ Trois électrovannes ;
- ✚ Bâche à eau anti scalant ;
- ✚ Bâche à eau à chlore ;
- ✚ PD1 (pompe de bac anti scalant) ;
- ✚ PD2 (pompe de bac a chlore).



FigIII.1. Synoptique de la station de traitement des eaux.

### III.3. Principe de fonctionnement de l'unité de traitement des eaux

La station doit pouvoir fonctionner en 05 modes sélectionnables à partir du pupitre de dialogue homme-machine (HMI) par l'opérateur (Mode production, lavage, CIP, By-pass et manuel) en plus d'une page paramètres ou l'en peut régler les consignes de pression ainsi que les temporisations.

#### III.3.1 Mode production

##### III.3.1.1 Démarrage de la procédure

###### Conditions :

- Sélecteur sur : ON
- Niveau bas citerne eau brute (FL1) : OK
- Niveau haut citerne eau traitée (FL2) : OK
- Consigne de pression BP (C1) : OK
- Consigne de pression HP (C2) : OK

###### Procédure de démarrage de production

-EV1 (Q1) : ON

-Temporisation = 2 Secondes

→ La pompe eau brute (Q6) : ON

→ -La pompe HP(Q7) : ON

-PD1(Q4) : ON

-PD1(Q5) : ON

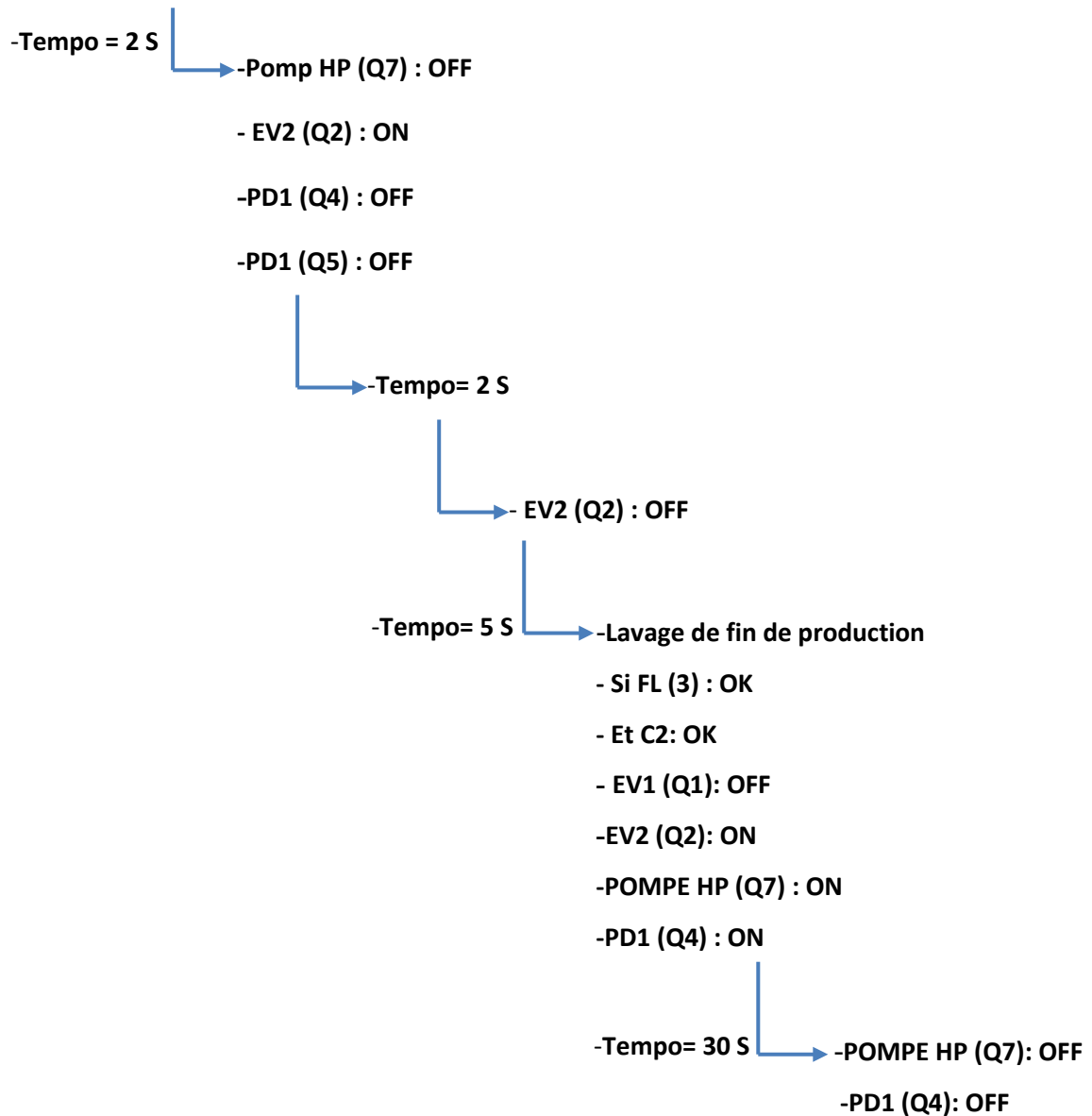
##### III.3.1.2 Fin de production et lavage

###### Conditions :

- Niveau bas citerne eau brute (FL1) : **NON OK**
- OU**
- Niveau haut citerne eau traitée (FL2) : **NON OK**
- OU**
- Sélecteur sur : **OFF**

### **Procédure de fin production**

La pompe eau brute (Q6) : OFF





### III.3.2 Mode lavage

#### ✚ Conditions :

- FL3 : OK
- C2 : OK

#### ✚ Procédure de lavage

- EV1 (Q1) : OFF
- EV2 (Q2) : ON
- Tempo = 30 S → -POMPE HP (Q7) : ON
- PD1 (Q4) : ON

### III.3.3 Mode CIP

#### ✚ Conditions :

- FL3 : OK
- C2 : OK

#### ✚ Procédure de CIP

- EV1 (Q1) : OFF
- Tempo = 30 S → -POMPE HP (Q7) : ON
- PD1 (Q4) : ON

### III.3.4 Mode BYPASS

#### ✚ Conditions

- Niveau bas citerne eau brute (FL1) : OK
- Niveau haut citerne eau traitée (FL2) : OK

#### ✚ Procédure de BYPASS

- EV3 (Q3) : ON
- POMPE d'eau brute (Q6) : ON
- PD2 (Q5) : ON

### III.3.5 Mode manuel

Démarrage et arrêt manuellement des équipements

- EV1 (Q1)
- EV2 (Q2)
- EV 3 (Q3)
- PD1 (Q4)
- PD2 (Q4)
- Pompe HP (Q7)

#### - Paramètres

Réglage des consignes C1 et C2, et réglages des tempos.

### III.4. le schéma de puissance et de commande

Le schéma de puissance, commande et de signalisation de la station de traitement des eaux considérée sont représentés respectivement par les figures III.2 et III.3.

#### III.4.1. Le schéma de puissance

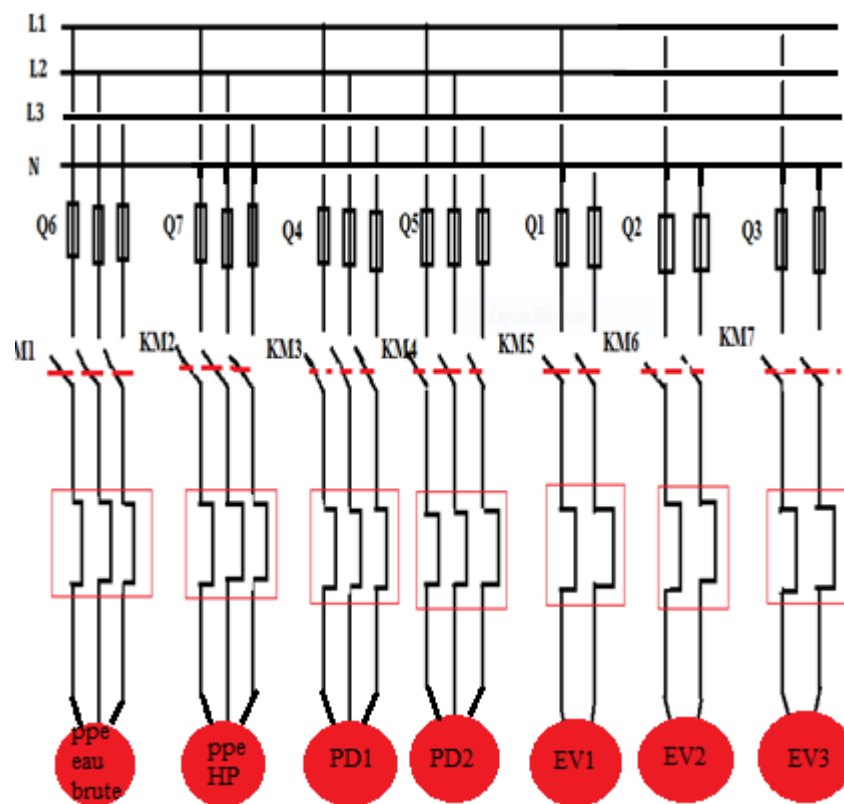
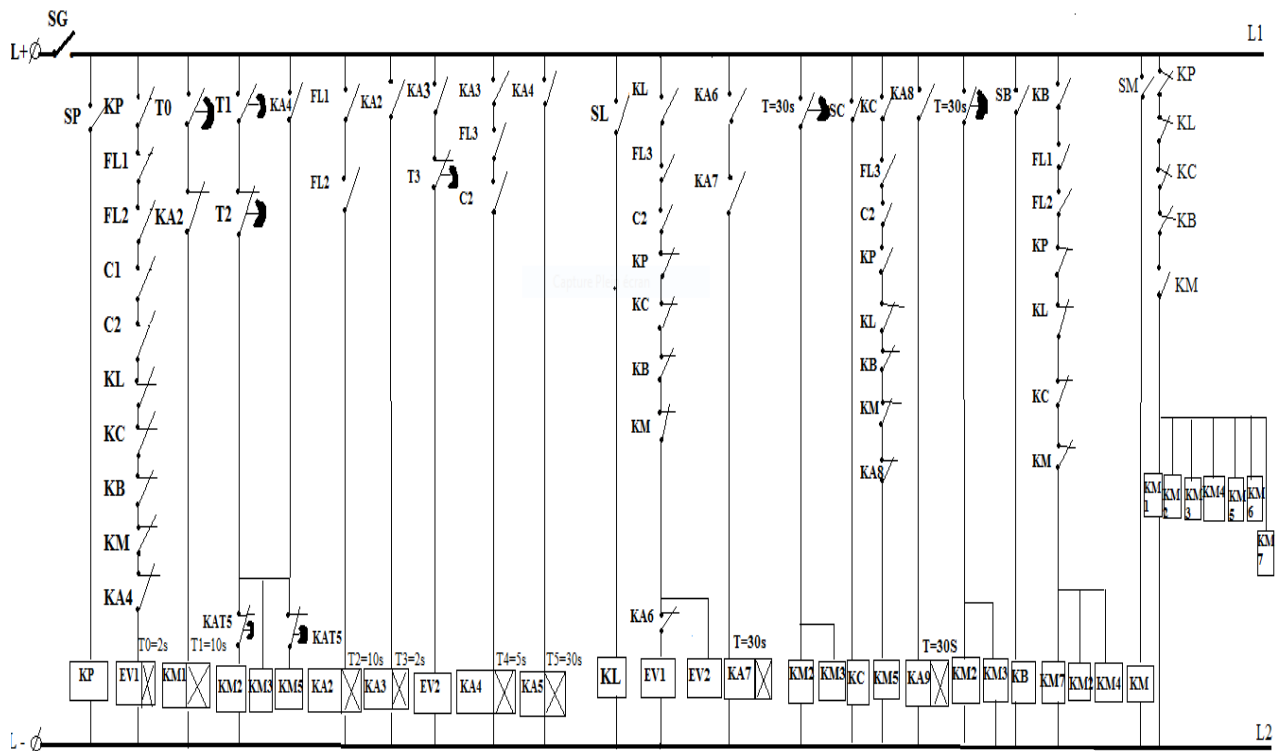


Fig.III.2. Schéma de puissance.

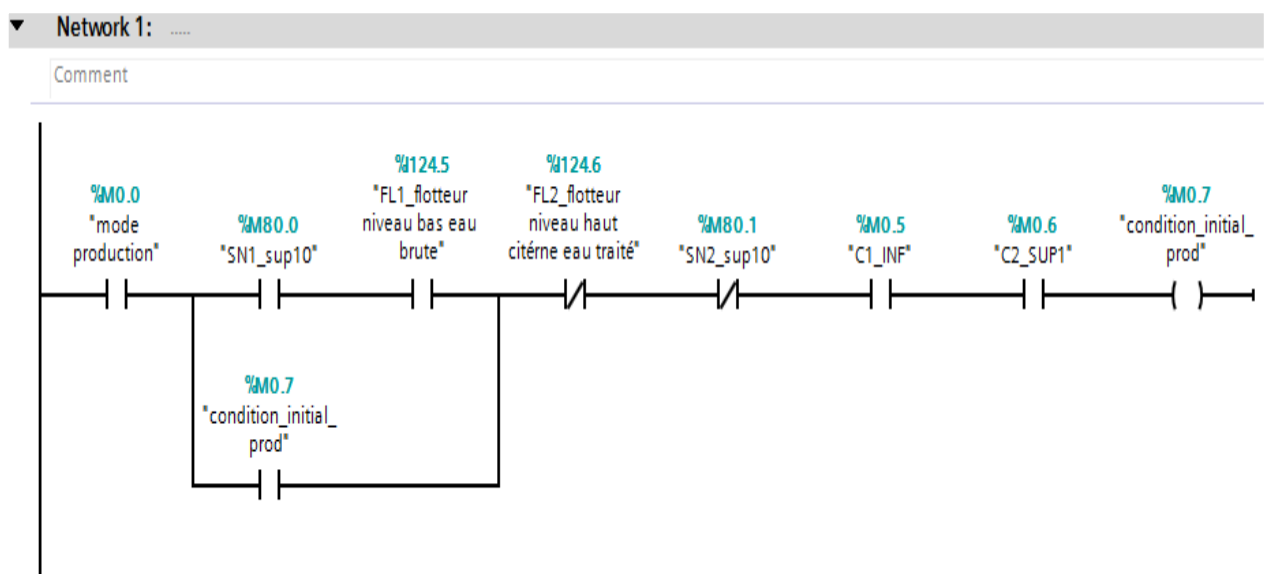
### III.4.2. Le Schéma de commande



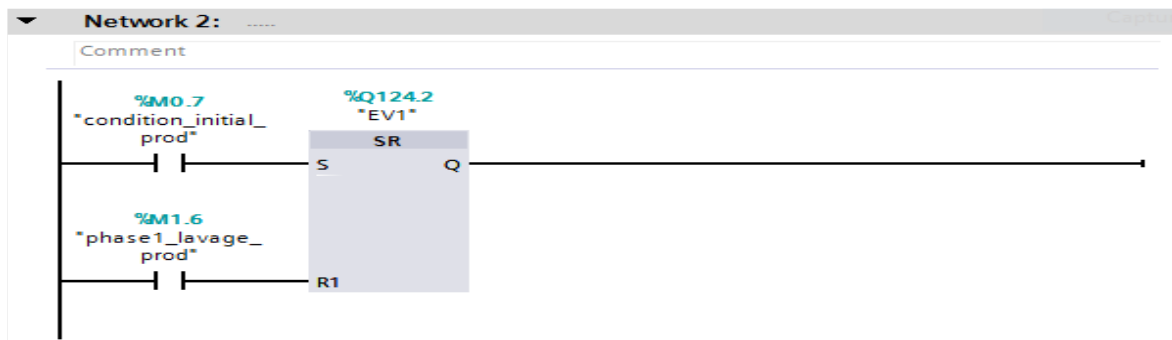
FigIII.3. Circuit de commande d'unité de traitement des eaux.

