

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'enseignement supérieur

Et de la recherche scientifique

ECOLE NATIONALE POLYTECHNIQUE D'ALGER



المدرسة الوطنية المتعددة التقنيات  
Ecole Nationale Polytechnique

Département Automatique



Mémoire de Projet de Fin d'Etudes

Pour l'obtention du Diplôme d'Ingénieur d'Etat en Automatique

---

## Développement des techniques de commande et supervision d'une station de pompage

---

**KAMEL Rania Fairouz**

Sous la direction de M. El Madjid BERKOUK professeur à l'ENP Alger

Présenté et soutenu publiquement le 09/07/2023

**Membres De Jury :**

---

Président	M.Mohamed Seghir Boucherit	Professeur	ENP
Promoteur	M .El madjid Berkouk	Professeur	ENP
Examineur	M.Djamel Boudana	Professeur	ENP

---



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'enseignement supérieur

Et de la recherche scientifique

ECOLE NATIONALE POLYTECHNIQUE D'ALGER



المدرسة الوطنية المتعددة التقنيات  
Ecole Nationale Polytechnique

Département Automatique



Mémoire de Projet de Fin d'Etudes

Pour l'obtention du Diplôme d'Ingénieur d'Etat en Automatique

---

## Développement des techniques de commande et supervision d'une station de pompage

---

**KAMEL Rania Fairouz**

Sous la direction de M. El Madjid BERKOUK professeur à l'ENP Alger

Présenté et soutenu publiquement le 09/07/2023

**Membres De Jury :**

---

Président	M.Mohamed Seghir Boucherit	Professeur	ENP
Promoteur	M .El madjid Berkouk	Professeur	ENP
Examineur	M.Djamel Boudana	Professeur	ENP

---

تهدف هذه المذكرة إلى تطوير محطة للضخ تستخدم في التجارب العملية. تم تصميم هذه المحطة بطريقة تسمح بإنشاء عدة مواصفات للتلاعب. تتألف المحطة من خزانين يتم تغذيتهما بواسطة مضخة عبر صمامات كهرومغناطيسية. بالإضافة إلى ذلك ، يتم استخدام محرك الخلط مع بداية تدريجية.

تم التحكم في المحطة بواسطة نظامين أوتوماتيكيين من شنايدر وسيمنز. من خلال هذه الأنظمة ، سيتم تنفيذ عدة أنظمة تحكم صناعي. ستستخدم هذه المحطة كنموذج لفهم مفهوم دليل دراسة وضعيات التشغيل والإيقاف GEMMA . سيتم تقديم حلول للحصول على تشطيب دقيق للمحطة وضمان مستوى عالٍ من الأمان أثناء استخدامها.

كلمات مفتاحية : لوغو! ، واجهات برمجة التطبيقات ، شنايدر ، زيليو لوجيك ، إيزيبيلدر برو ، كيبسرفر إكس.

## Abstract

This thesis aims to develop a pumping station intended for use in practical exercises. This station is designed to allow the creation of multiple specifications for manipulations. It consists of two reservoirs fed by a pump through solenoid valves. Additionally, an agitator paired with a progressive starter is utilized.

The station is controlled by two automation systems from Schneider and Siemens. Through these systems, several industrial control systems will be implemented. This station will serve as a model to understand the concept of the Guide for the Study of Operating and Shutdown Modes (GEMMA).

Proposed solutions will ensure a meticulous finish for the station and guarantee a high level of safety during its operation.

**Keywords :** LOGO!, APIs, Schneider, Zelio Logic, Easybuilder Pro, KepserverEX.

## Résumé

Ce mémoire vise le développement d'une station de pompage destinée à être utilisée dans les travaux pratiques. Cette station est réalisée de manière à permettre la création de plusieurs cahiers des charges pour les manipulations. Elle est composée de deux réservoirs alimentés par une pompe à travers des électrovannes. De plus, un agitateur associé à un démarreur progressif est utilisé.

Elle est commandée par deux automates des gammes Schneider et Siemens. À travers ces automates, plusieurs systèmes de contrôle industriel seront réalisés. Cette station sera utilisée comme modèle pour comprendre la notion de Guide d'étude des Modes de marche et d'arrêt (GEMMA).

Des solutions seront proposées pour obtenir une finition soignée pour la station et garantir un niveau de sécurité élevé lors de son utilisation.

**Mots clés :** LOGO!, APIs, Schneider, Zelio Logic, Easybuilder Pro, KepserverEX.

# *Remerciements*

*Je tiens à remercier dieu de m'avoir donné la force morale, physique et l'aide pour accomplir ce modeste travail.*

*je tiens à exprimer mes sincères remerciements à mon encadreur, M. El Madjid BERKOUK, pour son soutien, son expertise et ses précieux conseils tout au long de ce projet. Sa disponibilité et sa bienveillance ont grandement contribué à son aboutissement.*

*J'adresse également mes remerciements chaleureux aux membres du jury de la soutenance pour avoir consacré leur temps et leur attention à évaluer ce travail.*

*Je souhaite également exprimer ma gratitude envers l'ensemble des enseignants du département d'automatique. Leurs connaissances et leur dévouement ont été essentiels dans ma formation et ont joué un rôle majeur dans la réalisation de ce projet.*

*Enfin, je tiens à remercier toutes les personnes qui ont contribué de près ou de loin à la réussite de ce travail. Leur soutien et leur encouragement ont été précieux tout au long de ce parcours.*

## *Dédicace*

*Je dédie ce travail :*

*À la femme la plus chère au monde, qui a fait de moi ce que je suis  
aujourd'hui Ma mère*

*À mon cher Frère pour son soutien durant toutes mes  
années d'études*

*À mes deux petites sœurs Nerdjes et Maria, ainsi qu'aux membres  
de ma famille*

*À mes amies de l'école pour leur  
encouragement durant mes années d'études à l'ENP*

# Table des matières

Liste des tableaux

Table des figures

Liste des Abréviations

**Introduction Générale** **13**

**1 Présentation de la maquette de la Station de Pompage** **15**

1.1	Introduction . . . . .	15
1.2	Description du Prototype de La Station de Pompage . . . . .	15
1.2.1	Les Equipements de La Maquette . . . . .	16
1.2.2	Schémas de Raccordement . . . . .	18
1.2.3	Representation Synoptique de la Station . . . . .	19
1.3	Cahiers des Charges et Grafjets . . . . .	19
1.3.1	1er Cahier de Charge . . . . .	19
1.3.2	2éme Cahier de Charge . . . . .	21
1.4	Conclusion . . . . .	23

**2 Commande et Supervison de la Station de Pompage avec Automate Schneider** **24**

2.1	Introduction . . . . .	24
2.2	Module Programmable Zelio Logic . . . . .	24
2.3	La Programmation du Zelio Logic . . . . .	25
2.3.1	La Programmation sur L'Écran LCD . . . . .	25
2.3.2	La Programmation avec le Logiciel Zelio Soft . . . . .	27
2.4	Commande et Supervision de la Station de Pompage avec Automate Zelio Logic . . . . .	33
2.4.1	Schémas de Câblage des Entrées et Sorties . . . . .	33
2.4.2	Affectation des Variables . . . . .	33
2.4.3	La Commande de la Station de Pompage avec les Touches Z . . . . .	34
2.4.4	La Supervision de la Station de Pompage à travers l'écran LCD . . . . .	34
2.5	Conclusion . . . . .	36

**3 Commande et Supervison de la Station de Pompage avec Automate SIEMENS** **37**

3.1	Introduction . . . . .	37
3.2	Module Programmable LOGO! . . . . .	37
3.3	Le Module LOGO! TDE . . . . .	38
3.4	LOGO!Soft Comfort . . . . .	39
3.4.1	Interface de Programmation . . . . .	39
3.4.2	Configuration du Matériel . . . . .	40
3.4.3	Déclaration des Variables . . . . .	40
3.4.4	Simulation du Programme . . . . .	41
3.4.5	Transfert du Programme . . . . .	41
3.4.6	Test en Ligne . . . . .	41

3.5	Commande et Supervision de la Station de Pompage avec Automate LOGO! 0BA8 . . . . .	42
3.5.1	Schémas de Câblage des Entrées et Sorties . . . . .	42
3.5.2	Table d'Affectation . . . . .	42
3.5.3	Paramètre Réseau du LOGO! . . . . .	42
3.5.4	La Commande et La Supervision de la Station de Pompage avec LOGO! TDE . . . . .	43
3.5.5	La Commande et La Supervision de la Station de Pompage avec WebServer . . . . .	48
3.5.6	La Commande et La Supervision de la Station de Pompage avec Excel et le KepserverEX . . . . .	50
3.6	Conclusion . . . . .	53
<b>4</b>	<b>La Commande et la Supervision de la Station de Pompage avec Easybuilder Pro</b>	<b>54</b>
4.1	Introduction . . . . .	54
4.2	Présentation du Logiciel EasyBuilder Pro . . . . .	54
4.2.1	Interface de Projet . . . . .	54
4.2.2	Barre d'outils . . . . .	55
4.2.3	Création d'un nouveau projet . . . . .	56
4.2.4	Configuration Matérielle . . . . .	57
4.2.5	Déclaration des Variables . . . . .	58
4.2.6	Déclaration des Alarmes . . . . .	58
4.2.7	Acquisition des données . . . . .	59
4.2.8	Simulation du programme . . . . .	60
4.2.9	Transfert du programme d'EasyBuilder Pro vers l'IHM . . . . .	61
4.2.10	Les Avantages d'utilisation le logiciel Easybuilder Pro . . . . .	61
4.3	La Commande et la Supervision avec Easybuilder et l'automate Zelio Logic . . . . .	62
4.3.1	Les Vues de l'IHM . . . . .	62
4.3.2	La Gestion des utilisateurs de la Sécurité . . . . .	65
4.3.3	Communication entre EasyBuilder Pro et Zelio Logic . . . . .	66
4.4	La Commande et la Supervision avec Easybuilder et l'automate LOGO! . . . . .	69
4.4.1	Point d'accès aux services techniques TSAP . . . . .	69
4.4.2	Configuration des paramètres réseau dans LOGO!Soft Comfort . . . . .	70
4.4.3	Configuration des paramètres réseau dans EasyBuilder Pro . . . . .	72
4.4.4	Comparaison entre la supervision Zelio Logic et LOGO! . . . . .	72
4.5	Commande et la supervision avec Easybuilder Pro et Zelio Logic et LOGO! . . . . .	74
4.6	Guide d'Etude de Mode de Marche et d'Arrêt GEMMA . . . . .	75
4.6.1	Le GEMMA de la station de Pompage . . . . .	76
4.6.2	La commande et supervision de GEMMA dans Easybuilder Pro . . . . .	78
4.7	Systèmes de Contrôle Industriel . . . . .	78
4.7.1	La communication entre LOGO! et S7 1200 . . . . .	81
4.7.2	La communication entre Easybilder Pro et S7 1200 . . . . .	83
4.8	Conclusion . . . . .	85
<b>5</b>	<b>Conception de l'armoire de la station de pompage</b>	<b>86</b>
5.1	Introduction . . . . .	86
5.2	Séparation de l'alimentation électrique des automates programmables . . . . .	86
5.3	L'ajout d'équipements électriques à la station de pompage . . . . .	87
5.4	Schémas électrique de la station de pompage . . . . .	88
5.5	Conclusion . . . . .	106
	<b>Conclusion Générale</b>	<b>107</b>
	<b>Bibliographie</b>	<b>108</b>



Annexe A Généralités sur les Automates Programmables Industriels	109
Annexe B Les Caractéristiques Techniques Des Equipements Electriques de la Station	116
Annexe C Les Caracteristiques des Automates Programmables Utilisés	124
Annexe D Présentation de Logiciel KepsverEX	128
Annexe E Les Programmes du Contrôle de La Station de Pompage	134
Annexe F Schéma Electrique	137
Annexe G Guide d'Etude de Mode de Marche et d'Arret GEMMA	138
Annexe H Support de TP	141

# Liste des tableaux

2.1	Table d'Affectation des Entrees et Sortie sur Zelio Logic . . . . .	33
3.1	Table d'Affectation des Variables . . . . .	42
5.1	Prix des équipements électriques . . . . .	88

# Table des figures

1.1	Maquette de la station de pompage . . . . .	15
1.2	Equipements de l'installation . . . . .	16
1.3	Equipements Electriques . . . . .	17
1.4	Les Actionneurs . . . . .	18
1.5	Principe de câblage de station de pompage . . . . .	18
1.6	Representation Synoptique de la Station . . . . .	19
1.7	Grafcet de Manipulation 1 . . . . .	20
1.8	Grafcet de Sécurité . . . . .	21
1.9	Grafcet de Conduite . . . . .	22
1.10	Grafcet de Fonctionnement Normal 1 . . . . .	23
1.11	Grafcet de Fonctionnement Normal 2 . . . . .	23
2.1	Automate Zelio Logic SR2 B201BD . . . . .	24
2.2	Les boutons de Navigation . . . . .	25
2.3	Exemple de Saisie des Eléments . . . . .	27
2.4	Etape 1 . . . . .	28
2.5	Etape 2 . . . . .	28
2.6	Etape 3 . . . . .	28
2.7	Etape 4 . . . . .	29
2.8	Interface de programmation avec Langage LAADER . . . . .	29
2.9	Interface de programmation avec Langage FBD . . . . .	30
2.10	Les Etapes de Simulations . . . . .	30
2.11	Les Etapes de Transfert vers Module . . . . .	31
2.12	Mettre le Module en Mode Stop . . . . .	31
2.13	Connection et diagnostique du Module . . . . .	32
2.14	Fenêtre de dialogue d'un module Zelio . . . . .	32
2.15	Schémas de Câblage du Zelio Logic . . . . .	33
2.16	Commande du Système a travers les Touches de Fonctions . . . . .	34
2.17	La Supervision sur l'écran LCD . . . . .	34
2.18	Visualisation des Etats des Sorties . . . . .	35
2.19	Configuration des Message Textes . . . . .	35
2.20	Vue Face LCD . . . . .	36
2.21	Vue 2 LCD . . . . .	36
2.22	Vue 3 LCD . . . . .	36
3.1	Automate LOGO! 0BA8 . . . . .	38
3.2	Afficheur LOGO! TDE et LOGO! . . . . .	38
3.3	Interface de Programme de LOGO!Soft Comfort . . . . .	39
3.4	La configuration du matériel de LOGO!Soft Comfort . . . . .	40
3.5	Déclaration des variables sur LOGO!Soft Comfort . . . . .	40
3.6	Transfert du Programme Pc- > Module . . . . .	41
3.7	Câblage des Entrées et Sorties . . . . .	42
3.8	l'Adresse IP du LOGO! . . . . .	43

3.9	Paramètres réseau de LOGO! TDE	43
3.10	Saisir l'adresse IP de LOGO!	43
3.11	Configuration de matériel de LOGO! TDE	44
3.12	Commmande avec LOGO! TDE	44
3.13	Texte de Message	45
3.14	Paramètre LOGO! TDE	45
3.15	Configuration supervision du LOGO! TDE	45
3.16	La Supervision du 1er cahier des charge	46
3.17	Vue 1 LOGO!TDE	46
3.18	Vue 2 LOGO!TDE	46
3.19	Vue 3 LOGO!TDE	47
3.20	Le rétroéclairage de LOGO! TDE	47
3.21	Le rétroéclairage TDE	47
3.22	Activation du Web Server	48
3.23	Activation du Web Server	48
3.24	Création du Mot de Passe de Web Server	48
3.25	Commmande avec LOGO! TDE	49
3.26	Supervision des Etats sur le Navigateur Web	49
3.27	Commmande et la supervision avec LOGO! sur Web Server	49
3.28	Commmande et la supervision avec LOGO! TDE sur Web Server	50
3.29	Déclaration des variables dans l'excel	51
3.30	Déclaration des variables Excel dans KepserverEX	51
3.31	Déclaration des variables LOGO! dans KepserverEX	52
3.32	Liaison entre Excel et LOGO! KepserverEX	52
3.33	Page d'accueil	53
3.34	Vue 2 Excel	53
4.1	Interface de Projet EasyBuilder	55
4.2	Barre d'édition	55
4.3	Barre de Projet	55
4.4	Barre d'Objet	55
4.5	Barre Données/Historiques	56
4.6	Barre IIoT/Énergie	56
4.7	Barre Vue	56
4.8	Barre Outil	56
4.9	Création d'un nouveau projet	57
4.10	Configuration Matérielle	57
4.11	Configuration des paramètres de communication de Zelio Logic et LOGO!	58
4.12	Déclaration des variables dans Easybuilder Pro	58
4.13	Déclaration des alarmes dans Easybuilder Pro	59
4.14	Configuration des message d'alarmes dans Easybuilder Pro	59
4.15	Configuration des message d'alarmes dans Easybuilder Pro	60
4.16	Simulation du programme sur EasyBuilder Pro	60
4.17	Transfert du programme vers IHM physique	61
4.18	Page d'accueil	62
4.19	Vue du système en état Normal et en défaut	62
4.20	Vue du progression	63
4.21	Vue du Grafcet	63
4.22	Vue d'alerte	64
4.23	Vue de la gestion des utilisateurs	64
4.24	Quelques vues du guide de l'utilisateur	64
4.25	Menu de Navigation	65

4.26	Paramètres de sécurité . . . . .	66
4.27	Vue d'accès des utilisateurs . . . . .	66
4.28	Message d'avertissement . . . . .	66
4.29	paramètres de communication entre zelio et EasyBuilder Pro . . . . .	67
4.30	Les blocs de communications sur Zelio Logic . . . . .	67
4.31	Les adresses compatibles avec Zelio Logic . . . . .	68
4.32	Les adresse des bloc $SL_{In}$ et $SL_{Out}$ . . . . .	68
4.33	Les adresse des bloc $SL_{In}$ et $SL_{Out}$ . . . . .	69
4.34	Représentation Client/Serveur . . . . .	70
4.35	Connexion Ethernet . . . . .	70
4.36	Liaison client/Serveur . . . . .	70
4.37	Paramètres de connexion client/serveur . . . . .	71
4.38	Configuration réseaux du client/serveur . . . . .	71
4.39	Connexion client avec le serveur . . . . .	71
4.40	Réseaux client/serveur . . . . .	72
4.41	Configuration des paramètres réseaux EasyBuilder Pro . . . . .	72
4.42	Variable lire/écrire dans Easybuilder Pro . . . . .	73
4.43	Paramètres des timers et des compteurs LOGO!Soft . . . . .	73
4.44	Vue système LOGO! . . . . .	73
4.45	Vue Système . . . . .	74
4.46	Vue Grafcet . . . . .	75
4.47	Feuille de GEMMA . . . . .	76
4.48	GEMMA de la station de pompage . . . . .	77
4.49	Commande et supervision de GEMMA avec EasyBuilder Pro . . . . .	78
4.50	Système de contrôle avec API . . . . .	79
4.51	Système de contrôle avec IHM . . . . .	79
4.52	Système de contrôle avec SCADA . . . . .	80
4.53	Prototype de pompage . . . . .	80
4.54	Grafcet du prototype de pompage . . . . .	81
4.55	Communication entre LOGO! et S7 1200 . . . . .	82
4.56	Communication entre KepserverEx et S7 1200 . . . . .	83
4.57	Lien des variables de LOGO! et S7 1200 . . . . .	83
4.58	Communication entre Easybuilder Pro et S7 1200 . . . . .	84
4.59	Vue S7 1200 sur Easybuilder Pro . . . . .	84
4.60	Alarmes S7 1200 sur EasyBuilder Pro . . . . .	84
4.61	système DCS . . . . .	85
5.1	Schéma synoptique de séparation de l'alimentation électrique des APIs . . . . .	87
5.2	Schéma électrique de séparation de l'alimentation électrique des APIs . . . . .	87
5.3	Vue 3D de l'armoire électrique . . . . .	88
5.4	Vue de face de l'armoire électrique . . . . .	89
5.5	Vue interieur de l'armoire électrique . . . . .	89

# Liste des Abréviations

<b>AC</b>	Courant alternatif
<b>ADEPA</b>	Agence pour le Développement de la Productique Appliquée
<b>API</b>	Automate Programmable Industriel
<b>BF</b>	Boucle fermée
<b>CAN</b>	Convertisseur numérique-analogique
<b>CAN</b>	Convertisseur analogique-numérique
<b>COM</b>	Communication
<b>CPU</b>	Central Processing Unit
<b>Da</b>	Dinar algérien
<b>DC</b>	Courant continue
<b>DCS</b>	Distributed Control System
<b>DDE</b>	Dynamic Data Exchange
<b>DNP</b>	Distributed Network Protocol
<b>E/S</b>	Entrée/Sortie
<b>FBD</b>	System Fonctionnel Bloc
<b>GC</b>	Grafcet de conduite
<b>GEMMA</b>	Guide d'Etude de Mode de Marche et d'Arret
<b>GFN</b>	Grafcet de fonctionnement normal
<b>GRAF CET</b>	Graphe Fonctionnel de Commande Etapes Transitions
<b>ICS</b>	Industrial Control Systems
<b>IHM</b>	Interface homme machine
<b>IOT</b>	Internet of Things
<b>IP</b>	Internet Protocol
<b>IP</b>	Indice de protection
<b>LCD</b>	Liquid Crystal display
<b>MPI</b>	Multi-Point Interface
<b>OPC</b>	OLE pour le contrôle des processus
<b>PC</b>	Personal computer
<b>PLC</b>	Programmable Logic Controller
<b>PWM</b>	Pulse Width Modulation (Modulation à Largeur d'impulsion)
<b>RAM</b>	Random Access Memory
<b>SCADA</b>	Supervisory Control And Data Acquisition
<b>SFC</b>	Sequential Function Chart
<b>SNMP</b>	Simple Network Management Protocol
<b>TCP</b>	Transmission Control Protocol
<b>TOR</b>	Tout Ou Rien
<b>TSAP</b>	Technical Service Access Point
<b>USB</b>	Universal Serial Bus

# Introduction Générale

De nos jours, l'automatisation joue un rôle primordial dans l'amélioration de la productivité et de la qualité des services dans divers systèmes. Elle permet d'optimiser l'utilisation des ressources et de réduire les erreurs humaines. L'automatisation peut être mise en place à différents niveaux, en fonction de la complexité du système ou du processus étudié.

La mise en œuvre de l'automatisation nécessite l'expertise d'un ingénieur en automatique. Ce professionnel doit posséder des connaissances fondamentales en automatisation industrielle, notamment la maîtrise des différents automates, une analyse approfondie des projets, la capacité à proposer plusieurs solutions pour le contrôle et la supervision, ainsi que la mise en place de la communication entre différents systèmes, qui représente souvent un défi majeur dans l'industrie. Grâce à ces compétences de base, l'ingénieur est en mesure d'automatiser efficacement n'importe quel système.

Pour acquérir ces connaissances de base, il est essentiel de les intégrer dans le parcours universitaire. C'est pourquoi le département d'automatique de l'ENP (École Nationale Polytechnique) a mis en place une maquette de la station de pompage en utilisant des automates programmables tels que Siemens et Schneider. Cette maquette est destinée à être utilisée ultérieurement dans le cadre des travaux pratiques en laboratoire, en tant que système d'apprentissage didactique pour l'automatisation. La conception de cette maquette vise à appliquer les connaissances acquises en informatique industrielle et à répondre à un certain nombre d'exigences de formation professionnelle.

Dans le présent rapport, nous exposons cinq grands chapitres décrivant les volets principaux de notre projet :

Le premier chapitre se concentre sur la description du système étudié, en mettant en évidence une présentation détaillée de la maquette de la station de pompage. Nous décrivons les équipements de l'installation afin d'avoir une compréhension approfondie de la maquette, ce qui facilite également la proposition de cahiers des charges adaptés au système. À la fin du chapitre, nous présentons les objectifs de notre projet ainsi que les deux cahiers des charges proposés pour répondre à nos objectifs.

Le deuxième chapitre est consacré au contrôle et à la supervision de la station avec l'automate Zelio Logic de la gamme Schneider Electric. Dans ce chapitre, nous abordons en premier lieu la description de cet automate, les méthodes de programmation associées, et enfin, nous expliquons comment réaliser la commande et la supervision de la station en détail, conformément au premier cahier des charges.

Le troisième chapitre est réservé au contrôle et à la supervision de la station avec l'automate LOGO! de la gamme Siemens. Nous présentons une description détaillée de l'automate LOGO! et de son logiciel de programmation LOGO!Soft Comfort. Ensuite, nous présentons trois méthodes différentes de commande et de supervision de la station : l'utilisation de l'afficheur LOGO! TDE de Siemens, le WebServer et le KepserverEX, conformément au premier cahier des charges.

Dans le quatrième chapitre, nous abordons une autre technique de commande et de supervision de la station qui sera appliquée pour la première fois dans le département d'automatique. Cette

méthode repose sur le contrôle et la supervision de la station avec Easybuilder Pro. Nous décrivons également le logiciel, présentons les vues créées et expliquons les avantages de cette méthode par rapport à celle présentée précédemment. Cette méthode sera appliquée aux deux cahiers des charges proposés, et à travers cette méthode, plusieurs notions telles que les systèmes de contrôle industriel et le GEMMA seront expliquées. Différents types de communication seront également détaillés.

Le cinquième et dernier chapitre est réservé à la conception de l'armoire de la station de pompage. L'objectif de ce chapitre est de développer le système en termes de câblage afin de résoudre les lacunes électriques de l'installation. Nous proposerons quelques modifications tout en préservant l'intégrité du système. Nous présenterons un schéma électrique détaillé ainsi que le design souhaité de l'armoire finale.

L'objectif principal de ce projet est d'acquérir des compétences en informatique industrielle en réalisant des manipulations simples, claires et accessibles à tous les étudiants grâce à l'utilisation d'API. Les API permettront de contrôler les actionneurs et de programmer des manipulations à l'aide de logiciels spécifiques à chaque automate.



# Chapitre 1

## Présentation de la maquette de la Station de Pompage

### 1.1 Introduction

Ce chapitre a pour objectif de présenter la station de pompage qui fait l'objet de ce projet de fin d'études. Il comprend une description approfondie des équipements qui composent la station ainsi qu'une représentation de leur raccordement, ce qui permet de faciliter la réalisation des cahiers des charges appropriés. Il inclut également une présentation synoptique de la station qui permet de mieux comprendre son fonctionnement. Enfin, ce chapitre présente les cahiers des charges qui guideront les travaux à venir dans les autres chapitres.

### 1.2 Description du Prototype de La Station de Pompage

La station de pompage illustrée ci-dessous est un dispositif éducatif qui permet aux étudiants d'apprendre différentes techniques de commande et de supervision, principalement à travers les deux automates LOGO! Siemens et Zelio Logic Schneider.

Cette maquette a été conçue de manière à offrir la possibilité de réaliser différents cahiers de charge et de manipuler différentes configurations avec la station telle quelle (sans modifications au niveau du montage et du câblage). Cela permet d'acquérir des compétences et de faire face à des problématiques souvent rencontrées dans l'industrie, telles que la communication automate-IHM, automate-automate.

Elle permet également de développer l'esprit d'ingénieur des étudiants, en les encourageant à chercher plusieurs solutions pour répondre à un cahier des charges, une situation ou une problématique donnée, et à choisir la meilleure parmi elles.



FIGURE 1.1 – Maquette de la station de pompage

## 1.2.1 Les Equipements de La Maquette

Les équipements de la station sont choisis en fonction des paramètres de réglage, de leur disponibilité sur le marché et de leur coût.

### 1.2.1.1. Les Equipements d'Installation

La figure 1.2 illustre les équipements d'installation suivants :

- 1 **Support métallique** : Le prototype repose sur un support métallique 1.50 m sur 1m et d'une hauteur de 1.50m.
- 2 **Réservoir** : Réservoir de 50cm sur 20cm sur une hauteur de 0.20m qui sert à réserver de l'eau, et fait partie du procédé.
- 3 **Cuve A** : Cuve A de 15cm sur une hauteur de 30cm.
- 4 **Cuve B** : Cuve B de 15cm sur une hauteur de 25cm.
- 5 **Robinet** : Deux robinets d'arrêt à opercule DN40, assurent la flexibilité du prototype pour avoir plusieurs choix de différentes manipulations.
- 6 **Eléments de tuyauteries** : en PEHD (Polyéthylène Haute Densité) Tigre de diamètre DN40.

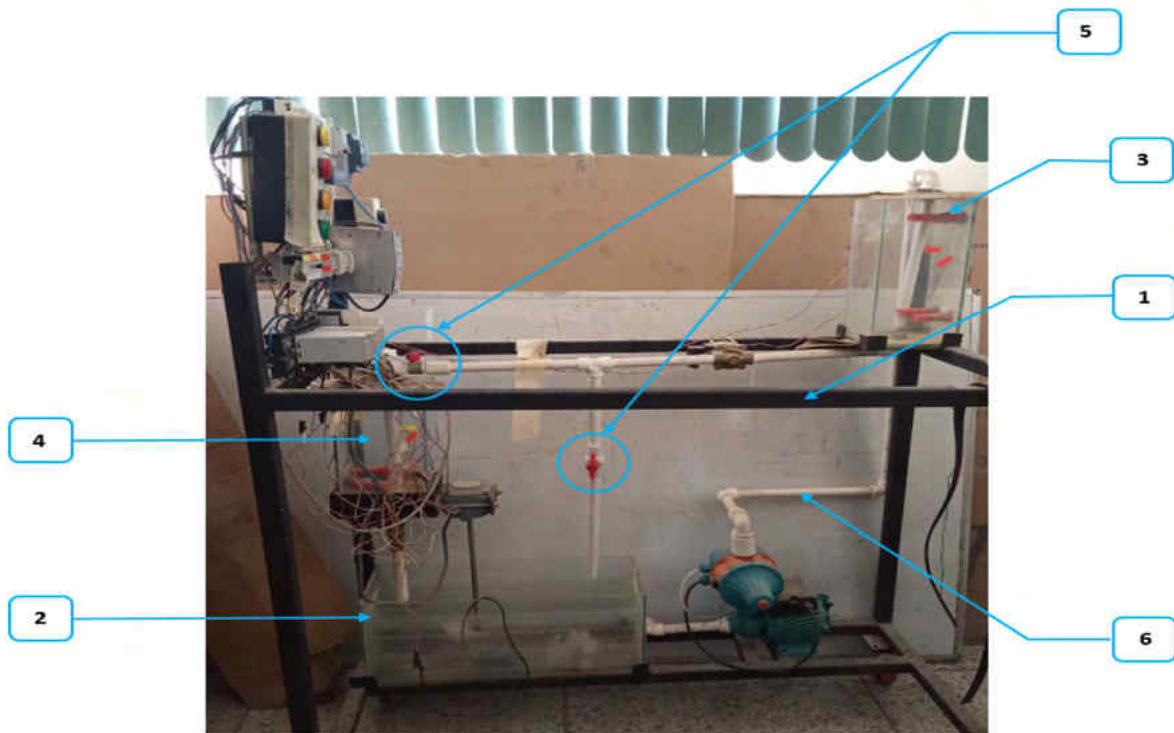


FIGURE 1.2 – Equipements de l'installation

### 1.2.1.2. Les Equipements Electriques

#### a. Dispositifs de commande et de protection(Figure 1.3)

- 1 **Automate LOGO** : Automate LOGO! (0BA8) Siemens, pour l'exécution des cahiers des charges et le contrôle de la station.
- 2 **Automate Zelio** : Automate Zelio de Schneider Electric (SR2 B201BD),pour l'exécution des cahiers des charges et le contrôle de la station.
- 3 **Afficheur TDE** : C'est un afficheur LCD de la famille Siemens, qui se raccorde avec le module LOGO!.

- 4 **Arduino** : Une carte de développement qui réalise des calculs pour mesurer le niveau d'eau.
- 5 **2 Disjoncteurs Magnétothermique HARVEST** : Un disjoncteur de 220V pour assurer la sécurité de toute l'installation, et un de 24V pour la sécurité des équipements de 24V.
- 6 **Relais Electromécanique** : Deux relais électromagnétiques 8 broches servant comme interface de sortie entre l'automate et les actionneurs (pompe et l'agitateur).
- 7 **Relais de Niveau** : Quatre relais de niveau JYB-714C alimenté en 220V et muni chacune de 2 bougies de sonde pour détecter les niveaux haut et les niveaux bas des réservoirs A et B.
- 8 **Alimentation Redressé Stabilisée** : Une alimentation stabilisée Harvest electric HAS-120-24 alimenté en monophasé 220 V et qui délivre du 24V.
- 9 **Démarrateur Progressif** : ATS01N03FT SCHNEIDER qui améliore les performances de démarrage du mélangeur en permettant un démarrage progressif et contrôlé.
- 10 **Boutonnerie** : Deux boutons poussoir verts normalement fermés **Reset** et **Marche** .
- 11 **Signalisation** : Quatre voyant (jaune, vert, orange et rouge).
- 12 **18 Borniers électriques** : Ils sont utilisés pour faciliter les connexions électriques dans les systèmes et les équipements électriques.
- 13 **Capteurs Ultrasons** : Deux capteurs ultrason hc-05 sont utilisés pour la mesure du niveau.
- 14 **Circuit convertisseur** : Deux circuits qui reçoit un signal PWM de la carte arduino et délivre un signal analogique vers les automates.
- 15 **Répartiteurs de tension** : Deux répartiteurs de tension pour le 24 V et 220 V.



FIGURE 1.3 – Equipements Electriques

## b. Les Actionneurs(Figure 1.4)

- 1 **Pompe** :Une pompe centrifuge (QB50) TAIFU, couplé à un moteur asynchrone monophasé alimenté à 220 V La hauteur manométrique maximale de cette pompe est de 20 m(Plaque signalétique).
- 2 **Agitateur** :Un mélangeur(Agitateur) OBIMEX alimenté en 220 V pour agiter le liquide.
- 3 **Electrovannes** :Deux électrovannes HARVEST (ZSC-1D) DN40 TOR alimentées en 24V DC, qui servent à régler l'écoulement d'eau.



FIGURE 1.4 – Les Actionneurs

### 1.2.2 Schémas de Raccordement

Le principe de câblage de la station de Pompage est montré dans la figure ci-dessous :

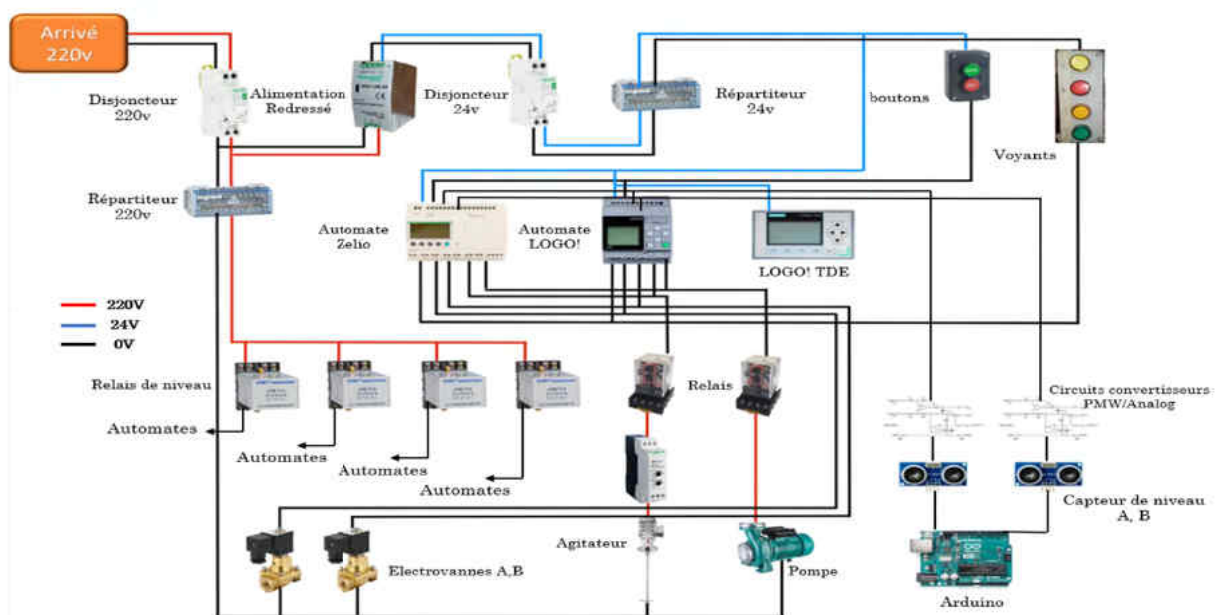


FIGURE 1.5 – Principe de câblage de station de pompage

### 1.2.3 Représentation Synoptique de la Station

La représentation synoptique de la station de pompage vise à rendre la compréhension de son principe de fonctionnement facile et accessible à tous. Elle permet de visualiser l'ensemble des éléments constitutifs de la station de pompage d'une manière claire et concise.

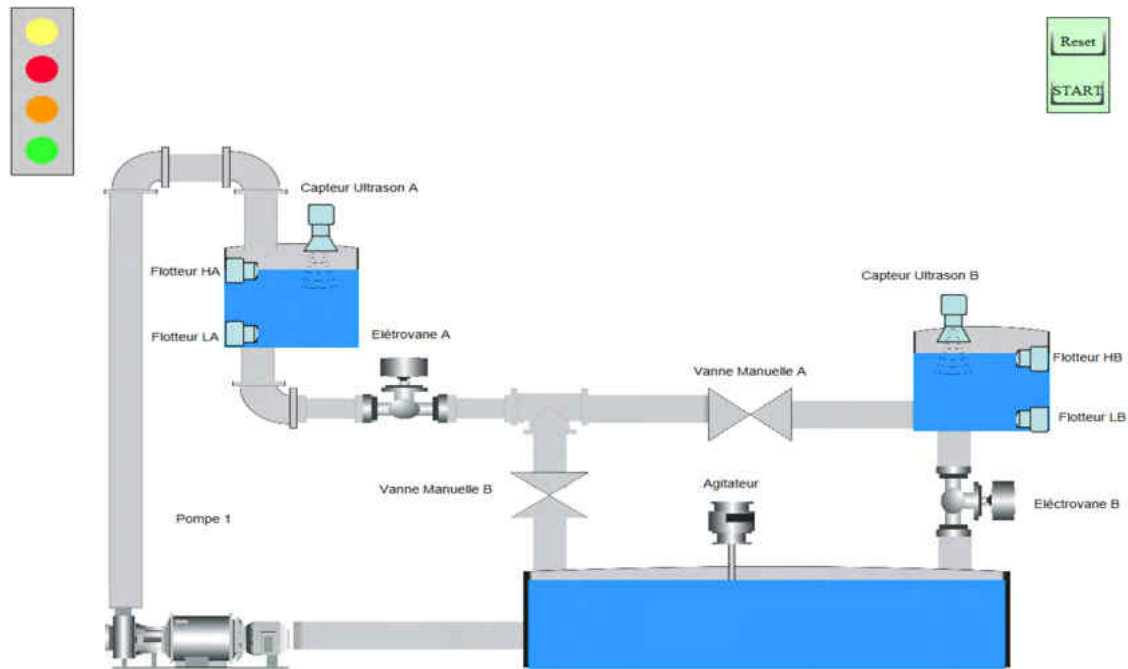


FIGURE 1.6 – Représentation Synoptique de la Station

## 1.3 Cahiers des Charges et Graficets

Ce projet de fin d'études a pour objectif de réaliser les travaux suivants :

- Commande et supervision avec les deux automates Zelio et LOGO!
- Compréhension des niveaux des systèmes de contrôle industriel.
- Application du Guide d'Étude de Mode de marche et d'arrêt GEMMA.
- Réalisation d'un système DCS.
- Proposer des solutions pour le développement de la station au niveau du câblage.

Pour cela, deux cahiers des charges sont proposés, chacun ayant un objectif spécifique.

### 1.3.1 1er Cahier de Charge

L'objectif de cette manipulation est de se familiariser avec le système, d'apprendre la manipulation des variables TOR, l'utilisation des compteurs, des timers et de développer différentes techniques de commande et de supervision en utilisant les deux automates LOGO! et Zelio Logic.

Dans cette partie, nous allons réaliser la manipulation avec l'automate Zelio Logic, puis avec l'automate LOGO!.

#### 1.3.1.1 Énoncé du cahier des charges

Lorsque la station est à l'arrêt, en appuyant sur le bouton "marche" on aura quatre scénarios possibles.