### III.5. Conversion du schéma électrique établi en langage à contact

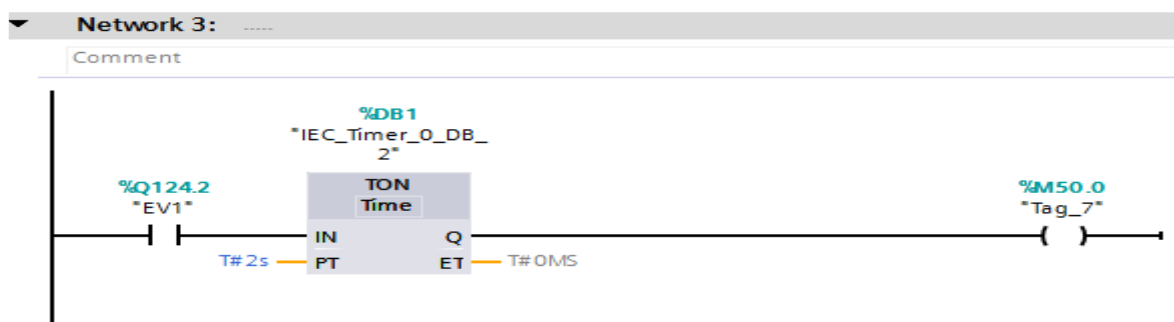
Projet → PLC\_1 → [CPU314-2 DP] → Program Blocks → **Production [FC1]**



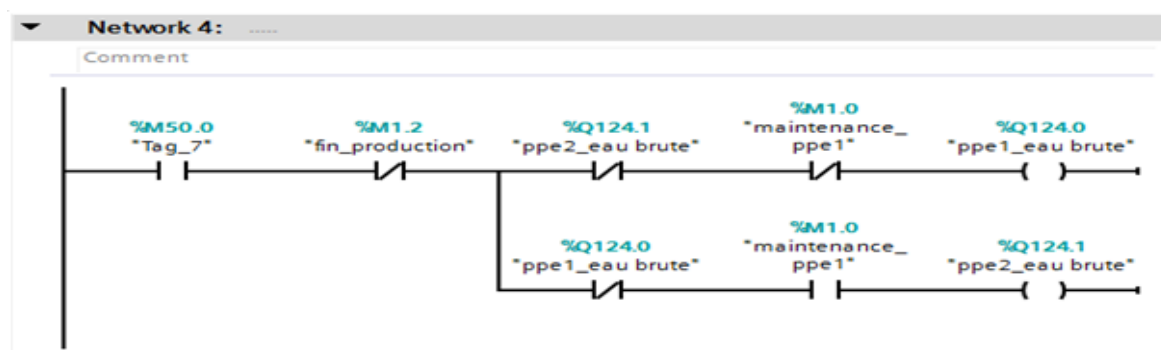
FigIII.4. les conditions initiales de démarrage production.



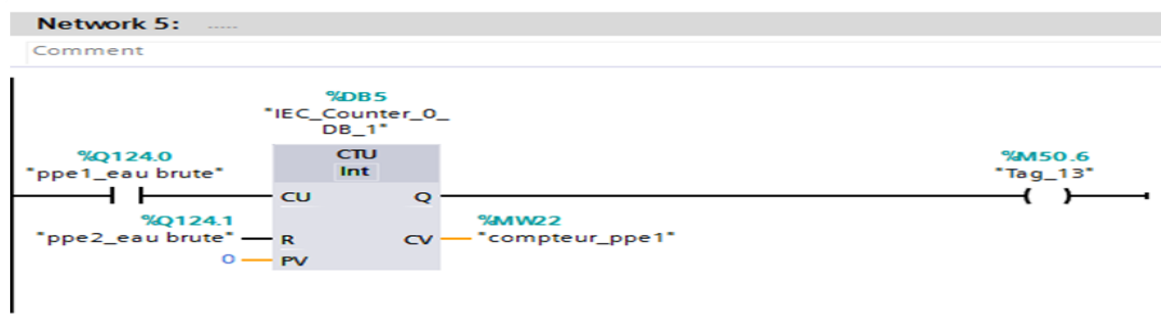
FigIII.5. l'EV1 en marche.



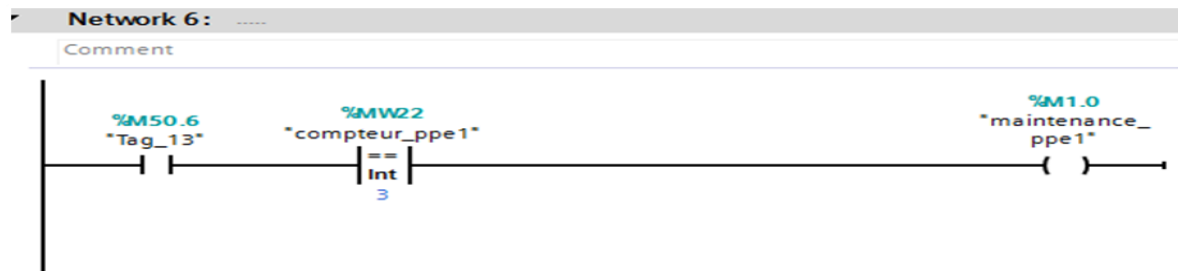
FigIII.6. temporisation pour le fonctionnement d'EV1.



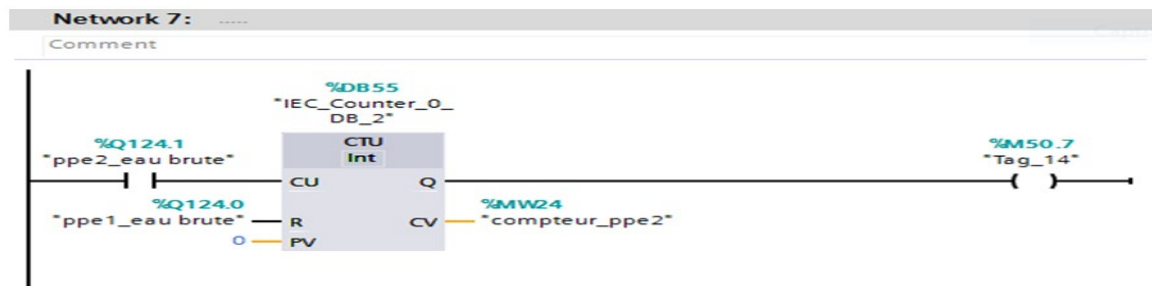
FigIII.7. les pompes d'eau brute en marche.



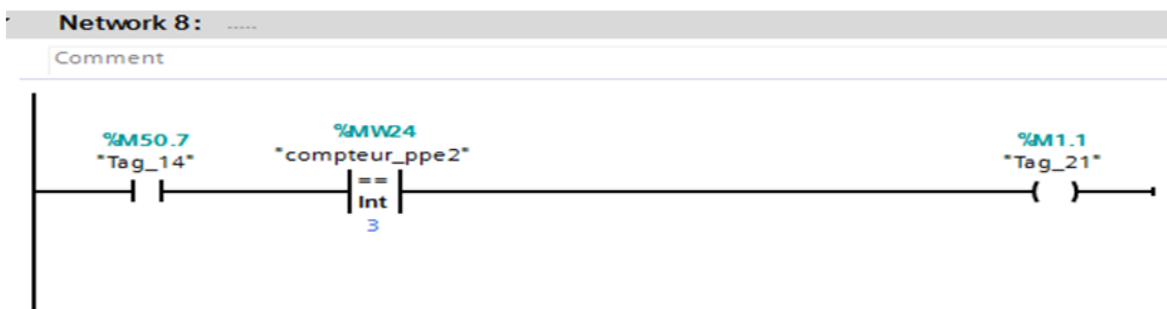
FigIII.8. Compteur de niveau d'eau dans ppe1.



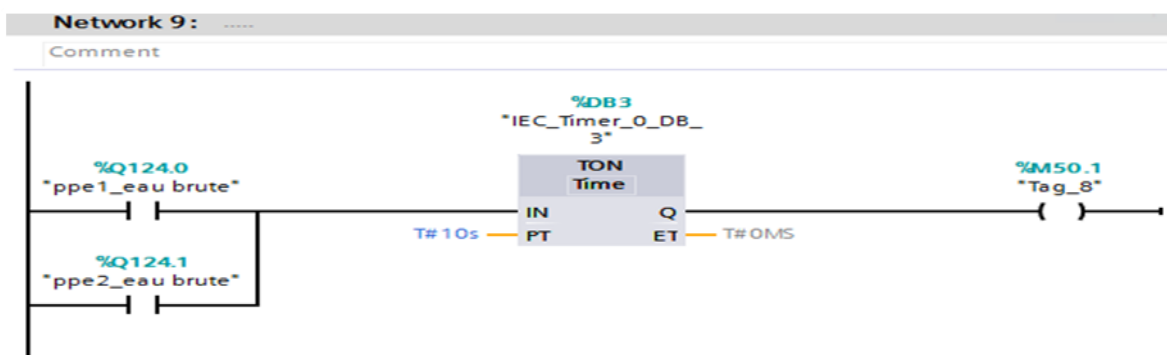
FigIII.9. Maintenance de ppe1.



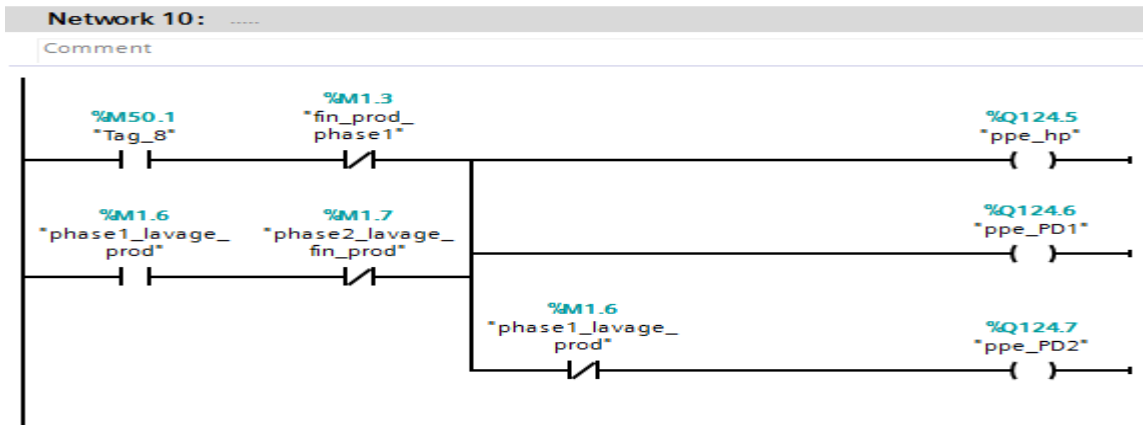
FigIII.10. compteur de niveau d'eau dans ppe2.



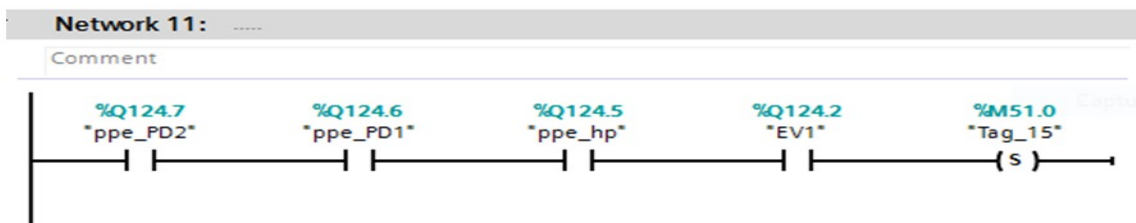
FigIII.11. Maintenance de ppe2.



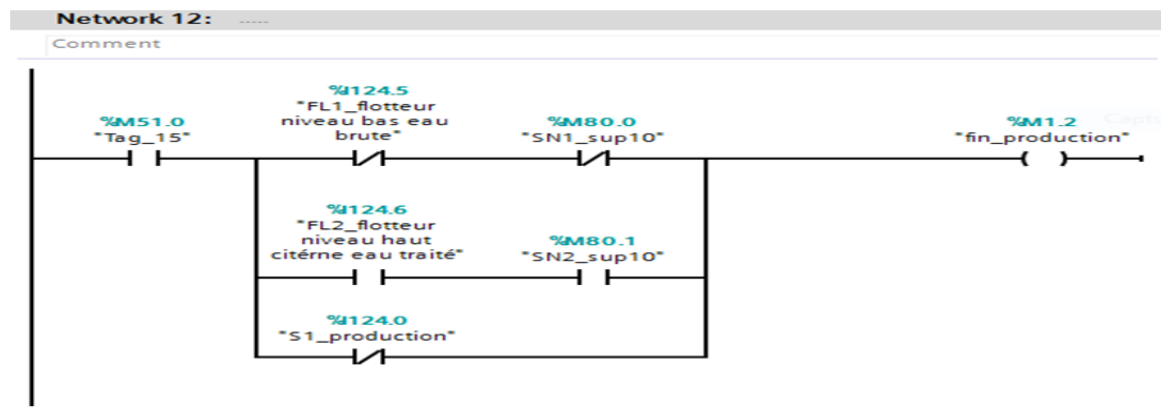
FigIII.12. temporisation pour le fonctionnement des ppe1 et ppe2.



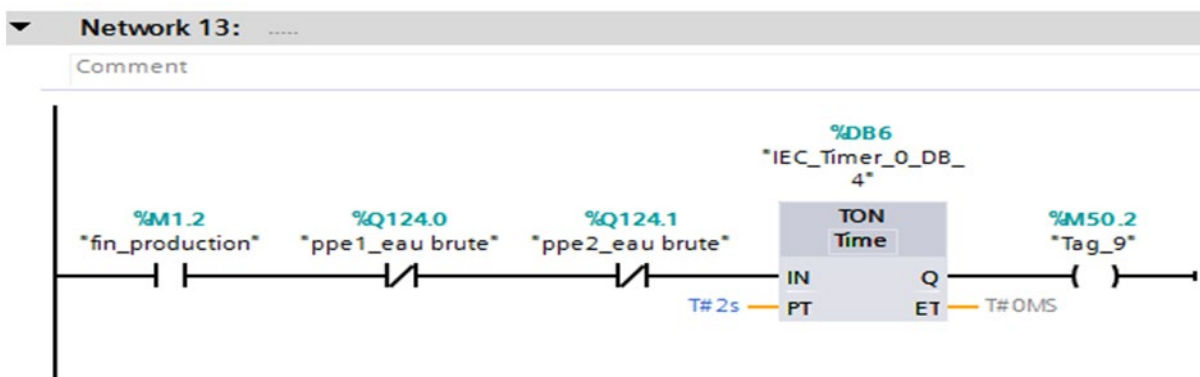
FigIII.13. les pompe HP, PD1 et PD2 en marche.



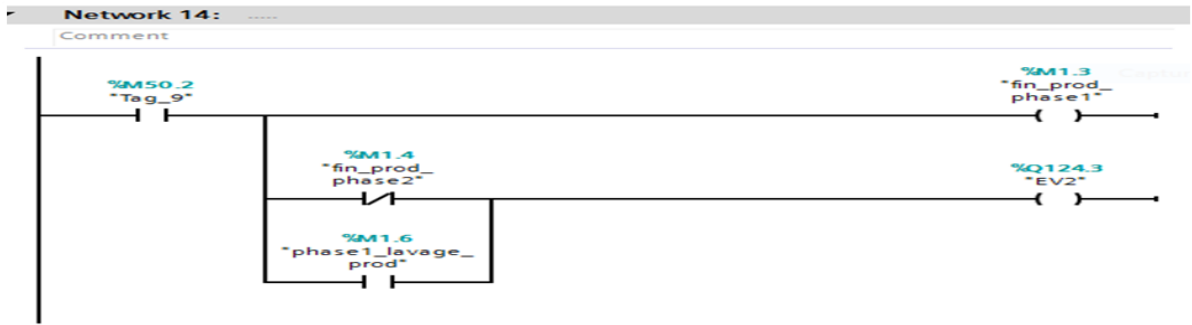
FigIII.14. Fin de production.



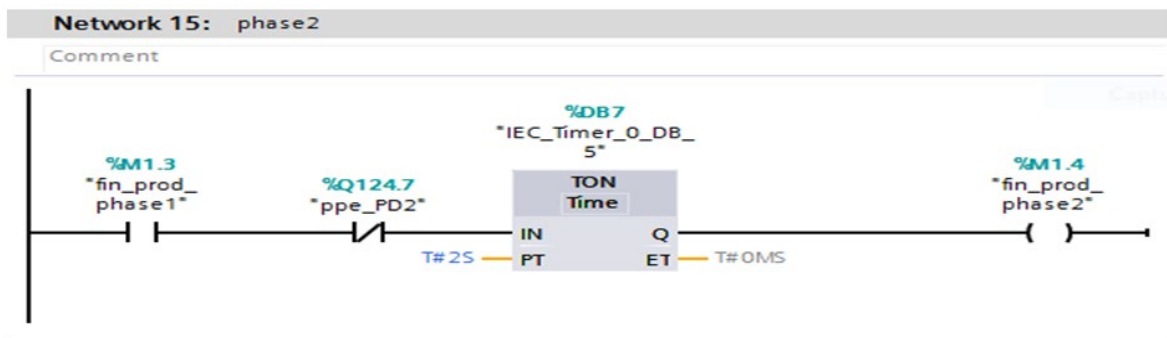
FigIII.15. Les conditions de fin de production et lavage.



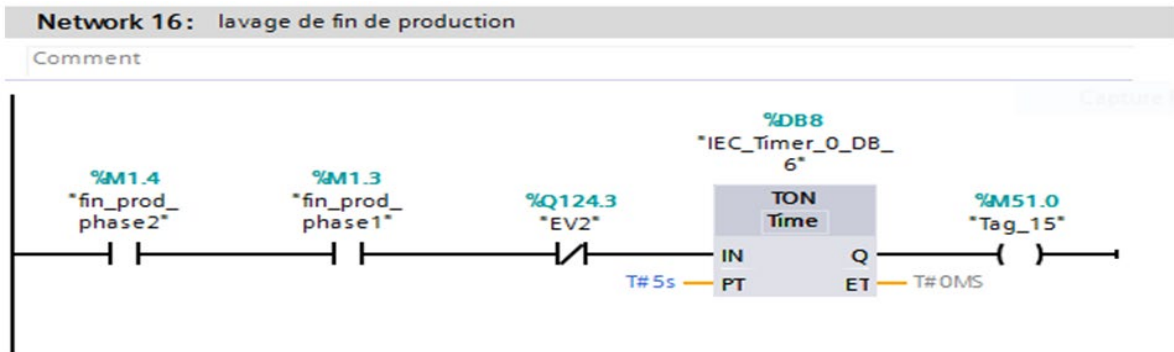
FigIII.16. temporisation pour arrêt les pompe d'eau brute.



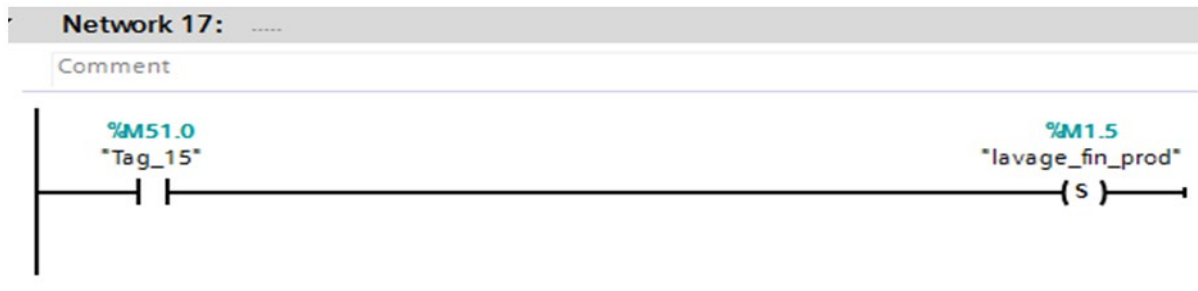
FigIII.17. EV2 et phase 1 en marche.



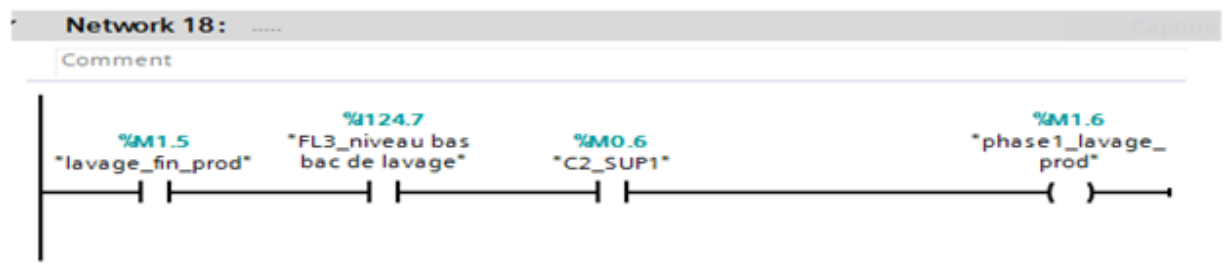
FigIII.18. Fin production en Phase 2.



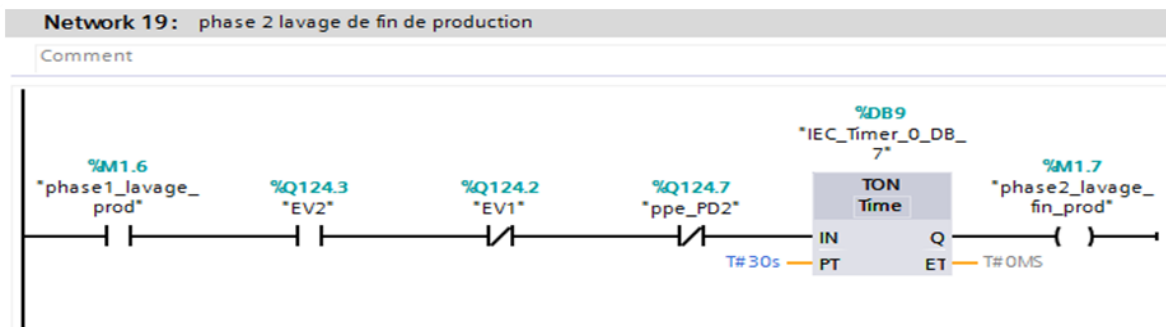
FigIII.19. lavage de fin production.



FigIII.20. set de lavage de fin production.

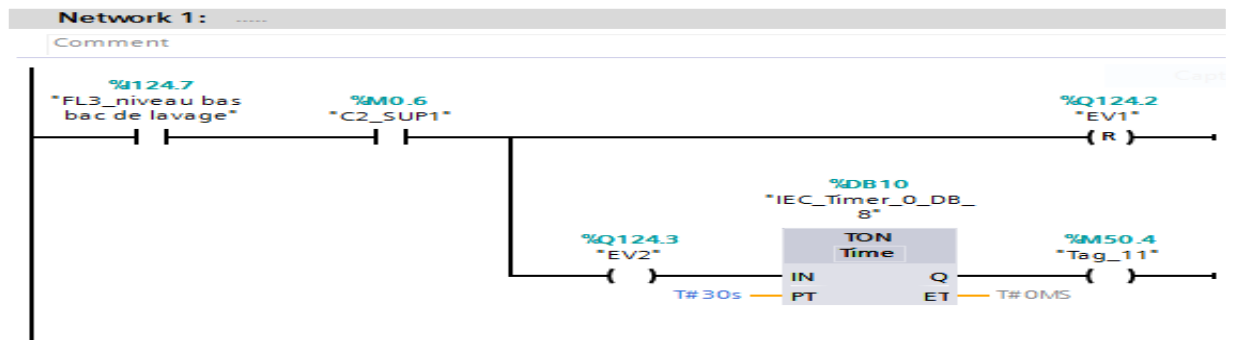


FigIII.21. Lavage de fin de production Phase 1.

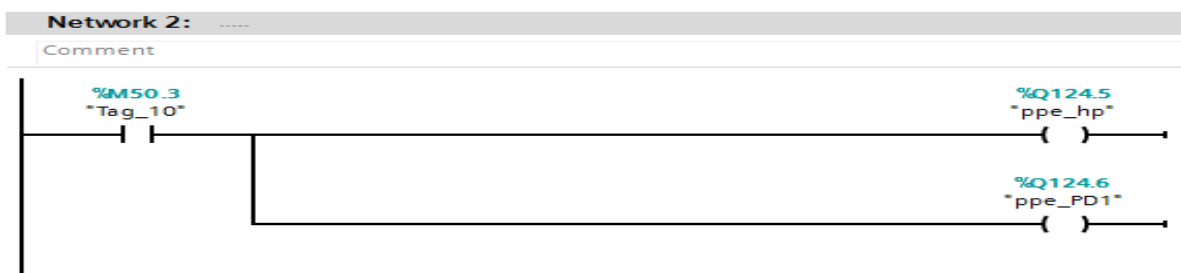


FigIII.22. Lavage de fin de production dans Phase 2 .

✚ Projet→PLC\_1→ [CPU314-2 DP] →Program Blocks→ **Lavage[FC2]**

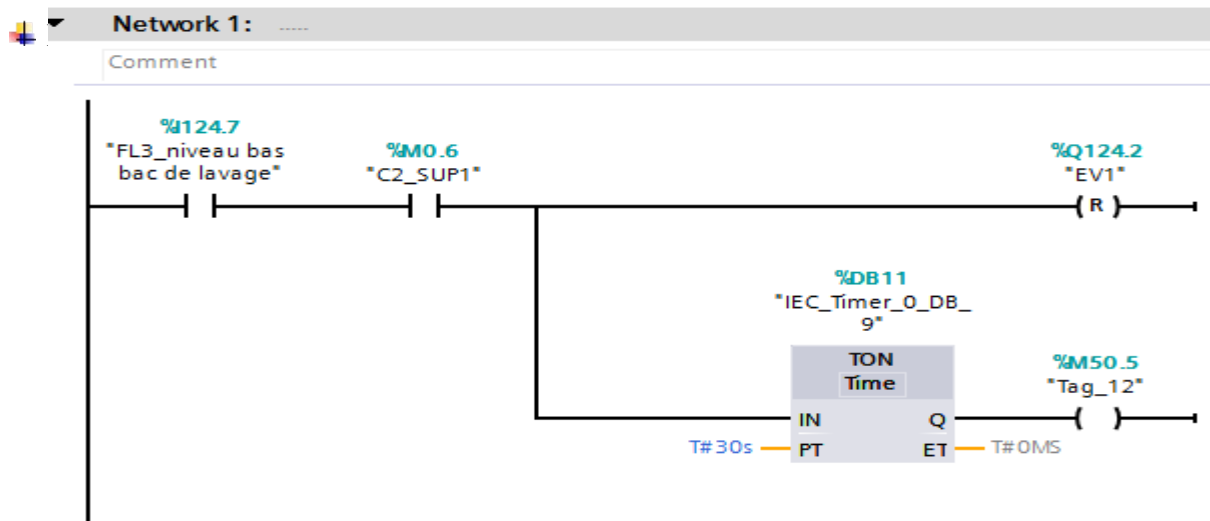


FigIII.23. Reset EV1 et set EV2.

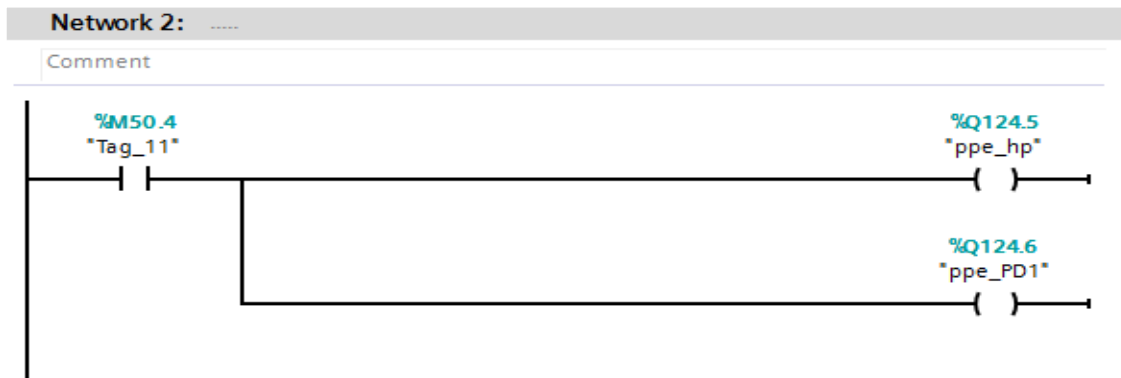


FigIII.24. Les pompe HP et PD1 en marche.



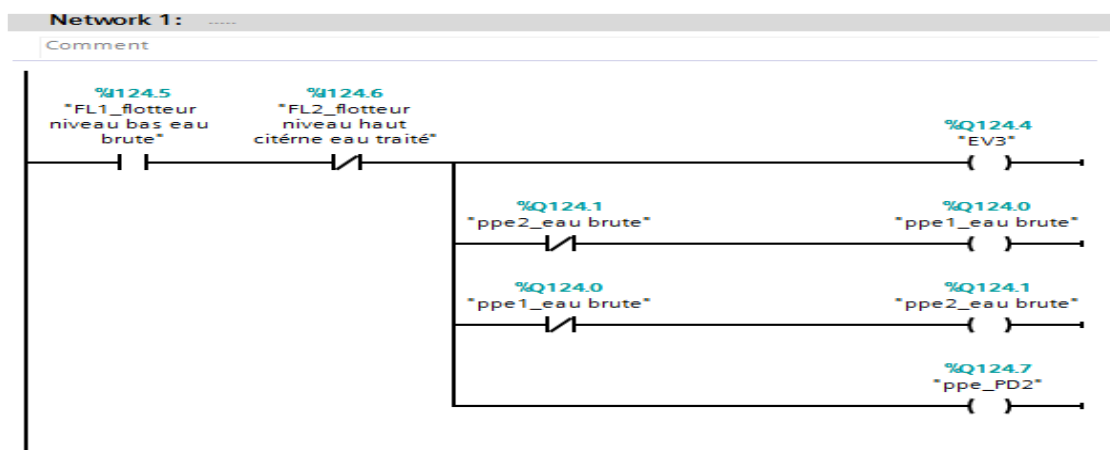


FigIII.25. Reset d'ev1.



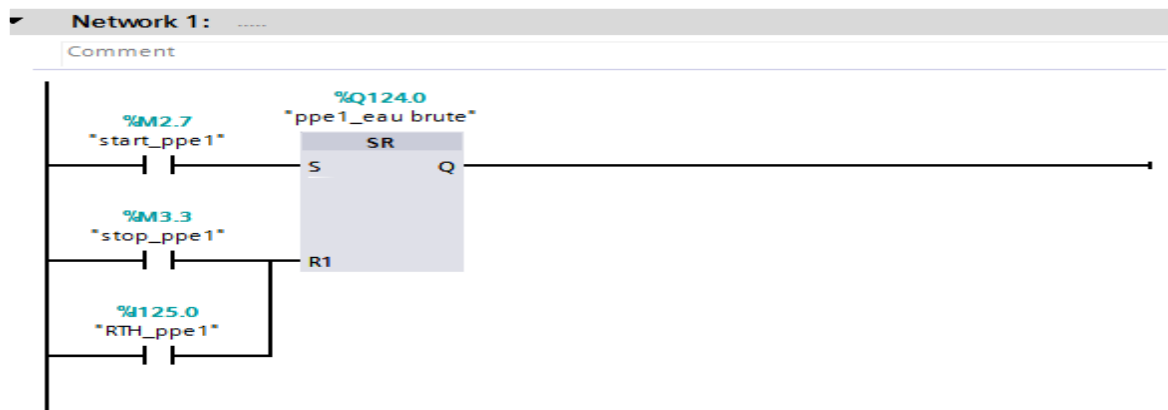
FigIII.26. Les pompe HP et PD1 en marche.