- Si le niveau haut de la cuve A n'est pas atteint, la pompe démarre pour ramener l'eau à la cuve A jusqu'à ce que le niveau haut de la cuve A "Ha" soit atteint.
- Si le niveau haut de la cuve B n'est pas atteint, l'électrovanne EVA s'ouvre pour déverser l'eau de la cuve A dans la cuve B jusqu'à ce que le niveau haut de la cuve B "Hb" soit atteint.
- Si les niveaux Ha et Hb ne sont pas atteints, l'électrovanne EVA s'ouvre et la pompe démarre pour ramener l'eau aux deux cuves. Lorsque le niveau haut de la cuve B est atteint, l'électrovanne EVA se ferme et la pompe reste toujours fonctionnel pour ramener l'eau à la cuve A jusqu'à ce que le niveau haut de la cuve A soit atteint.
- Si les deux niveaux sont atteints, l'électrovanne EVB s'ouvre pour déverser l'eau de la cuve B dans le réservoir. Une fois la cuve B vide, l'agitateur démarre pour une durée de 10 secondes. Ce cycle se répète 3 fois avant que le processus ne s'arrête.

**NB :**

- Il est à noter qu'avant de démarrer le processus, la vanne manuelle A doit être ouverte et la vanne manuelle B doit être fermée.
- Si le niveau d'eau de la cuve A est supérieur à 28 cm et/ou le niveau d'eau de la cuve B est supérieur à 23 cm, le système reviendra automatiquement à son état initial.

### 1.3.1.2. Grafcet de la Manipulation

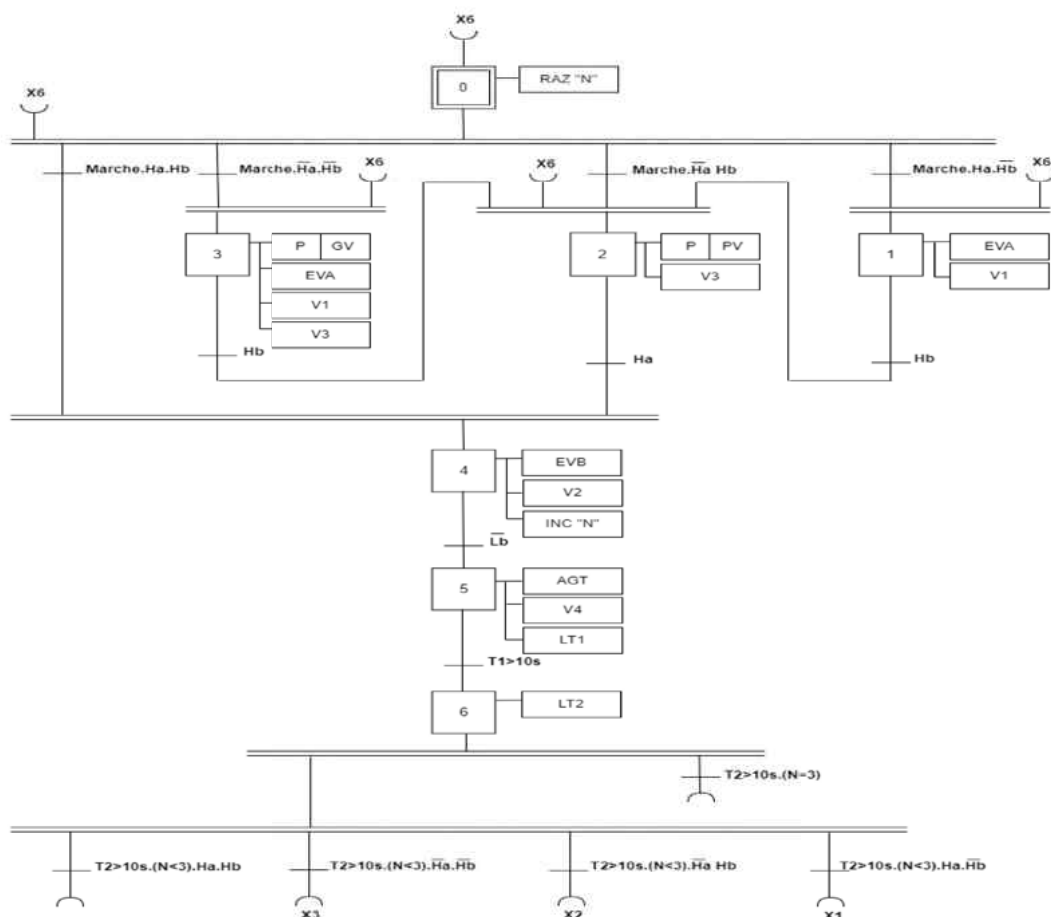


FIGURE 1.7 – Grafcet de Manipulation 1

### 1.3.2 2ème Cahier de Charge

Le cahier des charges a pour objectif de commander et superviser la station de pompage en utilisant simultanément les deux automates : Zelio Logic et LOGO!, chacun étant chargé d'effectuer une tâche spécifique clairement définie.

En outre, il vise à familiariser l'équipe de projet avec les trois types de Grafcet souvent utilisés dans les automatisations complexes et haut de gamme : le Grafcet de Sécurité (GS), le Grafcet de Conduite (GC) et le Grafcet de Fonctionnement Normal (GFN). La maîtrise de ces types de Grafcet sera essentielle pour la réalisation du GEMMA de notre station.

#### 1.3.2.1 Énoncé du cahier des charges

En appuyant sur le bouton de démarrage, la station de pompage démarrera le processus de remplissage des cuves A et B en eau. Une fois que les niveaux Ha et Hb sont atteints, le système passera à l'opération de déversement de l'eau de la cuve B dans le réservoir. Une fois que le niveau Lb est atteint, l'agitateur sera activé et fonctionnera pour une durée t1, qui sera définie par l'utilisateur via l'interface homme-machine (IHM). Une fois que l'agitateur est désactivé, le système reviendra à son état initial.

En cas de niveau d'eau de la cuve A supérieur à 28 cm et/ou le niveau d'eau de la cuve B supérieur à 23 cm, une alerte sera déclenchée et le système reviendra automatiquement à son état initial après une durée t2.

Avant de démarrer notre système, un test de marche sera appliqué pour vérifier que tous les actionneurs fonctionnent en cliquant sur la touche Z1 du Zelio Logic. Lorsque Z1 est maintenue, les 4 actionneurs s'activeront

#### 1.3.2.2 Grafcet de Sécurité GS

Le Grafcet de Sécurité (GS) est spécialement conçu pour prendre des mesures de sécurité immédiates afin de protéger les personnes et les équipements en cas d'urgence. Il permet de gérer efficacement les arrêts d'urgence, les défauts et les situations dangereuses. Le grafcet de sécurité de notre station est illustré ci-dessous :

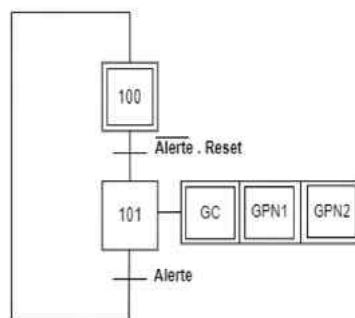


FIGURE 1.8 – Grafcet de Sécurité

#### 1.3.2.3. Grafcet de Conduite GC

Le Grafcet de Conduite (GC) est utilisé pour faciliter la compréhension du fonctionnement d'un système automatisé en fournissant une description sous forme de séquences d'étapes et de transitions

entre ces étapes. Les étapes correspondent aux différentes phases du processus, tandis que les transitions déterminent les conditions nécessaires pour passer d'une étape à une autre.

Dans le cas de notre station de pompage, le Grafcet de Conduite (GC) comprend deux phases du processus. La phase 1 est réservée à la tâche qui sera réalisée avec LOGO!, tandis que la deuxième phase est réservée à la tâche qui sera réalisée avec Zelio Logic.

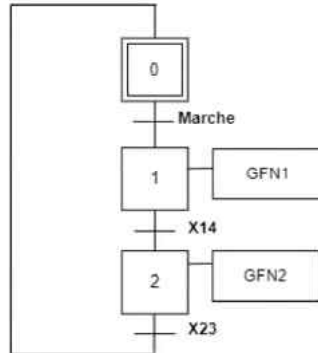


FIGURE 1.9 – Grafcet de Conduite

#### 1.3.2.4. Grafcet de Fonctionnement Normal GFN

Le Grafcet de Fonctionnement Normal est un outil graphique qui permet de représenter le comportement d'un système automatisé en conditions normales de fonctionnement. Dans le cas de la station de pompage, deux Grafcets de Fonctionnement Normal sont nécessaires : l'un pour l'automate LOGO! **GFN1** et l'autre pour l'automate Zelio Logic **GFN2**.

Chaque Grafcet de Fonctionnement Normal décrit les différentes étapes et transitions que la station de pompage traverse en conditions normales. Ces étapes comprennent le démarrage de la pompe, la vérification des niveaux d'eau, le pompage de l'eau de la cuve A vers la cuve B et l'arrêt de la pompe lorsque le niveau d'eau de la cuve B atteint un certain seuil, l'activation de l'agitateur, l'ouverture et la fermeture des électrovannes. Grâce à ces Grafcets, les automates peuvent être programmés pour effectuer les tâches nécessaires en temps voulu et garantir le fonctionnement efficace de la station de pompage.

##### a. Grafcet de Fonctionnement Normal 1 GFN1

Le GFN1 représente la tâche assurée par l'automate LOGO! qui est responsable du processus de remplissage des cuves A et B. Lorsque l'étape  $X_1$  est activé, trois scénarios sont possibles :

1. Si les niveaux  $H_a$  et  $H_b$  ne sont pas atteints, l'électrovanne EVA s'ouvre et la pompe démarre pour ramener l'eau aux deux cuves. Lorsque le niveau haut de la cuve B est atteint, l'électrovanne EVA se ferme et la pompe reste toujours fonctionnelle pour ramener l'eau à la cuve A jusqu'à ce que le niveau haut de la cuve A soit atteint. À ce stade, l'étape  $X_2$  du GC s'active grâce à l'étape  $X_{14}$  du GFN1 et le processus revient à son état initial.
2. Si le niveau haut de la cuve B n'est pas atteint, l'électrovanne EVA s'ouvre pour déverser l'eau de la cuve A dans la cuve B jusqu'à ce que le niveau haut de la cuve B " $H_b$ " soit atteint.



3. Si le niveau haut de la cuve A n'est pas atteint, la pompe démarre pour ramener l'eau à la cuve A jusqu'à ce que le niveau haut de la cuve A "Ha" soit atteint, l'étape  $X_2$  du GC s'active grâce à l'étape  $X_{14}$  du GFN1 et le processus revient à son état initial.

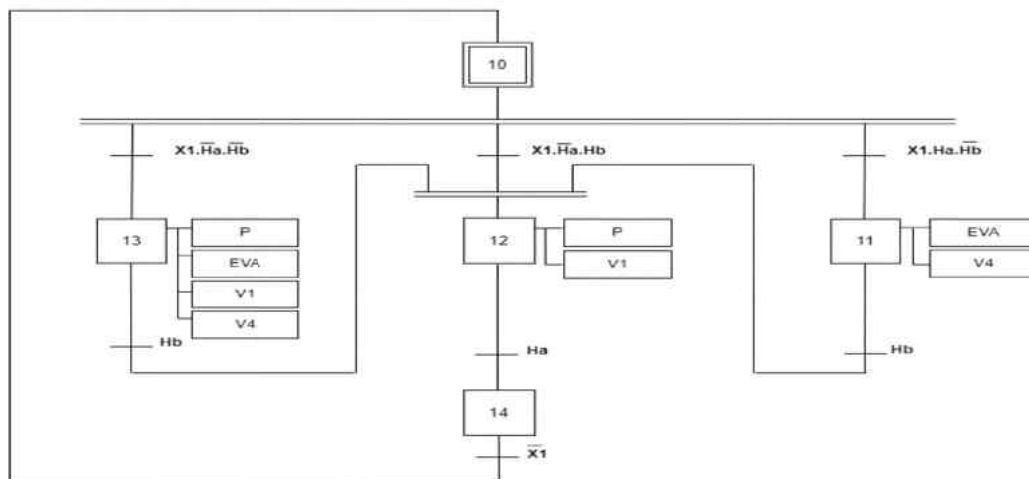


FIGURE 1.10 – Grafcet de Fonctionnement Normal 1

### b. Grafcet de Fonctionnement Normal 2 GFN2

Le GFN2 représente la tâche assurée par l'automate Zelio Logic qui est responsable du processus de diversification de l'eau dans le réservoir et de l'activation de l'agitateur. Lorsque l'étape  $X_2$  est active, l'électrovanne EVB s'ouvre pour diverser l'eau dans le réservoir. Lorsque le niveau Lb est atteint, l'agitateur démarre pour une durée  $t_1$  qui sera définie par l'utilisateur. Une fois que l'agitateur est désactivé, le processus revient à son état initial après une durée  $t_2$ .

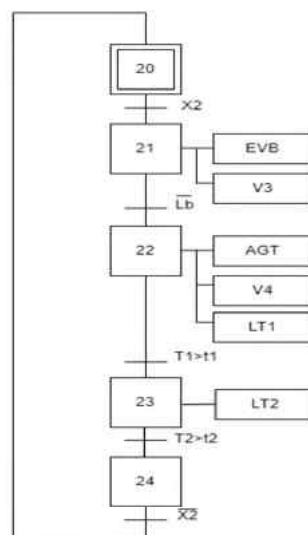


FIGURE 1.11 – Grafcet de Fonctionnement Normal 2

## 1.4 Conclusion

En conclusion, ce chapitre a permis de présenter la station de pompage qui fera l'objet de l'étude de ce projet de fin d'études. Nous avons pu décrire en bref les équipements qui composent la station, ainsi que leur principe de raccordement. Ce chapitre a permis également de présenter le cahier des charges qui servira de guide pour les travaux à réaliser dans la suite.

# Chapitre 2

## Commande et Supervision de la Station de Pompage avec Automate Schneider

### 2.1 Introduction

L'objectif principal de ce chapitre est de réaliser le contrôle et la supervision de la station de pompage en utilisant l'automate Zelio Logic. Pour cela, nous commencerons par présenter le module Zelio Logic, en décrivant ses fonctionnalités clés et en expliquant ses méthodes de programmation associées. Cette présentation permettra aux utilisateurs de se familiariser avec le module, de comprendre comment l'utiliser de manière optimale et de leur offrir une diversité de choix en matière de programmation. De plus, nous aborderons une technique de commande et de supervision en utilisant le module Zelio Logic.

### 2.2 Module Programmable Zelio Logic

Le module Zelio Logic utilisé pour la manipulation de la maquette est celui de la référence **SR2 B201BD**. Ce type d'automate est communément utilisé dans l'automatisation des processus industriels simples. Ce type d'automate :

- Possède un écran LCD et des boutons pour naviguer facilement dans l'automate.
- Capable de recevoir des consignes TOR ou Analogique et capable de commander les sorties TOR.
- Il comprend également des fonctionnalités de base telles que le comptage, la temporisation et les fonctions mathématiques pour la programmation d'opérations logiques complexes.
- Si nécessaire, il peut être étendu avec un module d'extension de communication **SR2 COM01**.
- Sa programmation est simple et accessible, avec deux langages de programmation disponibles : le LADDER, le Bloc Fonctionnel (FBD) et la language SFC.



FIGURE 2.1 – Automate Zelio Logic SR2 B201BD

## 2.3 La Programmation du Zelio Logic

Pour la programmation de Zelio SR2 B201BD il y a deux méthodes, la première méthode est la programmation à travers son écran **LCD** et la 2ème méthode est la programmation PC à l'aide de logiciel **Zelio Soft**.

### 2.3.1 La Programmation sur L'Écran LCD

La programmation du Zelio SR2 B201BD sur son écran LCD peut être pratique pour les utilisateurs qui n'ont pas accès à un ordinateur. Cette méthode de programmation utilise exclusivement le langage LADDER et nécessite la navigation à travers différents menus et paramètres en utilisant les boutons du dispositif pour saisir les valeurs requises.

1. Bouton **Shift** Permet d'afficher le menu contextuel au-dessus des boutons Z.



2. Bouton **Menu/OK** permet de confirmer un menu, un sous-menu, un programme, un paramètre, etc.
3. Les boutons **Z** (de navigation) permettent de se diriger vers la gauche, le bas, le haut ou la droite.
4. **Menu de Navigation** permet d'accéder aux différents paramètres du module.

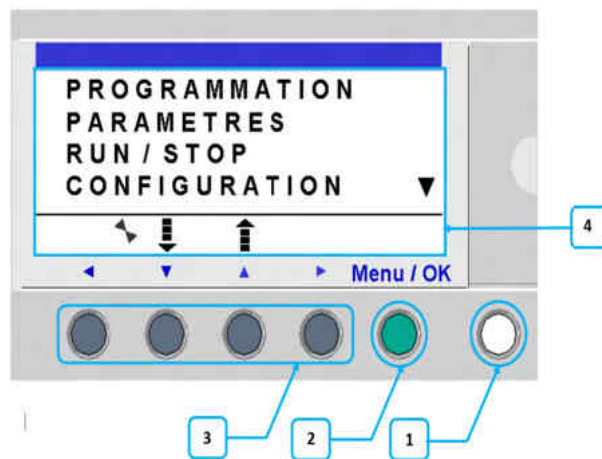
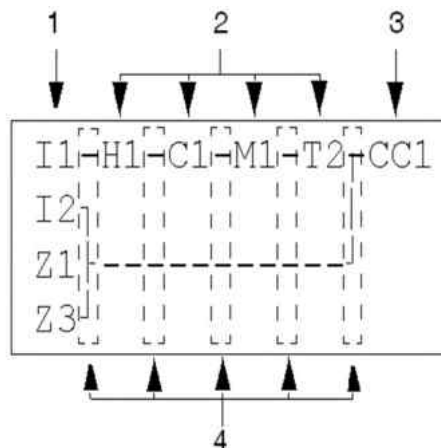


FIGURE 2.2 – Les boutons de Navigation

La programmation sur l'écran LCD permet la saisie de 120 lignes de schéma de commande et l'affichage de ces lignes se fait par groupes de 4. Comme illustré ci-dessous :



1. Colonne réservée aux contacts (conditions).
2. Colonne réservée aux contacts (conditions) ou aux liaisons.
3. Colonne réservée aux bobines (actions).
4. Colonne réservée aux liaisons.[7]

La saisie de ces éléments (contacts, bobines et liaisons) s'effectue selon les étapes suivantes : [7]

### 2.3.1.1 La Saisie d'un Contact ou une Bobine

Les étapes pour saisir un contact ou une bobine sont :

1. Positionnez le curseur clignotant à l'endroit souhaité, les boutons Z permettent de déplacer le curseur dans le sens des flèches de navigation.



2. Appuyez sur le bouton **Shift** pour afficher la sélection du menu contextuel.



Si on appuie simultanément sur **Shift** et sur l'une des touches (- et +), la première lettre de l'élément est insérée : I pour un contact et Q pour une bobine, suivie du numéro 1.

3. Pour choisir l'élément souhaité en appuyant simultanément sur **Shift** et + ou -, les différents éléments possibles que nous pouvons choisir sont :
  - Contact Normalement Ouvert : I,Z,N,M,Q,T,C,K,V,A,H,W,S.
  - Contact Normalement Fermé : i,z,n,m,q,t,c,k,v,a,h,w,s.
  - Les Bobines : I,Z,N,M,Q,T,C,K,V,A,H,W,S.[7]
4. Relâchez le bouton **Shift** pour accéder aux touches de navigation. Appuyez sur le bouton correspond a la direction droite → pour placer le curseur sur le chiffre 1 correspondant.
5. Appuyez simultanément sur le bouton **Shift** et + pour incrémenter le numéro de l'élément (2, 3, 4 à 9, A, etc.).
6. Relâchez le bouton **Shift** pour accéder aux boutons Z de navigation.[7]

### 2.3.1.2 La Saisie d'une Liaison

Les étapes pour la saisie d'une liaison :

1. Positionnez le curseur clignotant à l'endroit souhaité, les boutons Z permettent de déplacer le curseur dans le sens des flèches de navigation.



2. Appuyez sur le bouton **Shift** pour afficher la sélection du menu contextuel.



3. Pour tracer les connexions, on appuie simultanément sur **Shift** et une des touches ←↑↓→.
  - Shift et la touche → pour tracer une connexion jusqu'à la position du contact suivant ou jusqu'à la bobine en fin de ligne.
  - Shift et l'une des touches ↑ ↓ pour tracer des connexions perpendiculaires à la ligne précédente ou suivante.

4. On relâche le bouton **Shift** pour accéder aux boutons Z de navigation.[7]



FIGURE 2.3 – Exemple de Saisie des Eléments

**N.B :**

- Avant de programmer le module Zelio, nous devons le mettre en mode Stop en sélectionnant le menu **RUN/STOP**.
- Pour programmer le module Zelio, vous devez accéder au menu **PROGRAMMATION**. [7]

### 2.3.2 La Programmation avec le Logiciel Zelio Soft

La deuxième méthode de programmation est la programmation à l'aide d'un logiciel appelé **Zelio Soft**, qui est installé sur un ordinateur. Ce logiciel est conçu pour programmer tous les programmes des modules Zelio Logic. La programmation sur Zelio Soft présente certains avantages par rapport à la programmation sur LCD :

- Possibilité de choisir le langage de programmation ,LADDER, FBD ou SFC, pour une plus grande flexibilité.
- Affichage clair et organisé des données du programme et des paramètres pour une meilleure compréhension et une gestion plus facile.
- Possibilité de charger et de décharger rapidement et facilement des programmes, permettant ainsi de gagner du temps et de l'efficacité.
- Fonctionnalités de simulation, de monitoring et de supervision très utiles pour tester et surveiller le programme en cours d'exécution.
- Possibilité d'imprimer la documentation de l'application pour faciliter la communication et le partage d'informations avec d'autres membres de l'équipe.
- Assure les mêmes fonctions de temporisation et de comptage/decompage... similaires à celles de la programmation sur LCD.
- Mise en œuvre plus rapide et efficace, permettant d'économiser du temps et de l'argent.

#### 2.3.2.1 Création d'un Nouveau Projet Sur Zelio Soft

Afin de créer un nouveau projet sur Zelio Soft, nous devons suivre les étapes suivantes :

1. Ouvrir le logiciel Zelio Soft et choisir "Créer un nouveau programme".

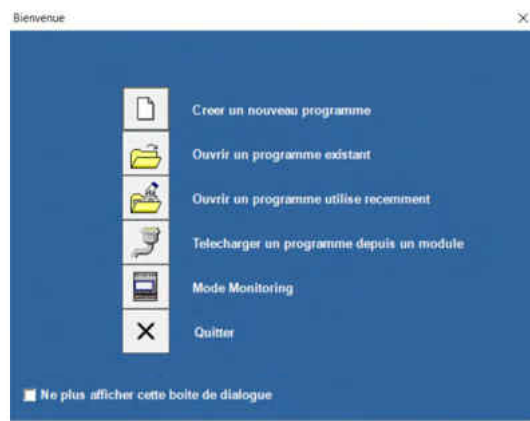


FIGURE 2.4 – Etape 1

2. Choisir le module adapté ,dans notre cas c'est **SR2 B201BD**, et cliquer sur **Suivant**.

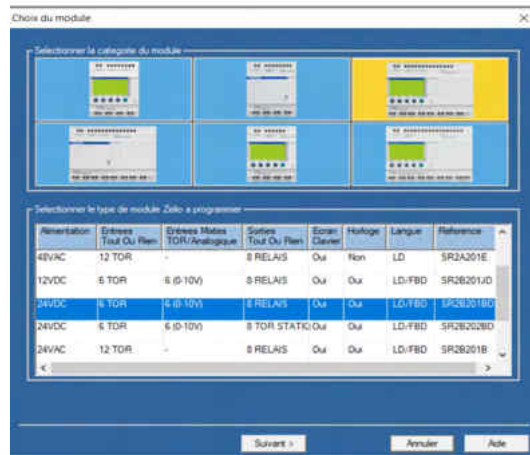


FIGURE 2.5 – Etape 2

3. Dans cette étape on peut ajouter une interface de communication **SR2 COM01** si n'essai en cliquant sur **Ajouter** puis sinon en passe a la prochaine etape en cliquant sur **Suivant**.



FIGURE 2.6 – Etape 3

4. Choisir le langage de programmation souhaité,**LADDER** ou **FBD** et cliquer sur **Suivant**.



FIGURE 2.7 – Etape 4

## 2.3.2.2 Langages de Programmation

### 2.3.2.2.1 Langage LADDER

La programmation en langage LADDER sur Zelio Soft présente les mêmes éléments (contacts, bobines et liaisons) que la programmation sur l'écran LCD.

L'interface de programmation comporte au maximum 240 lignes de code sans le module de communication SR2 COM01, et 120 lignes de code avec le module de communication SR2 COM01 sélectionné dans la configuration.

Chaque ligne peut contenir jusqu'à 5 contacts (zone de test) et une bobine (zone d'action). Les entrées, les sorties, les blocs fonctionnels, etc. sont disposés en bas de l'interface. Pour les utiliser, il suffit de glisser l'élément souhaité sur la zone de test ou d'action. Pour illustrer l'interface de programmation en langage ladder avec Zelio Soft, voici un exemple avec le module **SR2 B201BD** présenté dans la figure ci-dessous :

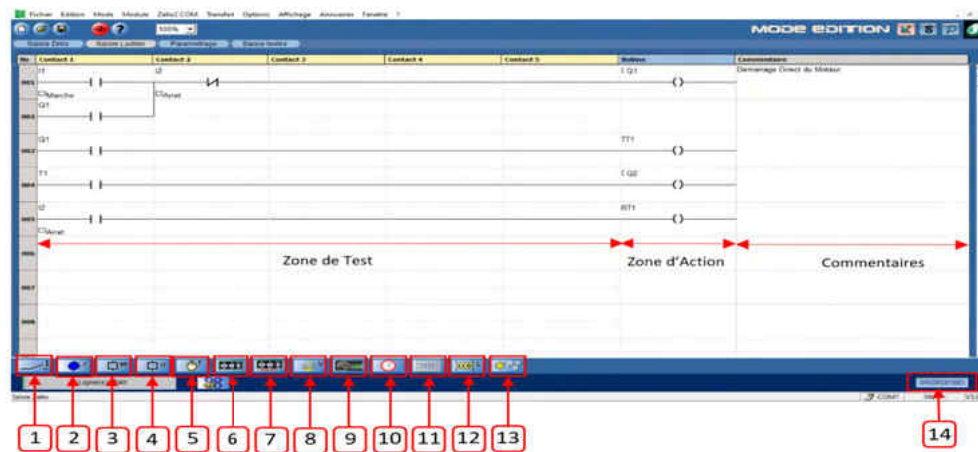


FIGURE 2.8 – Interface de programmation avec Langage LAADER

1. Les Entrées	8. Comparateurs de Compteurs
2. Les Boutons de Navigation Z	9. Comparateurs Analogiques
3. Memonto	10. Horloge
4. Sorties	11. Blocs Textes
5. Temporisateurs	12. Rétroéclairage LCD
6. Compteurs	13. Météo
7. Compteur Rapide	14. Référence du Module

### 2.3.2.2.2 Langage FBD

Le langage FBD repose sur des blocs fonctionnels prédéfinis, qui offrent une large gamme de fonctionnalités, comme illustré dans la figure ci-dessous :

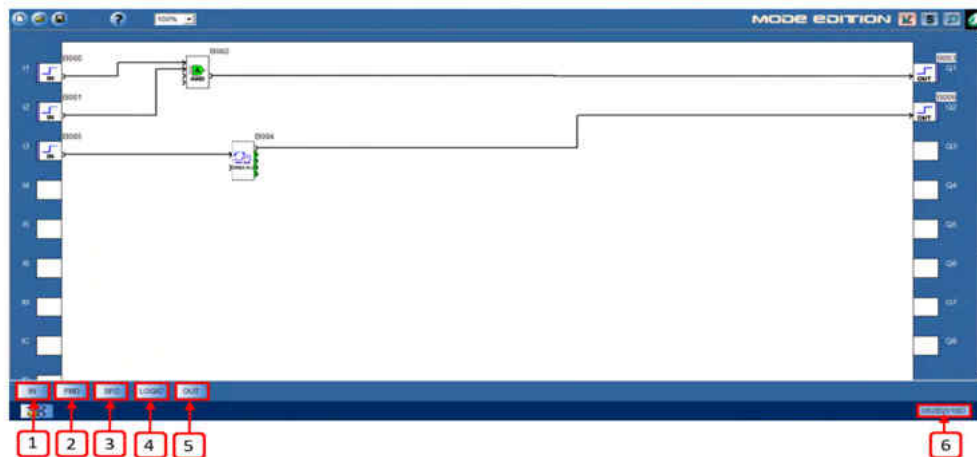


FIGURE 2.9 – Interface de programmation avec Langage FBD

1. Les Entrées	4. Portes Logiques
2. Les Blocs Fonctionnels : temporisateurs, compteurs, comparateurs..	5. Les Sorties
3. Fonctions SFC	6. Référence du Module

### 2.3.2.3 Simulation et Test du Programme

Pour simuler et tester le programme

1. On clique sur Simulation "S".
2. On clique Sur **Run**.
3. Pour arreter la simulation on clique sur **Stop**.

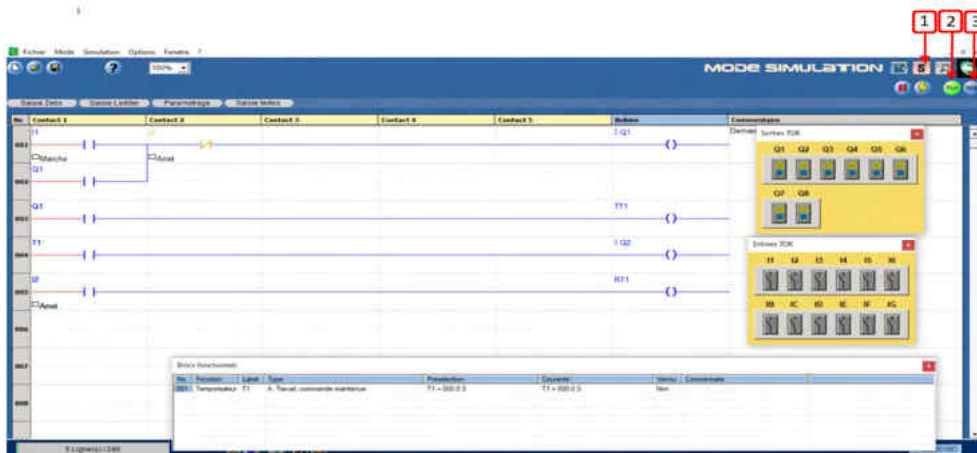


FIGURE 2.10 – Les Etapes de Simulations



## 2.3.2.4 Transfert du Programme

Il y'a deux types de transfert :

### 1.3.2.4.1 Transfert du Programme PC vers Module

Permet de transférer le programme sur Zelio Soft vers le Module Zelio logic en cliquant sur **Transfert** puis **Transférer Programme** et on choisit **PC-> Module**.

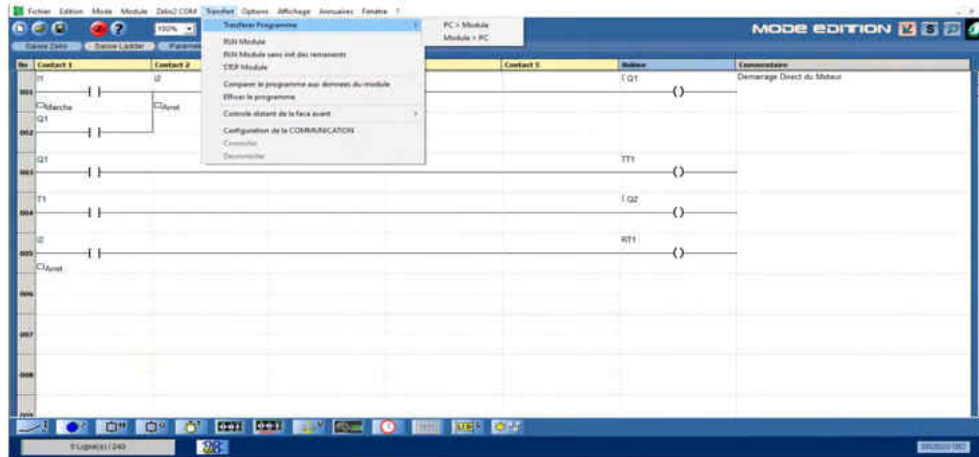


FIGURE 2.11 – Les Etapes de Transfert vers Module

Avant de transférer le programme il faut assurer que le module est en mode **STOP**, si le module connecté est en mode **RUN**, il faut donc le mettre en mode **STOP** en choisissant **Transfert** puis **STOP Module**.

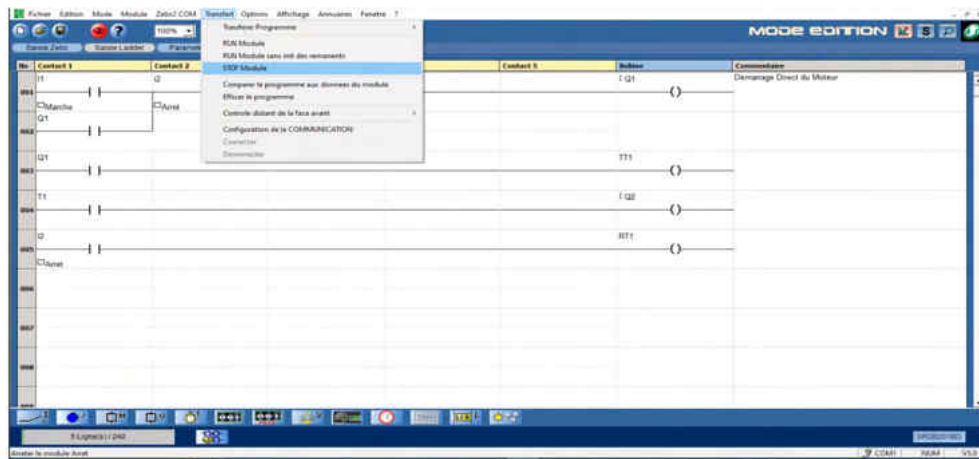


FIGURE 2.12 – Mettre le Module en Mode Stop

Si le type de module sélectionné au préalable n'est pas le même que celui qui est connecté, il faudra changer le type en cliquant sur **Module** puis choix du **module/programmation**, pour effectuer un diagnostic du module connecté il suffit de cliquer sur **Module** puis **Diagnostic du Module**.

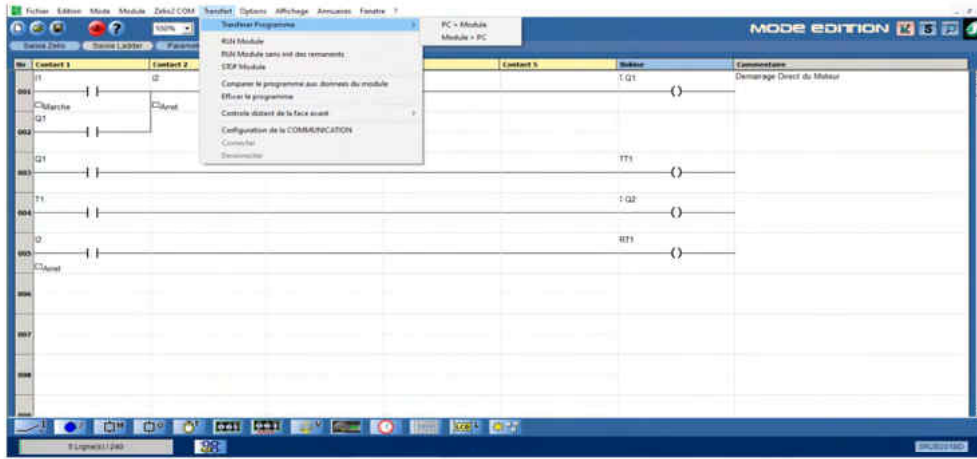


FIGURE 2.13 – Connection et diagnostic du Module

Une fois le module sélectionné et le module connecté s'accorde, la fenêtre de dialogue s'affiche comme suit :

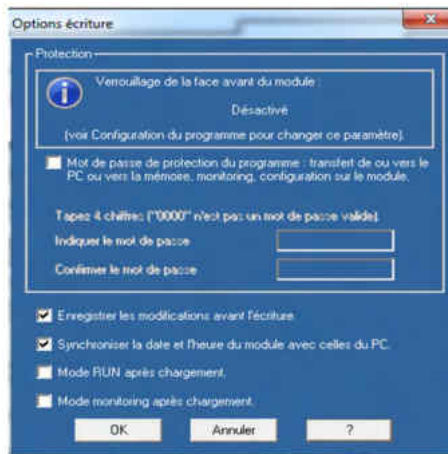


FIGURE 2.14 – Fenêtre de dialogue d'un module Zelio

Le programme présent dans le module avant le transfert est écrasé par le module. Pour mettre en marche le module à partir du logiciel, il suffit de cliquer sur **Transfert** puis **Module**. Toutefois, lorsque le transfert du programme est en cours, le module se met en mode **RUN** automatiquement.

#### 2.3.2.4.2 Transfert du Programme Module vers PC

Pour récupérer le programme existant dans le module, cliquez sur **Transfert**, puis **Module -> PC**. Le logiciel chargera alors le dernier programme présent dans le module. Si le programme du module est verrouillé, un code sera demandé par le module avant la récupération du programme.

#### 2.3.2.5 Mode de Passe de Programme

Le mot de passe sert à protéger l'accès à un programme. Lorsque le transfert du programme dans Zelio logic est en cours, la fenêtre option d'écriture s'ouvre et il suffit de cocher la case "**Mode de Passe de Protection du programme**". Une fois le mot de passe activé, il sera impossible d'écrire dans le module ni lire le programme sans connaître ce mot de passe. Le programme est ainsi protégé :

## 2.4 Commande et Supervision de la Station de Pompage avec Automate Zelio Logic

### 2.4.1 Schémas de Câblage des Entrées et Sorties

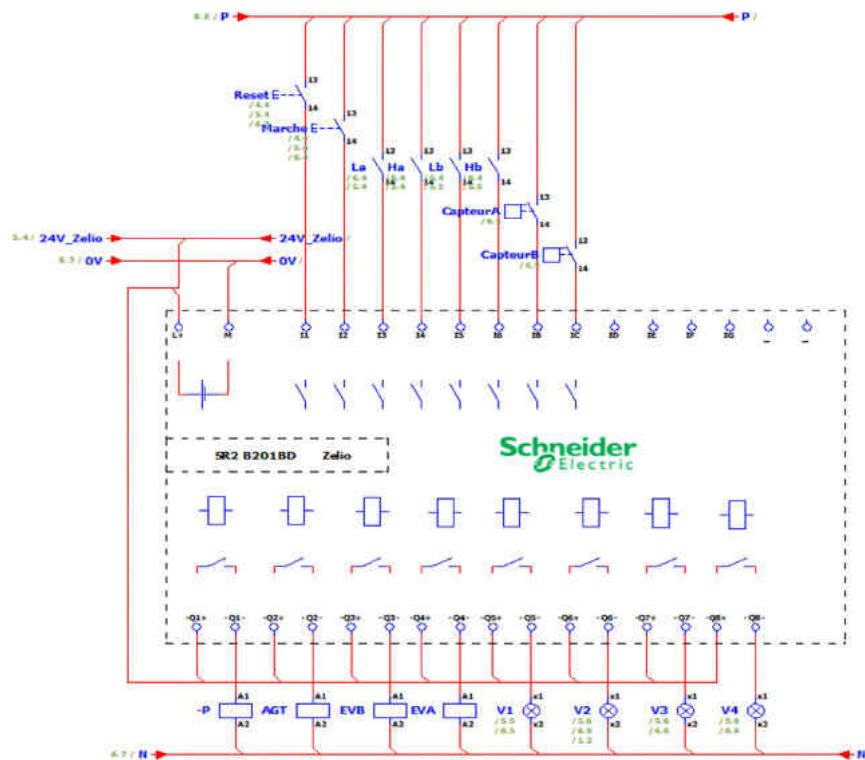


FIGURE 2.15 – Schémas de Câblage du Zelio Logic

### 2.4.2 Affectation des Variables

Les variables utilisées dans notre programme sont présentées dans le tableau suivant :

Adresses	Variables	Adresses	Variables	Adresses	Variables
<b>Les Entrées</b>		<b>Les Sorties</b>		<b>Mémentos</b>	
I1	Reset	Q1	Pompe	M1	X0
I2	Marche	Q2	Agitateur	M2	X1
I3	LA	Q3	EVB	M3	X2
I4	HA	Q4	EVA	M4	X3
I5	LB	Q5	V1	M5	X4
I6	HB	Q6	V2	M6	X5
IB	Capteur A	Q7	V3	M7	X6
IC	Capteur B	Q8	V4		

TABLE 2.1 – Table d'Affectation des Entrées et Sorties sur Zelio Logic

### 2.4.3 La Commande de la Station de Pompage avec les Touches Z

Il est possible de commander le système à travers les touches de fonction  $Z_1$ ,  $Z_2$ ,  $Z_3$  et  $Z_4$  du module Zelio logique en les utilisant comme **entrées numériques** dans le programme de commande. Elles peuvent être utilisées pour activer ou désactiver certaines fonctions du programme, pour modifier des paramètres ou encore pour déclencher des actions spécifiques.