Projet → PLC\_1 → [CPU314-2 DP] → Program Blocks → **BYPASS [FC4]**

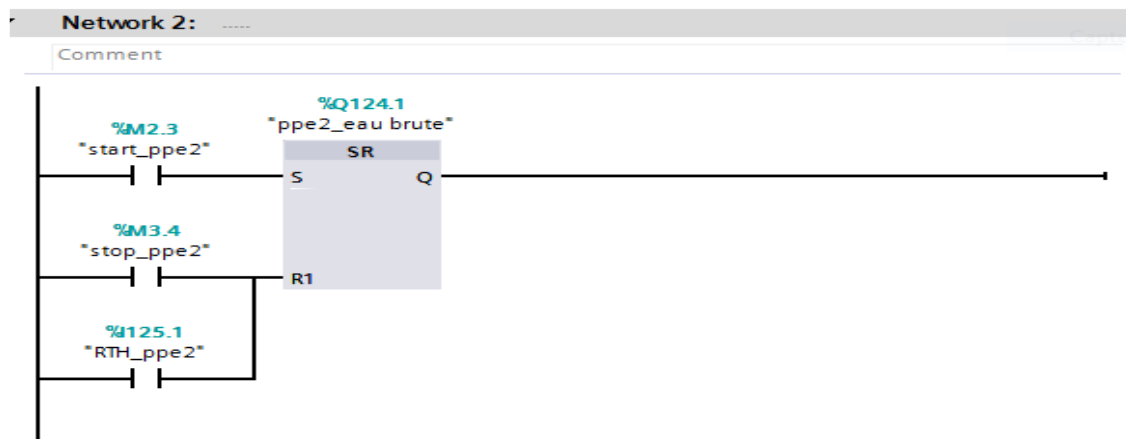


FigIII.27. EV3, pompe eau brute et PD2 en marche.

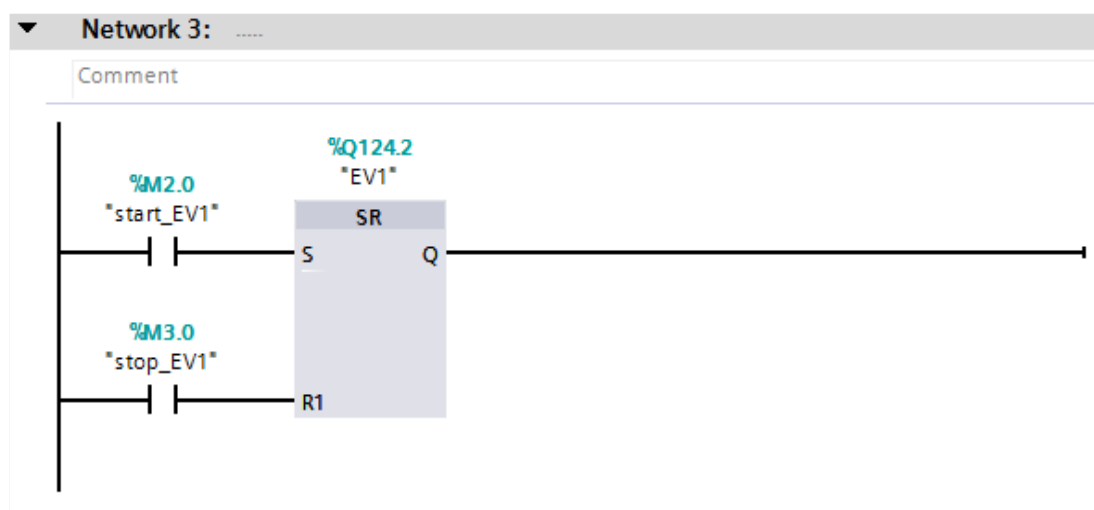
Projet→PLC\_1→ [CPU314-2 DP] →Program Blocks→ Manuel[FC5]



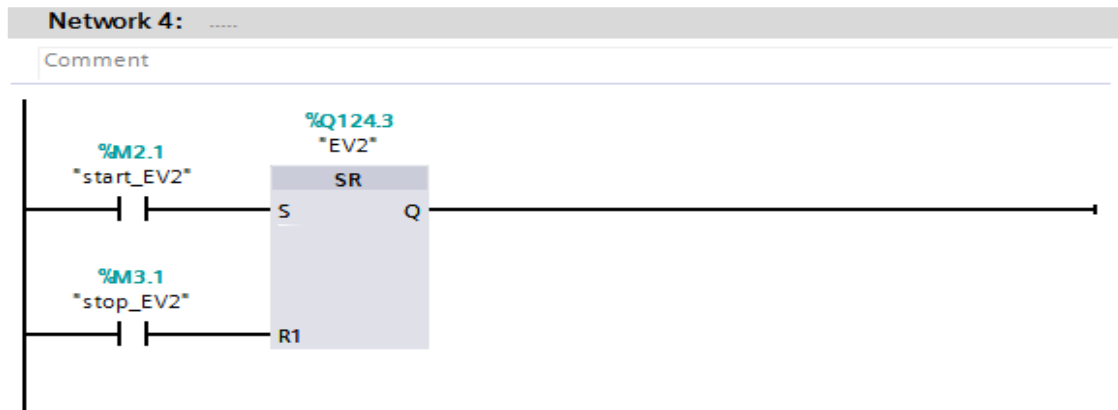
FigIII.28. Marche et arrêt de ppe1.



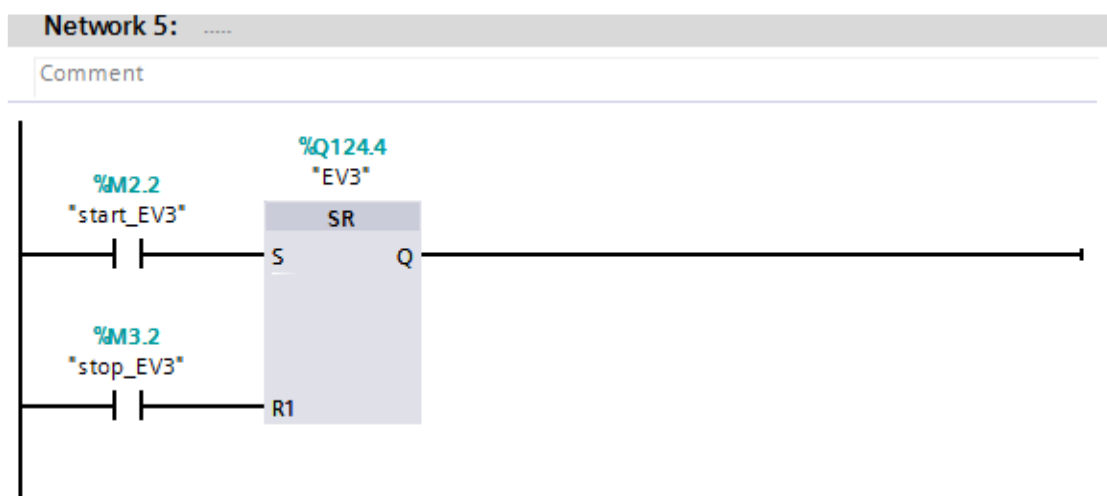
FigIII.29. Marche et arrêt de ppe2.



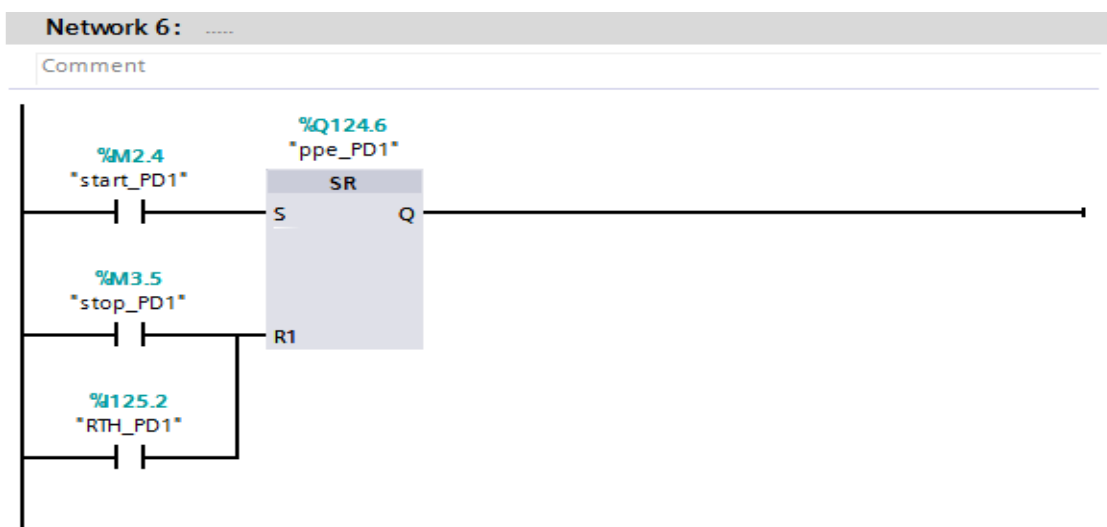
FigIII.30. Marche et arrêt d'EV1.



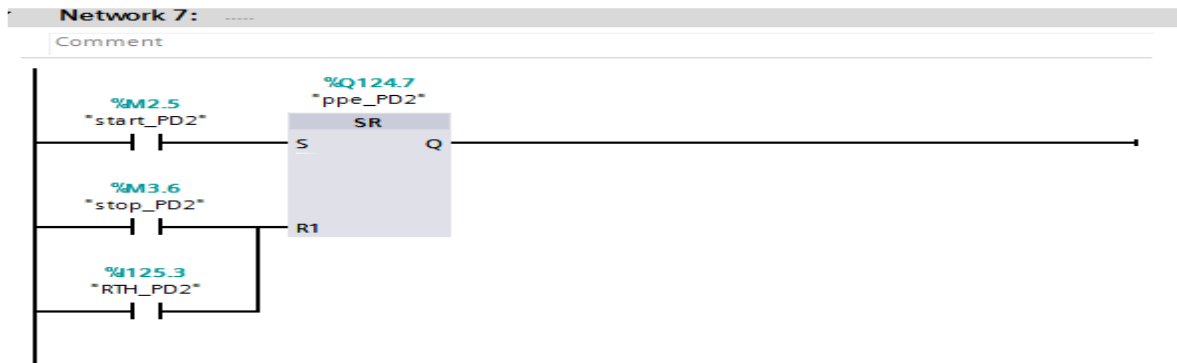
FigIII.31. Marche et arrêt d'EV2.



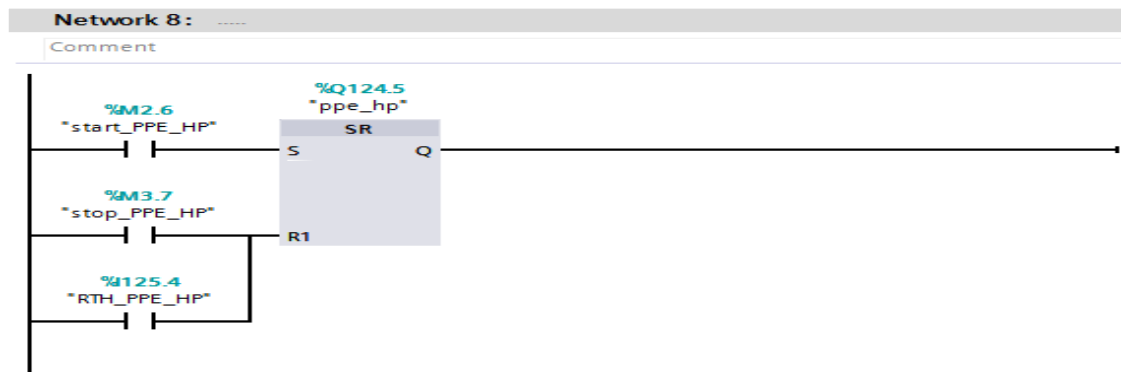
FigIII.32. Marche et arrêt d'EV3.



FigIII.33. Marche et arrêt de la pompe PD1.

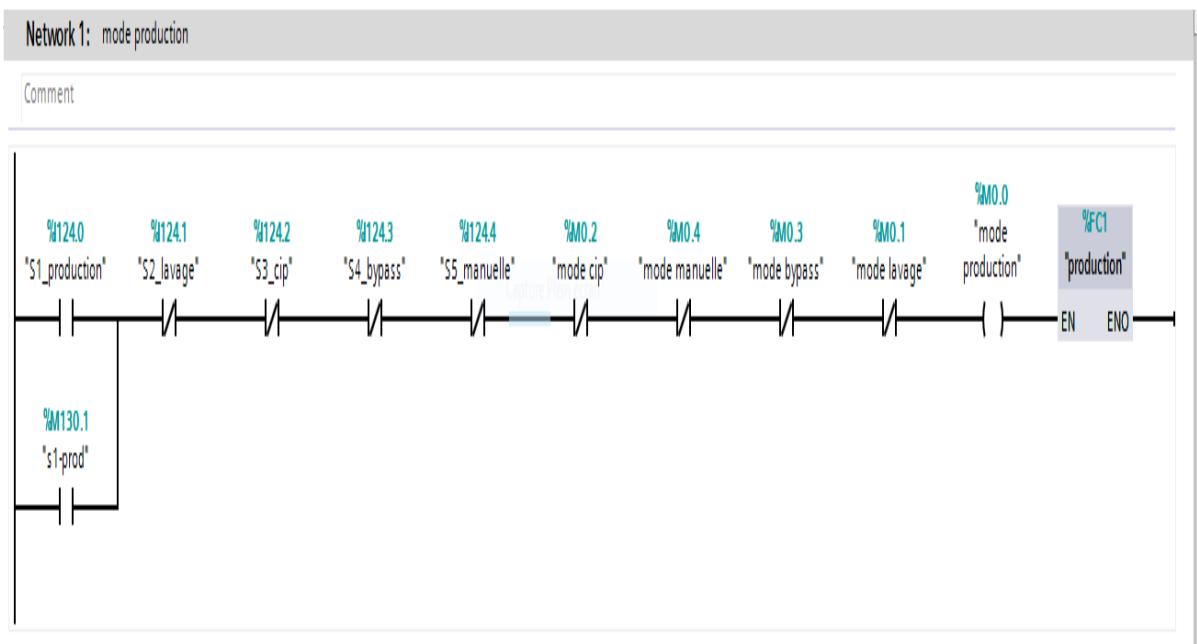


FigIII.34. Marche et arrête la pompe PD2.

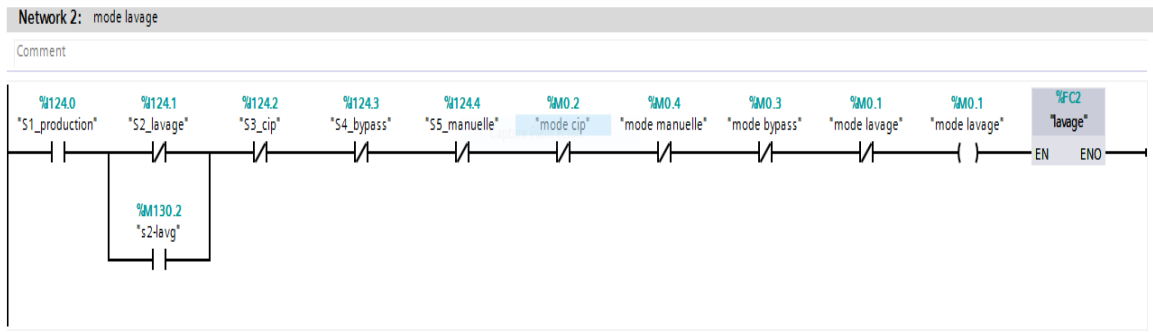


FigIII.35. Marche et arrêt de la pompe HP.

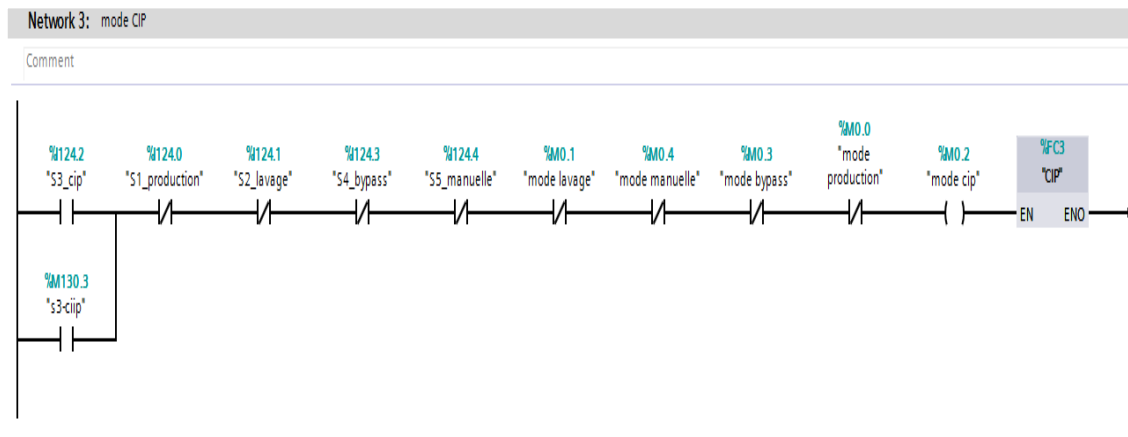
Projet → PLC\_1 → [CPU314-2 DP] → Program Blocks → Main[OB1]



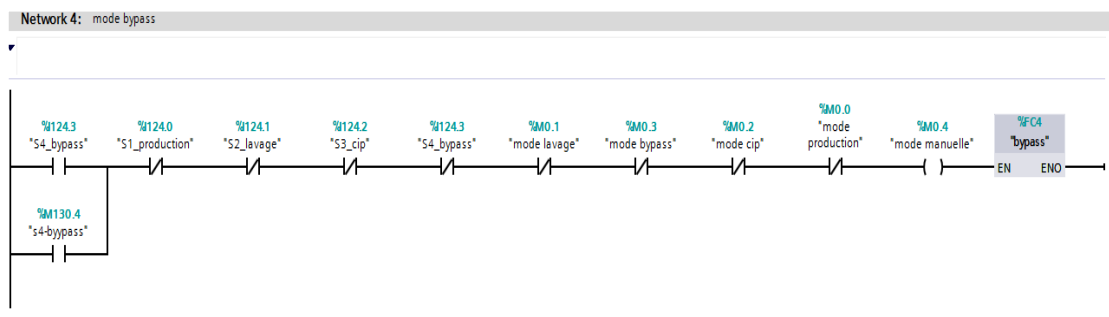
FigIII.36. Block FC1 de mode production.



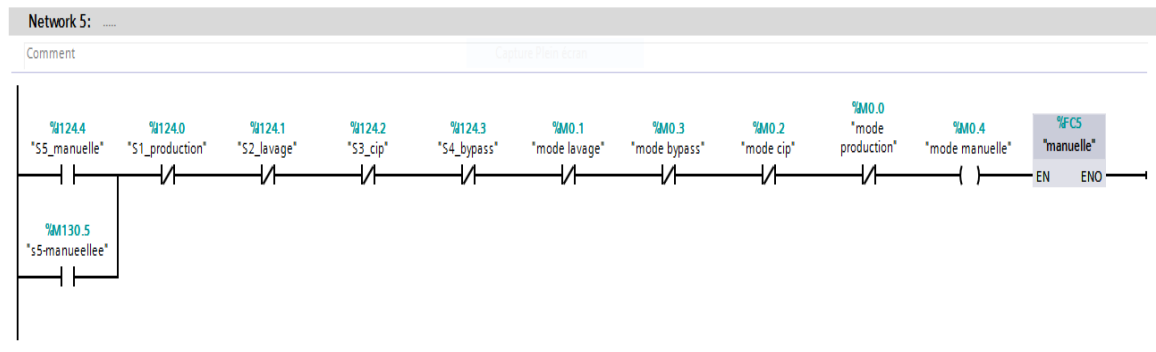
**FigIII.37.** Block FC2 de mode lavage.



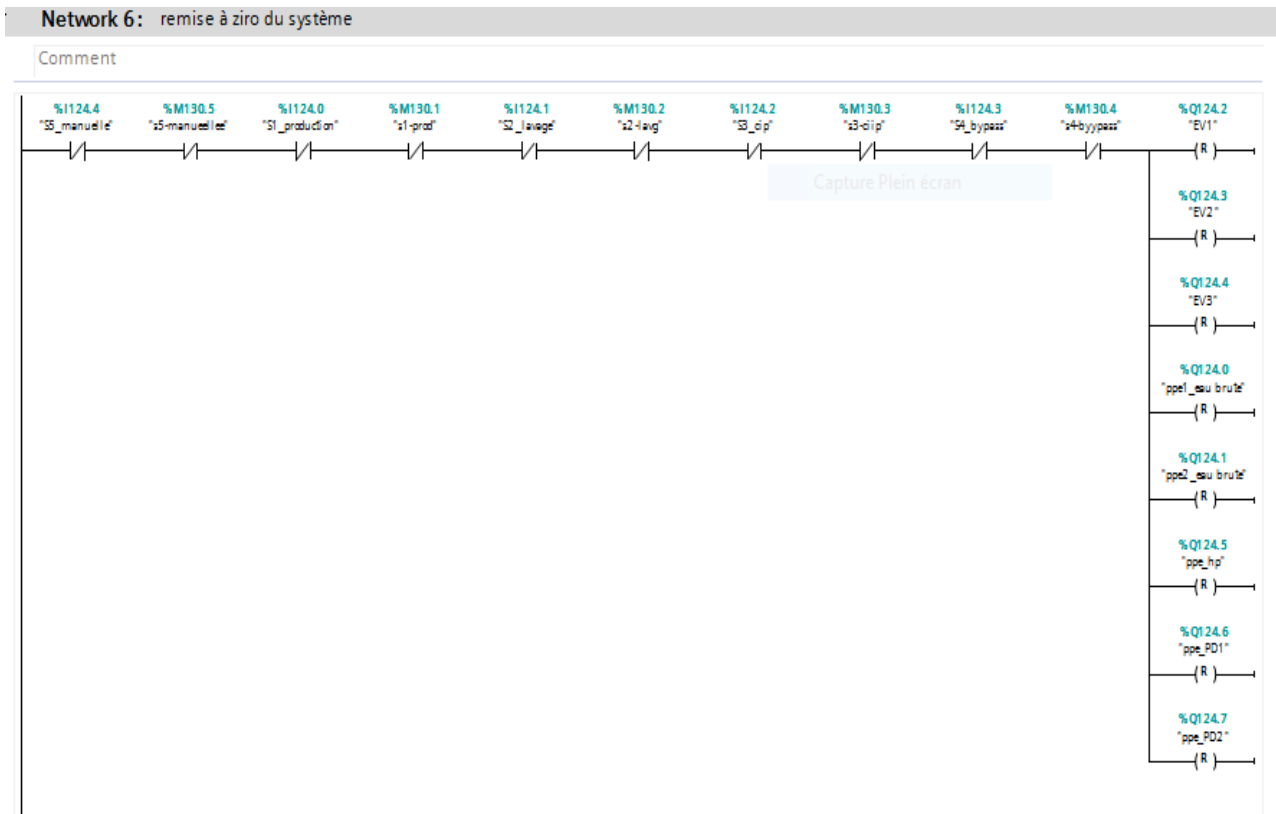
**FigIII.38.** Block FC3 de mode CIP.



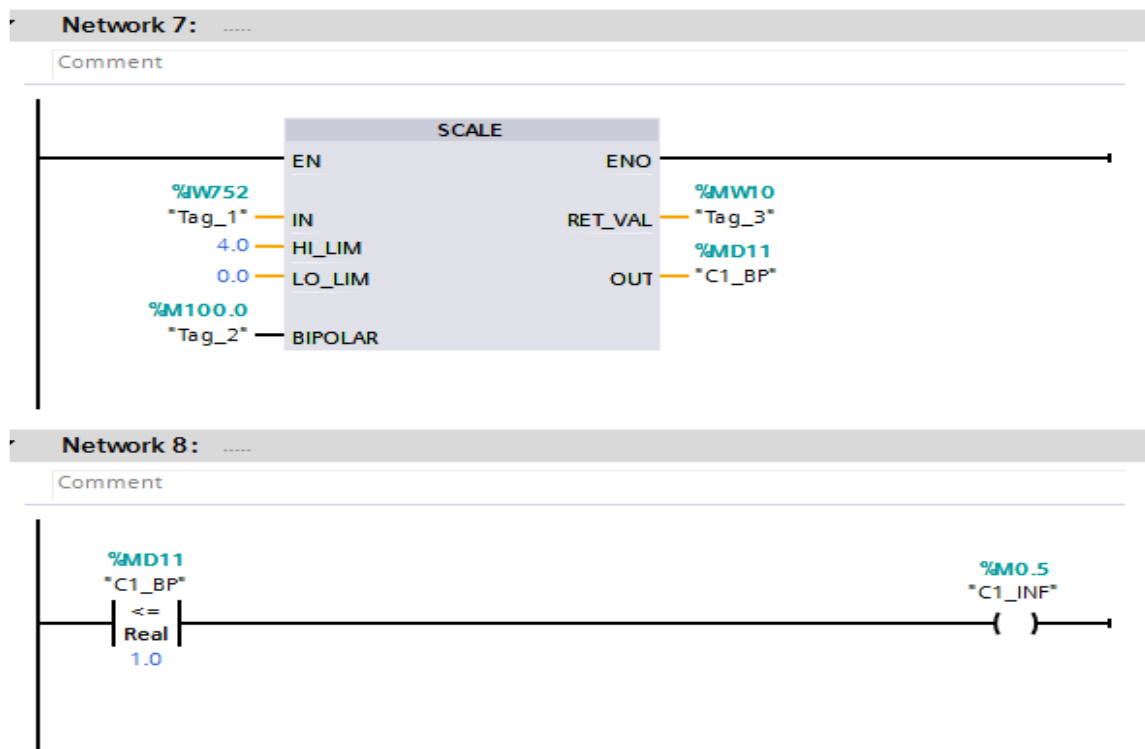
**FigIII.39.** Block FC4 de mode By-pass.



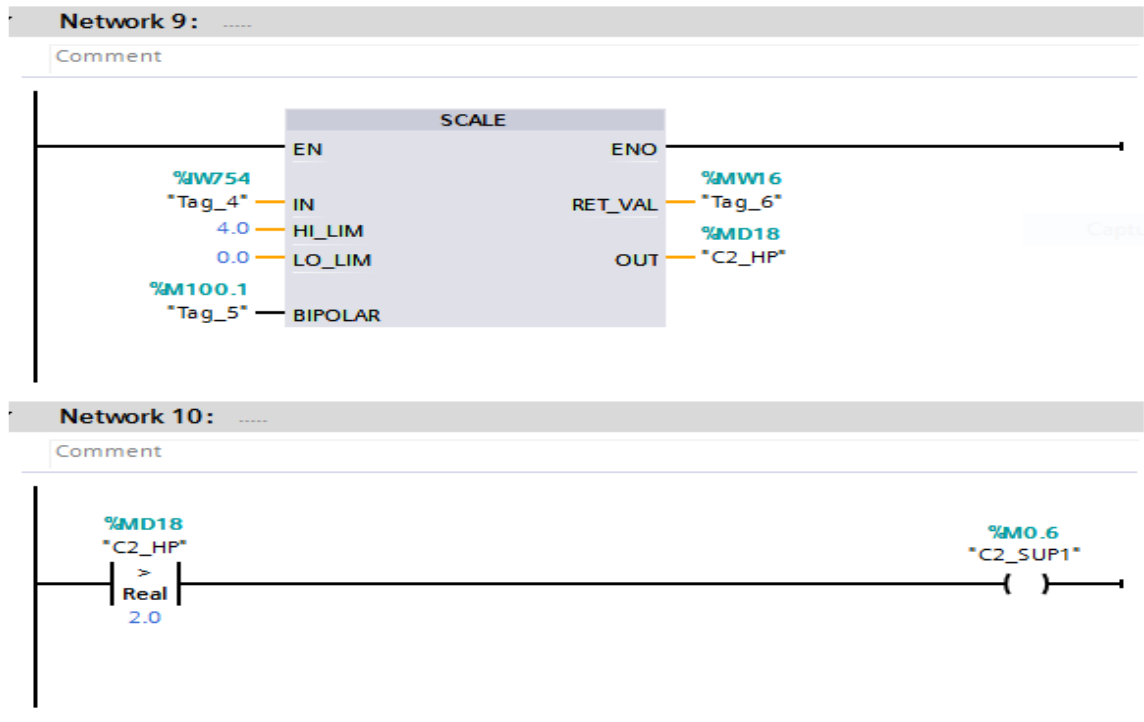
**FigIII.40.** Block FC5 de mode manuelle.



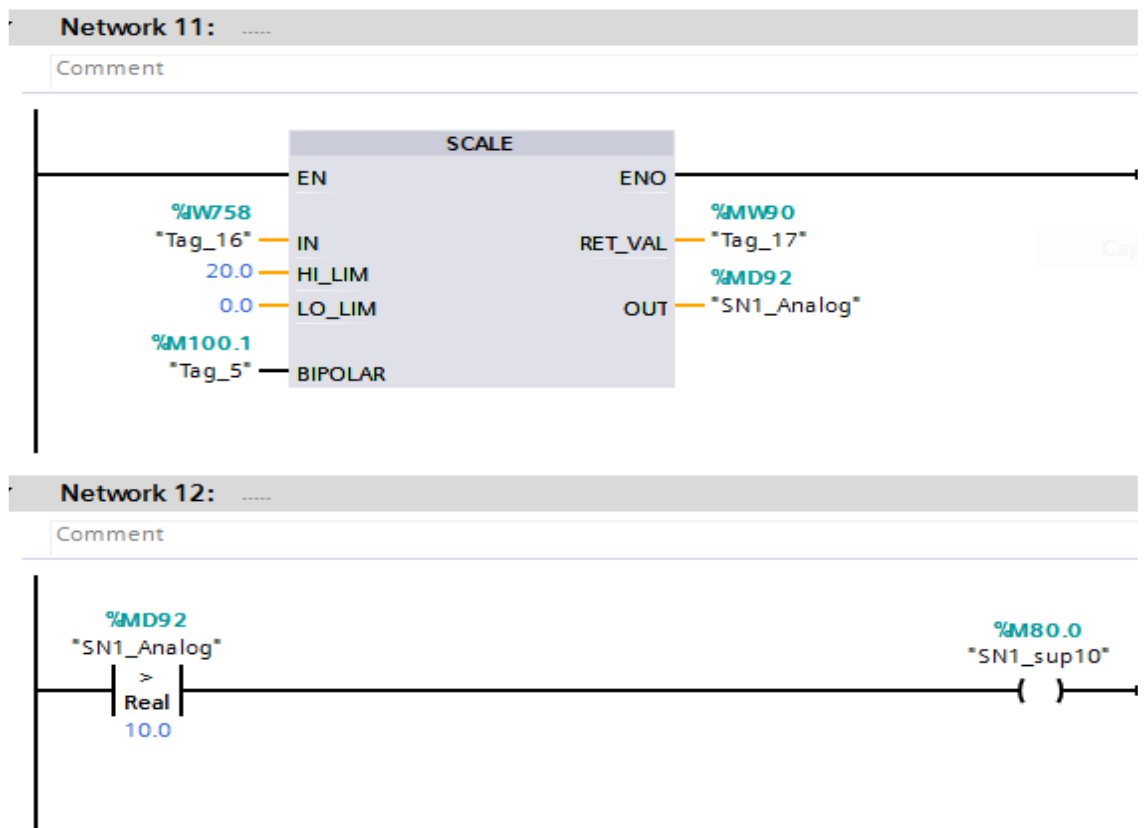
FigIII.41. Remise à zéro du système.