Dans le cas de notre simulation nous pouvons les utiliser pour forcé la manipulation de système, par exemple nous pouvons considérés la touche  $Z_1$  est l'entrée La,  $Z_2$  est Ha,  $Z_3$  est l'entrée Lb et  $Z_4$  est l'entrée Hb.

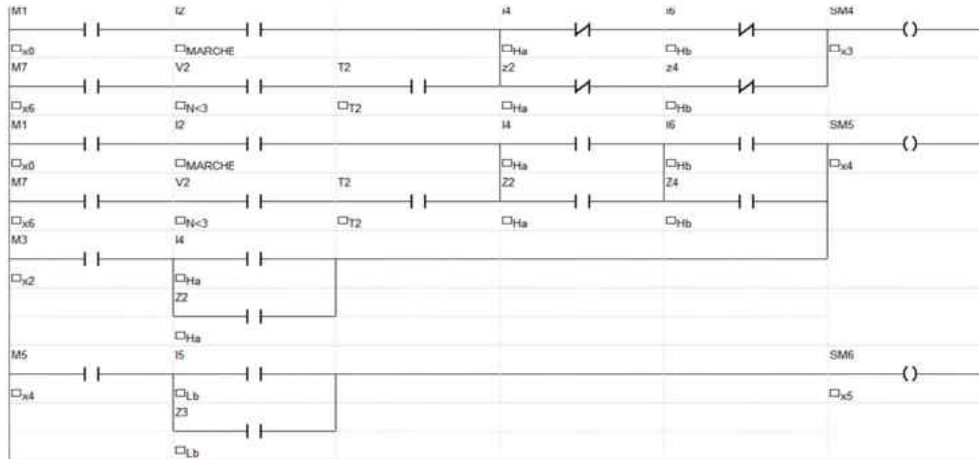


FIGURE 2.16 – Commande du Système a travers les Touches de Fonctions

### 2.4.4 La Supervision de la Station de Pompage à travers l'écran LCD

La supervision s'effectue en utilisant l'écran LCD intégré au module **Zelio Logic**, qui permet de visualiser les différents états des sorties, le langage de programmation utilisé ainsi que l'état actuel du module.

Le module Zelio Logic dispose de deux modes de fonctionnement : le mode **RUN** et le mode **STOP**. En mode RUN, le module est en fonctionnement normal, et les entrées/sorties peuvent être activées et désactivées selon les besoins de l'application. En mode STOP, le module est arrêté, et aucune entrée/sortie ne peut être activée.

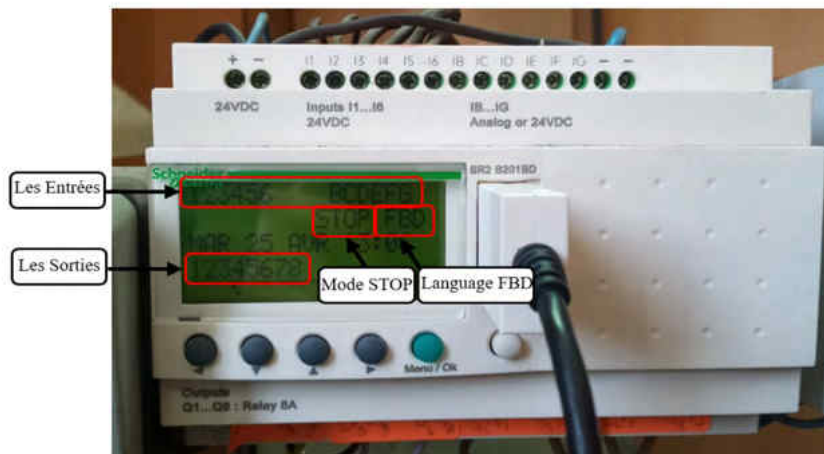


FIGURE 2.17 – La Supervision sur l'écran LCD

Lorsqu'une entrée/sortie est activée en mode **RUN**, le caractère correspondant à cette entrée/sortie devient en **Gras** sur l'écran LCD du module. Cela permet de visualiser rapidement l'état des différentes entrées/sorties et de s'assurer que les signaux sont bien transmis. La figure ci-dessous montre que l'entrée  $I_2$  et les sorties  $Q_1, Q_3, Q_6, Q_7$  sont activées :



FIGURE 2.18 – Visualisation des Etats des Sorties

Il est également possible de personnaliser l'affichage sur l'écran LCD en configurant d'autres éléments tels que les messages d'avertissement, les valeurs des timers et des compteurs, en utilisant le bloc fonctionnel "Text" **TX** comme illustrée ci dessous :

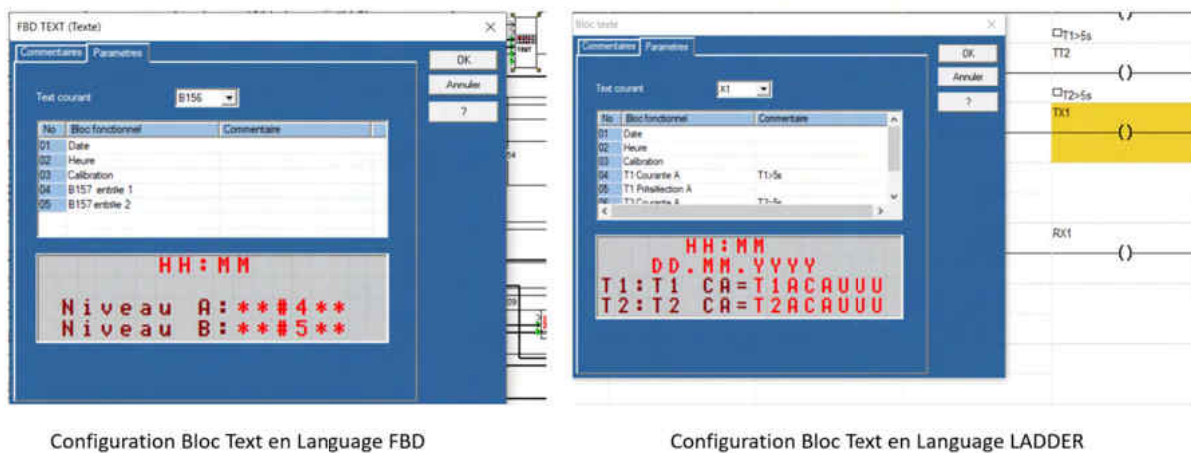


FIGURE 2.19 – Configuration des Message Textes

Il est possible de créer plusieurs vues en configurant plusieurs blocs de texte, en gérant leurs conditions d'activation et de désactivation.

Dans le cas de notre cahier des charges, nous avons créé 3 vues. La première page est la vue face de l'écran LCD qui permet de superviser l'état ON/OFF des entrées et des sorties. Cette vue est activée par les états  $X_0, X_1, X_2$  et  $X_3$ .



FIGURE 2.20 – Vue Face LCD

La deuxième vue nous permet de visualiser les valeurs des deux temporisations  $T_1$  et  $T_2$  ainsi que la valeur actuelle  $N$  du compteur. Cette vue est activée par les états  $X_4$ ,  $X_5$  et  $X_6$ .



FIGURE 2.21 – Vue 2 LCD

La troisième vue nous permet de surveiller le niveau d'eau dans les deux cuves A et B. Cette vue est activée en cliquant sur la touche  $Z_1$ .



FIGURE 2.22 – Vue 3 LCD

Le programme utilisé pour la commande et la supervision est détaillé dans l'Annexe E.

## 2.5 Conclusion

À travers ce chapitre, nous avons réussi à réaliser la commande et la supervision de la station de pompage en utilisant le module Zelio Logic. Au cours du développement de notre projet, nous avons appris les méthodes de programmation du Zelio Logic, telles que la commande du système à travers ses touches de fonctions. Enfin, nous avons détaillé comment nous pouvons réaliser la supervision à travers son écran LCD.

# Chapitre 3

## Commande et Supervision de la Station de Pompage avec Automate SIEMENS

### 3.1 Introduction

L'objectif de ce chapitre est de réaliser la commande et la supervision de la station de pompage en utilisant l'automate Siemens LOGO! et en appliquant diverses techniques. Pour cela, nous commencerons par présenter les modules LOGO! et LOGO! TDE, ainsi que leurs fonctionnalités respectives. Ensuite, nous allons exploiter l'utilisation du logiciel **Logo!Soft Comfort** qui sera utilisé pour programmer le module LOGO!. Enfin, nous examinerons les différentes techniques de commande et de supervision. Chacune de ces techniques sera appliquée au premier cahier des charges.

### 3.2 Module Programmable LOGO!

L'automate programmable industriel LOGO! fait partie de gamme de Siemens. Il est conçu pour les petites applications industrielles et offre une grande variété de fonctionnalités pour la commande de processus de production, la surveillance de température, la commande d'éclairage et bien d'autres applications.

La gamme du module LOGO qui sera utiliser dans notre manipulation est **LOGO! 0BA8**, elle comprennent les fonctionnalités suivantes :

- Un écran LCD rétroéclairé intégré pour naviguer dans les paramètres de l'automate et pour l'affichage des messages et des données.
- Des entrées/sorties intégrées pour la connexion à différents capteurs et actionneurs.
- Des fonctions de temporisation, de comptage et de mathématiques pour la programmation d'opérations logiques complexes.
- Une communication intégrée pour la connexion à des réseaux industriels.

En plus de ces fonctionnalités de base, l'automate LOGO! 0BA8 peut être étendu avec différents modules d'extension pour ajouter des fonctionnalités supplémentaires, telles que :

- Des extensions entrées/sorties supplémentaires pour la connexion à plus de capteurs et d'actionneurs.
- Des modules d'extension de communication pour la connexion à des réseaux spécifiques ou pour la communication sans fil.
- Des modules d'extension de commande spéciale : la surveillance de température, d'humidité, la commande de moteurs pas-à-pas,..ect

Ces modules d'extension permettent à l'automate LOGO! 0BA8 d'être adapté aux besoins spécifiques de l'application industrielle, tout en conservant sa simplicité et sa facilité d'utilisation.



FIGURE 3.1 – Automate LOGO! 0BA8

### 3.3 Le Module LOGO! TDE

Le module TDE est un accessoire compatible avec les modules logiques Siemens LOGO! qui fournit un écran supplémentaire plus grand que celui de LOGO!. Cet écran permet d'afficher des informations sur les processus en temps réel, de surveiller et de contrôler les processus. Le module TDE est équipé :

- D'une touche **Enter** permettant d'accéder à un menu ou de confirmer un paramètre.
- D'une touche **ESC** pour accéder aux paramètres de LOGO! TDE et quitter un menu ou des paramètres.
- Quatre touches de fonction (F1, F2, F3 et F4) et de 4 touches de curseur (C1, C2, C3 et C4) programmables. Grâce au logiciel LOGO!Soft Comfort, nous pouvons configurer ces touches programmables comme des entrées dans notre programme de commande, ce qui nous permet d'effectuer diverses actions telles que le démarrage et l'arrêt d'un processus, ainsi que la navigation dans les menus.

Le module TDE peut communiquer au module LOGO! via un câble **Ethernet**, et il est alimenté directement par le module LOGO!. Cela signifie qu'il n'a pas besoin d'une source d'alimentation supplémentaire pour fonctionner. Il est spécialement conçu pour être utilisé dans des environnements industriels où une interface simple et fiable est nécessaire pour surveiller et contrôler les processus.

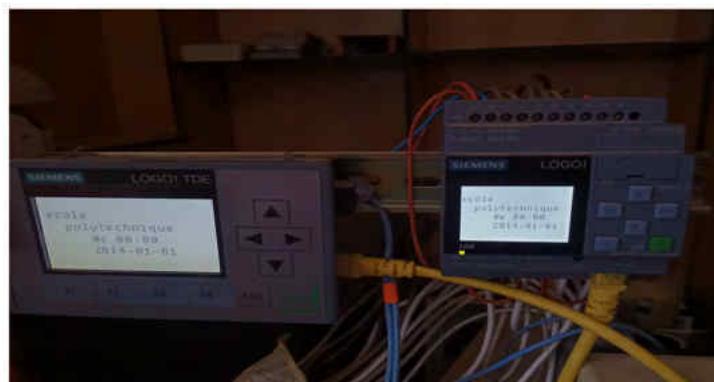


FIGURE 3.2 – Afficheur LOGO! TDE et LOGO!



## 3.4 LOGO!Soft Comfort

LOGO!Soft Comfort est un logiciel de programmation utilisé pour créer, configurer et programmer les modules logiques Siemens LOGO!. Il offre une interface qui permet aux utilisateurs de créer et de modifier facilement des programmes logiques à l'aide de diagrammes de blocs fonctionnels (FBD) ou de diagrammes en échelle (LADDER). Le logiciel comprend une bibliothèque de blocs fonctionnels pré-construits qui peuvent être utilisés pour simplifier les tâches de programmation.

Une fois qu'un programme a été créé, LOGO!Soft Comfort peut être utilisé pour simuler l'exécution du programme, permettant aux utilisateurs de vérifier son comportement avant de le télécharger sur le module LOGO!. Le logiciel comprend également des outils de diagnostic qui peuvent être utilisés pour résoudre les problèmes et déboguer les programmes.

### 3.4.1 Interface de Programmation

L'interface de programmation de Logo!Soft Comfort est conviviale et facile à utiliser. Dès que nous ouvrons le programme, un nouvel environnement de travail est automatiquement créé, cela nous permettant de commencer à programmer immédiatement.

L'interface est divisée en plusieurs sections, chacune avec sa propre fonctionnalité :

1. **La zone de travail** : C'est la section principale, où nous pouvons écrire et éditer notre code.
2. **Barre de menus** : Elle contient des options pour créer, enregistrer et charger des projets, ainsi que des options pour personnaliser le programme.
3. **La barre d'outils** : Elle contient des icônes pour des actions courantes telles que l'ouverture ou l'enregistrement d'un projet, la sauvegarde, l'impression et la compilation.
4. **Barre de sélection du mode** : Cette barre permet de choisir entre deux modes de programmation : le mode diagramme et le mode projet réseau.

Dans le mode diagramme, on peut créer des diagrammes pour représenter le code, tandis que dans le mode projet réseau, nous pouvons créer des réseaux de communication pour communiquer avec d'autres équipements.

5. **Section de diagramme** : Permet de créer des diagrammes pour représenter graphiquement le code.
6. **Section des instructions** : Cette section contient une liste d'instructions et de fonctions prédéfinies qu'on peut utiliser pour écrire le code. Pour les utiliser, il suffit de faire glisser ces instructions dans la zone de travail.

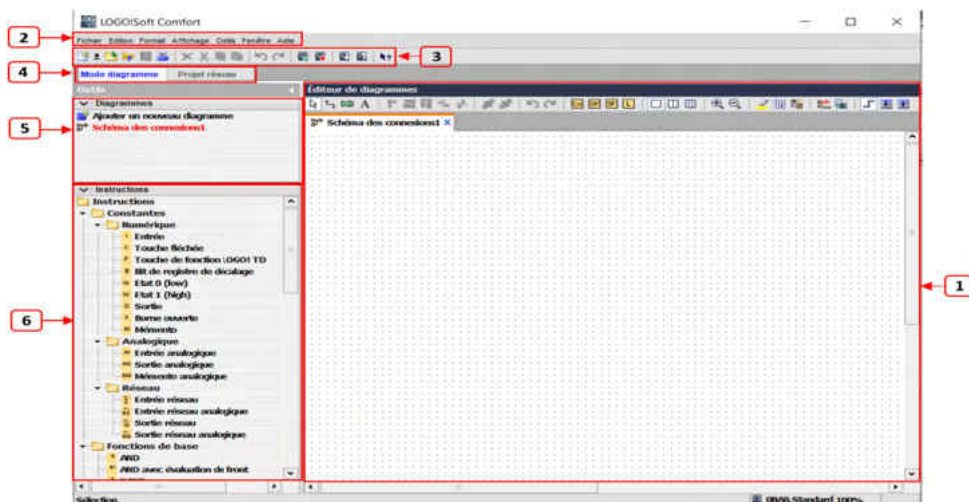


FIGURE 3.3 – Interface de Programme de LOGO!Soft Comfort

### 3.4.2 Configuration du Matériel

Afin de configurer le matériel sur LOGO!Soft Comfort nous suivons les étapes suivantes :

1. Choisir le mode **Projet réseau**.
2. Cliquer sur **Ajouter un nouveau appareil**.
3. Choisir l'automate souhaité ,dans notre cas, c'est **OBA8!** et saisissez les paramètres de connexion appropriés pour le module, tels que l'adresse IP.

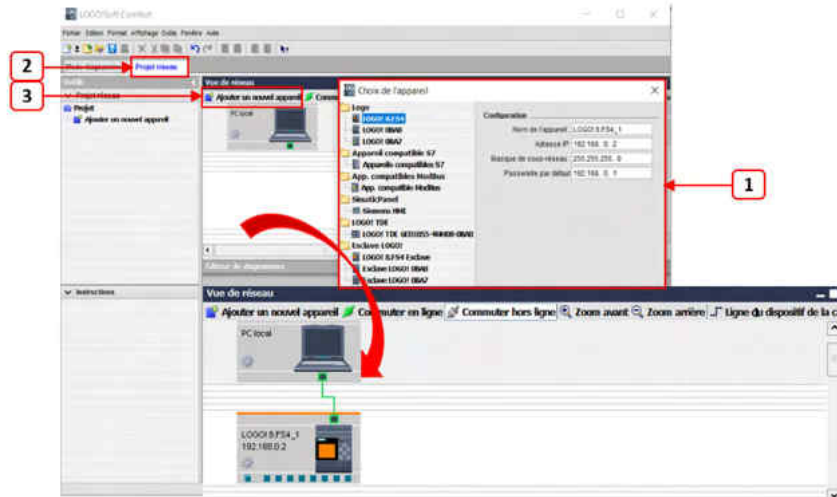


FIGURE 3.4 – La configuration du matériel de LOGO!Soft Comfort

### 3.4.3 Déclaration des Variables

Pour déclarer les variables, nous restons toujours dans le mode **Projet réseau** et nous suivons les étapes suivantes :

1. Aller vers **Paramètres**.
2. Choisir **Noms des E/S**.
3. Ecrire le nom de la variable devant l'adresse appropriée.

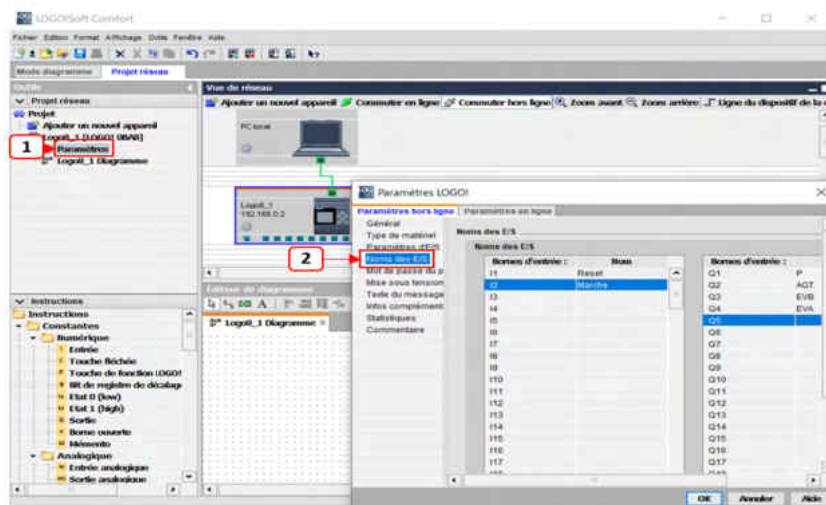



FIGURE 3.5 – Déclaration des variables sur LOGO!Soft Comfort



### 3.4.4 Simulation du Programme

Après l'enregistrement du schéma de connexion, il est possible de démarrer la simulation, en cliquant sur l'icône . En utilisant la simulation de programme sur LOGO!, il est possible de tester et de modifier les paramètres d'un programme de commande. Cette méthode permet d'optimiser le programme de commande pour s'assurer de son fonctionnement opérationnel et optimal.

### 3.4.5 Transfert du Programme

#### 3.4.5.1 Transfert du Programme Pc-> Module

Une fois que le test du programme dans la simulation LOGO! Soft Comfort est effectué, on transfère le programme vers le module suivant les étapes suivantes :

1. Cliquer sur l'icône .
2. Cliquer sur **Actualiser**  pour chercher l'appareil et sélectionner l'appareil identifié.
3. Cliquer sur **Test**.

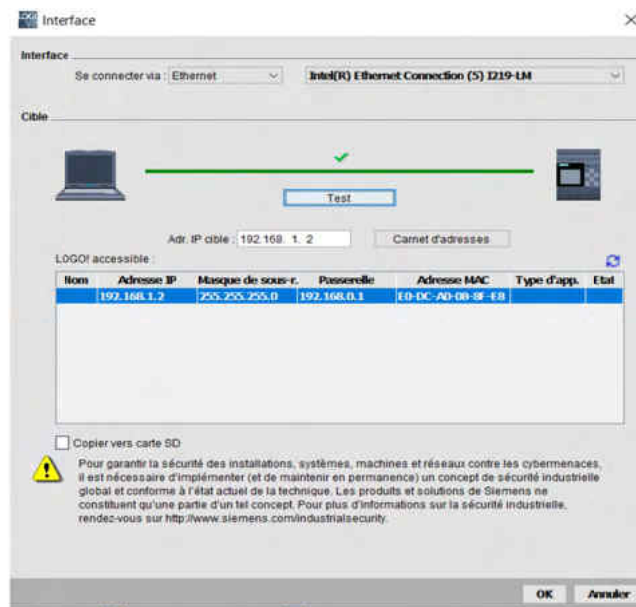




FIGURE 3.6 – Transfert du Programme Pc- > Module

#### 3.4.5.2 Transfert du Programme Module vers PC

Pour transférer le programme **Module vers PC**, on clique sur l'icône .

### 3.4.6 Test en Ligne

Pour charger le programme, nous devons procéder à un test en ligne : Le bouton de test en ligne  permet de tester le programme de commande associé au module LOGO!. Les états des entrées et sorties et des connexions logiques s'affichent.

### 3.5 Commande et Supervision de la Station de Pompage avec Automate LOGO! 0BA8

#### 3.5.1 Schémas de Câblage des Entrées et Sorties

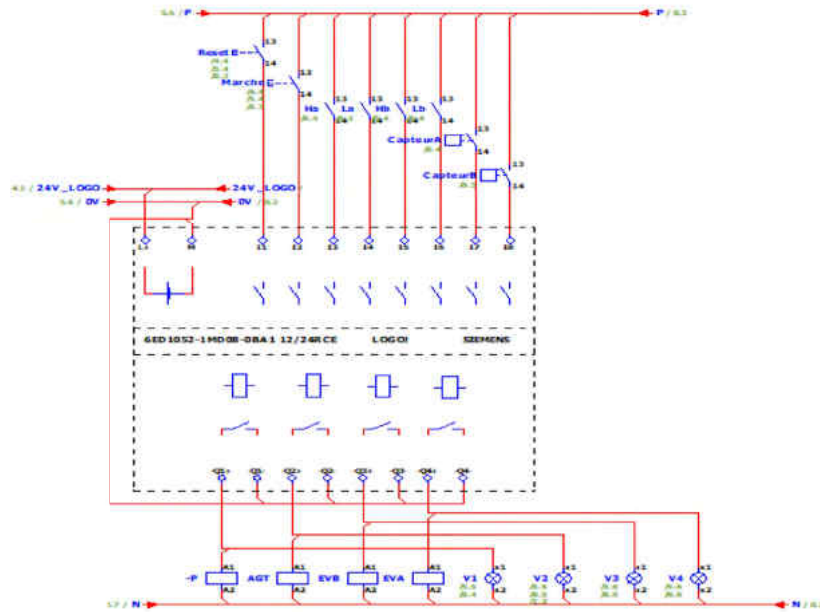


FIGURE 3.7 – Câblage des Entrées et Sorties

#### 3.5.2 Table d’Affectation

Les entrées et les sorties de notre système sont montrée dans le tableau ci-dessous :

Adresses	Variables	Adresses	Variables	Adresses	Variables
<b>Les Entrées</b>		<b>Les Sorties</b>		<b>Mémentos</b>	
I1	Reset	Q1	Pompe_V1	M1	X0
I2	Marche	Q2	Agitateur_V2	M2	X1
I3	LA	Q3	EVB_V3	M3	X2
I4	HA	Q4	EVA_V4	M4	X3
I5	LB			M5	X4
I6	HB			M6	X5
AI1	Capteur A			M7	X6
AI2	Capteur B				

TABLE 3.1 – Table d’Affectation des Variables

#### 3.5.3 Paramètre Réseau du LOGO!

Pour accéder aux paramètres de l’automate LOGO!, voici les étapes à suivre.

1. Cliquer sur la touche **ESC**.
2. Choisir le menu **Réseau(Network)** et cliquer sur la touche **OK**.
3. Choisir le menu **Adresse IP(IP Adresse)** et cliquer sur la touche **OK**.

Après avoir suivi les étapes, un menu affichera les paramètres réseau, où il sera possible de les modifier ou de les conserver tels quels.



FIGURE 3.8 – l'Adresse IP du LOGO!

### 3.5.4 La Commande et La Supervision de la Station de Pompage avec LOGO! TDE

Pour commander et superviser la station de pompage avec LOGO! TDE, il est nécessaire de configurer les paramètres de réseau entre l'automate LOGO! et l'afficheur LOGO! TDE.

Tout d'abord, il faut récupérer les paramètres réseau de LOGO! TDE, il faut accéder au menu **Paramètre TDE** et sélectionner le sous-menu **Réseau TDE**. Une fois les paramètres affichés, il est possible de les modifier ou de les conserver tels quels.



FIGURE 3.9 – Paramètres réseau de LOGO! TDE

Ensuite, il est important de s'assurer que LOGO! TDE communique correctement avec l'automate LOGO!. Pour cela, il faut accéder au menu **Sélectionner LOGO!** et saisir l'adresse ID de LOGO!. Dans notre cas, l'adresse IP est **168.192.1.2**.



FIGURE 3.10 – Saisir l'adresse IP de LOGO!

Enfin, il convient d'effectuer la configuration matérielle. Pour ce faire, il faut ajouter un nouveau appareil et sélectionner le module LOGO! TDE, puis saisir les paramètres réseaux correspondants.

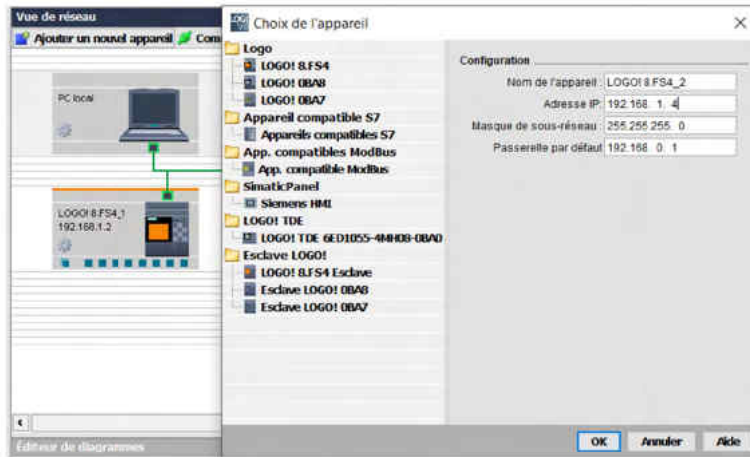



FIGURE 3.11 – Configuration de matériel de LOGO! TDE

### 3.5.4.1 La Commande de la Station avec LOGO! TDE

Il est possible de commander le système avec LOGO! TDE, à travers les touches de fonction  $F_1$ ,  $F_2$ ,  $F_3$  et  $F_4$  en les utilisant comme entrées numériques dans le programme de commande, pour effectuer des actions de contrôle telles que l'activation ou la désactivation de certaines fonctionnalités.

Leur utilisation permet d'économiser l'utilisation de commutateurs et d'entrées, tout en offrant la possibilité d'intervenir manuellement dans le programme de commande.

Afin d'utiliser ces touches comme des entrées, nous devons les configurer dans le programme principal sur LOGO!Soft Comfort, en utilisant le bloc fonctionnel **Touche de fonction LOGO!**

**TD** . Par exemple, il est possible de les utiliser pour activer les quatre sorties  $Q_1$ ,  $Q_2$ ,  $Q_3$  et  $Q_4$  afin de tester s'ils fonctionnent correctement avant de démarrer le système.

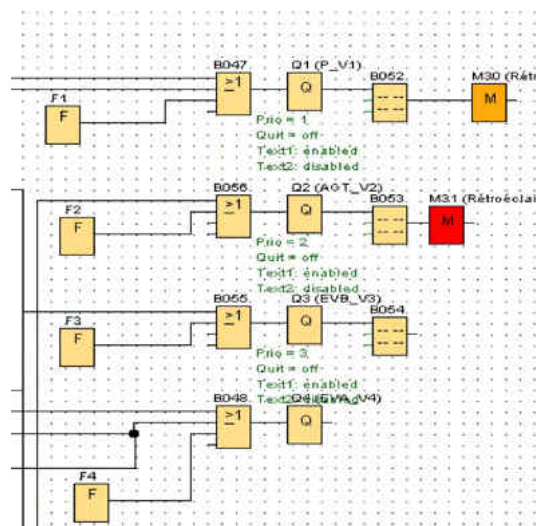


FIGURE 3.12 – Commande avec LOGO! TDE

### 3.5.4.2 La Supervision de la Station avec LOGO! TDE

La supervision avec LOGO! TDE permet de surveiller le système en temps réel grâce à l'interface LCD intégrée. À l'aide du LOGO! TDE, il est possible de visualiser les états des différentes entrées

et sorties, de surveiller les valeurs des capteurs, d'afficher des alarmes ou des messages d'erreur. La configuration de ce module s'effectue dans notre programme de commande avec LOGO!Soft Comfort à travers le bloc fonctionnel **Texte de message** trouvé dans la section **Instructions**.



FIGURE 3.13 – Texte de Message

Pour configurer l'afficheur du LOGO! TDE, nous devons accéder aux paramètres du bloc en double-cliquant dessus. Nous pouvons insérer la date, l'heure, les entrées et sorties, les variables TOR et analogiques, les timers, les compteurs, etc., tout en respectant la taille de l'afficheur TDE (6 lignes et 20 caractères par ligne).

Pour que cette configuration s'affiche sur le LOGO! TDE, il faut sélectionner la case **LOGO! TD**, ou **les deux** si nécessaire (sélectionner la case "Ser.Web" dans le cas de la supervision sur le serveur Web).

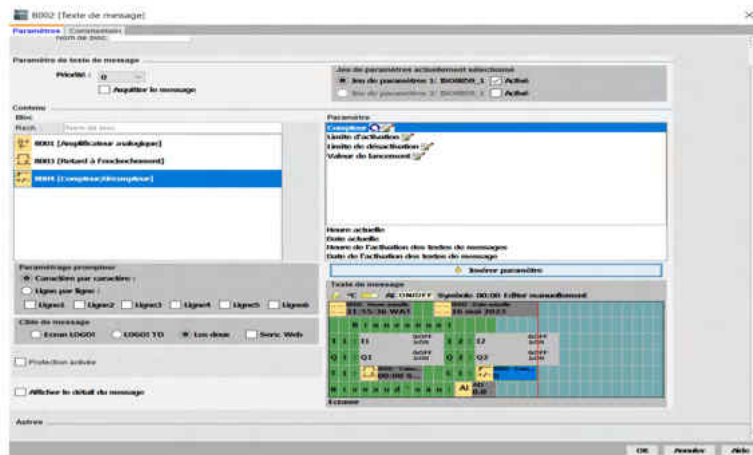


FIGURE 3.14 – Paramètre LOGO! TDE

Dans le cas de notre 1er cahier des charges nous pouvons superviser sur LOGO! TDE les sorties telles que la pompe, l'agitateur, l'électrovanne B et l'électrovanne A, et que les voyants V1, V2, V3 et V4. De même, nous pouvons surveiller les entrées HA (high a), LA (low a), HB (high b) et LB (low b), ainsi que les temporisateurs t1 et t2 et l'état du compteur.

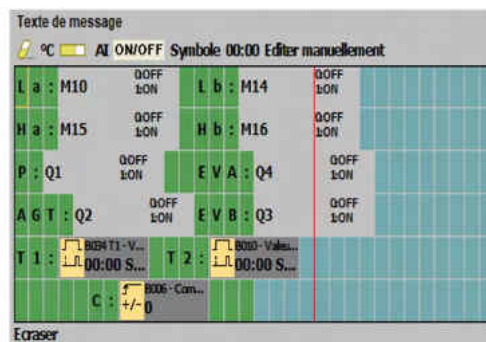


FIGURE 3.15 – Configuration supervision du LOGO! TDE

La supervision du système sur LOGO! TDE du système à l'état 5 est le suivant :

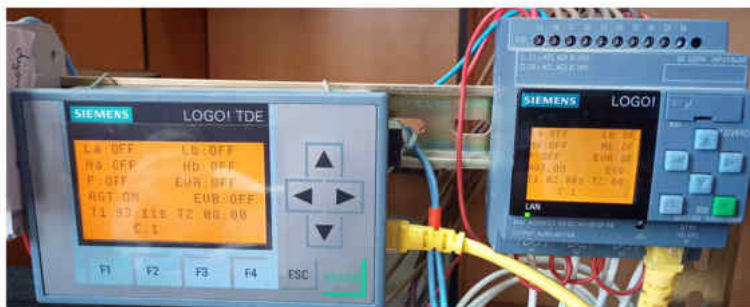


FIGURE 3.16 – La Supervision du 1er cahier des charge

Avec LOGO! TDE, il est possible de créer plusieurs vues sur l'interface en configurant plusieurs blocs de texte. Il est essentiel de bien gérer leurs paramètres d'activation et de désactivation.

Dans le cas du premier cahier des charges, nous avons cree 3 vues. La premiere vue est la vue principale qui s'affiche en premier. Cette vue est activée par les états  $X_0$ .



FIGURE 3.17 – Vue 1 LOGO!TDE

La deuxieme vue est la vue de supervision qui permet de superviser l'état ON/OFF des entrées et des sorties, les valeurs des deux temporisations  $T_1$  et  $T_2$  ainsi que la valeur actuelle  $N$  du compteur. Cette vue est activée par les états  $X_1, X_2, X_3, X_4, X_5, X_6$ .



FIGURE 3.18 – Vue 2 LOGO!TDE

La troisième vue nous permet de surveiller le niveau d'eau dans les deux cuves A et B. Cette vue est activée en cliquant sur la touche  $F_1$ .



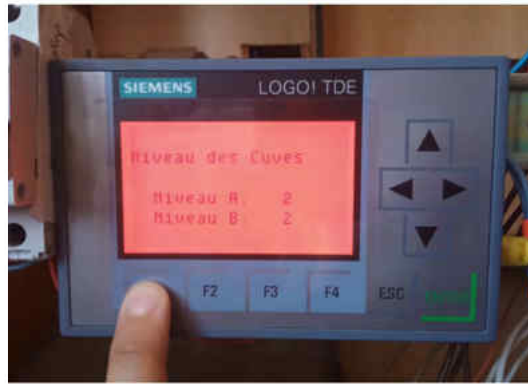


FIGURE 3.19 – Vue 3 LOGO!TDE

Aussi nous pouvons contrôler le rétroéclairage du LOGO! TDE en ajoutant un memento de couleur spécifique au bloc du texte, par exemple si en ajoutant le memento30 nous aurons un rétroéclairage orange sur LOGO! TDE.

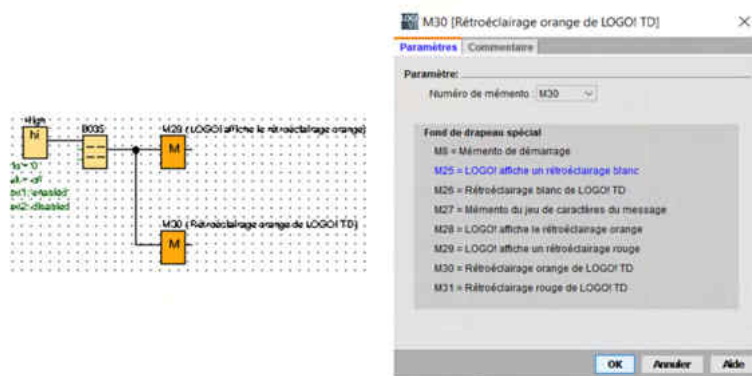


FIGURE 3.20 – Le rétroéclairage de LOGO! TDE

D'où il est possible de joindre chaque sorties a un memento de couleur en cohérence avec les voyants utilisés auparavant dans l'installation. La figure ci-dessous représente l'affectation de la couleur orange à l'activation du voyons  $V_3$  et l'affectation de la couleur rouge à l'activation des sorties  $Q_4$  et  $Q_4$ .



FIGURE 3.21 – Le rétroéclairage TDE

### 3.5.5 La Commande et La Supervision de la Station de Pompage avec WebServer

Le module LOGO! de Siemens possède un navigateur web intégré qui permet de commander et superviser le système à travers un PC, une tablette ou un smartphone. Pour accéder au navigateur web, il faut d'abord configurer le bloc de fonction **Texte de message**, comme montré précédemment. Cette fois-ci, il faut sélectionner la case **Web Server** pour activer le serveur web.

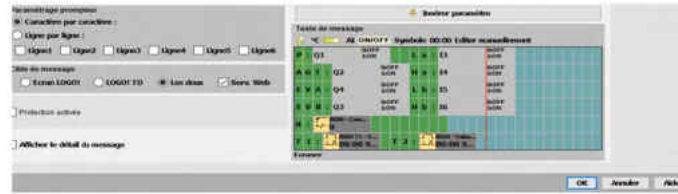


FIGURE 3.22 – Activation du Web Server

Ensuite, il faut se connecter avec le module réel pour activer les paramètres en ligne, et le contrôle d'accès s'effectue seulement en mode d'arrêt du module.

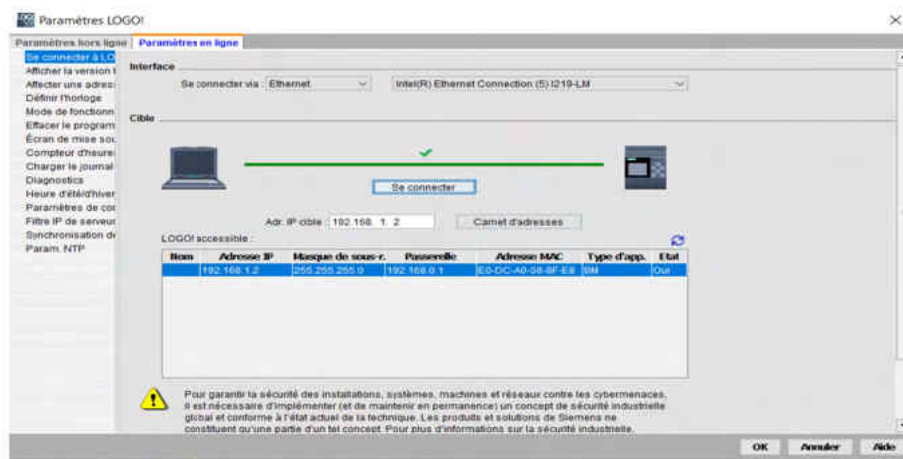


FIGURE 3.23 – Activation du Web Server

Après avoir activé les paramètres en ligne, nous pouvons accéder au menu des **Paramètres de contrôle d'accès** pour activer l'accès au navigateur web avec un PC et créer un mot de passe.