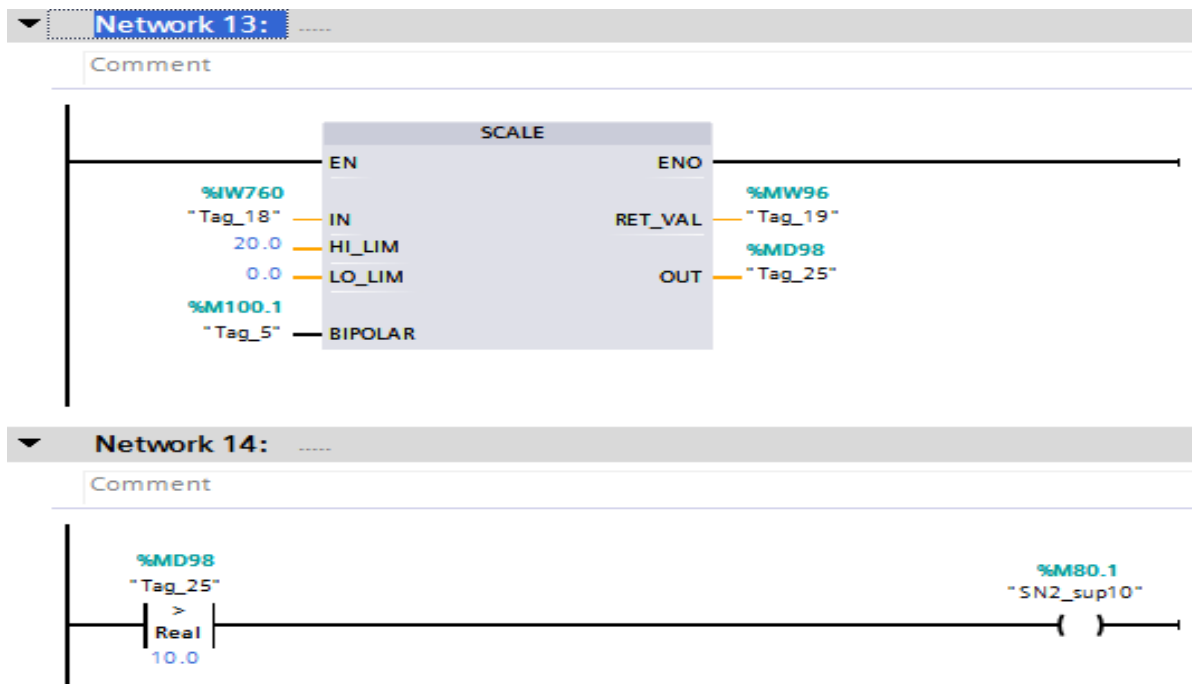
FigIII.42. Echèle analogique de capteur de pression C1.



FigIII.43. Echèle analogique de capteur de pression C2.



FigIII.44. Echèle analogique de capteur analogique SN1.



FigIII.45. Echèle analogique de capteur analogique SN2.

### III.6. Conclusion

Dans ce chapitre et après l'étude et une bonne analyse du cahier de charge conformément à la description donnée et après l'identification des entrées et sorties du système considéré. On a procédé à résoudre le problème posé tout en donnant la solution sous forme d'un schéma électrique (logique câblée) en premier temps et après on a remplacé le dispositif classiques de commande et de signalisation de la station considérée par l'utilisation et l'implémentation du programme dans un automate programmable Industriel en utilisant le langage à contact comme outil de programmation et aussi à l'aide des logiciels propre à l'automate (TIA portale V15).



# *CHAPITRE IV*

*Programmation Et Supervision*

## IV.1. Introduction

La supervision est une forme avancée de dialogue homme-machine qui permet de visualiser en temps réel l'évolution de l'état d'un système automatisé afin que les opérateurs puissent se surveiller et maintenir le système à un point de fonctionnement optimal avec un maximum de transparence. Cette transparence s'obtient au moyen de l'Interface Homme-Machine (IHM). Un système IHM constitue l'interface entre l'homme (opérateur) et le processus (la station de traitement des eaux).

Ce chapitre est réservé à la supervision et à la simulation de l'unité de traitement des eaux considérée en utilisant le langage PLCSIM qui permet de tester les programmes développés, surtout que TIA PORTAL offre différentes possibilités de test, telles que la visualisation et la supervision, avec le support du logiciel WINCC. La réalisation d'une IHM nous permettra un meilleur contrôle du processus, le diagnostic rapide d'éventuelle panne.

## IV.2. La supervision

La supervision industrielle consiste à surveiller l'état de fonctionnement d'un procédé pour l'amener à son point de fonctionnement optimal. Le but c'est de disposer en temps réel d'une visualisation de l'état d'évolution des paramétrés du processus, grâce à des vues préalablement créés et configurés, et à l'aide d'un logiciel adéquat. Ce logiciel est une entité capable d'offrir à l'opérateur des informations utiles, ce qui lui permet de prendre des bonnes décisions appropriées à ses objectifs telle que la conduite du procédé, cadence de la production, qualité des produits et sécurité des biens et des personnes. Le logiciel a essentiellement pour mission, de collecter les données (acquisition et stockage) et les mettre en forme (traitement) afin de les présenter à l'opérateur (supervision) [11].

En informatique industrielle, la supervision des procédés est un pupitre de commande évolué. Elle permet de surveiller et de contrôler l'exécution de tâches du procédé. Un logiciel de supervision fonctionne généralement sur un ordinateur en communication, via un réseau local industriel, avec un ou plusieurs équipements électroniques, Automate Programmable Industriel ou ordinateurs de commande directe (commande numérique). Un logiciel de supervision est composé d'un ensemble de pages (d'écrans), dont l'interface opérateur est présentée sous la forme d'un synoptique.

L'application peut être monoposte ou multiposte. Dans le cas d'une supervision multiposte l'architecture peut être de type redondante (redondance), de type client-serveur (un poste gère les communications pour les autres) et les deux en même temps. Ce système

assure un rôle de gestionnaire d'alarmes, d'événements déclenchés par des dépassements de seuils, pour attirer l'attention de l'opérateur et d'enregistrement d'historique de défauts, de temps de fonctionnement, d'alarmes, de paramètres prédéterminés [12].

### IV.3. WinCC sous TIA Portal

Le WinCC (TIA portal) est un logiciel d'ingénierie pour la configuration de pupitres Simatic, de PC industriel Simatic et de PC standard par le logiciel de visualisation. Le WinCC sous le TIA portal fait partie d'un nouveau concept d'ingénierie intégré qui offre un environnement d'ingénierie homogène pour la programmation et la configuration de solution de commande, de visualisation d'entraînement, c'est le logiciel pour toutes les applications IHM allant de solutions de commande simples avec basic panels aux applications SCADA pour système multipostes basé sur PC [13].

### IV.4. Variables HMI

On distingue deux types de variables, les variables externes et les variables internes :

- Les variables externes permettent de communiquer et d'échanger des données entre les composants d'un processus automatisé, entre un pupitre opérateur et un automate.
- Les variables internes ne possède aucun lien avec l'automate, elles sont enregistrées dans la mémoire du pupitre.

### IV.5 Création de la supervision

Pour bien contrôler le processus, l'opérateur a besoin d'avoir le maximum de transparence, ce qu'il lui permet de bien superviser et contrôler l'installation, cela est possible avec l'interface homme machine (HMI). Le contrôle du processus de traitement des eaux qui est assuré par le système d'automatisation.

Le pupitre de supervision une fois sous réseau permet :

- De visualiser l'état des actionneurs (pompes, vannes) et des capteurs (pression, niveau, température, pression).
- D'afficher les alarmes.
- D'agir sur les pompes et les vannes.
- La détection des défauts
- La surveillance du processus à distance
- Le diagnostic et le traitement des alarmes

## IV.5.1 Insertion de l'automate

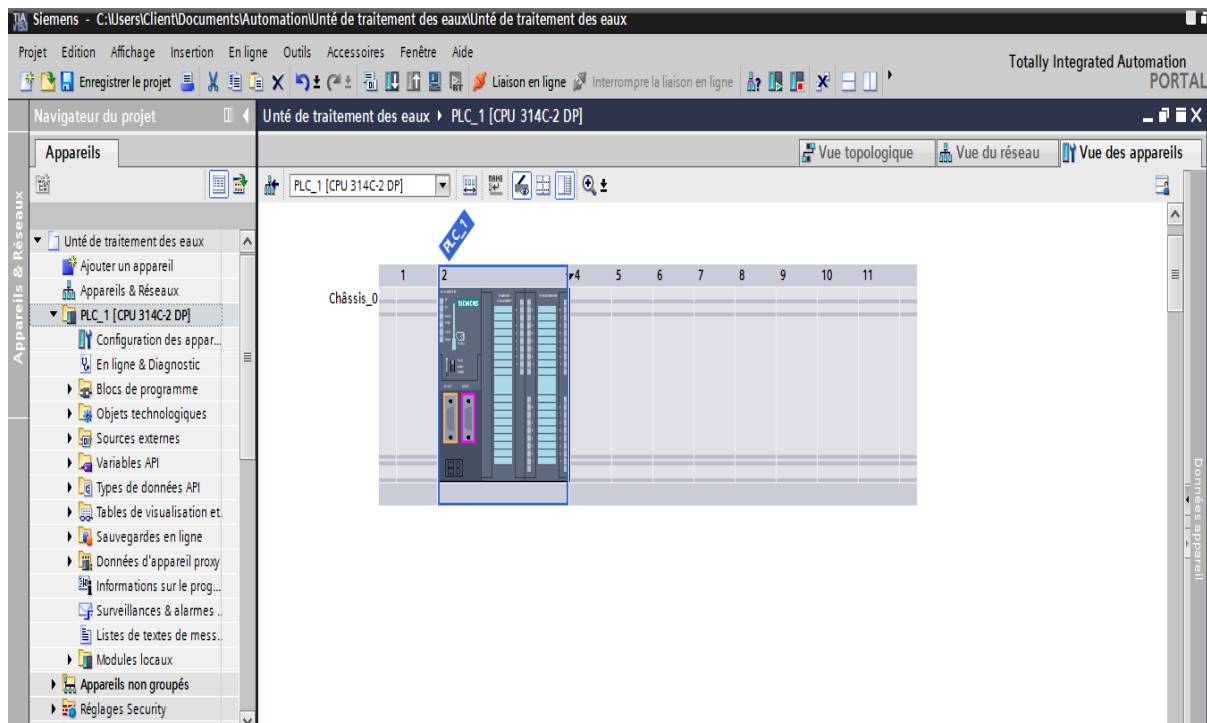


Fig. IV.1. Insertion ou l'ajout de l'automate choisi-S7-314-2DP.

## IV.5.2 Sélection du pupitre opérateur ou HMI de visualisation(Supervision)

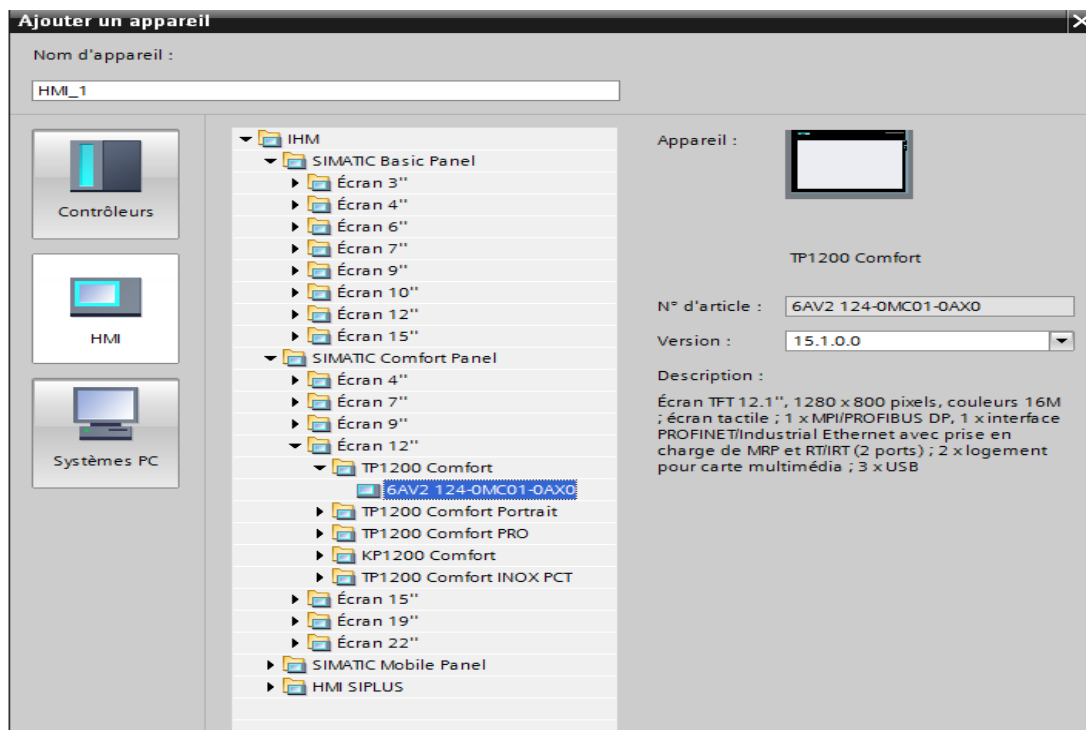


Fig. IV.2. Choix de pupitre de visualisation.

### IV.5.3. Présentation du pupitre opérateur utilisé

Le pupitre utilisé est un pupitre de type TP900 confort tactile qui offre de nombreuses possibilités d'application, une performance élevée et un excellent rapport qualité-prix, il prend en charge les tâches suivantes :

- Réduire le travail de configuration par l'utilisation des blocs d'affichages.
- Archivage des alarmes et des données de production.
- Emission automatique de compte-rendu d'état et de production au changement d'équipe.
- Bascule de la machine du mode manuel en mode automatique
- Gestion des opérations avec des droits d'accès différents.
- Présenter les états d'exploitation, Les valeurs actuelles du process de production et les alarmes de l'automate et d'autre part, décommander de manière aisée l'installation de production.
- Commutation de la langue entre le français et l'anglais.

#### IV.5.3.1. Caractéristiques SIMATIC HMI TP900 Comfort

- Écran tactile
- Commande tactile
- Écran TFT 9" (grand écran)
- 16 Mio. Couleurs
- Interface PROFINET
- Interface MPI/PROFIBUS-DP
- Mémoire de configuration : 12 MByte
- Windows CE 6.0 (Support-Microsoft inclusif (mises à jour de sécurité installé)
- Configurable à partir de WinCC Comfort V11



Fig. IV.2. Choix de pupitre de visualisation.

### IV.5.3.2 Configuration de L'HMI

Les figures suivantes décrivent les étapes de configuration d'un HMI

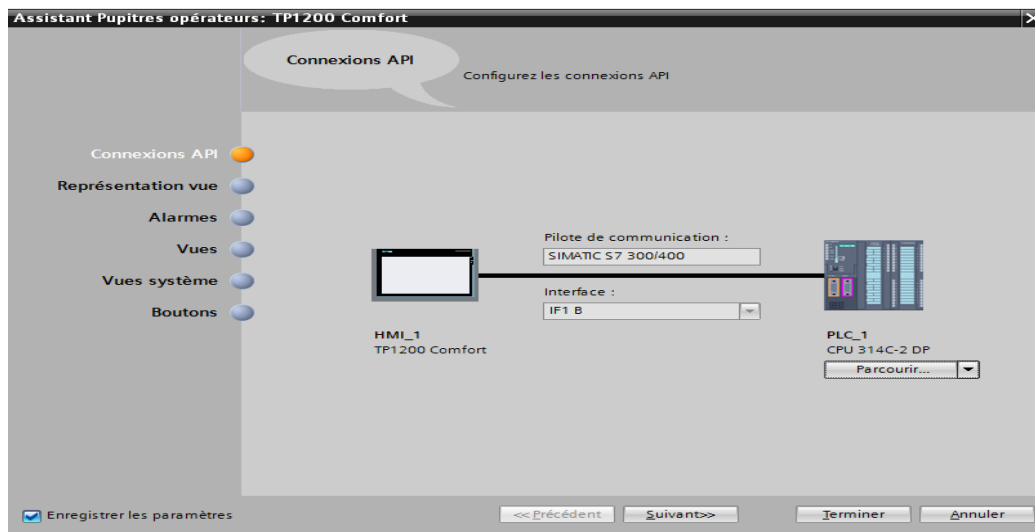


Fig. IV.4. Connexion HMI/API.

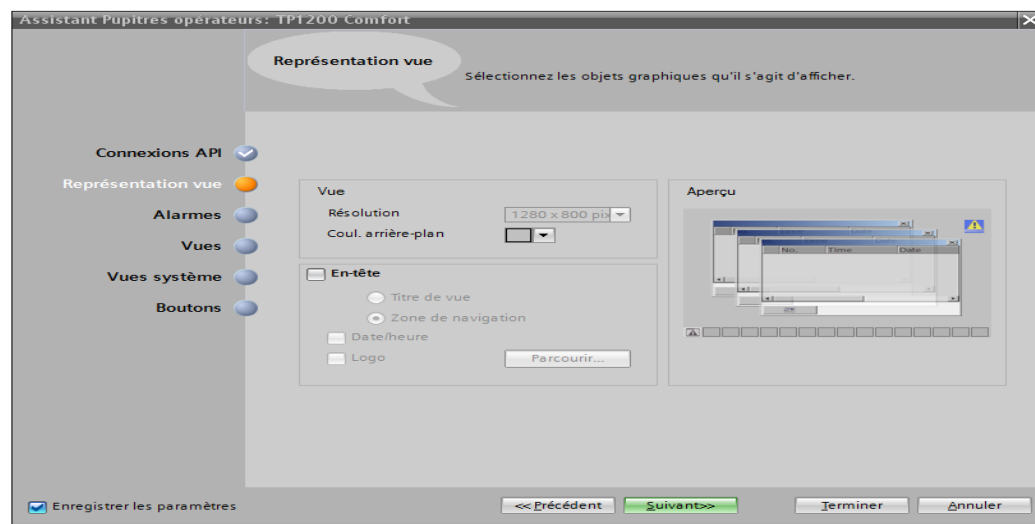


Fig. IV.5. Représentation.

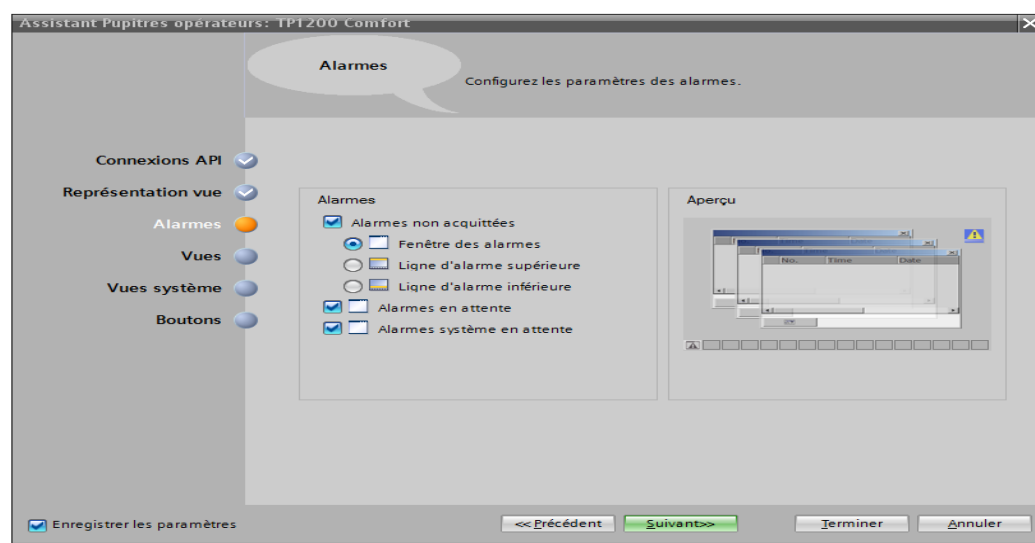


Fig. IV.6. Configurations des paramètres des alarmes.

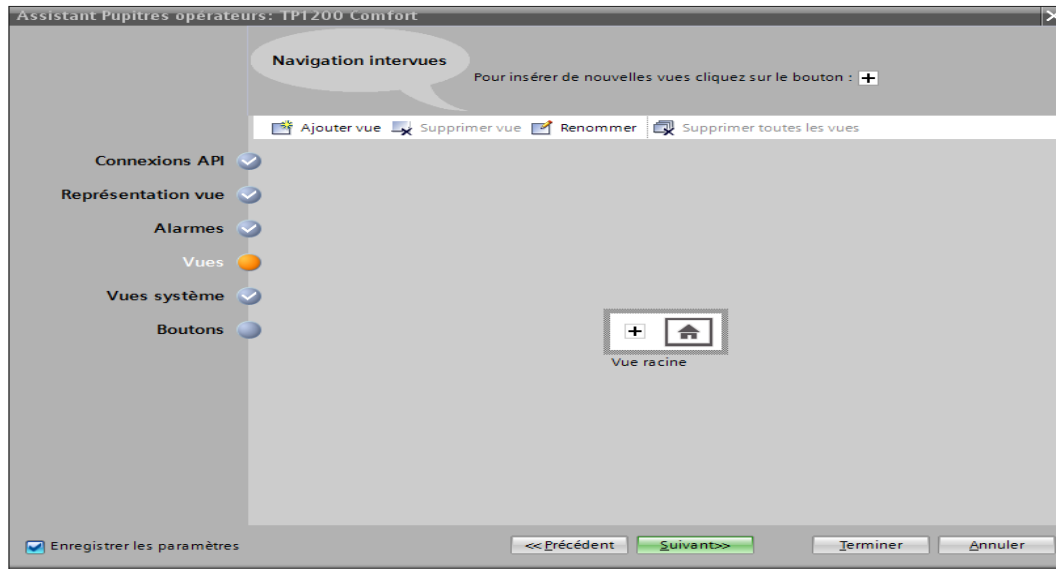


Fig. IV.7. Navigations inter vues.

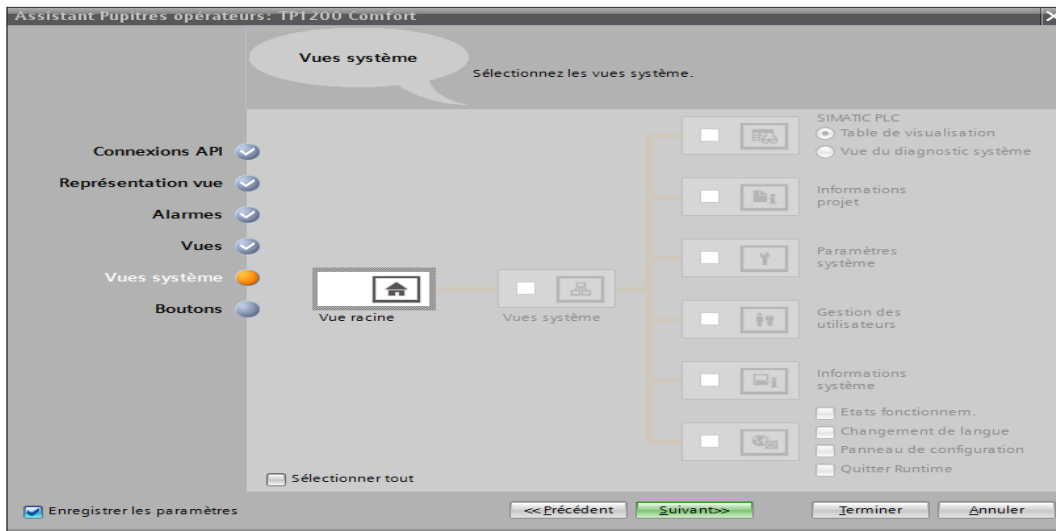


Fig. IV.8. Vue système.

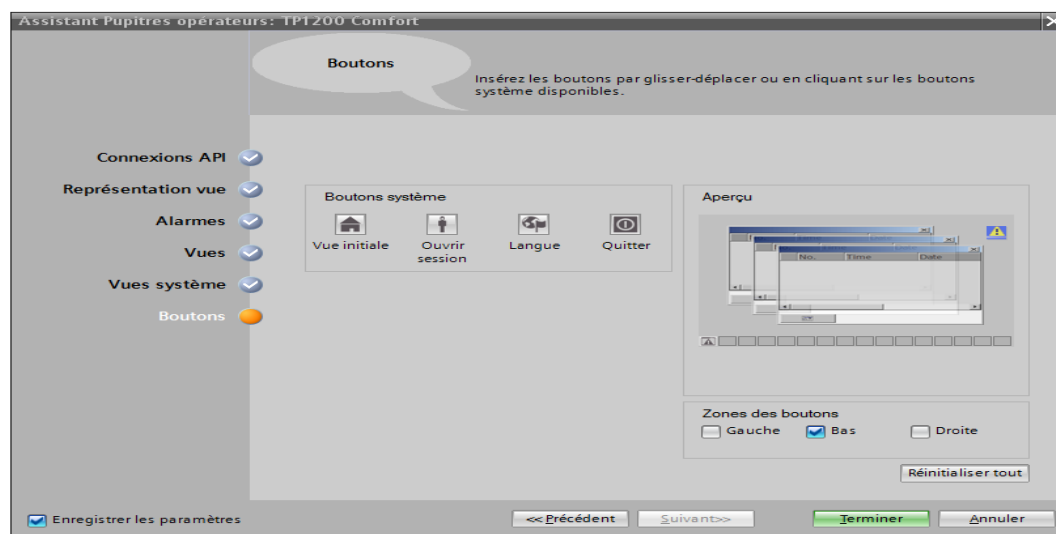


Fig. IV.9. Les boutons.

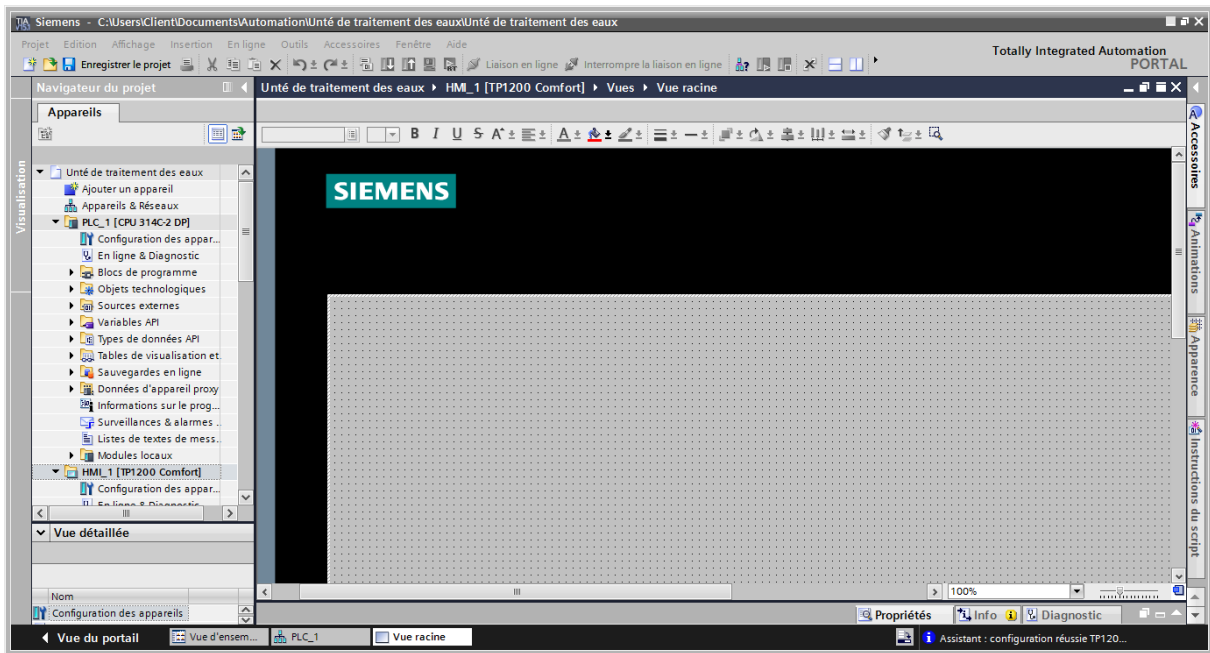


Fig. IV.10. Vue racine du projet

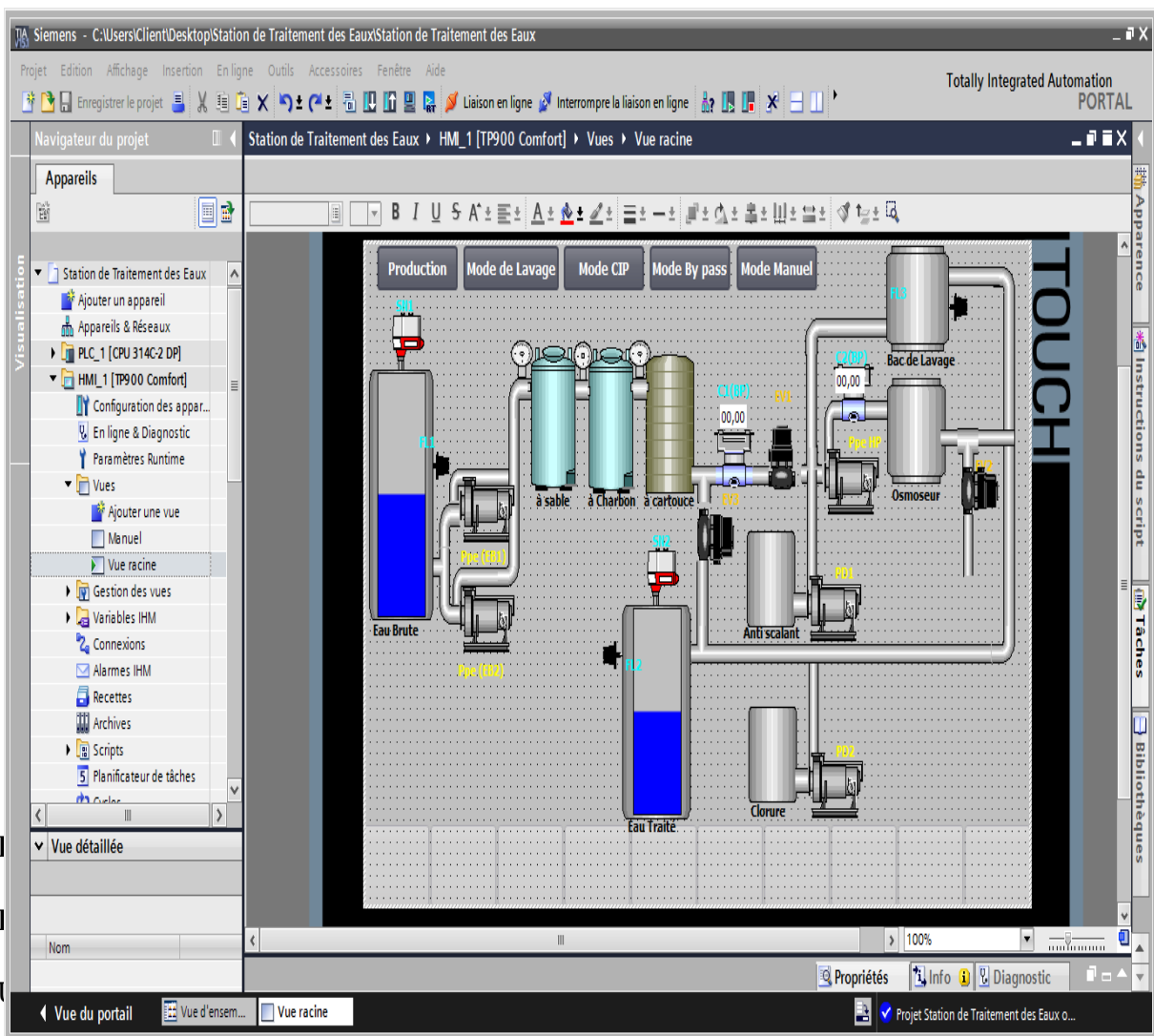


Fig. IV.11. HMI vue globale de l'unité de traitement des eaux.



## IV.6. Les variables HMI

Dans tous les programmes, il est nécessaire de définir une liste de variables qui seront utilisées lors de la programmation, pour lesquelles une table de variables est créée. L'utilisation de noms propres facilite la compréhension et l'utilisation des programmes. Ce type d'adressage est dit "relatif". Après le nom, nous définissons le type de données de la variable, puis l'adresse. La table de variables d'entrées et de sorties sont défini selon les variables du cahier de charges. Les tableaux suivant divisé en deux parties représentent tous les variables utilisés dans la supervision de la station de traitement des eaux avec leur types et

The figure displays two screenshots of the Siemens TIA Portal software interface, showing the 'Table de variables standard' (Standard Variable Table) for an HMI project. The table lists variables used in the supervision of a water treatment station, including their names, data types, connections, API names, API variables, and addresses.