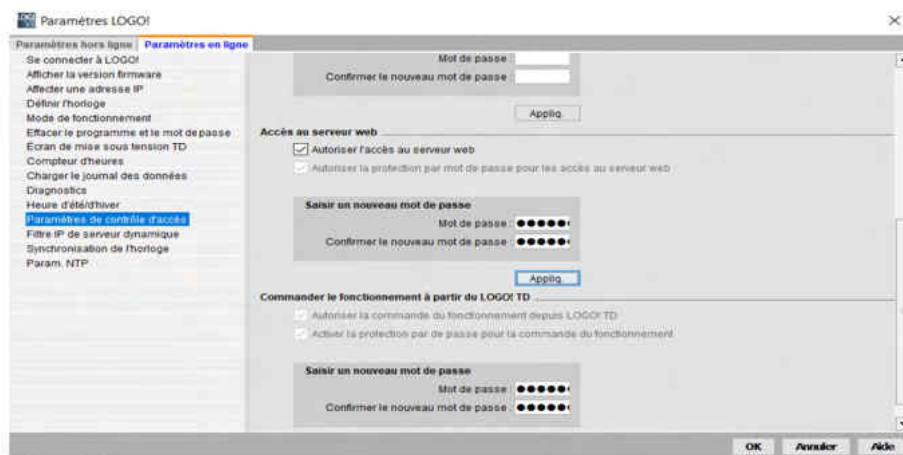


FIGURE 3.24 – Création du Mot de Passe de Web Server

Une fois que le navigateur web est activé, nous pouvons accéder via Internet à la page web en utilisant son adresse IP (192.168.1.2).

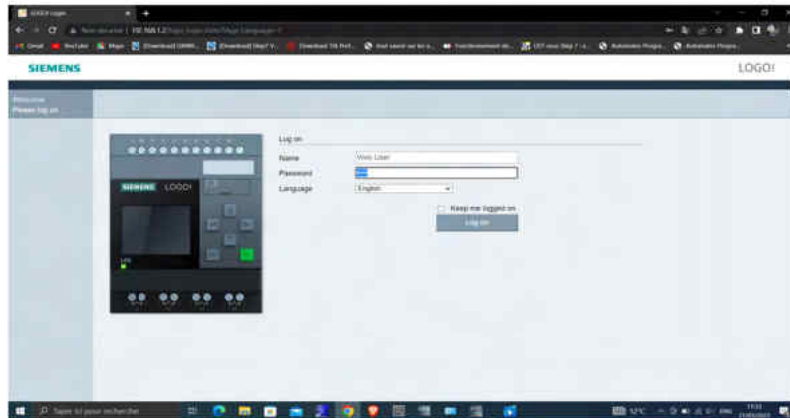


FIGURE 3.25 – Commande avec LOGO! TDE

Enfin, nous pouvons superviser les états du système.

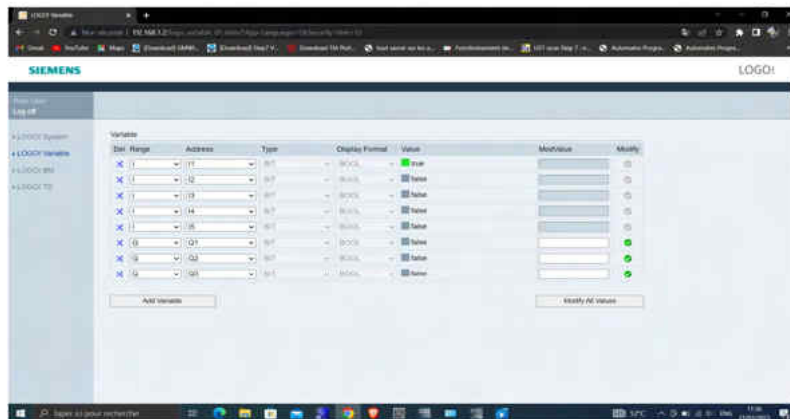


FIGURE 3.26 – Supervision des Etats sur le Navigateur Web

Le serveur offre une interface pour le module LOGO! et le module LOGO! TDE, où il est possible de commander et superviser le processus, et de bénéficier des mêmes fonctionnalités que LOGO! TDE à distance. Ainsi, il est possible de commander le système en utilisant les touches *F1*, *F2*, *F3*, et *F4*.



FIGURE 3.27 – Commande et la supervision avec LOGO! sur Web Server



FIGURE 3.28 – Commande et la supervision avec LOGO! TDE sur Web Server

Il est important de noter que la procédure spécifique pour activer le navigateur web et configurer le contrôle d'accès peut varier en fonction de la version de LOGO! que vous utilisez, ainsi que des autres composants matériels et logiciels de votre système. Il est donc recommandé de se référer à la documentation officielle de Siemens et de suivre les meilleures pratiques de sécurité pour configurer correctement l'accès via le navigateur web.

### 3.5.6 La Commande et La Supervision de la Station de Pompage avec Excel et le KepserverEX

Dans cette partie, nous allons utiliser Microsoft Excel et le logiciel KepserverEX afin de développer une autre technique de commande et de supervision.

KepserverEX est un logiciel de serveur OPC utilisé dans l'automatisation industrielle. Il permet de connecter différents équipements et systèmes industriels en utilisant des protocoles standard tels que OPC, Modbus, SNMP, etc. KepServerEX est largement utilisé dans les applications de supervision et de contrôle de processus dans les environnements industriels. KepserverEX collecte des données provenant de diverses sources, telles que des capteurs, des appareils de mesure, des automates programmables, des systèmes SCADA (Supervisory Control and Data Acquisition), etc. Il agrège ces données et les met à disposition pour les applications de supervision, de contrôle et d'analyse ultérieures.

Dans notre situation, Kepserver va lier LOGO! et Excel afin d'utiliser Excel comme interface homme-machine (IHM) qui recevra les données de LOGO! via KepserverEX et enverra des commandes marche/Reset à LOGO! à l'aide de KepserverEX.

Ainsi, Excel jouera le rôle d'interface homme-machine pour afficher les états du système, envoyer des commandes de marche et de réinitialisation, tandis que KepserverEX jouera le rôle de collecte des données et de protocole de communication entre LOGO! et l'interface Excel.

L'utilisation d'Excel comme interface de commande et de supervision présente certains avantages par rapport à l'utilisation de LOGO! TDE :

- L'interface avec Excel offre des fonctionnalités avancées pour l'analyse, le traitement et la visualisation des données, dépassant celles de l'interface LOGO! TDE. Par exemple, Excel permet la création de graphiques et de tableaux croisés dynamiques.
- Excel peut être facilement intégré à d'autres applications, améliorant l'interconnectivité et l'intégration des données de l'automate LOGO! avec d'autres systèmes.

- Les feuilles de calcul Excel peuvent être partagées avec d'autres utilisateurs, ce qui facilite le partage et la collaboration sur les données de l'automate.

De plus, l'utilisation d'Excel comme interface de commande et de supervision dans cette partie vise deux objectifs principaux :

- Le premier objectif est d'apprendre une nouvelle technique de communication fiable entre différents systèmes en utilisant le logiciel **KepserverEX**.
- Le deuxième objectif est d'assurer le fonctionnement de la supervision sur différents ordinateurs dotés de capacités de processeurs variables. En effet, certains PC ont une capacité de processeur limitée, ce qui peut les empêcher de démarrer certains logiciels.

### 3.5.7.1 Communication entre Excel et LOGO! avec KepserverEX

Pour réaliser la communication entre LOGO! et Excel avec KepServerEX, vous devez suivre les étapes suivantes :

1. Déclarer les variables à lire et à écrire à partir de LOGO! dans Excel. Pour ce faire, nous devons sélectionner la cellule correspondante dans Excel, cliquer avec le bouton droit de la souris, puis choisir **Define Name**. Ensuite, nous pouvons insérer le nom souhaité pour la variable.

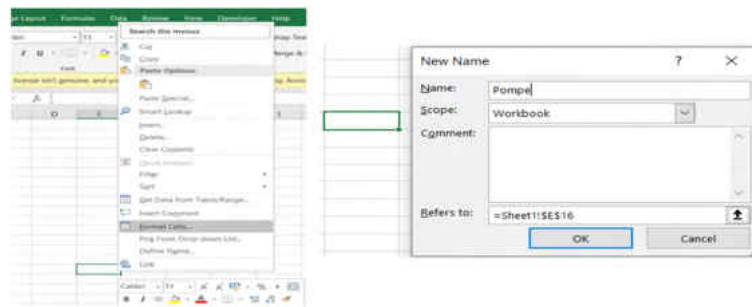


FIGURE 3.29 – Déclaration des variables dans l'excel

2. Déclarer les variables d'Excel dans le canal réservé à Excel (DDE) dans KepServerEX.

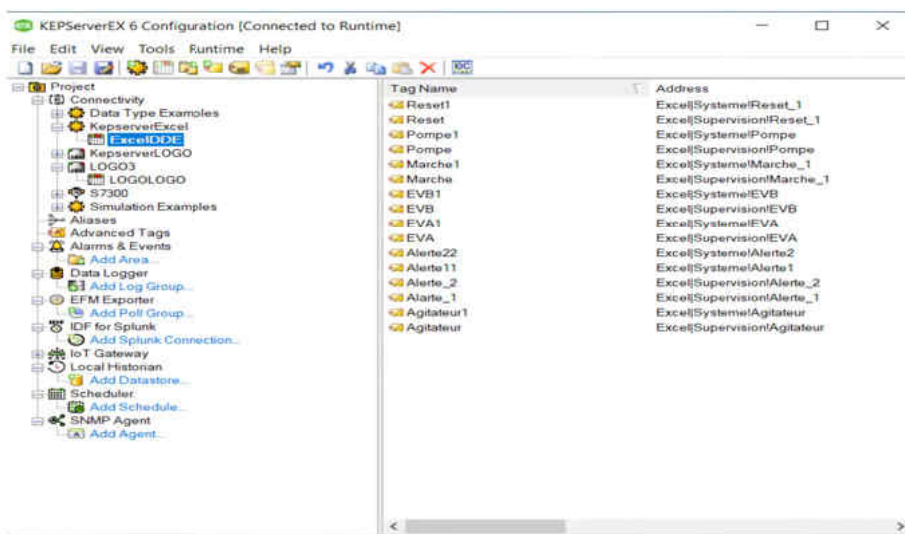


FIGURE 3.30 – Déclaration des variables Excel dans KepserverEX

3. Déclarer les variables de LOGO! que vous souhaitez lire ou écrire depuis Excel dans le canal réservé à LOGO! dans KepServerEX.

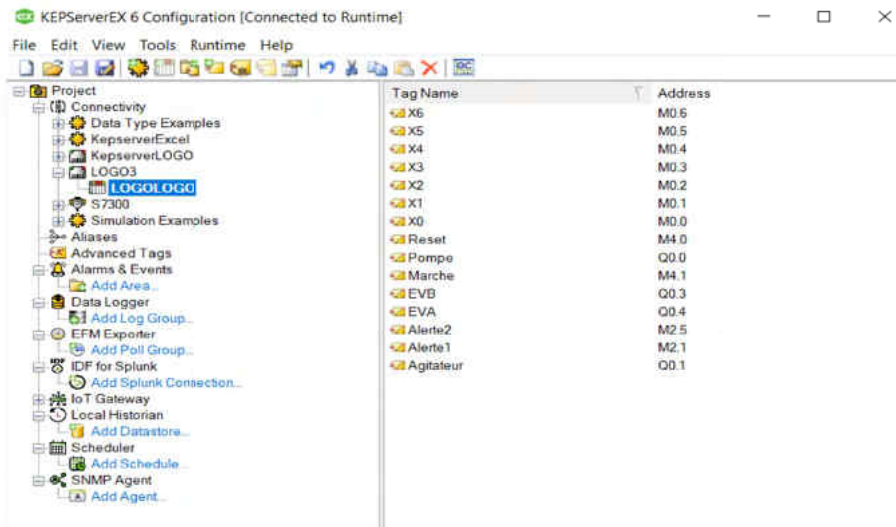


FIGURE 3.31 – Déclaration des variables LOGO! dans KepserverEX

4. Créer une liaison (lien) entre les variables correspondantes dans Excel et LOGO!.

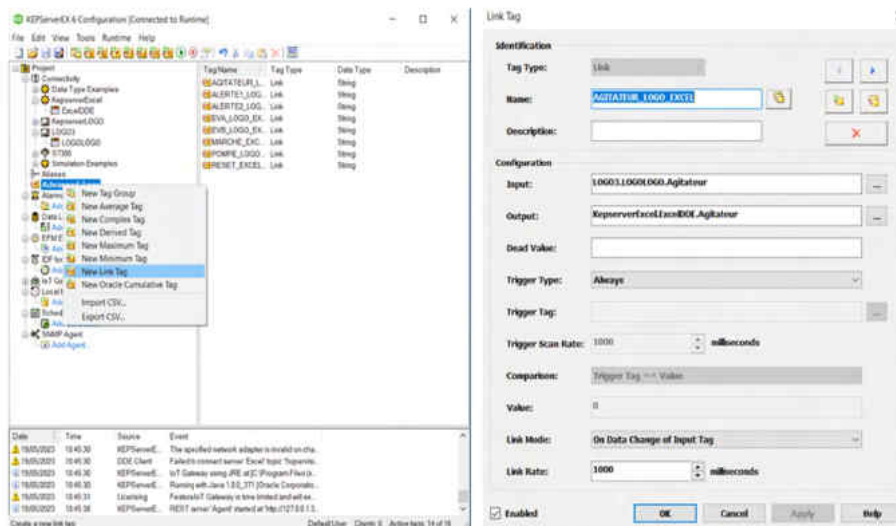


FIGURE 3.32 – Liaison entre Excel et LOGO! KepserverEX

### 3.5.7.2 Les vues réalisées à travers l'Excel et le KepserverEX

Avec Excel et Kepserver, nous avons réussi à créer deux vues (deux feuilles) : la première est la vue principale, et la deuxième est la vue de commande et de supervision du système.

La vue principale représente la page d'accueil où nous devons nous identifier afin d'accéder à la deuxième vue. Pour cela, nous devons remplir un formulaire en saisissant un nom d'utilisateur et un mot de passe spécifiques. Si le nom d'utilisateur et le mot de passe sont incorrects, l'accès est refusé.

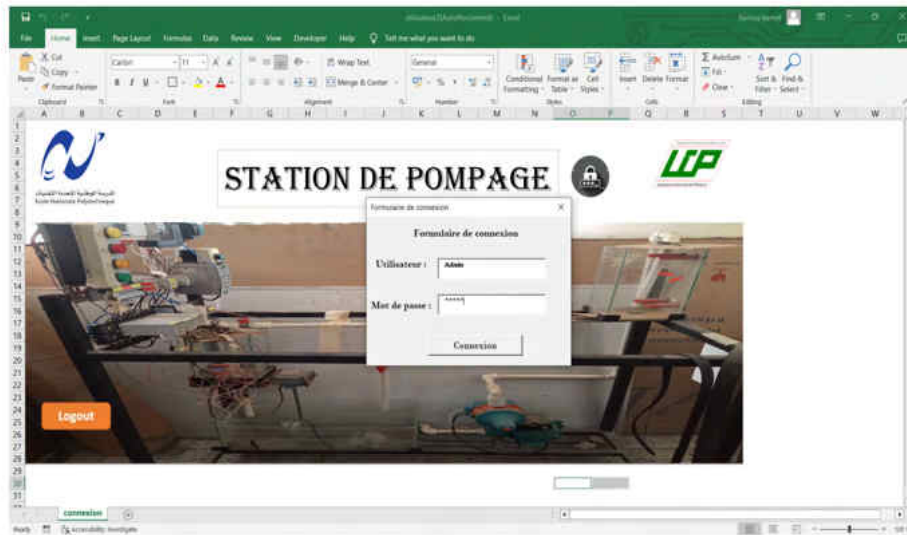


FIGURE 3.33 – Page d'accueil

Il la deuxième vue représente la vue de commande du système et la supervision de celui-ci. Elle permet de contrôler l'arrêt/marche des actionneurs et de surveiller les états du système.

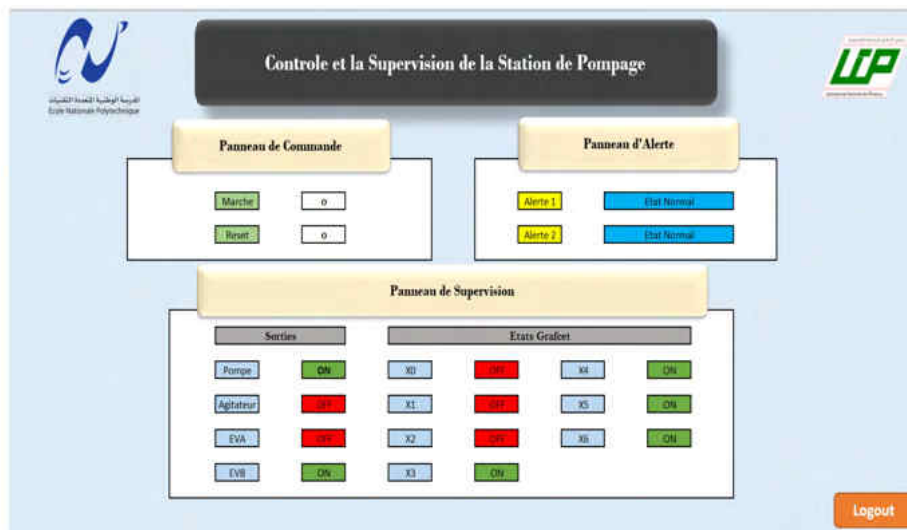


FIGURE 3.34 – Vue 2 Excel

Pour se déconnecter, il suffit de cliquer sur "Déconnexion" ou "Logout".

Le programme utilisé pour la commande et la supervision est détaillé dans l'Annexe E. Les détails d'utilisation du KepseverEX sont développés dans l'Annexe D.

### 3.6 Conclusion

Dans ce chapitre, nous avons réussi à mettre en place le contrôle de la station de pompage en nous familiarisant avec la gamme Siemens LOGO!. Nous avons également exploré en détail le module LOGO! TDE. Grâce à l'utilisation de différentes techniques, nous avons pu acquérir une compréhension approfondie des différentes méthodes de communication.

# Chapitre 4

## La Commande et la Supervision de la Station de Pompage avec Easybuilder Pro

### 4.1 Introduction

Les techniques de commande et de supervision vues dans les deux chapitres précédents peuvent présenter certaines limites en termes de fonctionnalités et de capacité de surveillance. Afin de remédier à ces limitations, nous allons introduire une nouvelle technique dans ce chapitre, en utilisant le logiciel Easybuilder Pro. Cette technique sera utilisée pour la première fois dans notre laboratoire d'automatique. De plus, ce chapitre abordera d'autres notions telles que le **GEMMA**, la communication **DCS** et **SCADA**, ainsi que leur relation avec Easybuilder.

### 4.2 Présentation du Logiciel EasyBuilder Pro

EasyBuilder Pro est un logiciel de conception d'interfaces homme-machine (IHM) développé par Weintek, une entreprise spécialisée dans les solutions IHM. Il est largement utilisé dans le domaine de l'automatisation industrielle pour créer des interfaces graphiques intuitives facilitant le contrôle et la surveillance des processus industriels. Les utilisateurs peuvent ajouter facilement des éléments tels que des boutons, des indicateurs et des graphiques, et configurer des fonctionnalités avancées telles que la communication avec des dispositifs externes et la gestion des alarmes.

#### 4.2.1 Interface de Projet

L'interface du projet dans EasyBuilder Pro est l'endroit où nous pouvons concevoir, configurer et gérer tous les éléments de votre interface homme-machine (IHM). Les principaux composants de l'interface du projet dans EasyBuilder Pro sont :

1. **Zone de travail** : C'est l'espace principal où nous concevons les éléments de l'interface homme-machine (IHM), plaçons et organisons les objets graphiques tels que les boutons, les indicateurs et les graphiques.
2. **Barre d'outils** : Elle propose une gamme de fonctionnalités telles que la création de boutons, la création de voyants, la déclaration d'alertes et le dessin de graphiques à barres, etc.
3. **L'arborescence de fenêtres** : Elle facilite la gestion et l'organisation du projet en nous permettant de voir toutes les fenêtres disponibles et d'interagir avec elles de manière pratique.
4. **Bibliothèque d'objets** : Cette partie de l'interface nous permet d'accéder à une variété d'objets préconfigurés, tels que les pompes, les moteurs, les symboles d'alertes, etc.



5. **Référence de l'IHM** : Elle affiche la référence de l'IHM choisie pour la configuration.

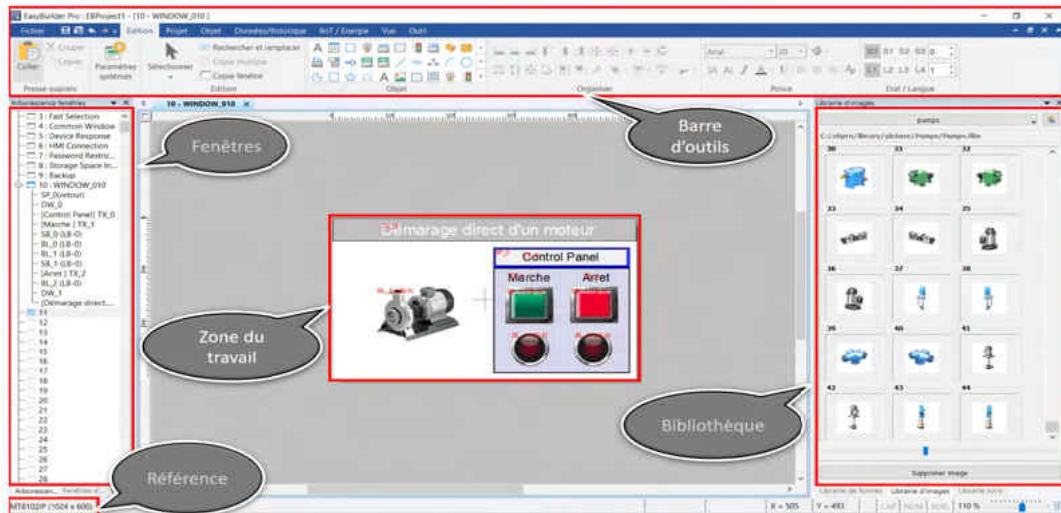


FIGURE 4.1 – Interface de Projet EasyBuilder

## 4.2.2 Barre d'outils

La barre d'outils dans EasyBuilder Pro est composée des outils suivants :

1. **Édition** : Nous pouvons utiliser cet outil pour éditer les objets présents dans la zone de travail, tels que le dimensionnement, l'alignement des objets, etc.



FIGURE 4.2 – Barre d'édition

2. **Projet** : Ce groupe d'outils nous permet de gérer notre projet dans son ensemble, notamment la compilation du projet, la simulation du fonctionnement, le transfert du programme du PC vers l'IHM, ainsi que la configuration des adresses.



FIGURE 4.3 – Barre de Projet

3. **Objet** : Ces outils sont dédiés à la gestion des objets de l'IHM. Nous pouvons créer des boutons, des voyants, configurer les actionneurs, les affichages numériques, etc.



FIGURE 4.4 – Barre d'Objet

4. **Données/Historique** : Nous pouvons utiliser ces outils pour déclarer des alarmes, effectuer l'acquisition de données et l'affichage des données historiques, etc.



FIGURE 4.5 – Barre Données/Historiques

5. **IIoT/Énergie** : Ces outils sont spécifiquement conçus pour la configuration des réseaux et de l'alimentation de l'IHM physique.



FIGURE 4.6 – Barre IIoT/Énergie

6. **Vue** : Nous pouvons ajuster les paramètres d'affichage de la vue de projet en utilisant ces outils. Cela inclut la régulation de la transparence, l'activation/désactivation de la grille, etc.



FIGURE 4.7 – Barre Vue

7. **Outils** : Ces outils nous permettent de nous connecter à d'autres logiciels, tels que Codsys, pour faciliter l'intégration et l'interopérabilité avec d'autres systèmes.



FIGURE 4.8 – Barre Outil

### 4.2.3 Création d'un nouveau projet

Pour créer un nouveau projet sur EasyBuilder, nous pouvons suivre les étapes suivantes :

1. Nous lançons le logiciel EasyBuilder Pro sur notre ordinateur.
2. Dans la barre de menu supérieure, nous cliquons sur **Fichier**.
3. Dans le menu déroulant, nous sélectionnons **Nouveau projet**.
4. Une nouvelle fenêtre s'ouvre, nous demandant de choisir le modèle d'IHM (Interface Homme-Machine) que nous souhaitons utiliser. Nous sélectionnons le modèle approprié en fonction de nos besoins.
5. Après avoir choisi le modèle d'IHM dans EasyBuilder, une fenêtre de configuration matérielle s'affiche.
6. Dans la fenêtre de configuration matérielle, nous pouvons spécifier les détails du matériel que nous souhaitons utiliser avec notre IHM, tels que les paramètres de communication, les adresses, les protocoles, ect.
7. Une fois la configuration matérielle terminée, nous cliquons sur **OK** et le projet est prêt à être utilisé.

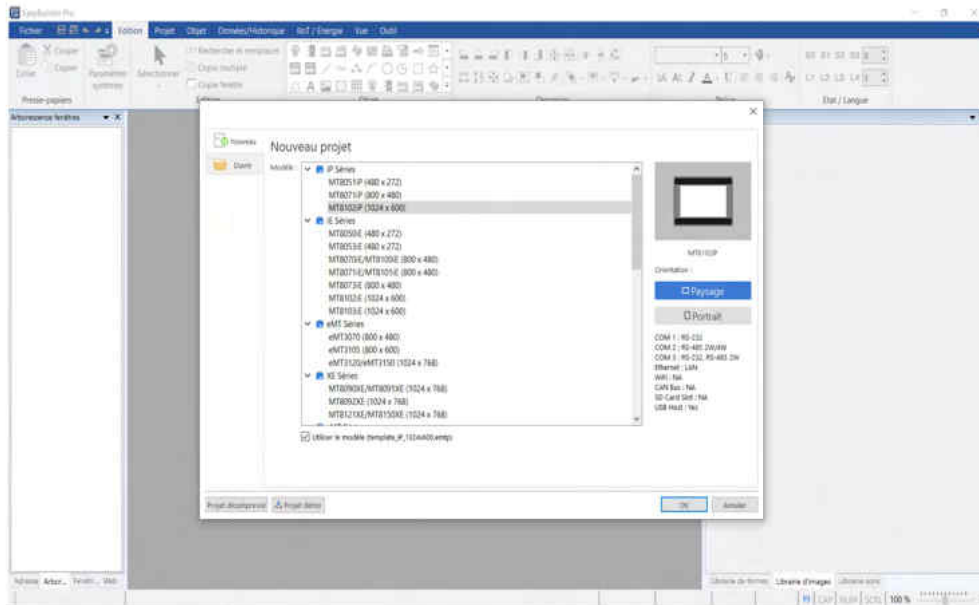


FIGURE 4.9 – Création d'un nouveau projet

#### 4.2.4 Configuration Matérielle

Les étapes de la configuration matérielle sont :

1. Dans la fenêtre de configuration matérielle, nous trouverons une liste des différents types de matériel pris en charge par EasyBuilder Pro. Nous sélectionnerons le type de matériel que nous utilisons ou souhaitons configurer.

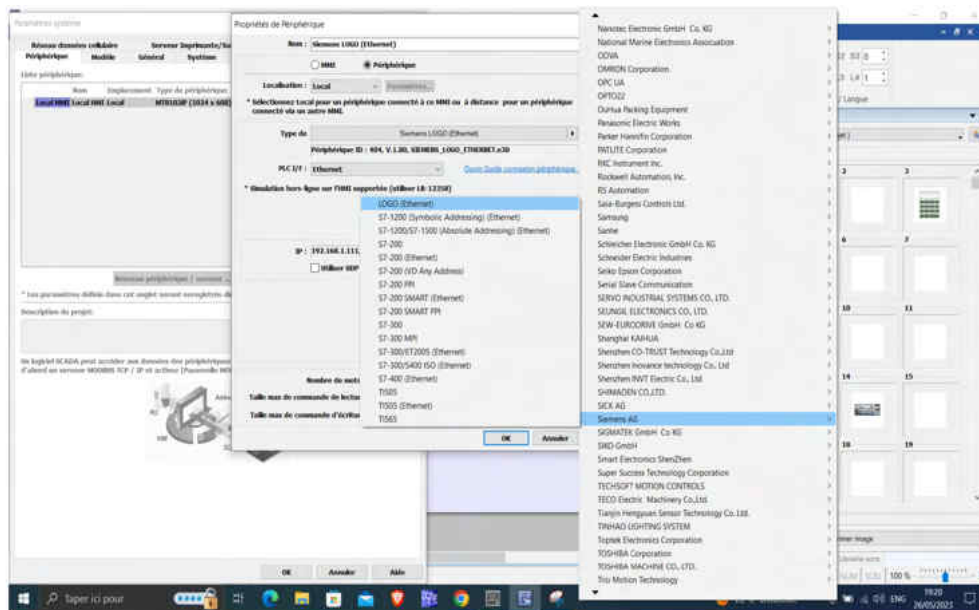


FIGURE 4.10 – Configuration Matérielle

2. Selon le type de matériel sélectionné, nous devons spécifier les détails de configuration spécifiques, tels que les paramètres de communication, les adresses, les protocoles, etc.
3. Une fois que nous avons configuré les paramètres matériels, nous cliquerons sur **OK** pour valider la configuration.

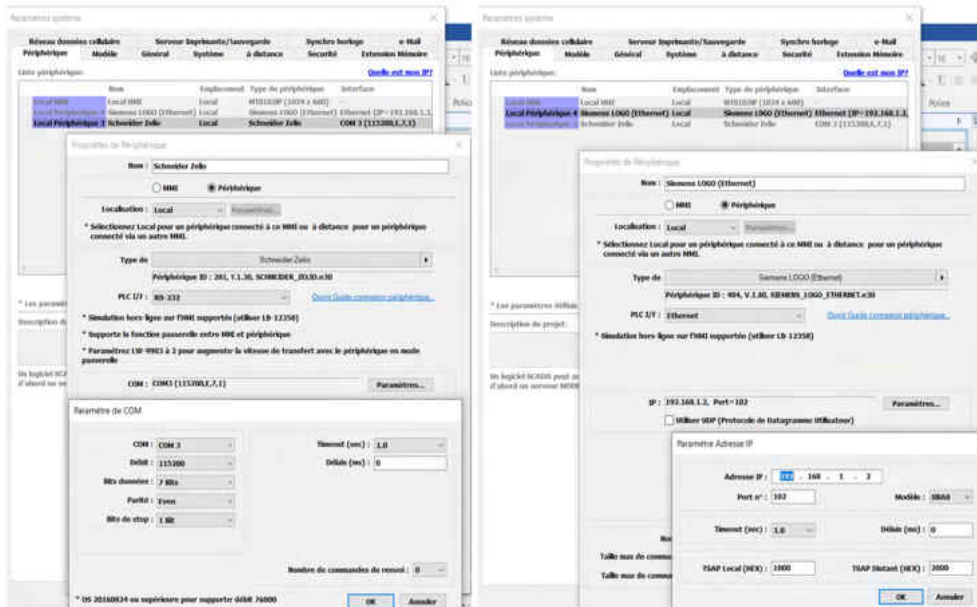


FIGURE 4.11 – Configuration des paramètres de communication de Zelio Logic et LOGO!

## 4.2.5 Déclaration des Variables

Dans EasyBuilder Pro, nous avons deux types de variables : les variables **bit** (Variables TOR) et les variables **mot** (Variables analogiques). Ces variables peuvent être lues ou écrites.

Pour déclarer une variable, nous suivons les étapes suivantes :

1. Accéder à la barre **Object**.
2. Sélectionner l'objet souhaité : voyons, bouton, diagramme, etc.
3. Configurer les paramètres de l'objet en sélectionnant le type de périphérique (l'automate) et l'adresse de la variable.

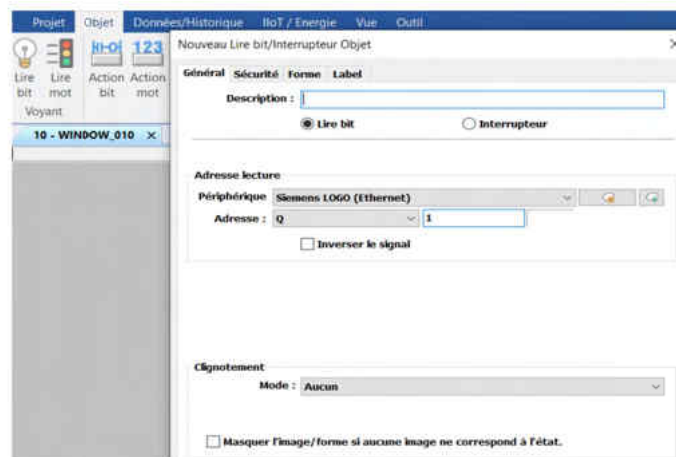


FIGURE 4.12 – Déclaration des variables dans Easybuilder Pro

## 4.2.6 Déclaration des Alarmes

Pour déclarer les alarmes dans Easybuilder Pro, nous suivons les étapes suivantes :

1. Accéder à la barre **Données/Historiques**.
2. Sélectionner l'icône **Déclaration des alarmes**.
3. Cliquer sur **Nouveau** pour déclaration une nouvelles alarmes, l'alarme peut être de type bit ou de type mot.
4. Configurer les paramètres de l'alarme en sélectionnant le type de périphérique (l'automate), l'adresse de la variable et la condition de l'enclenchement de cette alarme.

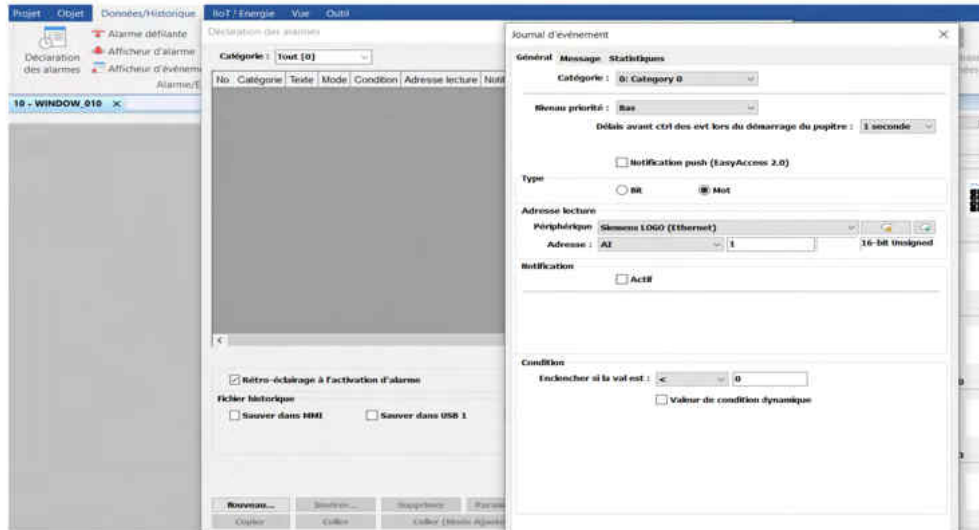


FIGURE 4.13 – Déclaration des alarmes dans Easybuilder Pro

5. Insérez le message souhaité à afficher lors d'une alarme en cliquant sur **message** dans la fenêtre du journal d'événement.



FIGURE 4.14 – Configuration des message d'alarmes dans Easybuilder Pro

## 4.2.7 Acquisition des données

L'acquisition des données dans Easybuilder Pro est réaliser en suivant les étapes suivantes :

1. Accéder à la barre **Données/Historiques**.
2. Sélectionner l'icône **Acquisition des données**.
3. Cliquez sur **Nouveau** pour déclarer une nouvelle donnée. Les données sont toujours de type mot.
4. Sélectionner le type du périphérique (l'automate) et l'adresse de la variable.
5. Sélectionnez le format de donnée souhaité et cliquez sur **OK**.

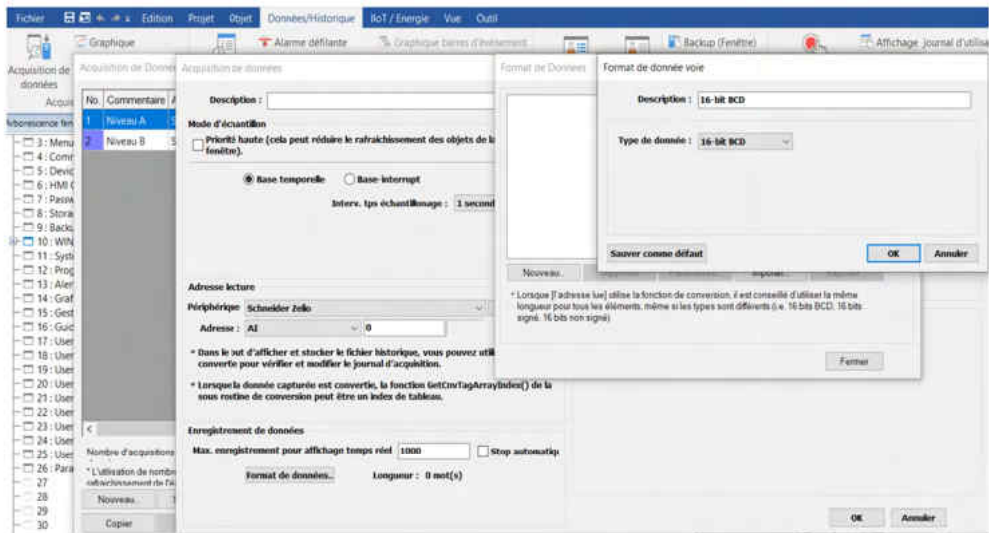


FIGURE 4.15 – Configuration des message d’alarmes dans Easybuilder Pro

## 4.2.8 Simulation du programme

Dans EasyBuilder, nous avons deux modes de simulation : la simulation en ligne et la simulation hors ligne.

1. **Simulation en ligne** : Ce mode de simulation nous permet de tester le programme de notre IHM en temps réel en étant connectés à l’automate. L’IHM peut interagir avec les données et les actions de l’automate, ce qui nous permet de valider le fonctionnement de l’ensemble du système.
2. **Simulation hors ligne** : Ce mode de simulation nous permet de vérifier la vue créée sans être connectés à l’automate. Cela nous permet de visualiser l’apparence et le comportement de notre IHM sans la nécessité d’une connexion matérielle. Nous pouvons interagir avec les objets de l’IHM et observer leur réaction pour nous assurer que la conception répond à nos attentes. Cependant, dans ce mode, les actions et les données de l’automate ne sont pas accessibles.

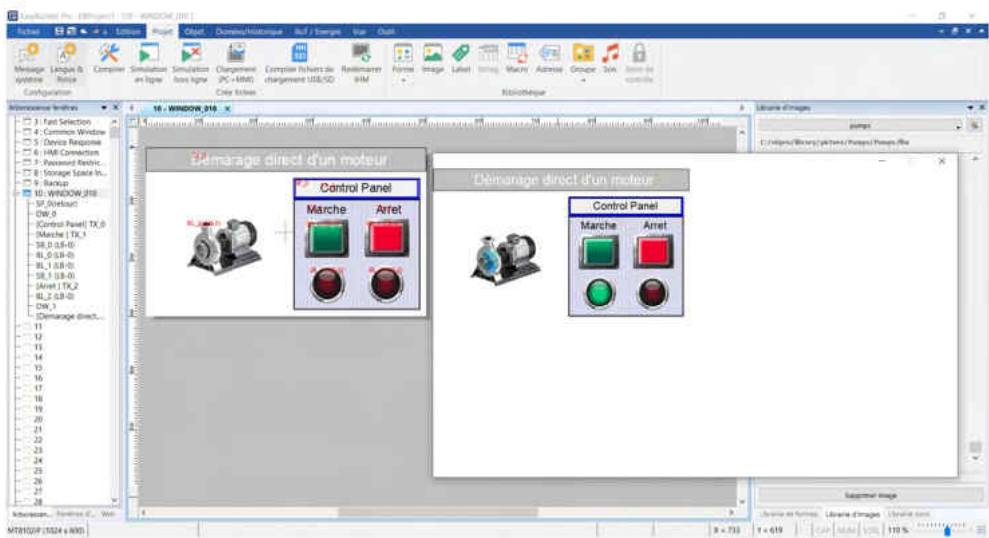


FIGURE 4.16 – Simulation du programme sur EasyBuilder Pro

## 4.2.9 Transfert du programme d'EasyBuilder Pro vers l'IHM

Afin de transférer le programme de logiciel Easybuilder Pro vers IHM physique nous devons :

1. Aller à la barre **Projet** cliquer sur l'icône **Chargement Pc** ->MMI
2. Saisir l'adresse IP de l'IHM physique par exemple **192.168.0.6**

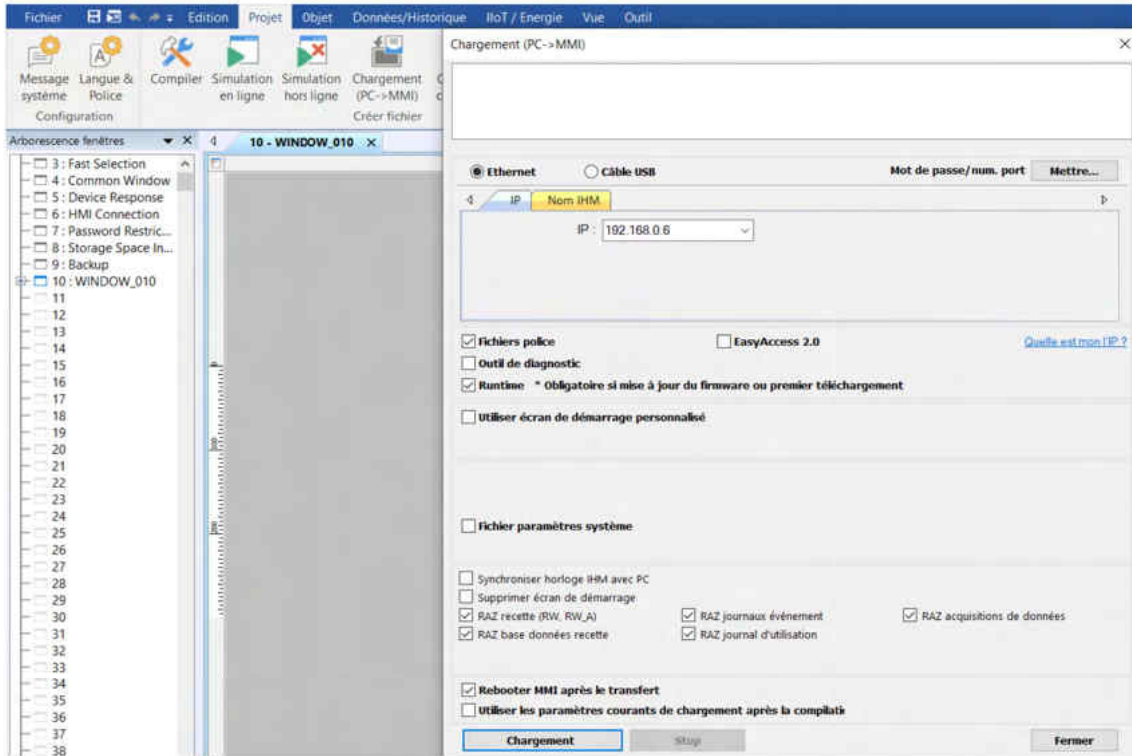


FIGURE 4.17 – Transfert du programme vers IHM physique

## 4.2.10 Les Avantages d'utilisation le logiciel Easybuilder Pro

Le choix du logiciel EasyBuilder Pro comme interface de commande et de supervision est basé sur plusieurs avantages :

- Tout d'abord, EasyBuilder Pro permet de commander notre système à travers notre PC, sans nécessiter une interface physique dédiée. Cela résout le problème de disponibilité d'une interface physique pour commander le système.
- De plus, EasyBuilder Pro est capable de communiquer avec une large gamme d'APIs, telles que Siemens, Schneider, ABB, etc. Cela évite le problème d'incompatibilité entre les interfaces des automates provenant de différentes gammes, car une interface standard est utilisée.
- Enfin, EasyBuilder Pro a la capacité de communiquer avec plusieurs APIs de différentes gammes et protocoles de communication simultanément. Cela facilite l'intégration de différents équipements et systèmes dans notre système global de commande et de supervision.

## 4.3 La Commande et la Supervision avec Easybuilder et l'automate Zelio Logic

Dans cette partie nous allons réaliser la commande et la supervision de la station selon le 1er cahier de charge en utilisant l'automate Zelio Logic.

### 4.3.1 Les Vues de l'IHM

L'interface IHM nous permet de créer des interfaces graphiques qui permettent aux utilisateurs de bien gérer le système. Pour notre station, nous avons créé 7 vues :

- La première vue est la page d'accueil, qui représente la vue principale de notre IHM et qui s'affiche en premier.

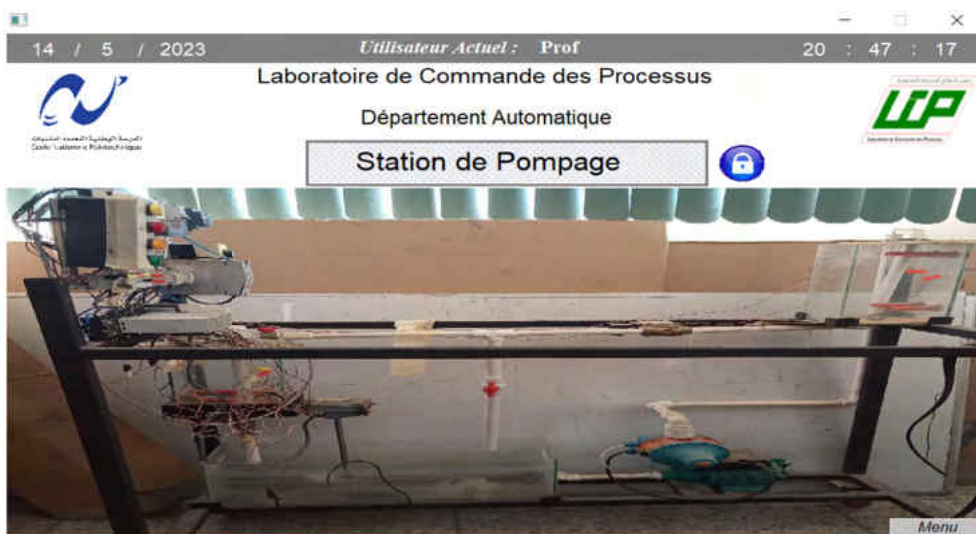


FIGURE 4.18 – Page d'accueil

- La deuxième vue est la vue du système, à partir de laquelle nous pouvons commander le système à travers les boutons Reset et Marche, suivre et surveiller le processus tel que l'activation des actionneurs, le niveau d'eau dans les cuves, et l'activation de l'alerte en cas de défaut au niveau du système. Il est également possible de sélectionner les valeurs de présélection des temporisations et du compteur. Cette vue représente une copie virtuelle de notre système réel.



FIGURE 4.19 – Vue du système en état Normal et en défaut



- La troisième vue est la vue de progression des niveaux d'eau dans les cuves A et B, où nous pouvons visualiser la variation du niveau de l'eau en fonction du temps.



FIGURE 4.20 – Vue du progression

- La quatrième vue est la vue du grafcet de notre système, cette vue permet de suivre l'activation et la désactivation des étapes, ce qui nous aide à comprendre le déroulement du grafcet du système.

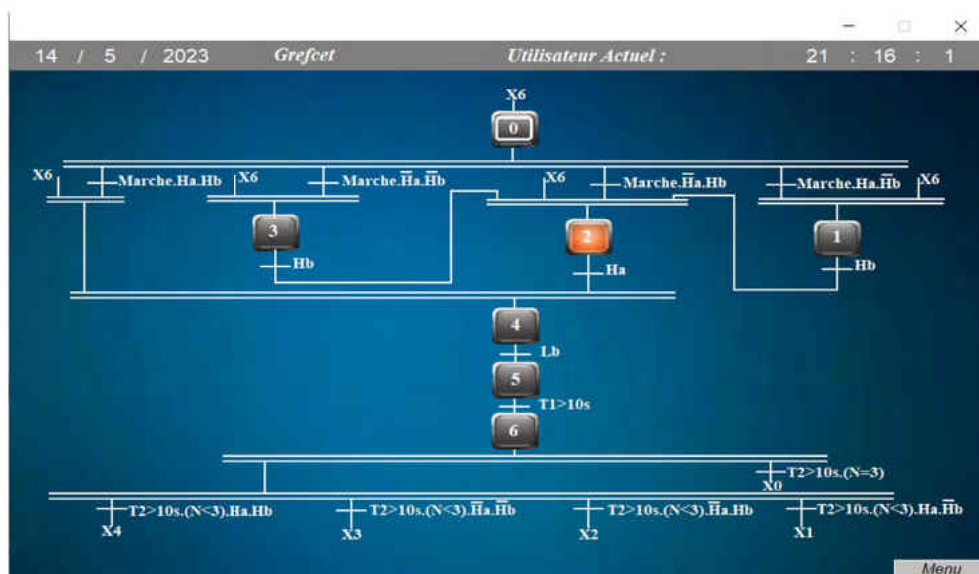


FIGURE 4.21 – Vue du Grafcet

- La cinquième vue est la vue de l'historique des alertes, qui permet d'archiver toutes les alertes survenues avec la date, l'heure et la cause.

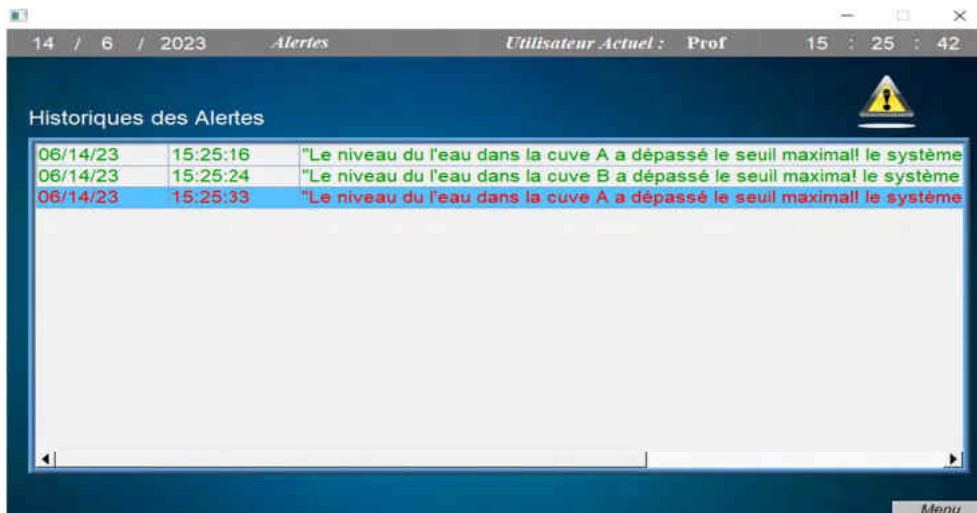


FIGURE 4.22 – Vue d'alerte

- La sixième vue est réservée à la gestion des utilisateurs. Elle permet d'enregistrer les accès des utilisateurs à l'interface avec la date, l'heure, la classe d'accès et le nom d'utilisateur.



FIGURE 4.23 – Vue de la gestion des utilisateurs

- La septième vue est le guide d'utilisation. Cette vue a pour objectif d'aider les utilisateurs à manipuler la supervision. Elle explique le fonctionnement du système, la gestion des vues et la sécurité de l'interface.



FIGURE 4.24 – Quelques vues du guide de l'utilisateur

Il est possible de naviguer entre les vues en utilisant le bouton "Menu" situé en bas de l'interface.

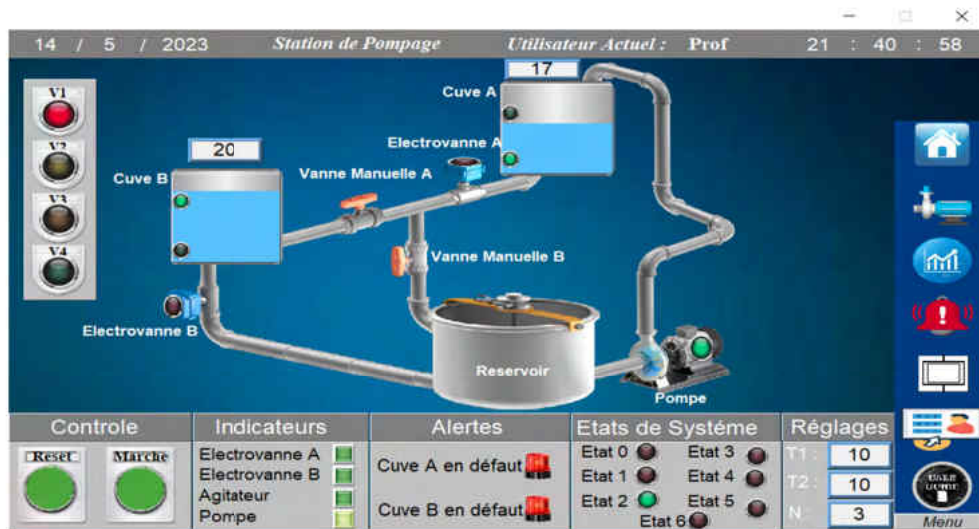


FIGURE 4.25 – Menu de Navigation

### 4.3.2 La Gestion des utilisateurs de la Sécurité

La gestion des utilisateurs et la sécurité sont des éléments clés de notre dispositif éducatif. En effet, notre station est destinée à être utilisée par des étudiants de différents niveaux d'études, et il est donc important de gérer l'accès des utilisateurs à l'interface pour assurer la sécurité du matériel et des personnes.

Nous avons ainsi classé les utilisateurs en quatre catégories :

- **La catégorie A** : les enseignants qui ont un accès total à l'interface et peuvent ainsi contrôler la station, modifier les paramètres et superviser le processus.
- **La catégorie B** : les étudiants de 5ème année qui ont accès à toutes les vues de l'interface, à l'exception de la vue de gestion des utilisateurs. Ils peuvent ainsi suivre le processus de la station et en comprendre le fonctionnement.
- **La catégorie C** : les étudiants de 4ème année qui ont les mêmes droits d'accès que les étudiants de 5ème année, à savoir l'accès à toutes les vues de l'interface, sauf celle de gestion des utilisateurs.
- **La catégorie D** : les étudiants de 3ème année qui ont un accès limité à l'interface. Ils peuvent uniquement superviser la station et visualiser les informations, mais ne peuvent pas commander la station à travers les boutons **Marche** et **Reset**.

Chaque utilisateur est identifié par un nom d'utilisateur et un mot de passe qui sont créés dans les paramètres de sécurité, comme illustré ci-dessous :

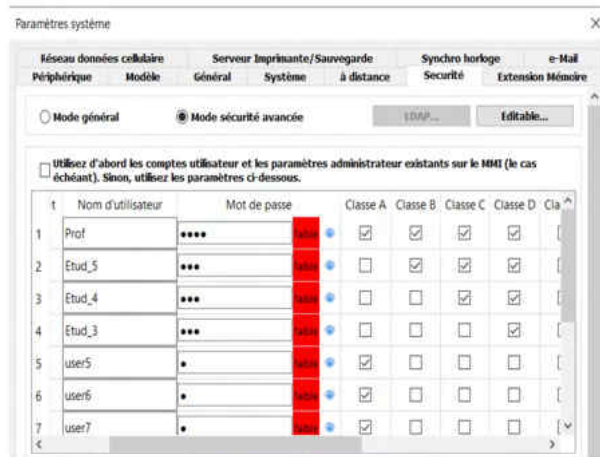


FIGURE 4.26 – Paramètres de sécurité

Pour s'identifier, l'utilisateur doit saisir son nom d'utilisateur et son mot de passe sur la page d'accueil, en cliquant sur l'icône de sécurité.



FIGURE 4.27 – Vue d'accès des utilisateurs

Si un utilisateur tente d'accéder à une page ou à une option de l'interface qui dépasse son niveau d'accès, un message d'avertissement s'affichera.

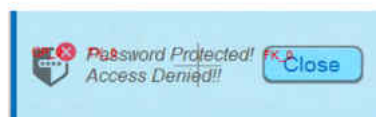


FIGURE 4.28 – Message d'avertissement

### 4.3.3 Communication entre EasyBuilder Pro et Zelio Logic

La communication entre le module Zelio Logic et EasyBuilder Pro s'effectue via le câble de communication SR2USB01. Cependant, il est important de noter que cette communication ne peut être établie avec succès que si les paramètres de communication sont correctement configurés dans EasyBuilder et Zelio Soft.

## a. Paramètres de communication d'EasyBuilder Pro

Les paramètres à configurer dans EasyBuilder Pro pour permettre la communication avec Zelio Logic sont les suivants :

1. Ouvrez EasyBuilder Pro et accédez aux paramètres du système et sélectionnez **Nouveau périphérique**.
2. Sélectionnez le type de périphérique **Schneider Electric Industries** puis **Zelio** dans la liste des options.
3. Choisissez le câble de communication approprié (par exemple, RS-232 ou USB) en fonction de votre configuration matérielle, dans notre cas, il s'agit du **RS-232**.
4. Sélectionnez le port de communication correspondant au câble utilisé (par exemple, COM1, COM2, etc.). Dans notre cas , il s'agit du port **COM 3**.

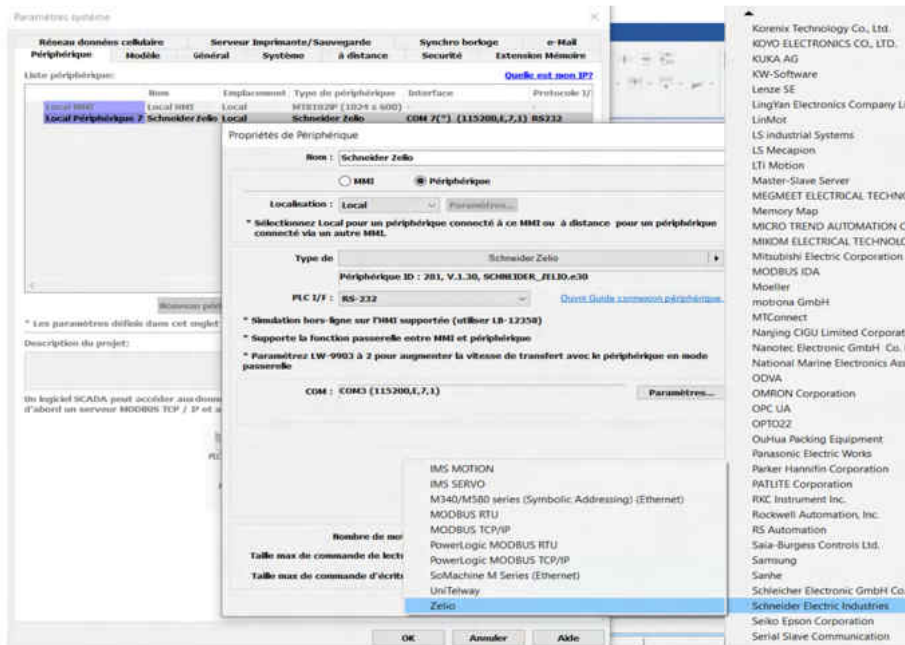


FIGURE 4.29 – paramètres de communication entre zelio et EasyBuilder Pro

## b. Paramètres de communication du Zelio Soft

Afin d'assurer une communication efficace entre le module Zelio Logic et EasyBuilder Pro, il est crucial de garantir la compatibilité des adresses entre les deux systèmes. Pour cela, il est essentiel d'utiliser les blocs de fonctions  $SL_{In}$  et  $SL_{Out}$  dans le programme principal qui sera transféré vers l'automate. Ces blocs de fonctions, propres au langage FBD, jouent un rôle essentiel dans l'échange d'informations entre les deux logiciels.

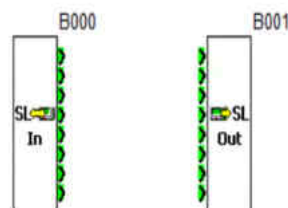


FIGURE 4.30 – Les blocs de communications sur Zelio Logic

EasyBuilder Pro dispose de 24 adresses  $SL_{In}$  de type mot, numérotées de 1 à 24. De même, il propose 24 adresses  $SL_{Out}$  de type mot, numérotées de 25 à 48. Cela signifie que EasyBuilder Pro peut envoyer au maximum 24 ordres vers Zelio Logic et recevoir 24 ordres de Zelio Logic.

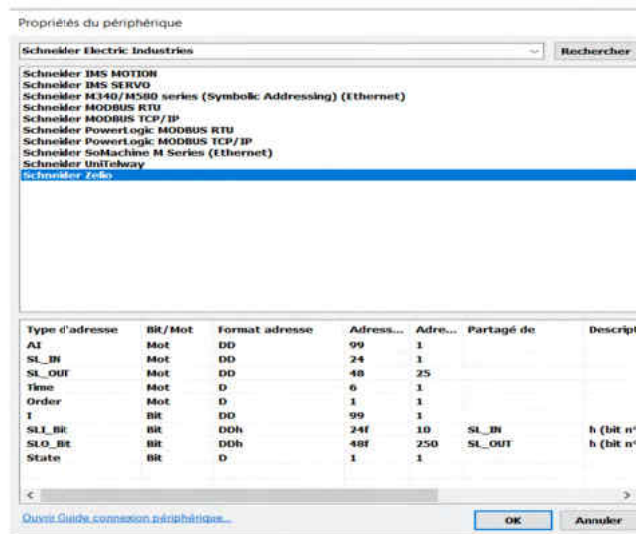


FIGURE 4.31 – Les adresses compatibles avec Zelio Logic

Dans Zelio Soft, les blocs fonctionnels  $SL_{In}$  et  $SL_{Out}$  sont conçus pour prendre en charge jusqu'à 8 connexions de type mot. Par conséquent, il est possible d'utiliser un maximum de 3 blocs  $SL_{In}$  avec les adresses 1 à 8, 9 à 16 et 17 à 24. De même, il est possible d'utiliser jusqu'à 3 blocs  $SL_{Out}$  avec les adresses 25 à 32, 33 à 40 et 41 à 48.

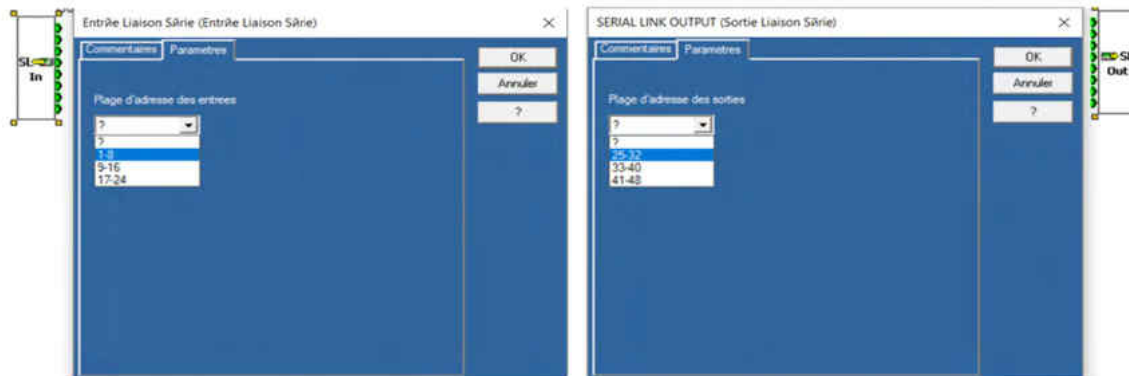


FIGURE 4.32 – Les adresse des bloc  $SL_{In}$  et  $SL_{Out}$

Comme expliqué précédemment, les variables  $SL_{In}$  et  $SL_{Out}$  sont de type mot, mais cela ne rend pas impossible de recevoir et d'envoyer des ordres de type bit. Ceci est possible en utilisant les blocs **CNA** (convertisseur numérique analogique) pour envoyer des ordres de type bit de Zelio Logic à Easybuilder, et le bloc **CAN** (convertisseur analogique numérique) pour recevoir des ordres de type bit d'Easybuilder à l'automate Zelio.

Pour envoyer un ordre bit de Zelio à Easybuilder, il suffit de connecter notre variable de type bit au premier bit du **CNA**. De même, pour recevoir un ordre bit d'Easybuilder Pro à Zelio, nous connectons le premier bit du bloc **CAN** à la sortie.

Il est important de noter qu'un bloc de **CNA** (respectivement **CAN**) est réservé pour une seule variable bit, comme illustré ci-dessous.

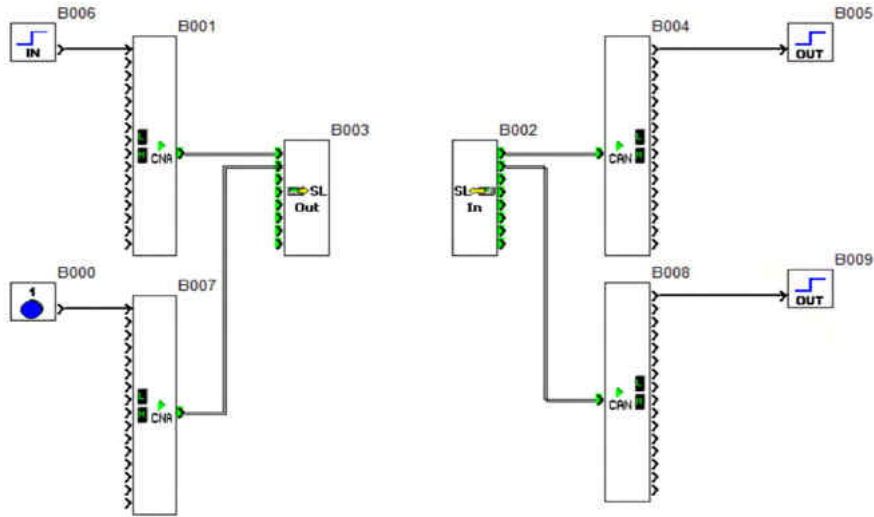


FIGURE 4.33 – Les adresse des bloc  $SL_{In}$  et  $SL_{Out}$

## 4.4 La Commande et la Supervision avec Easybuilder et l'automate LOGO !

L'objectif de cette partie est le même que celui de la partie précédente, à l'exception que nous allons travailler avec l'automate LOGO!. Les vues de cette partie sont également les mêmes que celles précédentes, à l'exception de la vue du système.

La communication entre l'automate LOGO! et EasyBuilder est différente de celle entre Zelio Logic et EasyBuilder. Ils peuvent communiquer via un câble **Ethernet**. Pour assurer une communication fiable et réussie, il est nécessaire de configurer les paramètres **TSAP** dans les deux logiciels, LOGO!Soft Comfort et EasyBuilder.

### 4.4.1 Point d'accès aux services techniques TSAP

Point d'accès aux services techniques (Technical Service Access Point) **TSAP**, est un identifiant unique utilisé pour établir une communication entre différents appareils.

Plus spécifiquement, un TSAP est utilisé dans les protocoles de communication pour permettre à des entités de réseau (par exemple, des équipements ou des logiciels) d'interagir les uns avec les autres. Chaque entité dispose d'un TSAP qui agit comme une adresse permettant de les identifier de manière unique dans le réseau.

L'utilisation de TSAP facilite l'échange de données et la communication entre les appareils connectés au réseau. Il permet de spécifier l'adresse de communication d'un appareil et de définir le point d'accès aux services techniques offerts par cet appareil.

Dans la configuration TSAP, il existe deux entités distinctes : le client et le serveur.

- Le client est l'entité qui initie la communication et envoie des demandes de services au serveur. Il est généralement celui qui souhaite accéder aux informations ou aux fonctionnalités offertes par le serveur.
- Le serveur, est l'entité qui répond aux demandes du client et fournit les services ou les informations demandés. Il est responsable de traiter les requêtes reçues du client et de fournir les

réponses correspondantes.

Dans notre cas, EasyBuilder Pro agit en tant que serveur et l'automate LOGO! joue le rôle de client .

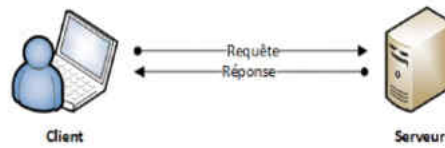


FIGURE 4.34 – Représentation Client/Serveur

## 4.4.2 Configuration des paramètres réseau dans LOGO!Soft Comfort

Pour établir une communication entre LOGO!Soft Comfort et EasyBuilder Pro, nous devons suivre deux étapes :

### 1. Créer une liaison client/serveur

a **Sélectionnez le mode diagramme**, puis cliquez sur l'option **Outils** dans la barre de menu et choisissez **Connexion Ethernet**.

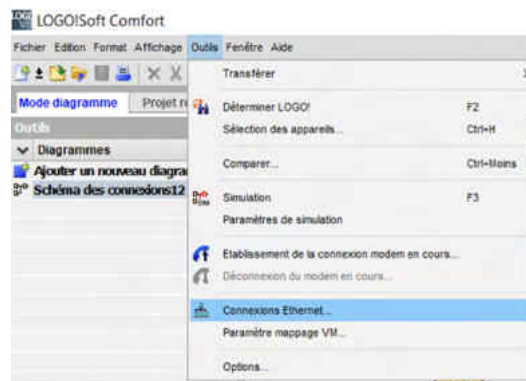


FIGURE 4.35 – Connexion Ethernet

b Créez une **Liaison serveur** en faisant un clic droit sur **Connexion Ethernet** et en sélectionnant **Ajouter une connexion au serveur**, puis choisissez **Connexion S7**.

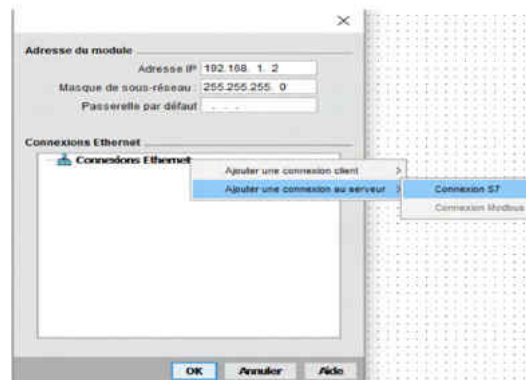


FIGURE 4.36 – Liaison client/Serveur



- c Configurez les paramètres de TSAP du client et du serveur. Par défaut, le TSAP du serveur est **20.00**, et vous pouvez choisir votre propre valeur pour le TSAP client, qui dans notre cas est **10.00**.

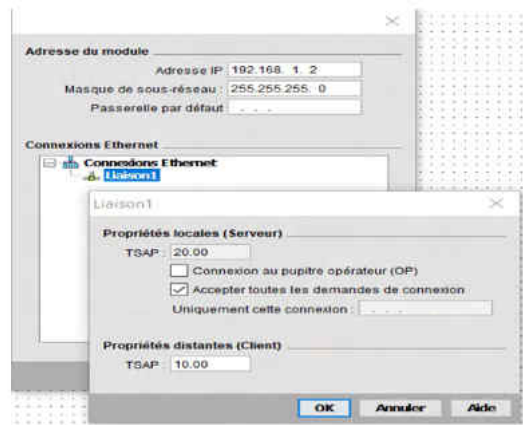


FIGURE 4.37 – Paramètres de connexion client/serveur

## 2. Créer un réseau client/serveur

- a Sélectionnez **Projet réseau** (où vous avez déjà effectué la configuration matérielle), puis faites un clic droit sur l'une des entrées réseau. Sélectionnez **Ajouter une connexion au serveur** et choisissez **Connexion S7**.



FIGURE 4.38 – Configuration réseaux du client/serveur

- b Cochez la case **Accepter toutes les demandes de connexion sur le serveur** et saisissez la valeur du TSAP du client, qui a déjà été configurée à l'étape 1 (c'est toujours **10.00**), dans la fenêtre qui s'affiche.

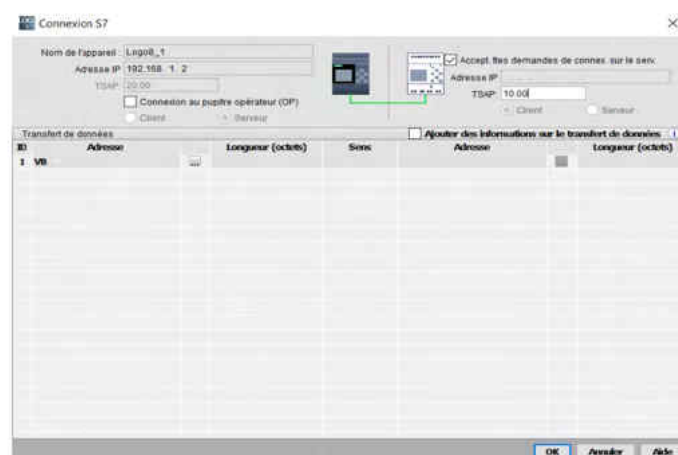


FIGURE 4.39 – Connexion client avec le serveur

Après avoir terminé ces étapes, un réseau sera créé, permettant ainsi la communication entre l'automate LOGO! et EasyBuilder Pro.

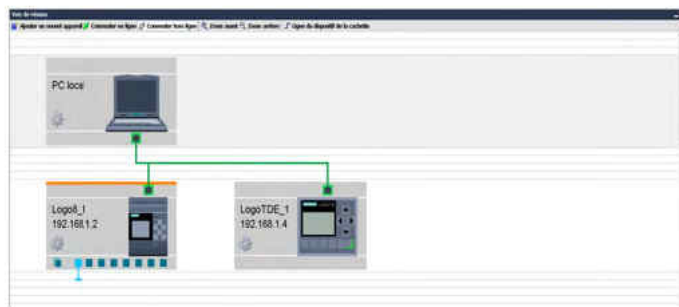


FIGURE 4.40 – Réseaux client/serveur

### 4.4.3 Configuration des paramètres réseau dans EasyBuilder Pro

Pour configurer les paramètres réseau dans EasyBuilder afin de permettre la communication avec le module LOGO! :

1. Accédez aux paramètres du système.
2. Sélectionnez **Siemens LOGO (Ethernet)** dans la liste des types de modules.
3. Accédez aux paramètres et saisissez l'adresse IP de l'automate (192.168.1.2) ainsi que les paramètres TSAP du client et du serveur qui ont déjà été configurés dans LOGO!Soft Comfort ( TSAP client =1000 et TSAP serveur =2000 ).

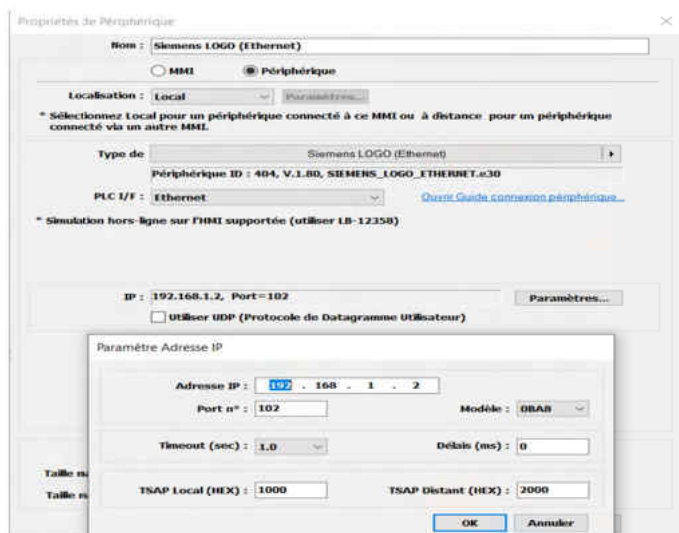


FIGURE 4.41 – Configuration des paramètres réseaux EasyBuilder Pro

### 4.4.4 Comparaison entre la supervision Zelio Logic et LOGO!

Dans le cas la supervision avec EasyBuilder Pro dans le chapitre 2, nous étions limités en termes du nombre de variables pouvant être utilisées et supervisées. Cependant, avec LOGO!, il n'y a pas de telles limitations. LOGO! offre une capacité plus étendue pour la gestion des variables, nous permettant ainsi de travailler avec un nombre plus important de variables et de les superviser de manière plus complète.

Type d'adresse	Bit/Mot	Format adresse	Adress...	Adre...	Partagé de	Desc
AI	Mot	DD	16	1		
AQ	Mot	DD	16	1		
AM	Mot	DD	64	1		
RA1	Mot	DD	64	1		
RAQ	Mot	DD	32	1		
VW	Mot	DDDD	1468	0		
VD	Mot	DDDD	1466	0		
VB	Mot	DDDD	1469	0		
RTC	Mot	D	7	1		
I	Bit	DD	64	1		
Q	Bit	DD	64	1		
M	Bit	DDD	112	1		
HI	Bit	DDD	128	1		
HQ	Bit	DDD	128	1		
V	Bit	DDDDo	14697	0		

FIGURE 4.42 – Variable lire/écrire dans Easybuilder Pro

Une autre différence notable entre les deux systèmes de supervision réside dans la configuration des valeurs de temporisations et des compteurs. Dans le cas de la supervision Zelio Logic, nous avons la possibilité de configurer ces valeurs directement via l'interface utilisateur. Cependant, avec LOGO!, cette possibilité n'est pas disponible en raison des caractéristiques des blocs fonctionnels dans LOGO!. Pour configurer ces blocs dans LOGO!, il est nécessaire d'accéder à leurs paramètres spécifiques, ce qui rend impossible de les modifier directement via l'interface utilisateur.

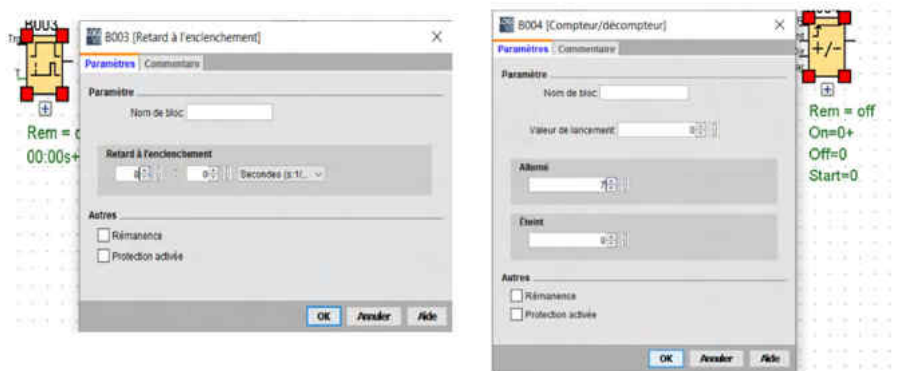


FIGURE 4.43 – Paramètres des timers et des compteurs LOGO!Soft

La vue système de LOGO! est illustrée ci-dessous :

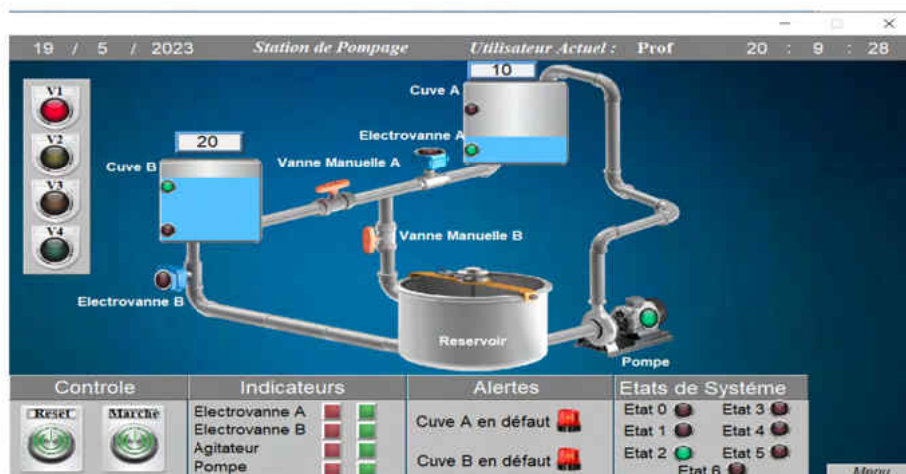


FIGURE 4.44 – Vue système LOGO!

## 4.5 Commande et la supervision avec Easybuilder Pro et Zelio Logic et LOGO!

La commande et la supervision avec EasyBuilder Pro dans cette partie diffèrent de celles des autres, où nous avons réalisé la commande et la supervision des automates Zelio Logic et LOGO! indépendamment. Nous avons remarqué que chaque IHM de chaque automate présente des différences en termes de commande et de supervision en raison des caractéristiques de communication propres à chaque automate avec EasyBuilder Pro, ainsi que des spécificités de leurs paramètres fonctionnels.

Dans cette partie, nous allons réaliser la commande et la supervision de la station de pompage en utilisant simultanément les deux automates avec EasyBuilder Pro. Chaque automate sera chargé d'une tâche spécifique, comme expliqué dans le 2<sup>ème</sup> cahier de charge dans chapitre 1.