**Table 1: Variables from the first screenshot**

Nom	Type de données	Connexion	Nom API	Variable API	Adresse
C1_BP	Real	HMI_Liaison_1	PLC_1	%M12	%M12
C2_HP	Real	HMI_Liaison_1	PLC_1	%M18	%M18
EV1	Bool	HMI_Liaison_1	PLC_1	%Q124.2	%Q124.2
EV2	Bool	HMI_Liaison_1	PLC_1	%Q124.3	%Q124.3
EV3	Bool	HMI_Liaison_1	PLC_1	%Q124.4	%Q124.4
FL1-Flotteur Niveau bas Eau Br...	Bool	HMI_Liaison_1	PLC_1	"FL1-Flotteur Niveau bas ..."	%I24.5
FL2-Niveau Haut Citerne Eau Tr...	Bool	HMI_Liaison_1	PLC_1	"FL2-Niveau Haut Citerm...	%I24.6
FL3-Niveau bas Bac de lavage	Bool	HMI_Liaison_1	PLC_1	"FL3-Niveau bas Bac de L...	%I24.7
Mode Bypass	Bool	HMI_Liaison_1	PLC_1	"Mode Bypass"	%M0.3
Mode CIP	Bool	HMI_Liaison_1	PLC_1	"Mode CIP"	%M0.2
Mode Lavage	Bool	HMI_Liaison_1	PLC_1	"Mode Lavage"	%M0.1
Mode Manuel	Bool	HMI_Liaison_1	PLC_1	"Mode Manuel"	%M0.4
Mode Production	Bool	HMI_Liaison_1	PLC_1	"Mode Production"	%M0.0
Ppe1-Eau Brute	Bool	HMI_Liaison_1	PLC_1	"Ppe1-Eau Brute"	%Q124.0
Ppe2-Eau Brute	Bool	HMI_Liaison_1	PLC_1	"Ppe2-Eau Brute"	%Q124.1
Ppe-HP	Bool	HMI_Liaison_1	PLC_1	"Ppe-HP"	%Q124.5
Ppe-PD1	Bool	HMI_Liaison_1	PLC_1	"Ppe-PD1"	%Q124.6
Ppe-PD2	Bool	HMI_Liaison_1	PLC_1	"Ppe-PD2"	%Q124.7
S1_Prod	Bool	HMI_Liaison_1	PLC_1	S1_Prod	%M30.1
S2_Lavage	Bool	HMI_Liaison_1	PLC_1	S2_Lavage	%M30.2
S3_CIP	Bool	HMI_Liaison_1	PLC_1	S3_CIP	%M30.3
S4_Bypass	Bool	HMI_Liaison_1	PLC_1	S4_Bypass	%M30.0
S5_Manuel	Bool	HMI_Liaison_1	PLC_1	S5_Manuel	%M30.4

**Table 2: Variables from the second screenshot**

Nom	Type de données	Connexion	Nom API	Variable API	Adresse
S5_Manuel	Bool	HMI_Liaison_1	PLC_1	%M130.4	%M130.4
SN1_Analog	Real	HMI_Liaison_1	PLC_1	%MD92	%MD92
SN2_Analog	Real	HMI_Liaison_1	PLC_1	%MD98	%MD98
Start_Ev1	Bool	HMI_Liaison_1	PLC_1	Start_Ev1	%M2.0
Start_ev2	Bool	HMI_Liaison_1	PLC_1	Start_ev2	%M2.1
Start_ev3	Bool	HMI_Liaison_1	PLC_1	Start_ev3	%M2.2
Start_PD1	Bool	HMI_Liaison_1	PLC_1	Start_PD1	%M2.4
Start_PD2	Bool	HMI_Liaison_1	PLC_1	Start_PD2	%M2.5
Start_PHP	Bool	HMI_Liaison_1	PLC_1	Start_PHP	%M2.6
Start_PPE1	Bool	HMI_Liaison_1	PLC_1	Start_PPE1	%M2.7
Start_PPE2	Bool	HMI_Liaison_1	PLC_1	Start_PPE2	%M2.3
Stop_EV1	Bool	HMI_Liaison_1	PLC_1	Stop_EV1	%M3.0
Stop_EV2	Bool	HMI_Liaison_1	PLC_1	Stop_EV2	%M3.1
Stop_EV3	Bool	HMI_Liaison_1	PLC_1	Stop_EV3	%M3.2
Stop_PD1	Bool	HMI_Liaison_1	PLC_1	Stop_PD1	%M3.5
Stop_PD2	Bool	HMI_Liaison_1	PLC_1	Stop_PD2	%M3.6
Stop_PHP	Bool	HMI_Liaison_1	PLC_1	Stop_PHP	%M3.7
Stop_PPE1	Bool	HMI_Liaison_1	PLC_1	Stop_PPE1	%M3.3
Stop_PPE2	Bool	HMI_Liaison_1	PLC_1	Stop_PPE2	%M3.4
<ajouter>					

Fig. IV.12. Les variables HMI.

## IV.7 Vue Marche Manuel de l'unité de traitement des eaux

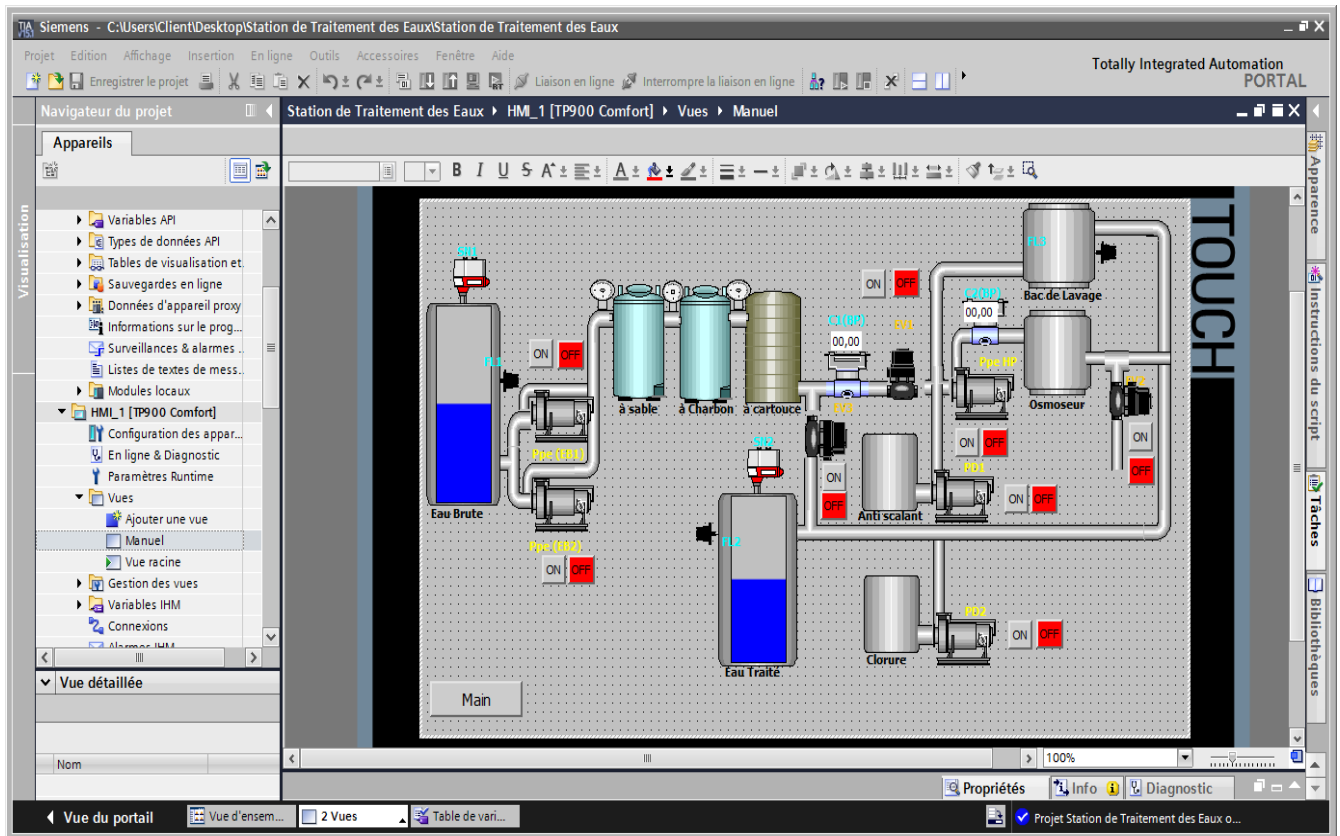


Fig. IV.13. Connexion HMI /API.

## IV.8 Connexion HMI / API

Nom	Pilote de communication	Mode de synchronisation de l'heure	Station	Partenaire	Noeud	En ligne	Commentaire
HMI_Liaison_1 <ajouter>	SIMATIC S7 300I400		Station S7300IE T20...	PLC_1	CPU 314C-2 DP, Int...	<input checked="" type="checkbox"/>	

**TP900 Comfort**  
Interface : IF1 B

**Appareil IHM**  
Type : SIMATIC  
Vitesse de transmission : 187500  
Adresse : 1  
Point d'accès : S7ONLINE  
Seul maître sur le bus :

**Réseau**  
Profil : MPI  
Adresse de station la plus élevée (HSA) : 31  
Nombre de maîtres : 1

**PLC**  
Adresse : 2  
Emplacement extension : 2  
Châssis : 0  
Fonctionnement cyclique :

Fig. IV.14. Vue Marche Manuel de l'unité de traitement des eaux.

## **IV.9. Conclusion**

Dans ce chapitre nous avons réalisé les vues de la supervision de la station de traitement des eaux qui permet de suivre l'évolution du procédé en temps réel. Nous avons constaté que le logiciel de supervision WinCC sous la plateforme Tia Portal V15.1 est très riche en options, il est très puissant dans les solutions globales d'automatisation car il assure un flux continu d'informations. Ses composants permettent d'intégrer sans problème les applications dont on a besoin, il combine l'architecture moderne Windows et la simplicité du logiciel de conception graphique et intègre tous les composants nécessaires aux tâches de visualisation et de pilotage. Il suffit d'imaginer le design de l'installation et tous les effets d'animation qui seront nécessaires pour bien apporter l'état réel de l'installation à l'opérateur avec plus d'informations à partir des messages configurés et l'attribution des différentes couleurs pour les états des différents objets.

L'utilisation du PLCSIM nous permet de tester les programmes développés, puisque TIA PORTAL offre différentes possibilités de test, telles que la visualisation et la supervision, avec le support du logiciel WINCC.

*CONCLUSION*

*CONCLUSION*

*GENERALE*

*GENERALE*

## Conclusion Générale

Les progrès technologiques de ces dernières années ont abouti au développement des automates programmables industriels et à une révolution conséquente de l'ingénierie de contrôle/commande et l'intégration des automates programmables pour le contrôle des différents processus industriels est aujourd'hui de plus en plus la solution la plus recherchée dans l'industrie vue la justesse du traitement numérique qu'il effectue pour générer la commande adéquate à tout moment dans toutes les conditions et renforce le degré de fiabilité de l'équipement et offre une très grande adaptabilité face aux évolutions de l'environnement.

Aujourd'hui, l'automate programmable n'est plus seulement une machine séquentielle mais il est beaucoup plus considéré comme un calculateur de processus grâce aux énormes progrès quant à la structure de base, la qualité et la diversité des outils proposés. Son intégration sur Fieldbus (Profibus, WorldFip), sur Ethernet (Standard TCP-IP), accroît ses possibilités et constitue un passage obligé pour augmenter la performance des processus. La maîtrise des systèmes de contrôle et de commande permet aux entreprises d'améliorer la qualité des produits, d'améliorer la productivité et de sécuriser l'ensemble des équipements.

Dans ce travail, on a proposé de remplacer la commande câblée de l'unité de traitement des eaux considérée par un automate programmable industriel afin d'assurer la sécurité, la rapidité et la fiabilité du system. Pour cela, notre choix est fixé sur l'utilisation l'automatisation programmable comme moyen de commande, de contrôle et de supervision et ce pour :

- La complexité de l'installation existante (logique électrique câblée).
- Le diagnostic des pannes et entretien relativement difficiles.
- Problème de l'inflexibilité (installation de contrôle figée).
- Efficacité minime et autres problèmes économiques tels que couts et d'entretien ...etc.

Pour cela, l'automate de la gamme SIMATIC S7- 314-D2P de SIEMENS est utilisé avec le logiciel de programmation TIA Portal V15, ce qui nous a appris plein de choses sur l'automatisation et acquérir une expérience réelle munie d'une grande capacité d'information, astuce et secret de l'industrie, en concluant que :

- L'étude et la compréhension du cahier de charges est une étape primordiale est nécessaire pour résoudre la problématique envisagée.
- Une bonne analyse du problème sert en premier lieu à déterminer les entrées et les sorties du système en question, ce qui permettra par la suite à réfléchir au choix

de l'automate programmable, de sa CPU et de tous les types des modules qui vont être utilisés.

- La mise en évidence de notre travail est basée sur le choix d'un API (S7-314-D2P) avec le logiciel de programmation TIA Portal.
- Le logiciel de programmation TIA Portal constitue le lien entre l'utilisateur et l'automate programmable (S7-314-D2P) car ce dernier ne peut gérer ces fonctionnalités sans logiciel bien définit.
- Les systèmes automatisés par l'API sont mis en sécurité, et ont plus de garanti, une rapidité et une fiabilité du système de contrôle automatique. Ainsi que la réduction du nombre de pannes.
- Grace à la présentation graphique offerte par les applications WINCC de PLC SIM, l'illustration de n'importe qu'elle partie du system industriel, ainsi que la localisation de la panne ou du défaut possible, sont simplifiées
- L'étude détaillée de la machine, nous a permet de toucher plusieurs disciplines l'informatique, automatisme et communication.
- Les différents langages de programmation facilitent le contrôle à partir de terminaux de programmation très simples et économiques surtout pour l'operateur et la maintenance.
- Ce travail nous a permis aussi, non seulement d'approfondir mes connaissances en automatisme mais aussi d'acquérir une expérience extrêmement valorisante de point de vue programmation des automates programmable par différents langages de programmation et spécialement l'automate S7-314-D2P de Siemens.

Enfin, nous pourrions dire que remplacer le dispositif classique de commande et de signalisation automatique d'une unité de traitement des eaux par un Automate Programmable Industriel révèle comme étant la solution qui subviendrait aux besoins du monde industriel future voire même actuel qui ne cesse de trop exiger pour ce qui est de qualité et quantité des produits.

# *BIBLIOGRAPHIQUE*



## ***Bibliographie***

	<i><b>Bibliographie</b></i>
[1]	MELLALI Sofiane, YOUSFI Iounis, Etude de l'automatisation et de la supervision d'un procédé de lavage de filtres Niagara à CEVITAL - TIA PORTAL V12 - , université de Bejaia, 2017
[2]	C.VRIGNON et M.THENAISIE, « l'automatisation », ISTIA, 17 octobre 2005.
[3]	Alain GONZAGA, « Les automates programmables industriels », 2004
[4]	Michel G, « Architecture et application des automates programmables industriels » DUNOD, Paris, 1987.
[5]	Hans Berger, "Automating with SIMATIC", second edition, 2003.
[6]	Siemens AG, «SIMATIC STEP 7 dans le portail Totally Integrated Automatio Portal », 2013.
[7]	Document de formation pour une solution complète d'automatisation TOTALLY INTEGRATED automation, édition 05/2004 (fr : 05/2005)
[8]	BENMOUMENE Hamza, AFFETTOUCHE Mouloud, '' AUTOMATISATION D'UNE STATION DE DEMINERALISATION D'EAU'', université de Tizi-Ouzou, 2013
[9]	<a href="http://fst.univ-guelma.dz/sites/default/files/field/Chapitre%203.%20API.pdf">http://fst.univ-guelma.dz/sites/default/files/field/Chapitre%203.%20API.pdf</a>
[10]	<a href="http://cv.automatismes.free.fr/cours%20portal/tia_portal_prise_en_main_1.pdf">http://cv.automatismes.free.fr/cours%20portal/tia_portal_prise_en_main_1.pdf</a>
[11]	Mohand Ameziane, Cheref Massinissa Benziane, '' Etude, automatisation et supervision de la ligne de lavage du verre feuilleté au niveau de l'unité Coater à l'entreprise MFG (Cevital)'' , Mémoire de Master en automatique industrielle, université de Tizi-Ouzou, 2018.
[12]	SIEMENS, documentation et manuel d' utilisation de TIA PORTAL V13 « SIEMENS, manuel produit TIA PORTAL V13 », 2013.
[13]	Manuels SIEMENS, « WinCC flexible 2008 », SIMATIC HMI, 2008.