Les tâches sont réparties entre les automates en fonction de leurs caractéristiques, tel que : l'automate LOGO! est chargé du remplissage des cuves A et B avec de l'eau. Il est également responsable de contrôler la marche et l'arrêt de la pompe et de l'électrovanne A. En revanche, l'automate Zelio Logic est chargé de contrôler la marche et l'arrêt de l'agitateur et de l'électrovanne B, ainsi que des valeurs des temporisations t1 et t2.

Grâce à EasyBuilder Pro, nous avons pu communiquer simultanément avec les deux automates et effectuer des opérations de lecture et d'écriture.

Les vues de l'historique des alertes, de l'historique de la gestion des utilisateurs et des tendances des variations du niveau de l'eau dans les cuves restent les mêmes que dans les parties précédentes. De plus, la gestion des utilisateurs reste identique.

Cependant, dans cette partie, nous avons effectué quelques modifications au niveau des vues du système et du Grafset. afin de répondre aux exigences du cahier des charges.

Dans la vue du système, nous avons apporté des ajustements pour mieux correspondre aux spécifications requises. Par exemple, nous avons ajouté la possibilité de régler les temporisations et d'afficher en temps réel les valeurs correspondantes. Ces modifications nous permettent de suivre plus précisément notre système.

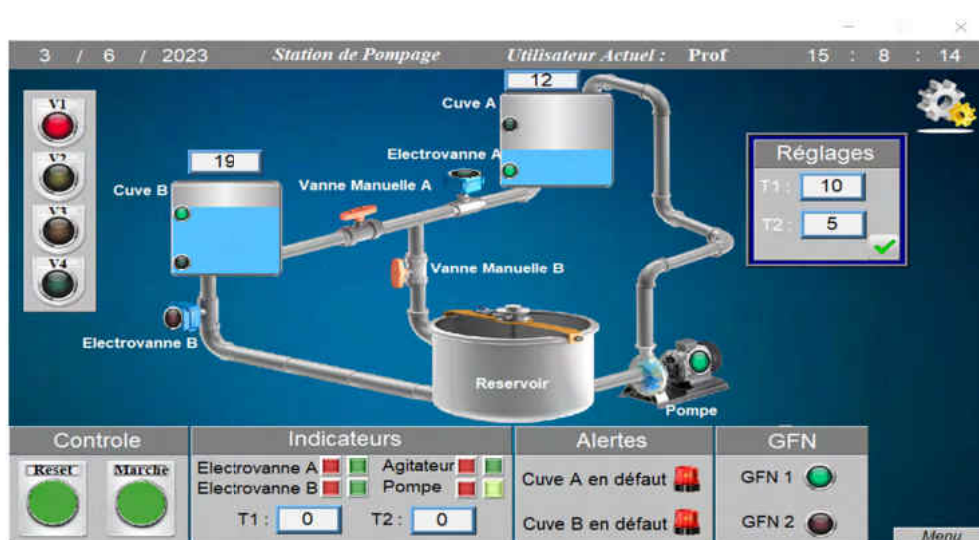


FIGURE 4.45 – Vue Système

Dans la vue Grafset, nous avons développé une vue spécifique à cette partie. Cette vue nous

permet de suivre les états des Grafquets GS, GC, GFN1 et GFN2. Ainsi, nous pouvons visualiser et analyser le fonctionnement et la progression des différents Grafquets dans notre système.

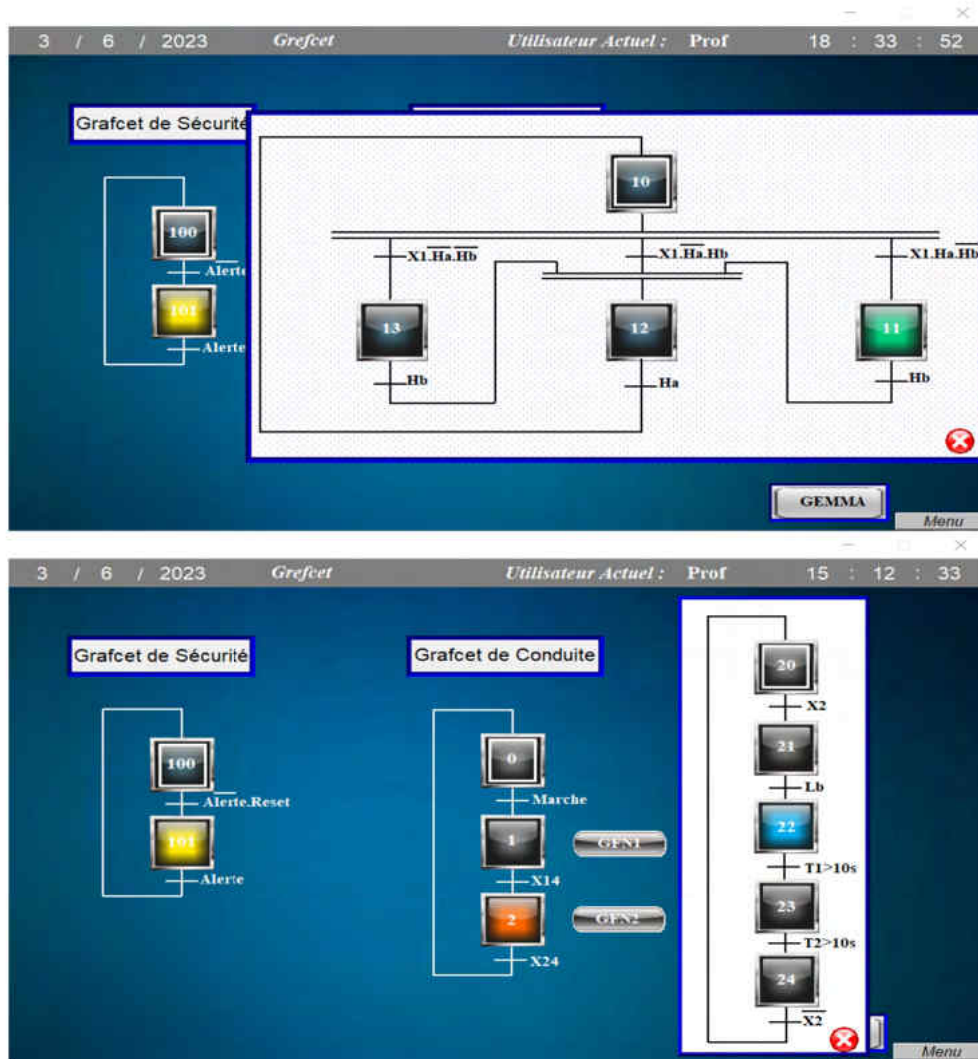


FIGURE 4.46 – Vue Grafcet

## 4.6 Guide d'Etude de Mode de Marche et d'Arret GEMMA

Dans le premier cahier des charges, la forme du Grafcet était réduite à son expression la plus simple. Il a permis de décrire le fonctionnement du système en production normale, ce qu'on appelle le **grafcet de fonctionnement normal GFN**.

Cependant, les systèmes automatisés rencontrent plusieurs situations en dehors de la situation de production normale, telles que les arrêts d'urgence causés par des pannes au niveau du processus ou des défaillances du système, le démarrage de la préparation du système avant le démarrage du processus de production, la vérification du fonctionnement du système étape par étape, etc.

Ces situations peuvent être résumées dans un seul outil graphique appelé le **e Guide d'Etude des Modes de Marche et d'Arrêt GEMMA**.

Le GEMMA est un outil graphique d'aide complémentaire au GRAFCET, développé par l'**Agence pour le Développement de la Productique Appliquée ADEPA** [1], en avril 1981. Il permet de définir les modes de marche et d'arrêt du système automatisé en décrivant tous les états du système,

ainsi que les transitions entre ces états. De plus, il permet de détailler les procédures à mettre en œuvre après l'analyse d'une défaillance ou d'un défaut de fonctionnement.

Le GEMMA est rempli progressivement lors de la conception du système.

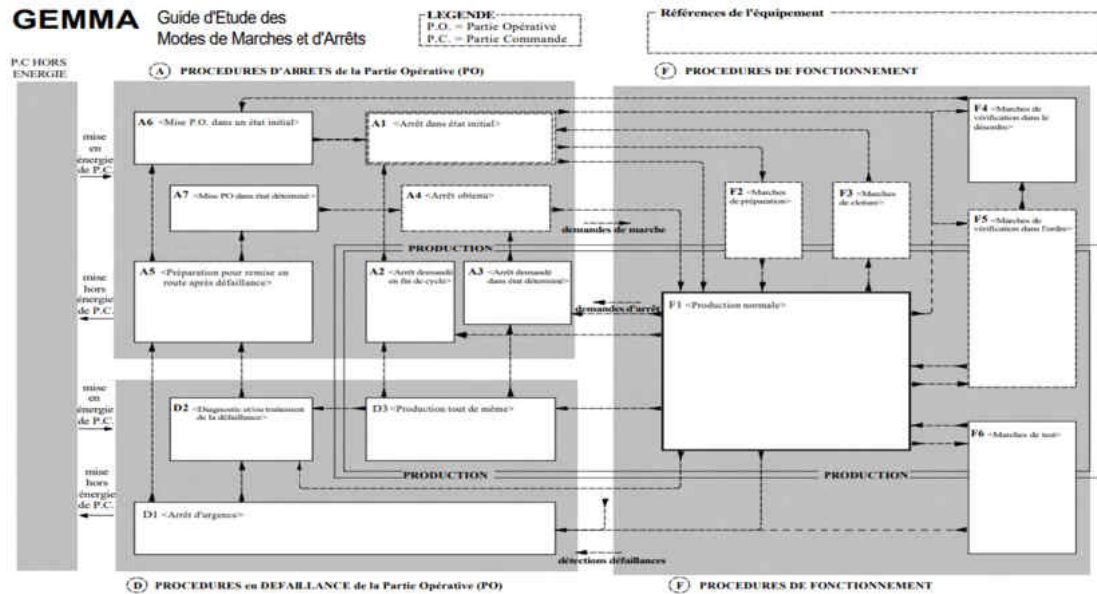


FIGURE 4.47 – Feuille de GEMMA

#### 4.6.1 Le GEMMA de la station de Pompage

Le GEMMA dispose plusieurs procédures, ce qui n'est pas obligatoire d'utiliser toutes les procédures dans le GEMMA. La sélection et l'utilisation des procédures dépendent des besoins spécifiques du système ainsi que des objectifs de fonctionnement. Chaque système peut nécessiter l'adaptation et la personnalisation des procédures du GEMMA pour répondre aux exigences spécifiques (Voir l'Annexe G ).Le GEMMA de la station de pompage comprend les procédures suivante :

- **A1** : Système en arrêt dans l'état initial.
- **A2** : Système en fin de cycle.
- **A5** : Vider les cuves A et B, ouvrir la vanne manuelle A et fermer la vanne manuelle B.
- **A6** : Condition initiale vérifier.
- **F1** : Mise en service du système selon le 2<sup>ème</sup> cahier des charges.
- **F4** : Activer simultanément la pompe, les électrovannes A et B, ainsi que l'agitateur pour vérifier leur fonctionnement avant démarrer le système, en cliquant sur le bouton Z1 du module Zelio Logic.
- **D1** : L'eau dans la cuve A et/ ou la cuve B a dépassé la limite maximale prédéfinie.
- **D2** : L'affichage de l'heure, la date et la raison de l'arrêt.

La feuille du GEMMA de notre station de pompage est montrée ci-dessous :

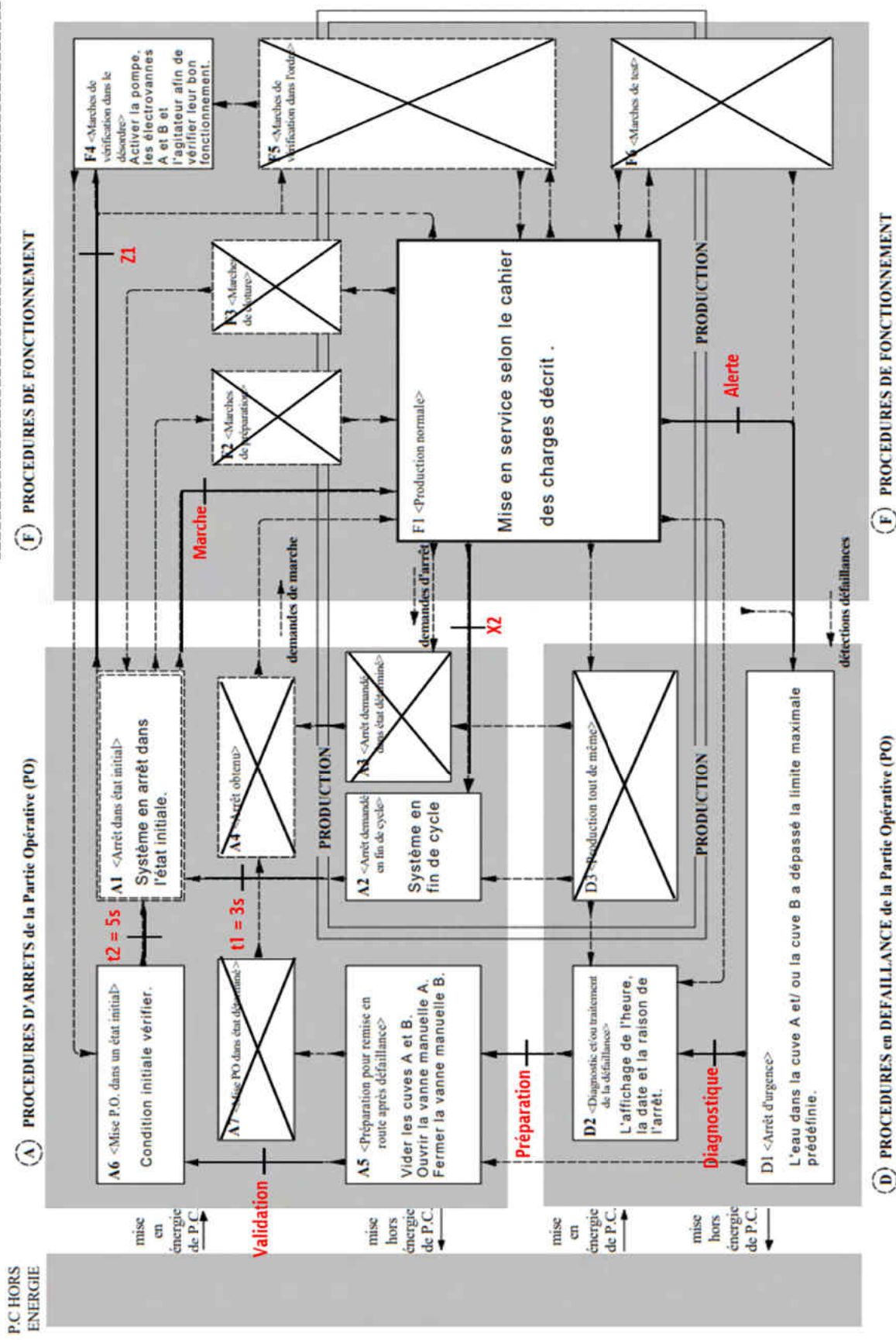


FIGURE 4.48 – GEMMA de la station de pompage

## 4.6.2 La commande et supervision de GEMMA dans Easybuilder Pro

Avec Easybuilder Pro, il est possible de superviser les différents états et commander le GEMMA comme montré ci-dessous

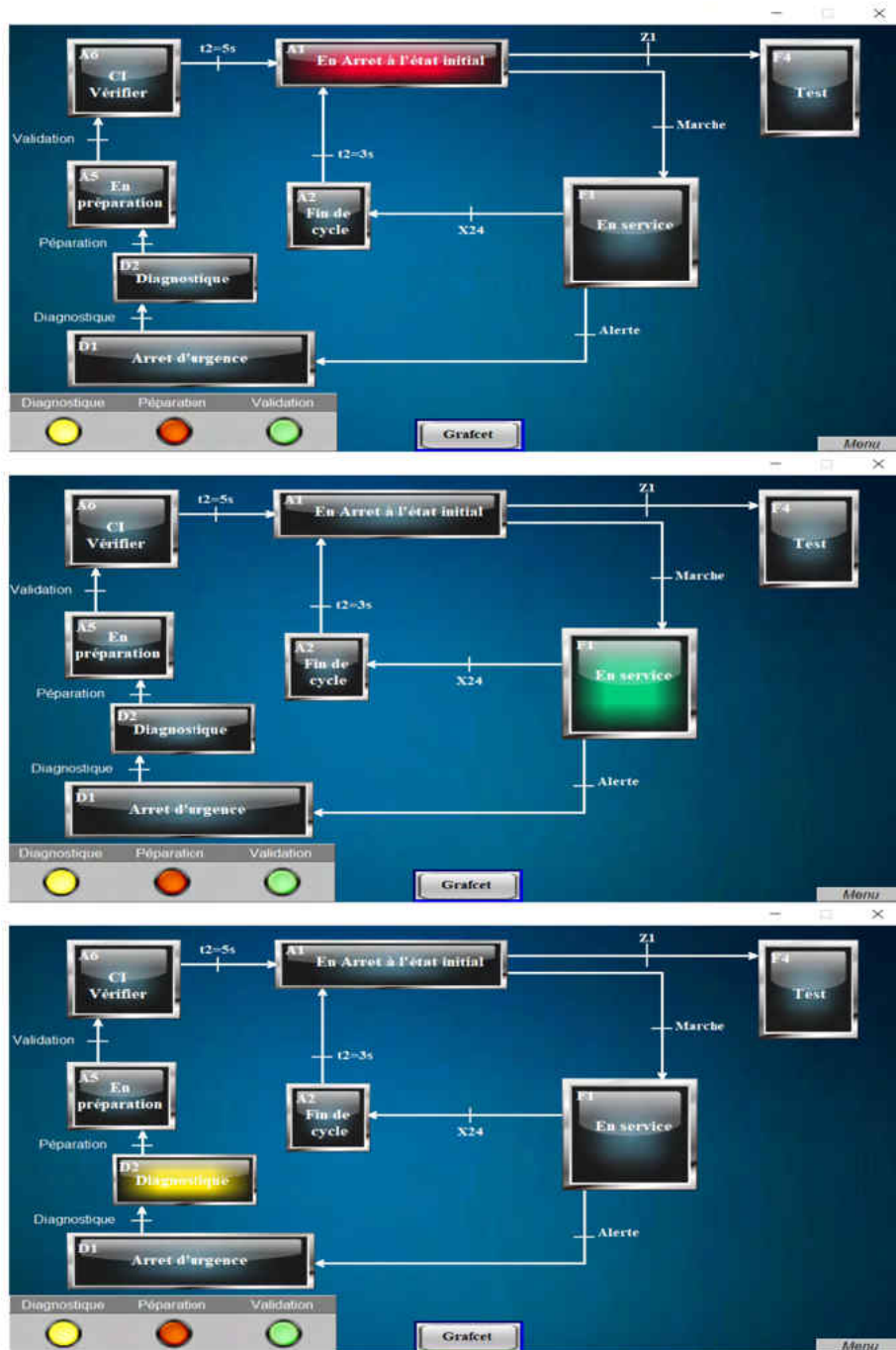


FIGURE 4.49 – Commande et supervision de GEMMA avec EasyBuilder Pro

## 4.7 Systèmes de Contrôle Industriel

Jusqu'à présent, nous avons mis en œuvre plusieurs techniques de commande et de supervision, regroupées sous le terme de **Systèmes de Contrôle Industriel** (ICS - Industrial Control Systems). Ces systèmes sont classés par niveau en fonction de leur complexité, de leurs avantages et de leurs besoins. À travers les chapitres et les parties précédentes, nous avons réalisé trois niveaux d'ICS :



- **Niveau 1 : Système de contrôle avec API (PLC - Programmable Logic Controller) :** Ce système permet un contrôle spécifique des tâches. Nous l'avons utilisé dans le chapitre 2 pour la commande et la supervision de la station avec l'automate Zelio Logic, conformément au premier cahier des charges.

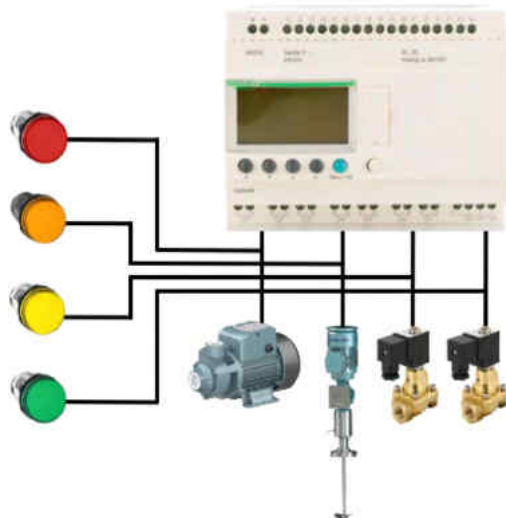


FIGURE 4.50 – Système de contrôle avec API

- **Niveau 2 : Système IHM (Interface Homme-Machine) :** Ce système fournit une interface permettant aux utilisateurs d'interagir avec le processus. Il fournit une présentation visuelle des informations qui facilite le contrôle, la surveillance et la gestion des processus. L'IHM peut utiliser un écran tactile, une page web ou un module LCD, comme nous l'avons vu dans le chapitre 3 pour la commande et la supervision de la station avec l'automate LOGO!, l'afficheur LOGO! TDE, le serveur web et Excel. De plus, nous avons également utilisé EasyBuilder Pro avec l'automate Zelio Logic et LOGO!, conformément au premier cahier de charge.

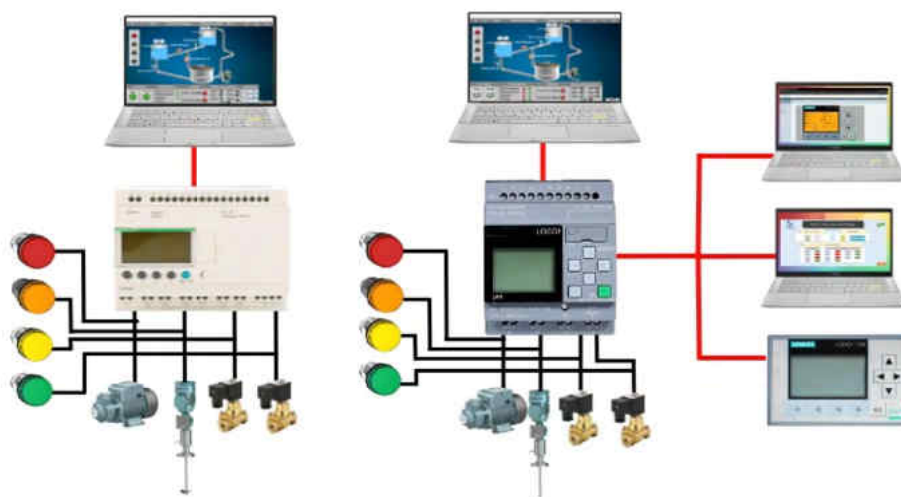


FIGURE 4.51 – Système de contrôle avec IHM

- **Niveau 3 : Système SCADA (Supervisory Control And Data Acquisition) :** Ce système offre une interface étendue par rapport à l'IHM. Il permet le contrôle, la supervision et l'acquisition d'informations à partir de plusieurs API. L'IHM peut être un sous-système du SCADA. Dans ce chapitre, nous avons utilisé le système SCADA pour la commande de la station en utilisant les automates LOGO!, Zelio Logic et EasyBuilder Pro, conformément au deuxième cahier de charge.

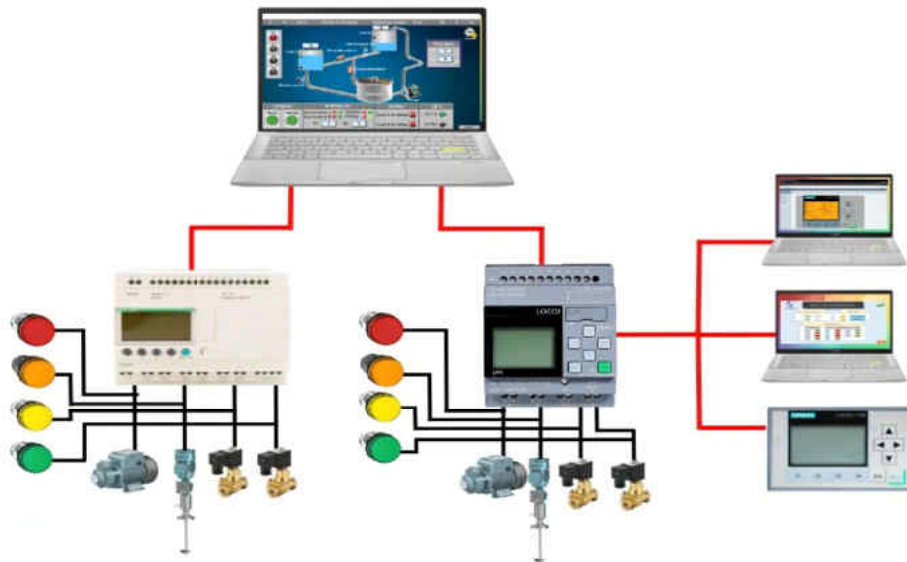


FIGURE 4.52 – Système de contrôle avec SCADA

Il existe un autre système de niveau supérieur appelé **DCS** (Distributed Control System).

Le DCS est un système de contrôle distribué utilisé pour gérer et contrôler de manière centralisée des processus complexes dans les installations industrielles. Contrairement au PLC, qui est souvent utilisé pour une tâche spécifique, le DCS est conçu pour gérer l'ensemble du processus de production. Il est composé de plusieurs unités de contrôle interconnectées qui communiquent entre elles pour superviser et contrôler les différentes parties du système.

Afin de mettre en place ce système, nous établirons une communication entre notre station de pompage et un autre prototype de pompage, ce qui nous permettra de gérer un autre système.

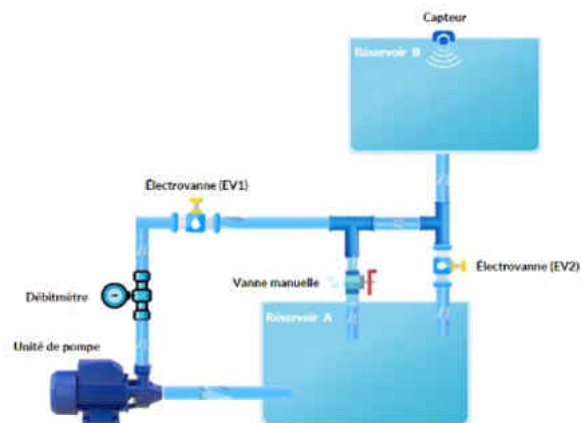


FIGURE 4.53 – Prototype de pompage

L'objectif de ce prototype est de réaliser le contrôle du niveau d'eau dans le réservoir A en boucle fermée (BF), de sorte que le niveau d'eau suive une consigne prédéfinie. Afin d'atteindre cet objectif, le système doit répondre au cahier de charge suivant :

Lorsque le système est à l'arrêt, en cliquant sur le bouton de démarrage, deux scénarios peuvent

se produire :

1. **Remplissage** : Si la consigne définie est supérieure au niveau H de l'eau dans le réservoir A, le voyant de remplissage s'allume pour indiquer que le système est en mode de remplissage. La pompe et l'électrovanne 1 (EV1) s'activent pour amener l'eau dans le réservoir A jusqu'à ce que la consigne devienne inférieure au niveau H de l'eau. L'opération de remplissage s'arrête alors et le système revient à son état initial.
2. **Vidange** : Si la consigne est inférieure au niveau H du réservoir, le voyant de vidange s'allume pour indiquer que le système est en mode de vidange. L'électrovanne 2 (EV2) s'ouvre pour évacuer l'eau vers le réservoir B jusqu'à ce que la consigne devienne supérieure au niveau H. L'opération de vidange s'arrête alors et le système revient à son état initial.

Si le système est en état de marche et que le bouton d'arrêt d'urgence est enclenché, le système s'arrête immédiatement.

Il est à noter que la consigne que le niveau d'eau doit suivre est calculée selon la relation suivante : Tolérance \* Référence. Ces valeurs sont définies par l'utilisateur à travers l'IHM. Le Grafcet de ce système est présenté ci-dessous :

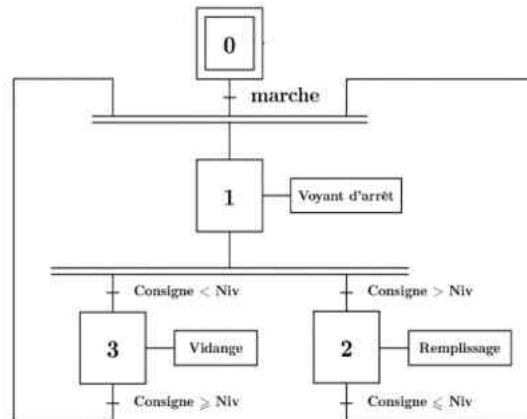


FIGURE 4.54 – Grafcet du prototype de pompe

Pour intégrer le prototype de pompe dans notre système DCS, nous devons :

1. Établir une communication entre l'automate LOGO! de notre station de pompage et l'automate S7 1200 du prototype de pompe afin d'échanger des informations entre eux.
2. Créer une nouvelle vue dans l'interface existante d'EasyBuilder Pro, qui répondre aux spécifications du deuxième cahier de charge. dans le cadre de la supervision et du contrôle du prototype de pompe.

#### 4.7.1 La communication entre LOGO! et S7 1200

L'objectif de cette communication est de réaliser les actions suivantes :

1. Permettre à l'automate LOGO! d'envoyer les consignes de **Marche** et **Reset** du prototype de pompe à l'automate S7 1200, en utilisant les boutons correspondants de la station de pompage.

- Permettre à l'automate LOGO! de recevoir les activations et désactivations des signaux X0, X1, X2 et X3 du grafcet du prototype de pompe, ainsi que les états d'arrêt et de marche des voyants, de la pompe et des deux électrovannes 1 et 2 à partir de l'automate S7 1200.
- Afficher ces informations reçues dans l'interface EasyBuilder afin de les visualiser et de superviser de manière claire et conviviale.

La figure ci-dessous illustre et clarifie l'échange entre l'automate LOGO! et S7 1200 :

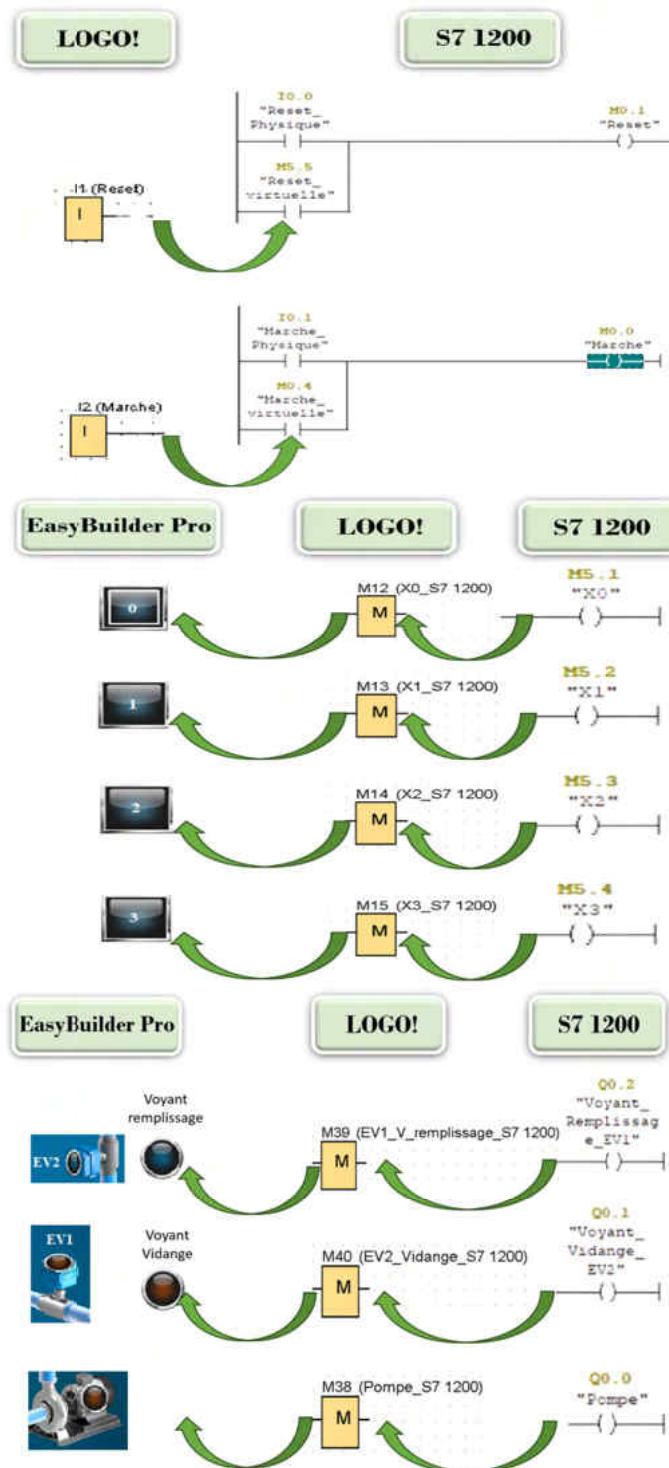


FIGURE 4.55 – Communication entre LOGO! et S7 1200

Cette communication est réalisée à travers KepserverEx en suivant les étapes suivantes :

1. Création d'un nouveau canal **Siemens TCP/IP Ethernet**.
2. Sélection d'un nouveau appareil **S7 1200** avec l'adresse IP **192.168.0.1**.
3. Déclaration des variables souhaitées à lire ou à écrire dans S7 1200.

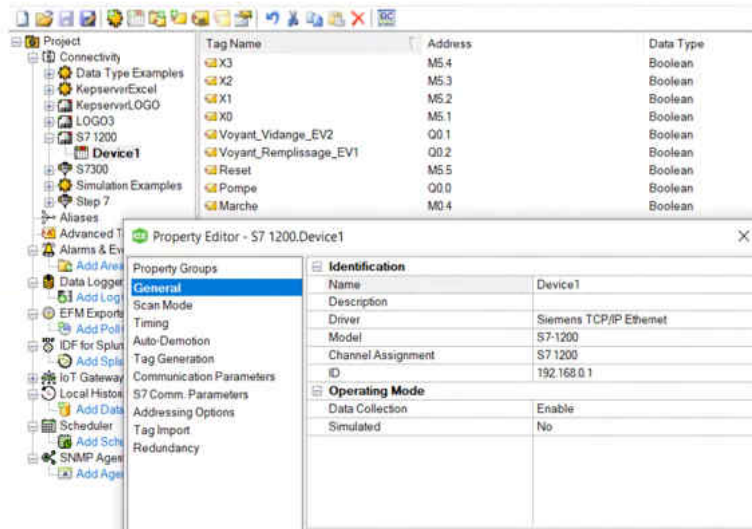


FIGURE 4.56 – Communication entre KepserverEx et S7 1200

4. Lier des variables correspondantes entre LOGO! et S7 1200, comme expliqué précédemment.

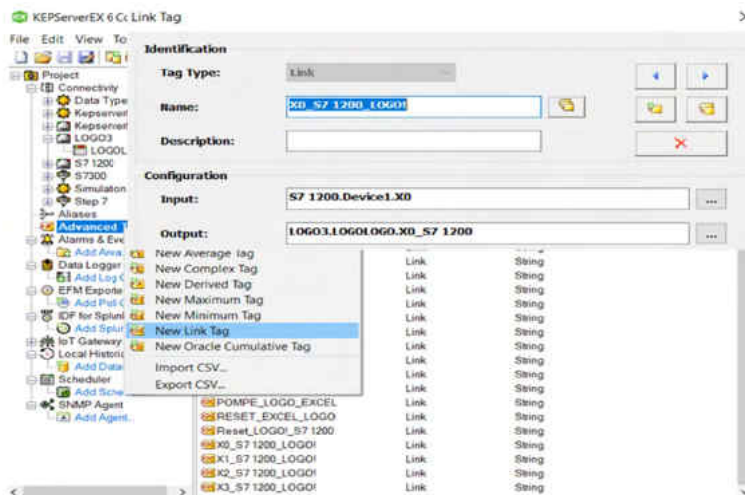


FIGURE 4.57 – Lien des variables de LOGO! et S7 1200

**NB :** Dans cette partie, nous avons uniquement relié les variables TOR, car les variables analogiques de l'automate LOGO! et de l'automate S7 1200 ne sont pas compatibles. Par conséquent, pour lire la variable analogique de S7 1200 (par exemple, le capteur de niveau dans la cuve A), nous devons communiquer directement avec EasyBuilder Pro.

#### 4.7.2 La communication entre Easybuilder Pro et S7 1200

Afin de commander et superviser les variables analogiques du prototype de pompage, telles que le niveau de l'eau dans le réservoir A et la saisie des valeurs de tolérance et de référence, nous devons communiquer directement entre l'automate S7 1200 et EasyBuilder Pro.

Pour établir cette communication, nous suivons les mêmes étapes précédentes utilisées pour la communication entre LOGO! et EasyBuilder Pro. Cependant, cette fois ci, nous sélectionnons le périphérique  $S71200/S71500_1$  et saisissons l'adresse IP **192.168.0.1**.

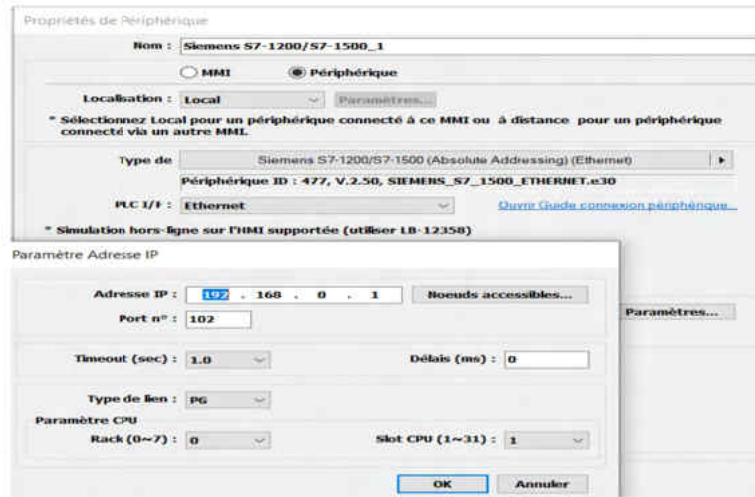


FIGURE 4.58 – Communication entre Easybuilder Pro et S7 1200

La vue réalisée du prototype de pompage dans EasyBuilder est présentée ci-dessous :

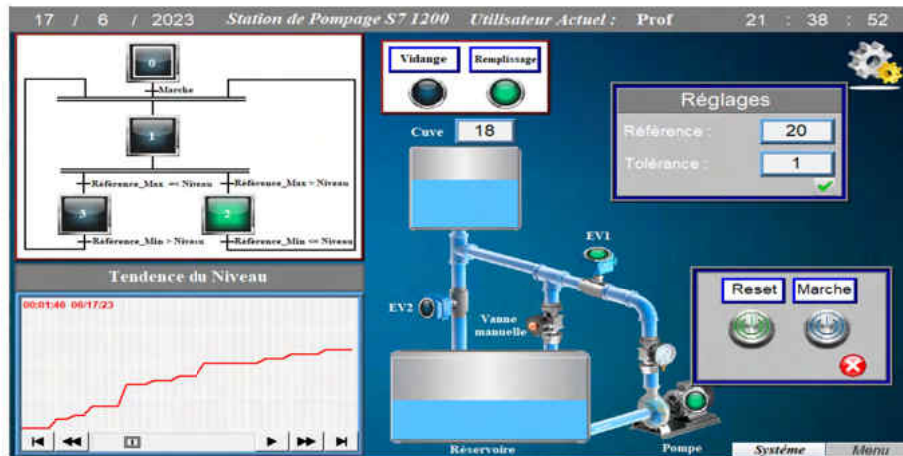


FIGURE 4.59 – Vue S7 1200 sur Easybuilder Pro

Il est également possible de superviser les alarmes du prototype de pompage, comme illustré ci-dessous :

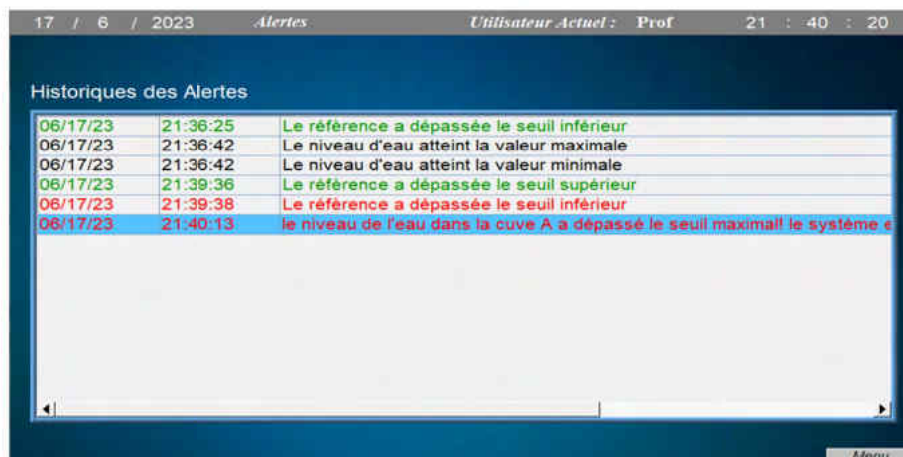


FIGURE 4.60 – Alarmes S7 1200 sur EasyBuilder Pro

Enfin, le système DCS que nous avons réalisé est présenté ci-dessous :

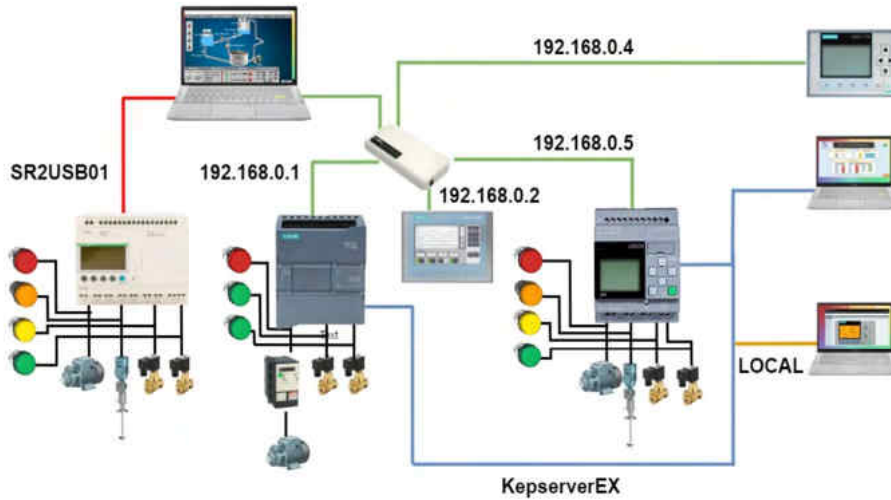


FIGURE 4.61 – système DCS

Il est à noter que les automates LOGO! et S7 1200, l'IHM du prototype de pompage et l'afficheur doivent être dans le même réseau afin d'établir le réseau DCS. Pour cela nous devons changer les adresses IP de l'automate LOGO! et LOGO! TDE. Les nouvelles adresses IP de LOGO! et LOGO! TDE respectivement sont : **192.168.0.5**, **192.168.0.4**. Ces équipements sont reliés à travers un switch.

## 4.8 Conclusion

À travers ce chapitre, nous avons exploré une nouvelle technique de commande et de supervision avec le logiciel EasyBuilder Pro. Grâce à cette méthode, nous avons été en mesure de réaliser différents systèmes de contrôle industriel, tels que le DCS (Système de Contrôle Distribué), le SCADA (Supervision, Contrôle et Acquisition de Données) et l'IHM (Interface Homme-Machine). Nous avons également abordé de nouvelles notions telles que le GEMMA (Guide d'Étude de Mode de Marche et d'Arrêt). De plus, nous avons appris des notions sur la communication entre l'automate et EasyBuilder, ainsi que la communication entre automates.

# Chapitre 5

## Conception de l'armoire de la station de pompage

### 5.1 Introduction

Ce chapitre se concentre sur le développement du câblage électrique de la station de pompage afin de résoudre les problèmes existants, tels que l'alimentation des automates et le câblage des équipements. Nous mettrons en œuvre une nouvelle conception en 3D de l'armoire électrique et créerons un schéma électrique amélioré. L'objectif principal est d'assurer la sécurité des personnes et des équipements dans la station. Parallèlement, nous profiterons de cette opportunité pour ajouter de nouveaux équipements électriques qui amélioreront les fonctionnalités et l'utilisation globale de la station de pompage.

### 5.2 Séparation de l'alimentation électrique des automates programmables

Lors de l'étude des cahiers des charges, nous avons remarqués que les automates s'activaient simultanément lorsque la station était alimentée. Cela n'a pas posé de problème lors de la manipulation du deuxième cahier des charges, car nous utilisons les deux automates en même temps. Cependant, pour le premier cahier des charges, cela posait un problème car il était nécessaire de vérifier si l'un des automates était en mode **STOP** avant de procéder à la manipulation. Par exemple, lors de l'utilisation de l'automate Zelio Logic pour la manipulation, il fallait s'assurer que l'automate LOGO! était en mode **STOP**, et vice versa. Le problème était que cette vérification n'était pas pratique, et parfois les utilisateurs pouvaient oublier de mettre l'automate correspondant en mode d'arrêt, ce qui pouvait entraîner des échecs dans la manipulation. Il est donc nécessaire de trouver une solution plus pratique pour éviter ce problème et garantir le bon fonctionnement des automates.

La solution pour résoudre ce problème consiste à ajouter deux sélecteurs à trois positions (une position d'arrêt, les deux autres position pour sélectionner un des deux choix). Le premier sélecteur permettra de choisir si nous travaillons avec un ou deux automates. En position **un automate**, nous pourrons sélectionner soit Zelio Logic, soit LOGO! à l'aide du deuxième sélecteur. Cela éliminera le besoin de vérifier manuellement si l'automate correspondant est en mode **STOP**, simplifiant ainsi le processus et réduisant les risques d'erreur et offre une plus grande flexibilité et facilite la manipulation de la station de pompage.



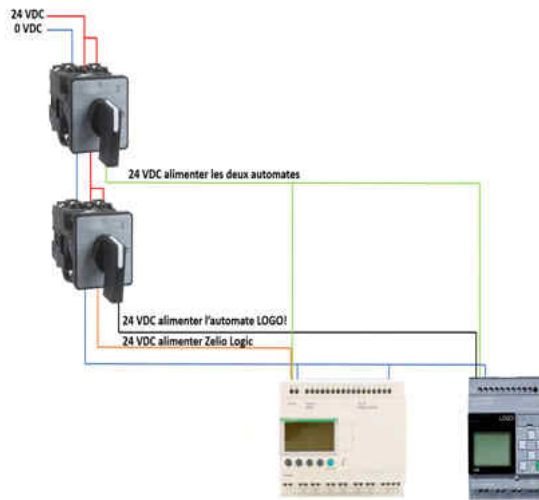


FIGURE 5.1 – Schéma synoptique de séparation de l'alimentation électrique des APIs

Le schéma électrique correspondant à cette solution est présenté ci-dessous :

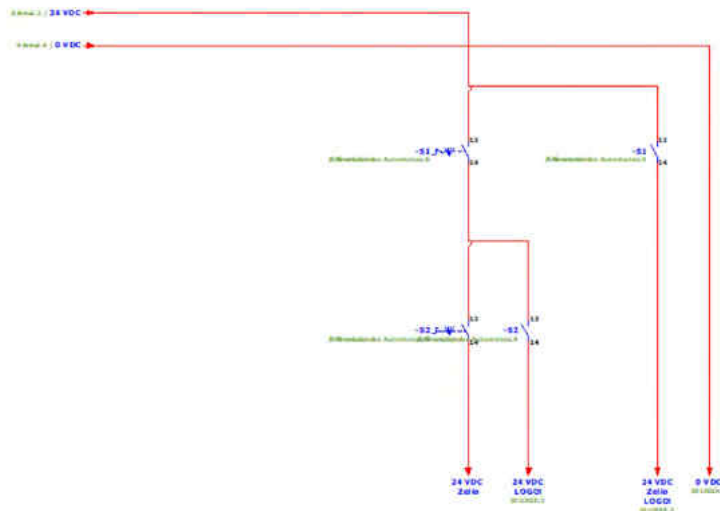


FIGURE 5.2 – Schéma électrique de séparation de l'alimentation électrique des APIs

### 5.3 L'ajout d'équipements électriques à la station de pompage

Afin d'améliorer les fonctionnalités de l'armoire, nous proposons d'ajouter les équipements suivants en plus des deux sélecteurs utilisés pour séparer l'alimentation des deux automates :

- **Un voyant 2 A blanc de 220 V pour indiquer la présence de tension** : ce voyant fournira une indication visuelle claire pour savoir si la tension est présente ou non.
- **Un fusible 2 A 220 V pour protéger le voyant** : ce fusible garantira la protection du voyant contre les surtensions et les courts-circuits, assurant ainsi sa sécurité et sa durabilité.
- **Un bouton d'arrêt d'urgence 220 V** : ce bouton permettra d'interrompre immédiatement toutes les opérations en cas de situation d'urgence, assurant ainsi la sécurité des personnes et

des équipements.

- **Un bouton poussoir normalement fermé de couleur rouge** : ce bouton pourra être utilisé comme un bouton d'arrêt, permettant de stopper les opérations rapidement lorsque nécessaire.

Le coût des équipements proposé est détaillé dans le tableau ci-dessous :

Produit	Prix
Voyant blanc	Entre 600 da et 700 da
bouton poussoir NC Rouge	Entre 600 da et 700 da
Douton d'arrêt d'urgence	Entre 1900 da et 2000 da
Fusible	Entre 680 da et 700 da
Port fusible	Entre 500 da et 600 da

TABLE 5.1 – Prix des équipements électriques

## 5.4 Schémas électrique de la station de pompage

Pour développer la station de pompage en termes de câblage, nous avons réalisés un schéma électrique détaillé et effectués la conception 3D de l'armoire proposée. Cette conception a été réalisée en respectant les normes afin de faciliter le câblage et d'obtenir une finition de qualité.

Nous avons également portés une attention particulière au choix de l'armoire. Nous avons opté pour une armoire avec un indice de protection IP66, ce qui signifie qu'elle est étanche à la poussière et protégée contre les jets d'eau puissants. Cette mesure vise à assurer la sécurité des personnes et la protection des équipements. Ainsi, nous créons un environnement sûr et fiable pour le fonctionnement optimal de la station de pompage.

La conception proposée de l'armoire pour améliorer la station, qui représente la disposition des équipements électriques, est présentée ci-dessous :

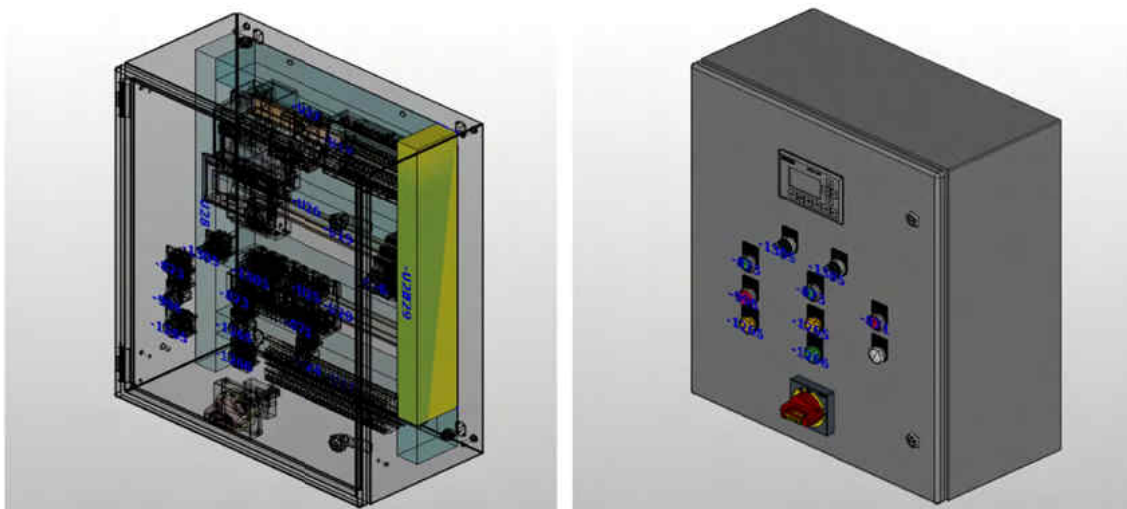


FIGURE 5.3 – Vue 3D de l'armoire électrique

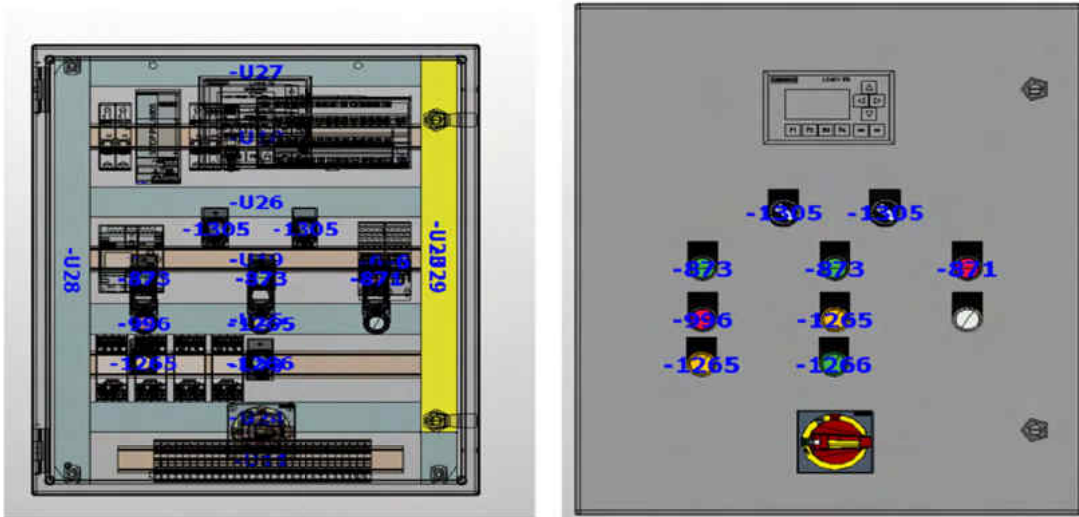


FIGURE 5.4 – Vue de face de l'armoire électrique

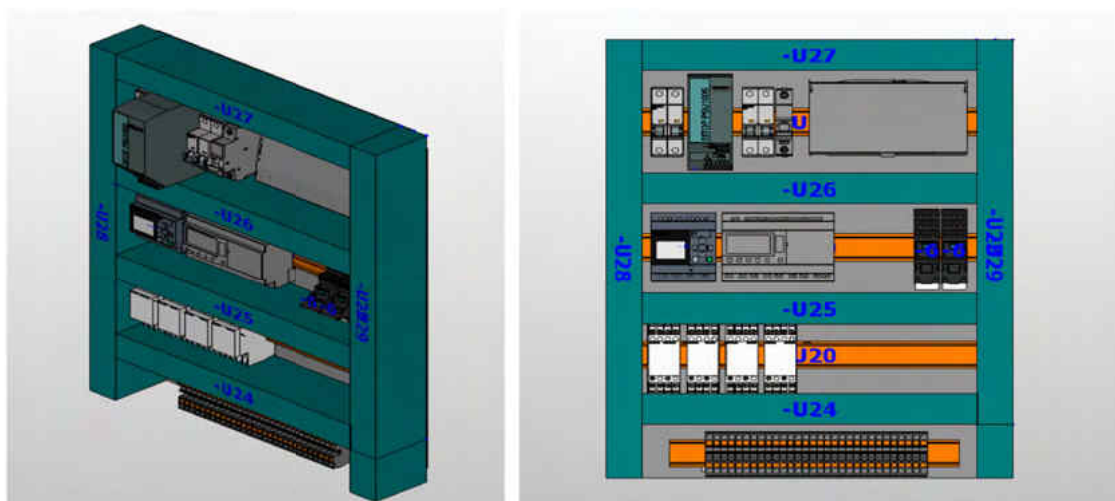




FIGURE 5.5 – Vue intérieur de l'armoire électrique

Le schémas électrique proposer pour la réalisation de l'armoire est montré ci-dessous :

0	1	2	3	4	5	6	7	8	9	
<p><b>CLIENT :</b></p>  <p>الجامعة الوطنية للتكنولوجيا Ecole Nationale Polytechnique</p>										
<p><b>CLIENT FINAL :</b></p>  <p>جامعة العلوم والتكنولوجيا Université de Sciences et de Technologie</p>										
<p><b>Ecole :</b></p> <p><b>Laboratoire de contrôle des processus</b> <b>Ecole nationale polytechnique d'alger</b></p>										
<p><b>SCHEMA :</b></p> <p><b>Schéma Multifilaire</b></p>										
<p><b>Projet :</b></p> <p><b>Station de Pompage</b></p>										
<p><b>CODE DE PROJET :</b> PFE_Station_de_Pompage_2023_KAMEL</p>										
<p><b>DESIGNATION DU SCHEMA :</b> SCHEMA MULTIFILAIRE DE L'ARMOIRE STATION DE POMPAGE</p>										
<p><b>DESIGNATION DU PROJET :</b> ARMOIRE STATION DE POMPAGE</p>										
<b>Date</b>	<b>Modifié par</b>	<b>Dessiné par</b>	<b>KAMEL Rania Fairouz</b>	<b>Approuvé par</b>						<b>FOLIO</b>
										<b>01</b>
										<b>14</b>



## CONDITION D'ENVIRONNEMENT

INSTALLATION  INTERIEURE  EXTERIEUR

INDICE DE PROTECTION DE L'ARM  IP42  IP54  IP66

TEMPERATURE AMBIANTE MIN  10°  15°  20°

TEMPERATURE AMBIANTE MAX  30°  35°  40°

HUMIDITE RELATIVE  50%  80%  90%

## ENTREE DE CABLES/AUXILIAIRE

Entrée de cable  BAS  HAUT

ACCES CABLE  AVANT  ARRIERE

PRESSE ETOUPE  OUI  NON  PLASTIQUE  METAL  AUTRE

ECLAIRAGE DE L'ARMOIRE  NON  OUI  FIN DE COURSE

VENTILATION DE L'ARMOIRE  FORCEE  NATURELLE

## ARMOIRE

TYPE D'ENVELOPPE  Acier  POLYSTER  INOX

DIMENSIONNEMENT DE L'ARMOIRE  
HAUTEUR = 600 mm  
LARGEUR = 500 mm  
PROFONDEUR = 250 mm

SOCLE ARMOIRE  OUI  NON

CHAUFFAGE ARMOIRE  OUI  NON

## DONNEES ELECTRIQUES

TENSION ASIGNEE  220V  400V  630V






















TENSION DE COMMANDE  24VDC  48VDC  220VDC  
 24VAC  48VAC  220VAC



REGIME DU NEUTRE  TT  TNS  TNC  IT

COURANT NOMINAL In = 630A

FREQUENCE ASSIGNEE  50Hz  60Hz

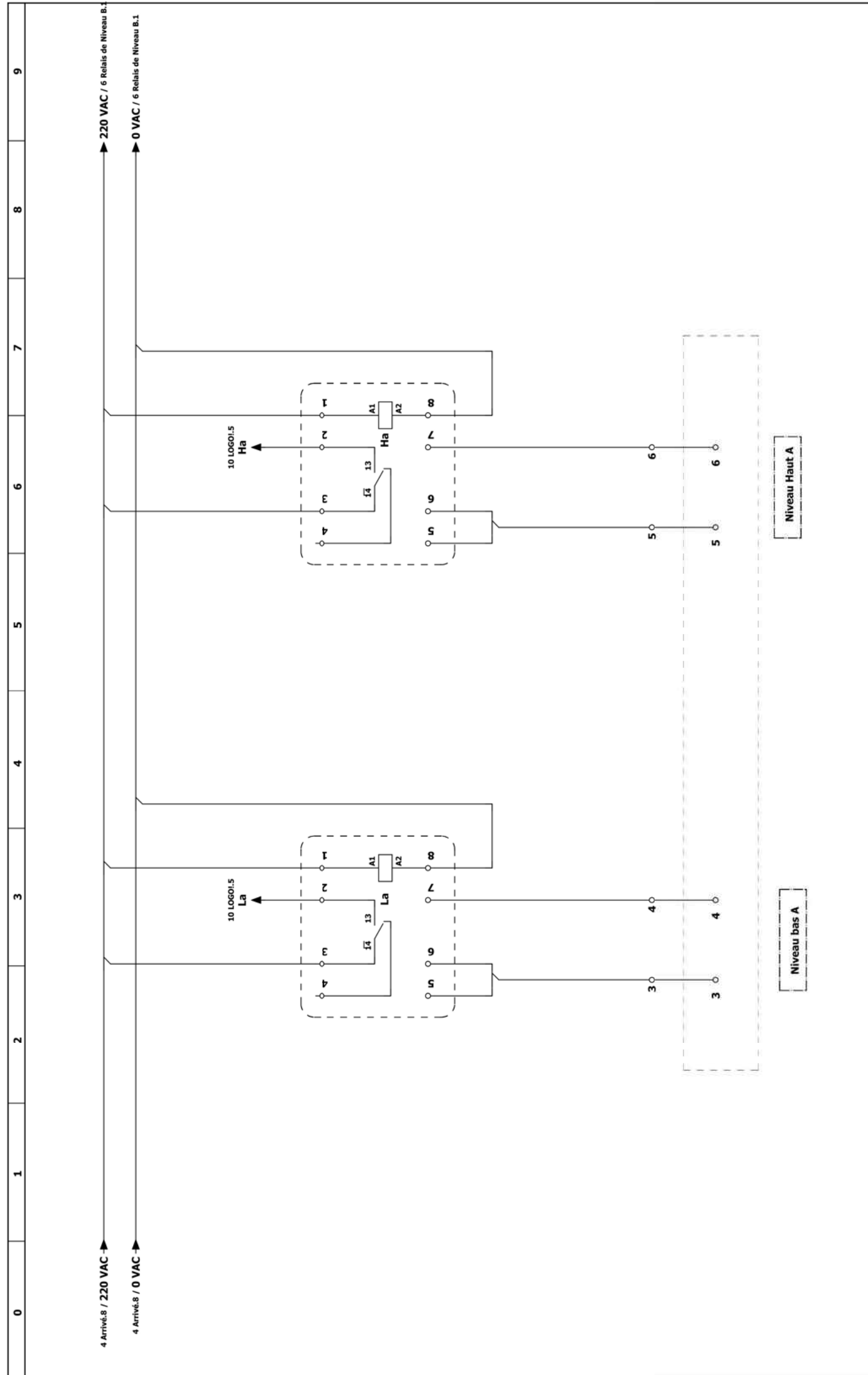
# IEC\_symbol

Electromechanical operating device, general / relay coil, general		Electromechanical operating device, with off delay		Electromechanical operating device, with pick-up delay	
Terminal with 1 connection points without saddle jumper connection points		Switch, NO contact, operation 3 Switching position		Miniature circuit-breaker, two-pole	
Terminal with 2 connection points without saddle jumper connection points		Solenoid valve, general		Emergency stop switch / Emergency stop pushbutton, NO contact, with pull-to-reset	
Ground, general		NC contact		Fuse circuit-breaker, single-pole	
Pushbutton, NC contact, operated by pushing		NC contact, with delayed opening		Change-over contact (2-path) with break point	
Emergency stop switch / Emergency stop pushbutton, NO contact		Rectifier, bridge, two-phase, secondary 2 connection points		DC motor, general	
Pushbutton, NO contact, operated by pushing		NO contact		Lamp / Indicator light, general	

 المدرسة الوطنية للتكنولوجيا Ecole Nationale Polytechnique	Date	Dessiné par	KAMEL Rania Fairouz	Approuvé par	DESIGNATION DU SCHEMA : <b>ARMOIRE STATION DE POMPAGE</b>	DESIGNATION DU FOLIO : <b>Liste des symboles</b>	 W.P. Wahab & Partners	FOLIO
					04	14		









4 Arrivé.8 / 220 VAC → 220 VAC / 6 Relais de Niveau B.1  
 4 Arrivé.8 / 0 VAC → 0 VAC / 6 Relais de Niveau B.1

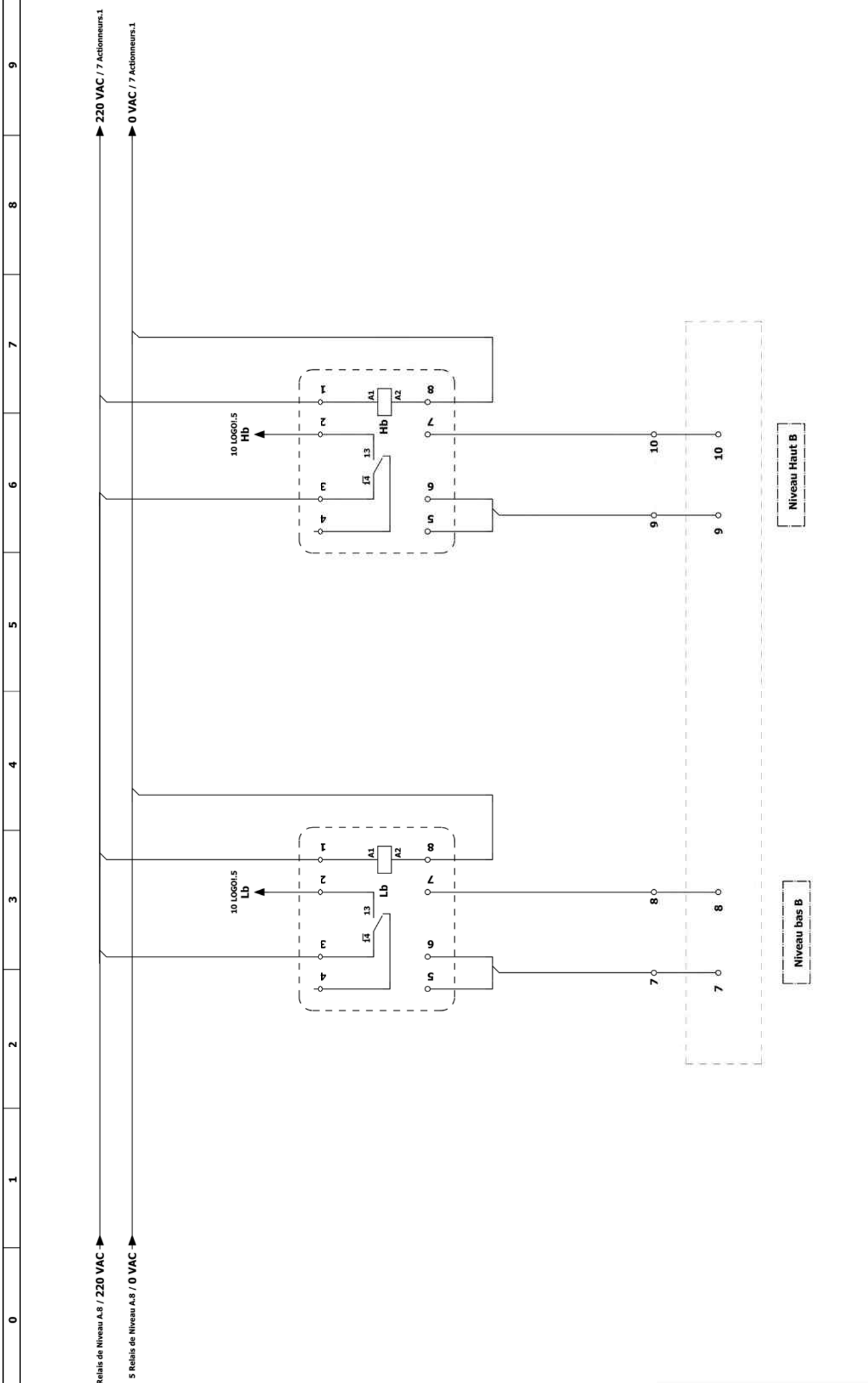
10 LOG01.5  
Ha



10 LOG01.5  
La

Niveau Haut A

Niveau bas A

 المدرسة الوطنية للتقنيات Ecole Nationale Polytechnique	Date	Dessiné par	KAMEL Rania Fairouz	Approuvé par	Relais de niveau A		FOLIO	06	14
					ARMOIRE STATION DE POMPAGE				

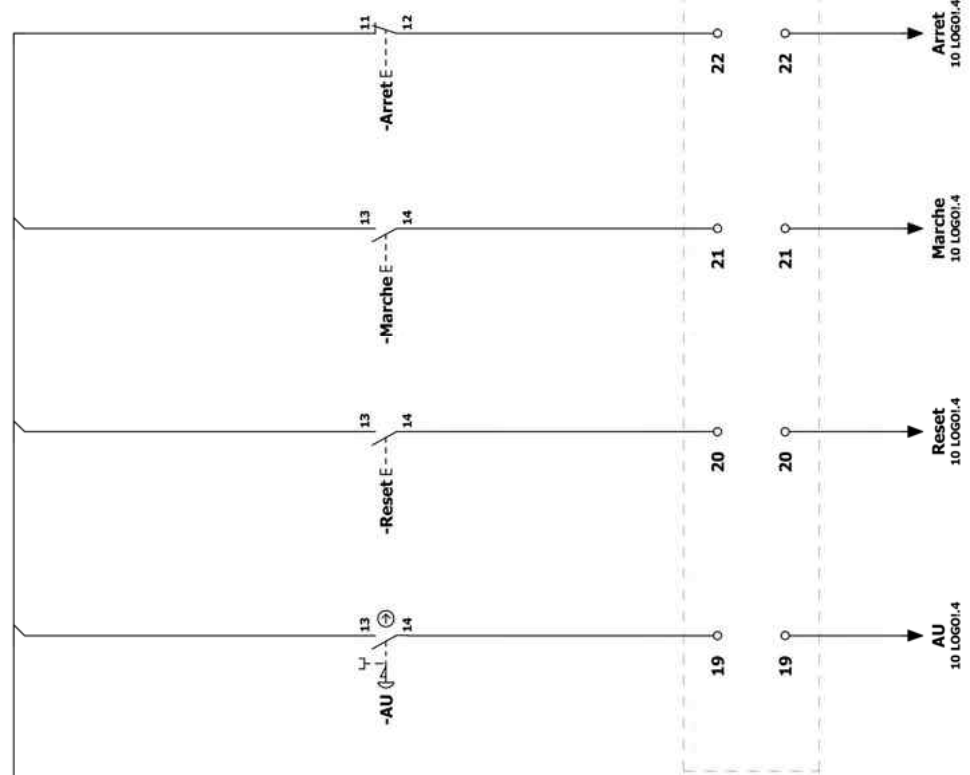


 المدرسة الوطنية للتكنولوجيا Ecole Nationale Polytechnique	Date	Dessiné par	KAMEL Rania Fairouz	Approuvé par		DESIGNATION DU FOLIO :	Relais de niveau B		FOLIO	07	14
						DESIGNATION DU SCHEMA :	ARMOIRE STATION DE POMPAGE				

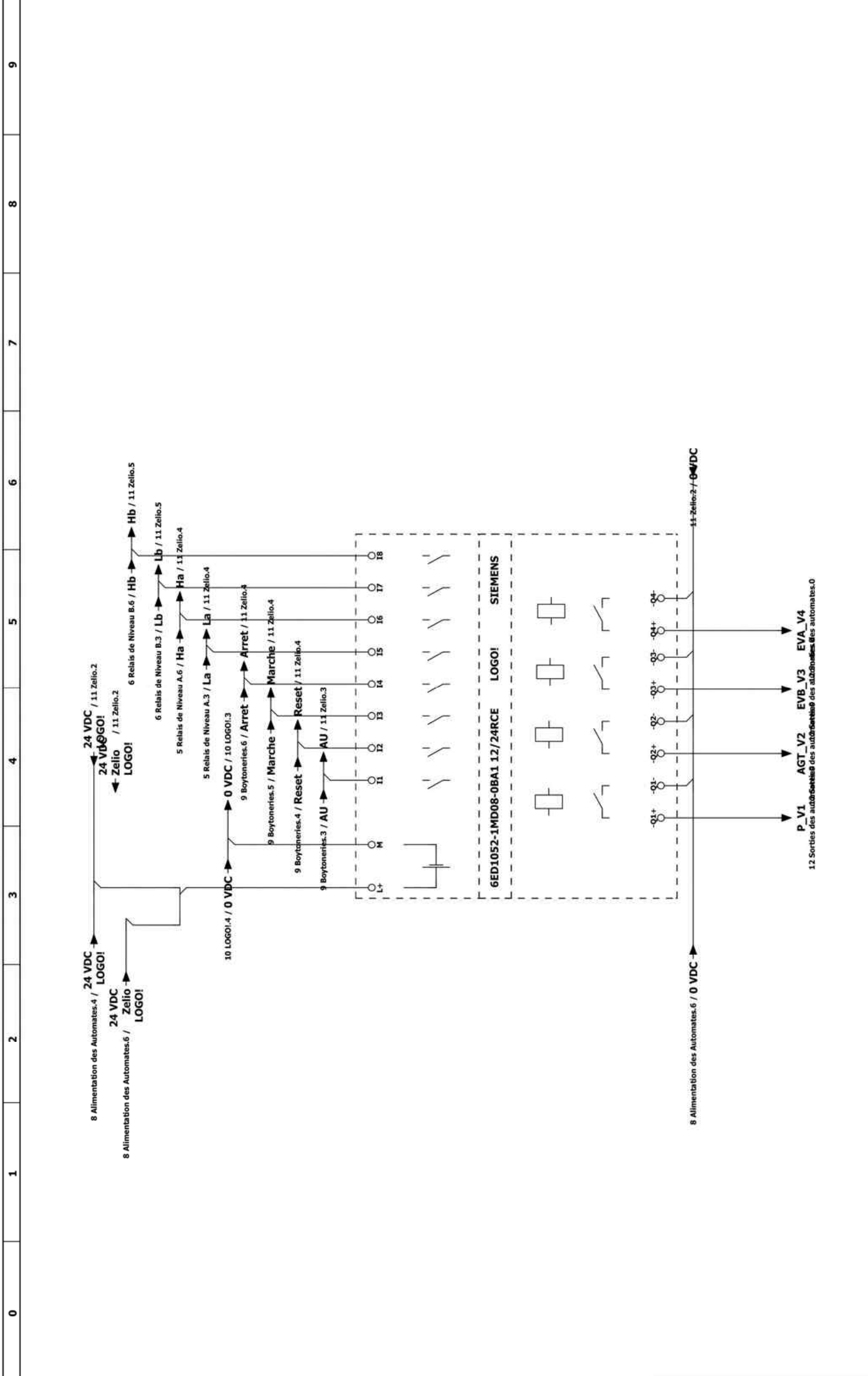






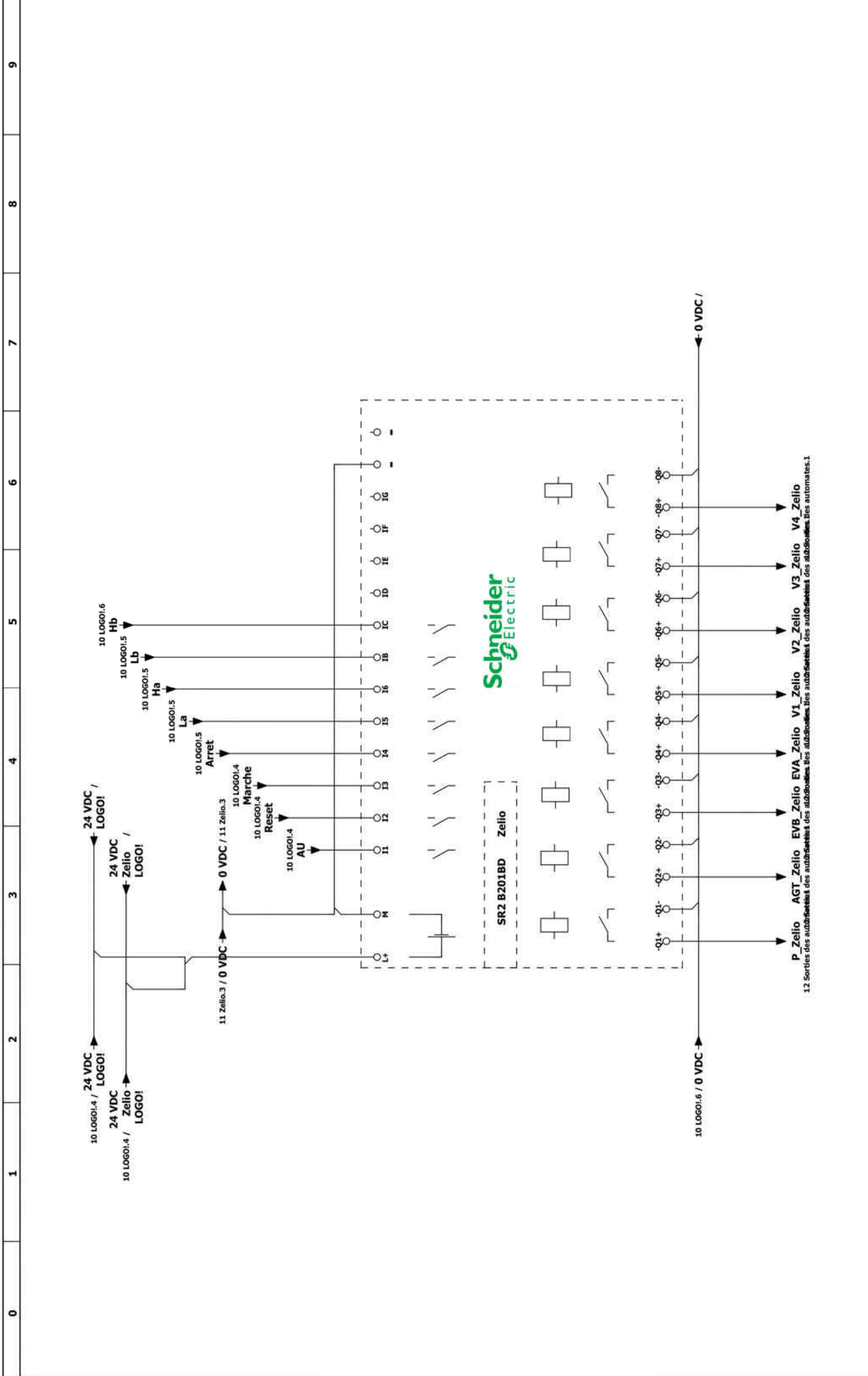
7 Actionneurs.9 / 220 VAC





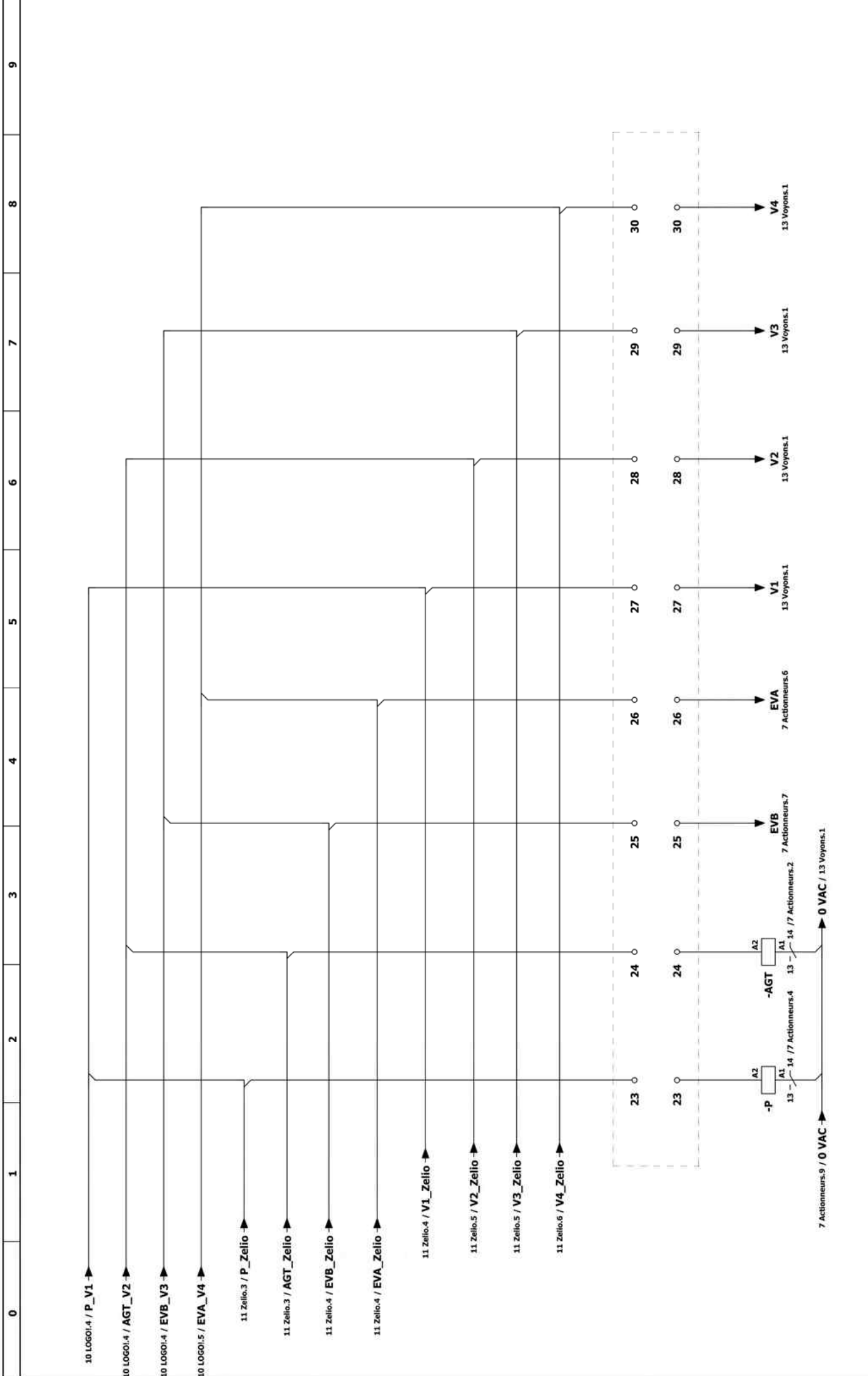
DESIGNATION DU FOLIO :	Boutonneries
DESIGNATION DU SCHEMA :	ARMOIRE STATION DE POMPAGE
Date	Dessiné par
	KAMEL Rania Fairouz
	Approuvé par





 المدرسة الوطنية للتقنية Ecole Nationale Polytechnique	Date	Dessiné par	Approuvé par	DESIGNATION DU FOLIO :	DESIGNATION DU SCHEMA :	 ARMOIRE STATION DE POMPAGE	FOLIO
		KAMEL Rania Fairouz		LOGO!	11.a		14



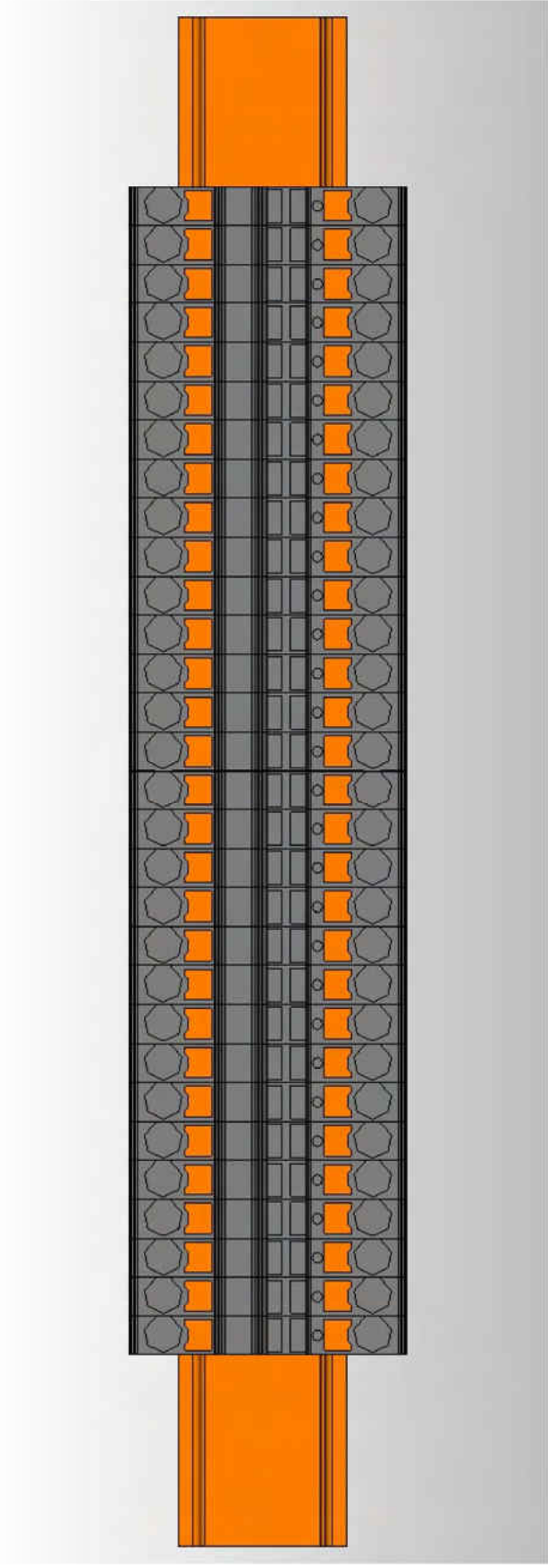
 المدرسة الوطنية للتكنولوجيا Ecole Nationale Polytechnique	Date	Dessiné par	Approuvé par	DESIGNATION DU FOLIO :	Zelio		FOLIO	11.b	14
		KAMEL Rania Fairouz		ARMOIRE STATION DE POMPAGE	DESIGNATION DU SCHEMA :				



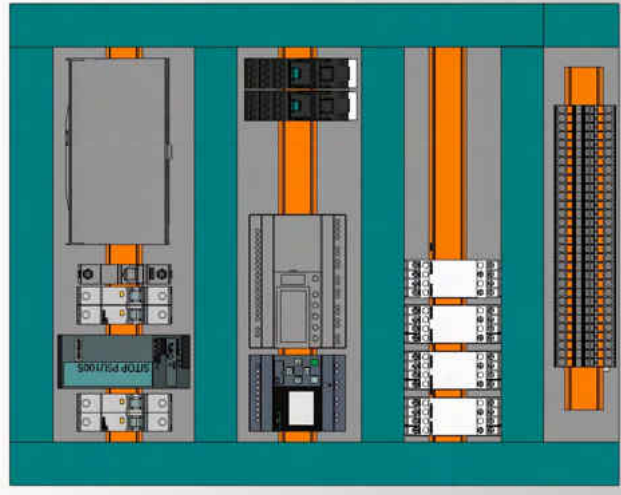
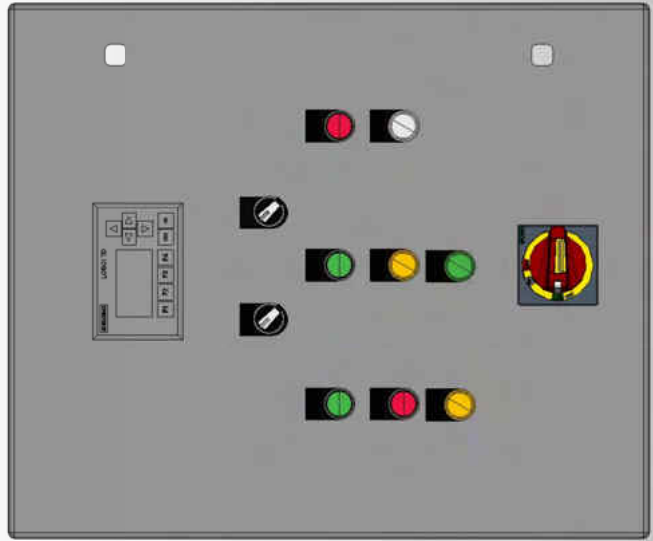
 المدرسة الوطنية للتقنيات Ecole Nationale Polytechnique	Date	Dessiné par	KAMEL Rania Fairouz	Approuvé par	DESIGNATION DU SCHEMA : <b>ARMOIRE STATION DE POMPAGE</b>	 Université de Saida Université Saida en Power	FOLIO	12	14
							DESIGNATION DU FOLIO : <b>Sorties des automates</b>		







DESIGNATION DU FOLIO :	Bournies		FOLIO	14_a
DESIGNATION DU SCHEMA :	ARMOIRE STATION DE POMPAGE		14	
Date	Dessiné par	KAMEL Rania Fairouz	Approuvé par	



Date

Dessiné par

KAMEL Rania Fairouz

Approuvé par

DESIGNATION DU FOLIO :

Vue Armoire

DESIGNATION DU SCHEMA :

ARMOIRE STATION DE POMPAGE

FOLIO

14\_b

14



## 5.5 Conclusion

Dans ce chapitre, nous avons réussi à formuler des propositions visant à améliorer la station de pompage en ce qui concerne le câblage et le matériel. Nous avons proposés une solution permettant de choisir le nombre et le type d'automates à utiliser en fonction des besoins de la manipulation, ce qui permet d'éviter les erreurs lors des opérations. De plus, nous avons suggéré d'ajouter quelques équipements électriques pour développer la station de pompage. Enfin, nous avons présenté la conception 3D de l'armoire proposée ainsi que son schéma électrique.

# Conclusion Générale

Les travaux pratiques jouent un rôle essentiel dans la formation et le développement des compétences de l'étudiant ingénieur. Dans cette optique, nous avons développé la station de pompage.

Au cours de la réalisation de ce projet, nous avons acquis de nouvelles connaissances et compétences. Grâce à notre étude préalable des automates Zelio dans le domaine de l'informatique industrielle au cours de notre parcours universitaire, nous avons pu avancer rapidement dans leur intégration dans la maquette de la station de pompage. Leur configuration, programmation, simulation et manipulation ont été réalisées sans trop de difficultés.

Cependant, l'intégration de l'automate LOGO! a été une nouvelle expérience pour nous, car c'était la première fois que nous manipulions cet automate. L'utilisation de celui-ci nous a permis d'en apprendre davantage sur cette gamme d'automates largement utilisée dans l'industrie. Il est important de souligner que l'apprentissage d'un nouveau logiciel de programmation est toujours bénéfique pour former un ingénieur automaticien compétent.

Nous avons réalisé plusieurs méthodes de commande et de supervision, à travers lesquelles nous avons approfondi plusieurs notions qui peuvent sembler complexes pour certains étudiants.

Enfin, nous avons réussi à proposer un nouveau schéma et une structure d'armoire pour notre station, permettant une meilleure organisation pour l'avenir.

Ce travail peut être développé davantage en incluant les aspects suivants :

1. La réalisation de la commande et de la supervision de la station à l'aide de WebServer, mais cette fois-ci en utilisant un téléphone mobile. Cette application nécessite l'utilisation d'un point d'accès Ubiquiti NanoStation Loco M2 pour permettre la communication entre le téléphone et l'automate LOGO!.
2. Établir une communication entre la station de pompage et une autre station qui permet de commander le déplacement aller-retour d'un chariot, contrôlé par l'automate S7 300.
3. Finaliser la réalisation de l'armoire proposée dans notre projet, en mettant en place la structure physique et le câblage conformément aux spécifications.
4. Ajouter des capteurs pour la mesure de débit de l'eau.

Nous espérons que ce travail contribuera à former une nouvelle génération d'ingénieurs en automatique bien préparés pour relever les défis de l'industrie et apporter des innovations dans le domaine de l'automatisation industrielle.

# Bibliographie

- [1] Patrick ABATI. Gemma. <https://sitelec.org/cours/abati/gemma.htm>, 07/11/2000.
- [2] Dr. HERIZI Abdelghafour. Cours automatismes industriels chapitre 4. <http://www.est-usmba.ac.ma/coursenligne/GE-S2-M8.1-Automatismes%20logiques%20Industriels-CRS-E1%20Hammoui.pdf>, 2023.
- [3] CHNT. Control relay wiring diagram for jyb-714, jyb-714b, jyb-714c 220v water drainage mode. <https://qcelectrics.co.nz/files/file-c2ba67fd9e400555f6a9bd882697f4ba/Chint%20Power%20Relays.pdf>.
- [4] Schneider Electric. Product datasheet characteristics ats01n103ft soft starter for asynchronous motor - ats01 - 3 a- 110..480v - 0.55..1.1 kw.
- [5] Schneider Electric. Zelio logic - cordon de raccordement usb pc - 3m. <https://www.se.com/fr/fr/product/download-pdf/SR2USB01>.
- [6] Schneider Electric. Zelio logic sr2b201bd fichetechnique du produit. <https://www.se.com/fr/fr/product/download-pdf/SR2B201BD>.
- [7] Schneider Electric. Zelio logic module logique manuel utilisateur. <https://www.electropoint.ch/web/content/70831>, août, 2017.
- [8] Schneider Electric. Démarreur progressif altivar ats01, Octobre, 2020.
- [9] Alain GONZAGA. les automates programmables industriels. [https://sitelec.org/download.php?filename=cours/automates\\_programmables\\_industriels.pdf](https://sitelec.org/download.php?filename=cours/automates_programmables_industriels.pdf).
- [10] Nicolas JOUVRAY. Langages de programmation pour systèmes automatisés : norme cei 61131-3. <https://www.techniques-ingenieur.fr/base-documentaire/automatique-robotique-th16/supervision-des-systemes-industriels-42396210/langages-de-programmation-pour-systemes-automatisees-norme-cei-61131-3-s8030/>, 10 ,mars ,2008.
- [11] kepsserver. Manuel kepware. <https://www.kepware.com/getattachment/e1943820-ef3c-4932-b055-4ef2a80ab863/kepsserverex-manual.pdf>, 2023.
- [12] OMRON. Data sheet general purpose relay mk. <https://docs.rs-online.com/14c0/0900766b81398813.pdf>.
- [13] SIEMENS. Data sheet 6ed1055-4mh08-0ba0. <https://docs.rs-online.com/a481/0900766b81642412.pdf>.

# Annexe A

## Généralités sur les Automates Programmables Industriels

### A.1.Introduction

Un système automatisé est un système qui fonctionne sans l'intervention humaine. Ces systèmes sont conçus pour exécuter des tâches répétitives ou des fonctions spécifiques en utilisant une combinaison de matériel et de logiciels. L'objectif principal d'un système automatisé est d'effectuer des tâches plus rapidement et avec plus de précision que les humains, ce qui permet d'améliorer les performances et les résultats. L'automatisation se fait principalement à l'aide des automates programmables, ce chapitre fournit une introduction sur les automates programmables, leurs architecture et leurs langages de programmation. Le chapitre décrit également les critères de choix d'un automate et quel gammes des automates SIEMENS.

### A.2. Les Automates Programmables Industriels

#### A.2.1. Définition

L'Automate Programmable Industriel (API ou PLC pour Programmable Logic Controller) est dispositif électronique, programmable par un personnel non informaticien, à l'aide d'un langage adapté, pour le stockage interne des instructions composant les fonctions d'automatisme. Ils sont généralement utilisés pour contrôler un large éventail d'applications, notamment les chaînes de montage, les machines d'emballage, les systèmes de convoyage et d'autres types de machines industrielles. Ils sont conçus pour exécuter des fonctions spécifiques, telles que la surveillance des entrées provenant de capteurs, le traitement des données et la commande des sorties vers les actionneurs.

#### A.2.2. Architecture des Automates Programmables Industriels

L'architecture des APIs peut être divisée selon deux aspects : l'aspect externe et l'aspect interne.

##### A.2.2.1. Aspect Externe

Les APIs peuvent être, suivant leur aspect extérieur, soit de type **compact** ou **modulaire**.

Les automates compacts sont une catégorie d'automates programmables industriels (API) qui sont conçus pour être de petite taille, économiques et faciles à utiliser, ils sont souvent utilisés pour des applications simples et à faible complexité, telles que la commande de machines-outils, de petits processus industriels et de systèmes d'alarme.

Les automates compacts sont généralement équipés d'une unité centrale de traitement (CPU), de modules d'entrée/sortie (E/S) et d'interfaces de communication, qui sont intégrés dans un seul

boîtier, ils peuvent recevoir des extensions en nombre limité. Ils sont souvent conçus pour être programmés à l'aide d'un logiciel de programmation spécifique fourni par le fabricant.

Parmi ces automates, citons notamment : LOGO de Siemens, ZELIO de Schneider.[9]



Figure A.1 : Automate Compact Zelio

Les automates de type modulaire sont une catégorie d'API qui se distinguent des automates compacts par leur modularité et leur capacité à être étendus en fonction des besoins de l'application.

Les automates modulaires, le processeur, l'alimentation et les interfaces d'entrées /sorties résident dans des unités séparées (modules) et sont fixées sur un ou plusieurs racks. Ils sont conçus pour être flexibles et évolutifs. Ils peuvent être configurés selon les besoins de l'application en ajoutant ou en retirant des modules d'entrée/sortie (E/S) ou des modules de traitement supplémentaires. Les modules peuvent être montés directement sur l'API ou déportés à distance.



Figure A.2 : Automate Modulaire Modicon M320

#### A.2.2.2. Aspect Interne

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

- **CPU** :Le processeur ou unité centrale de traitement (CPU, Central Processing Unit) contient le microprocesseur. La CPU interprète les signaux d'entrée et effectue les actions de commande conformément au programme stocké en mémoire, en communiquant aux sorties les décisions sous forme de signaux d'action.
- **Mémoire** :La mémoire contient le programme qui définit les actions de commande effectuées par le microprocesseur. Elle contient également les données qui proviennent des entrées en vue



de leur traitement, ainsi que celles des sorties.

- **Les interfaces d'entrées/sorties** : assurent l'intégration directe de l'automate dans son environnement industriel en réalisant les liaisons entre le processeur et le processus, elles peuvent être de type TOR ou analogique.
- **Alimentation** : L'unité d'alimentation est indispensable, elle assure la distribution d'énergie aux différents modules (nécessaire au processeur et aux modules d'entrées-sorties).
- **Interface de Communications** :  
Les interfaces de communication comprennent les consoles et les boîtiers tests.
  - **Les consoles** : Les consoles permettent la programmation, le paramétrage et les relevés d'informations. Ils peuvent également afficher le résultat de l'autotest comprenant l'état des modules d'entrées et de sorties, l'état de la mémoire, de la batterie, etc. Ils sont équipés (pour la plupart) d'un écran à cristaux liquides.
  - **Les boîtiers tests** : Les boîtiers de tests quand aux sont destinés aux personnels d'entretien ; ils permettent de visualiser le programme ou les valeurs des paramètres telle que l'affichage de la ligne de programme à contrôler, la visualisation de l'état des entrées et des sorties. [9]

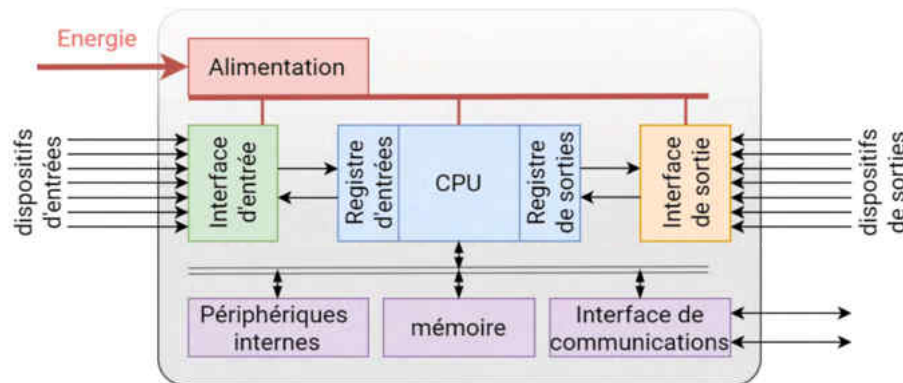


Figure A.3 : Architecture Interne de l'API

### A.2.3. Nature des Entrées et des Sorties des APIs

Les entrées et les sorties des Automates Programmables Industriels (API) peuvent être de différents types, selon leur application spécifique. Cependant, de manière générale, on peut distinguer deux types d'entrées/sorties pour les API : les entrées/sorties tout ou rien (TOR) et les entrées/sorties analogiques.

**Les entrées/sorties TOR** : des interfaces numériques qui permettent à l'API de détecter l'état de dispositifs de commutation tels que des boutons-poussoirs, des interrupteurs, des capteurs de proximité, des capteurs de niveau, des détecteurs de température, des détecteurs de pression, etc. Les entrées tout ou rien peuvent être configurées pour fonctionner en mode "tout" ou "rien", c'est-à-dire qu'elles peuvent détecter si un signal est présent ou non. Les sorties tout ou rien permettent à l'API de contrôler des dispositifs de commutation tels que des relais, des vannes, des moteurs, des pompes, etc.

**Les entrées/sorties analogiques** : sont des interfaces qui permettent à l'API de mesurer et de contrôler des grandeurs physiques continues telles que la température, la pression, le débit, la vitesse, etc. Les entrées analogiques peuvent être utilisées pour mesurer des signaux de tension ou de courant proportionnels à la grandeur physique mesurée, tandis que les sorties analogiques peuvent

être utilisées pour contrôler des dispositifs tels que des vannes proportionnelles, des variateurs de fréquence, etc.

### A.2.4. Critères de choix d'un API

Le choix de l'API est une des étapes les plus importantes dans la réalisation d'un projet. Plus les tâches à accomplir sont sensibles, plus ce choix devient critique. La sélection se fait selon différents critères :

- Le coût.
- Cahier des charges.
- Le temps de cycle.
- Le temps de cycle.
- Capacité de la mémoire.
- La capacité de traitement arithmétique.
- L'alimentation nécessaire pour alimenter l'automate.
- Nombre des entrées et des sorties.
- Le type des interfaces d'entrées et de sorties (TOR, Numérique, Analogique).
- Nombre de compteurs.
- Nombre de temporisateurs.
- Les langages de programmation.

## A.3. Les Langages de Programmes des APIs

Les langages de programmation des systèmes automatisés sont décrits dans deux normes complémentaires : la norme CEI 61131-3 et la norme CEI 61499. La norme CEI 61131-3 définit entre autres choses, cinq langages qui peuvent être utilisés pour la programmation d'applications d'automates programmables industriels (API). Les cinq langages se divisent en deux grandes catégories :

- Langages graphiques comme le SFC, FBD et le LD.
- Langages textuels tel que le ST et List.

### A.3.1. Langages graphiques

#### A.4.1.1 GRAFCET (SFC)

Le langage (sequential function chart), ou GRAFCET, est un langage graphique utilisé pour décrire les opérations séquentielles. Le procédé est représenté comme une suite connue d'étapes, 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 du GRAFCET sont :

- Deux étapes ne peuvent se suivre sans qu'il n'y ait de Transition entre eux.
- Deux Transitions ne peuvent se suivre sans qu'il n'y ait d'étapes entre eux.

Les composants de base (symboles graphiques) du langage SFC sont : [10]

- **Étapes et étapes initiales** : Une étape est représentée par un carré. Chacune des étapes est référencée par un nom inscrit dans son cadre. Ces informations constituent le niveau "1" de l'étape. La situation initiale d'un programme SFC est représentée par ses étapes initiales. Le symbole d'une étape initiale est un cadre double.
- **Transitions** : Une transition est représentée par une barre horizontale qui croise un arc de liaison. Chacune est référencée par un nom, inscrit à côté du symbole de la transition.

- **Liaisons orientées** : Une simple ligne est utilisée pour représenter une liaison orientée entre une étape et une transition. Quand l'orientation n'est pas explicitement donnée, le lien est orienté du haut vers le bas.
- **L'envoi à une étape** : Un symbole de renvoi peut être utilisé pour représenter un arc de liaison d'une transition vers une étape, sans tracer le lien. Il doit être référencé avec le nom de l'étape de destination.
- **L'action** : représentée dans un cadre à côté de l'étape dans le GRAFCET. elle est référencée selon l'action se produisant lors de l'activation de l'étape.

Le format graphique d'un programme GRAFCET est montré dans la figure suivante :

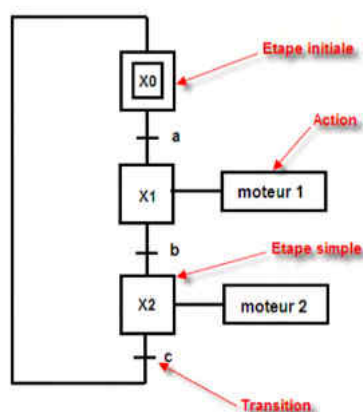


Figure A.4 : La Structure du Grafcet

### A.3.1.2. Langage FBD

Le diagramme FBD, décrit une fonction entre des variables d'entrée et des variables de sortie. Une fonction est décrite comme un réseau de fonctions élémentaires. Les variables d'entrée et de sortie sont connectées aux boîtes fonctions par des arcs de liaison. Une sortie d'une boîte peut être connectée sur une entrée d'une autre boîte.

Les variables d'entrée du diagramme FBD doivent être connectées aux entrées des boîtes fonctions. Le type de chaque variable doit être cohérent avec le type de l'entrée de la boîte correspondante. Une entrée peut être une expression constante, une variable interne ou externe.

Les variables de sortie du diagramme FBD doivent être connectées aux sorties des boîtes fonctions. Le type de chaque variable doit être cohérent avec le type de la sortie de la boîte correspondante.

Une sortie peut être une variable interne ou de sortie, ou le nom du programme édité (pour les fonctions seulement). Quand le nom de la fonction éditée est utilisé comme sortie du diagramme, il représente une assignation de la valeur retournée par la fonction.

Par exemple pour calculer la valeur DELTA d'une équation de deuxième ordre  $AX^2 + BX + C = 0$ , Avec  $DELATA = B^2 - 4AC$  on utilise le diagramme FBD suivant

### A.3.1.3. Langage LD

Le langage Ladder Diagram (LD) est une représentation graphique d'équations booléennes combinant des contacts (arguments d'entrée) avec des bobines (résultats de sortie). Il permet la description de tests et la modification de données booléennes à l'aide des symboles graphiques placés dans un

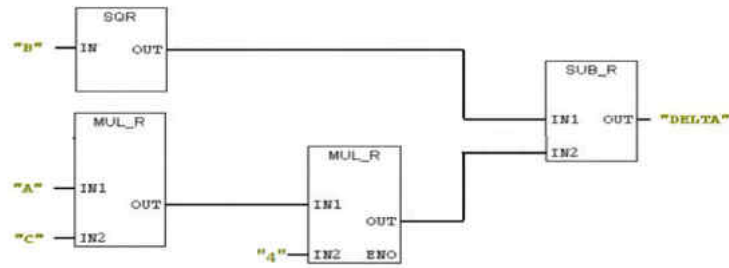


Figure A.5 : Exemple d'un Programme FBD

diagramme.

Les symboles graphiques LD sont organisés comme les éléments d'un schéma électrique à contacts, et ils sont connectés à gauche et à droite aux barres d'alimentation verticales par des segments de liens.

Chaque segment de lien peut prendre l'état booléen FALSE ou TRUE [10]. Cet état est le même pour tous les segments directement connectés ensemble.

On peut réaliser comme exemple l'équation logique :  $D = (A + B)\overline{C}$ , par la représentation LD montrée dans la figure ci-dessous :

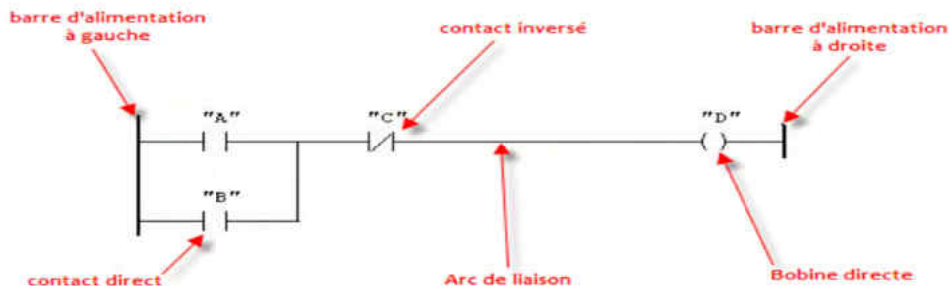


Figure A.6 : Exemple d'un Programme LA

## A.3.2. Les Langages Textuels

### A.3.2.1. Langage ST

Le langage ST (Structure Text) est un langage textuel de haut niveau, dédié aux applications d'automatisation. Il est principalement utilisé dans l'implémentation des procédures complexes et difficile à modéliser avec les langages graphiques. Le langage ST peut être utilisé pour la programmation des actions dans les étapes et les conditions associées aux transitions du SFC.

Un programme ST est une suite d'énoncés. Chaque énoncé se termine par un point virgule (";"). Les noms utilisés par le code source (identificateurs de variables, constantes, mots-clés du langage, etc.) sont délimités par des séparateurs passifs (espace, fin de ligne ou tabulation) ou par des séparateurs actifs, qui ont une signification bien définie et qui jouent le rôle d'opérateurs (par exemple, le séparateur > indique une comparaison "Plus grand que").

On peut citer plusieurs types d'énoncés standards du ST tels que : [10]

- Assignation (variable := expression ;).
- Appel de fonction.
- Appel de bloc fonctionnel.
- Énoncés de sélection (IF, THEN, ELSE, CASE, etc.).

- Enoncés d'itération (FOR, WHILE, REPEAT, etc.).
- Enoncés de contrôle (RETURN, EXIT, etc.).
- Enoncés spéciaux pour le lien avec d'autres langages, tels que SFC.

Pour la résolution d'une équation de deuxième ordre  $AX^2 + BX + C = 0$  on écrit le programme en langage ST montré dans la figure suivante :

```

Delta := B * B - 4 * A * C;
IF Delta < 0 THEN
    NSolution := 0;
ELSE IF Delta = 0 THEN
    NSolution := 1;
    X1 := (-B) / (2 * A);
ELSE
    NSolution := 2;
    X1 := ( -B + sqrt (Delta) ) / (2 * A);
    X2 := ( -B - sqrt (Delta) ) / (2 * A);
END_IF

```

Figure A.7 : Exemple d'un Programme ST

### A.3.2.2. Langage List

Le langage LIST, est un langage textuel de bas niveau. Les instructions travaillent 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 à son tour stocké dans le registre.

Un programme IL est une liste d'instructions. Chaque instruction doit commencer par une nouvelle ligne, et doit contenir un opérateur et un opérandes. Les opérateurs de langage list se defaire d'un automate a un autre, a titre d'exemple , on citer quelques opérateurs du langage IL d'un automate siemens :

- U : AND (ET).
- UN : NAND .
- O : OR (OU).
- ON : NOR.
- CALL : Appeler un bloc fonctionnel.
- = : Affecter le résultat a une sortie.
- L : Charger une variable.
- T : Transfer d'un résultat dans une variable.

On reprend l'exemple l'équation logique :  $D = (A + B)\overline{C}$ , on écrit le programme montré dans la figure suivante :

U(		
O	"A"	E0.0
O	"B"	E0.1
)		
UN	"C"	E0.2
=	"D"	A0.0

Figure A.8 : Exemple d'un Programme List avec automate siemens

# Annex B

## Les Caractéristiques Techniques Des Equipements Electriques de la Station

### B.1.Démarreur Progressive ATS 01N103FT

L'ATS 01N103FT est un démarreur progressif de la série Altistart 01 de Schneider Electric. Il est conçu pour fournir un démarrage en douceur et progressif des moteurs asynchrones triphasés. Il contribue à réduire le stress mécanique et les surtensions électriques qui peuvent survenir lors du démarrage d'un moteur. Cela entraîne une durée de vie plus longue de l'équipement et des coûts de maintenance réduits. Le dispositif dispose également d'une protection contre les surcharges et d'une protection contre la perte de phase pour assurer un fonctionnement sûr et fiable du moteur.

Il est destiné aux applications simples suivantes : Pompe, Tapis transporteurs, Ventilateurs, Portes automatiques..ect [8]



Figure B.1.1 :ATS 01N103FT

#### B.1.1. Description

l'ATS 01N103FT est équipé de :

- Un potentiomètre de réglage du temps de démarrage ( $P_1$ ).
- Un potentiomètre pour ajuster le seuil de tension de démarrage en fonction de la charge du moteur( $P_2$ ).
- 2 LED, une pour indiquer la mise en marche du demarreur et l'autre pour indiquer la presence de tension dans le moteur.
- 2 entrées 24V et 220V (CL2) pour le controle de la partie commande.

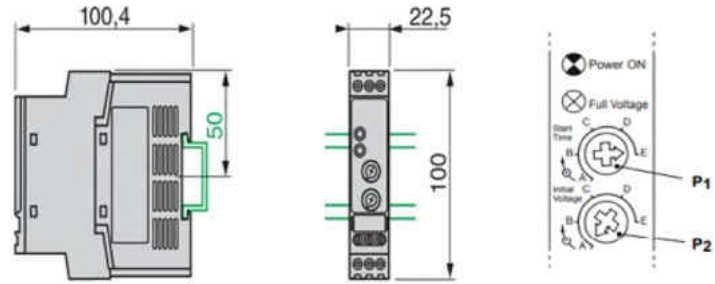


Figure B.1.2 : Dimension de l'ATS 01N103FT [4]

## B.1.2. Fiche Technique

Démarreurs progressifs pour moteurs de 0,37 à 11 kW									
Moteur						Démarreur			
Puissance moteur (1)		Courant nominal		Encombrements L x P x H		Référence (2)		Masse	
Monophasé	Triphasé								
230 V	110 V 230 V 230 V 400 V 460 V								
kW	HP kW HP kW HP	A		mm/ in.				kg/ lb	
Tension d'alimentation monophasée 110...230 V ou triphasée 110...480 V 50/60 Hz									
0,37	- 0,37 0,5 1,1 0,5	3		22,5 x 100,4 x 100/ 0,89 x 3,95 x 3,94		ATS01N103FT		0,160/ 0,353	
	- 0,55 - - 1,5								

Figure B.1.3 : Fiche Technique de l'ATS 01N103FT [8]

## B.1.3. Schéma de câblage monophasé

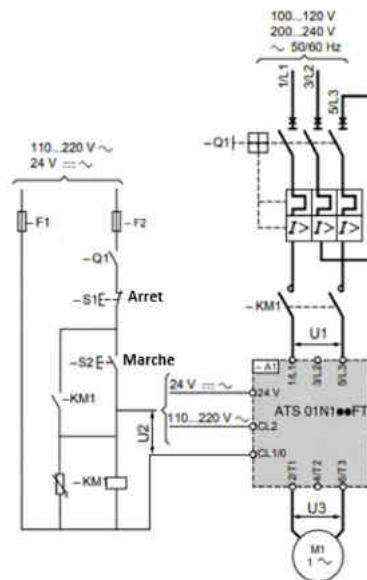


Figure B.1.4 : Câblage Monophasé de l'ATS 01N103FT [4]

## B.1.4. Diagramme Fonctionnel

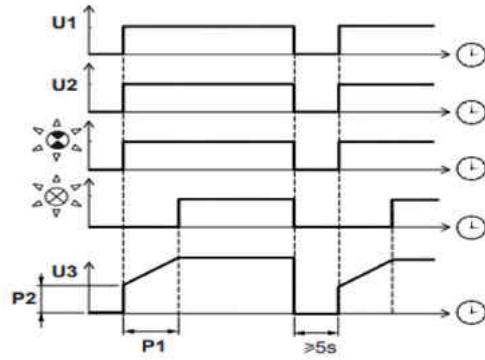


Figure B.1.5 : Diagramme Fonctionnel [4]

## B.2. Relais Electromécanique OMRON MK2P-I

Un relais électromécanique est un type de commutateur électrique qui est activé par un courant électrique pour contrôler un autre circuit. Il est constitué d'un électroaimant qui, lorsqu'il est alimenté en courant, attire un noyau mobile qui agit sur des contacts électriques pour ouvrir ou fermer un circuit électrique. Il peut être utilisé pour isoler un circuit électrique de forte puissance du circuit de commande de faible puissance. Il peut également être utilisé pour contrôler plusieurs circuits avec un seul signal de commande.

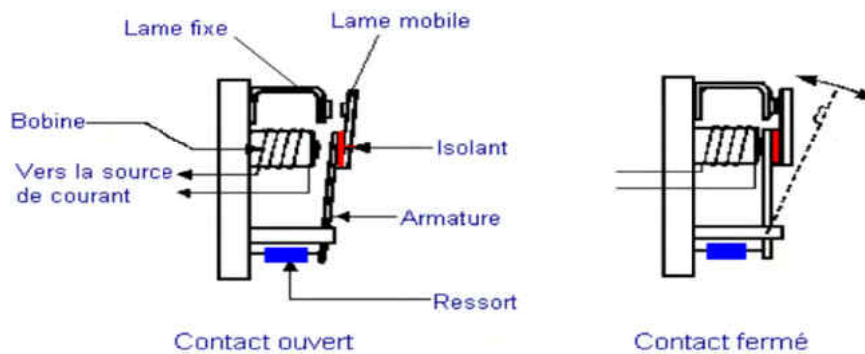


Figure B.2.1 : Schéma d'un relais électromécanique

### B.2.1. Séchema de Connection

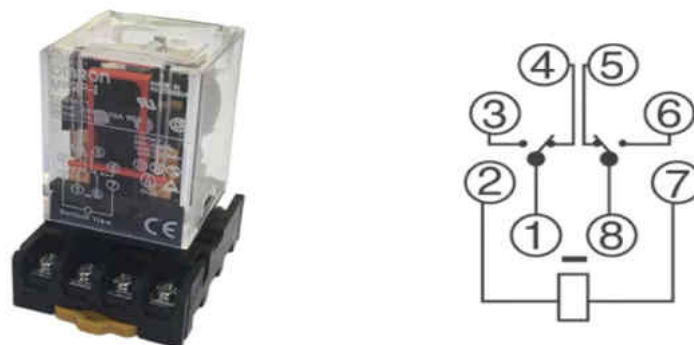


Figure B.2.2 : Schéma de connexion de OMRON MK2P-I [12]



## B.2.2. Montage et Dimesions

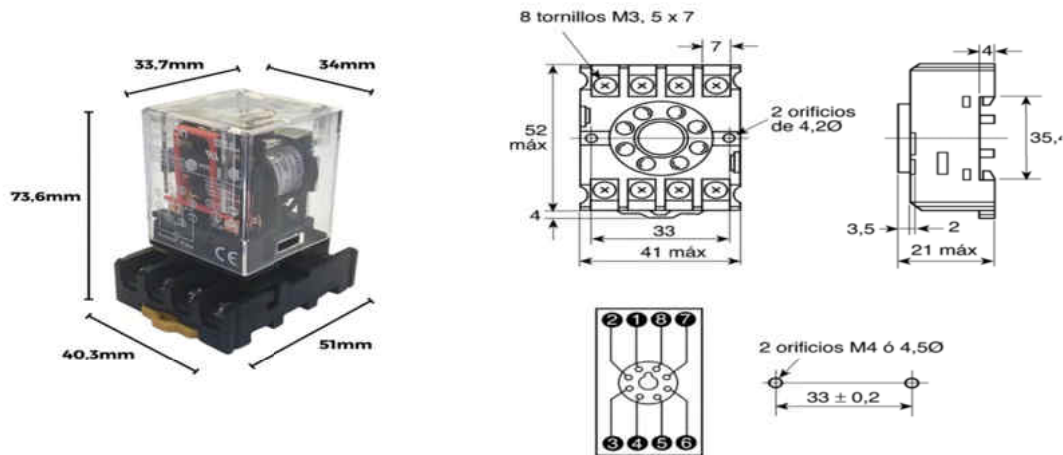


Figure B.2.3 : Les dimensions de OMRON MK2P-I [12]

## B.2.3. Fiche Technique

Produit	Relais Electromécanique
Tension de la bobine	24 VDC
Forme de contact du relais	2 Forme C (DPDT-NO, NC)
Résistance de la bobine	430 Ohms
Courant de la bobine	56 mA
Consommation d'énergie	1.5 W
Contact	250VAC /28V DC (10A (N.O),5A (N.C)) 250 VAC 7A GEN.cos $\phi$ =0.4

Tableau B.2.1. Fiche Techique de OMRON MK2P-I [12]

## B.3. Relais de Niveau CHINT JYB-714C

Le relais de niveau est un dispositif électronique utilisé pour surveiller le niveau d'un liquide ou d'un solide dans un réservoir ou une cuve. Il fonctionne en détectant le niveau du liquide à l'aide de sondes et en envoyant un signal électrique pour activer ou désactiver un dispositif de commande tel qu'une pompe, une vanne ou un moteur.



Figure B.3.1 : CHINT JYB-714C

### B.3.1. Dimensions et Montage

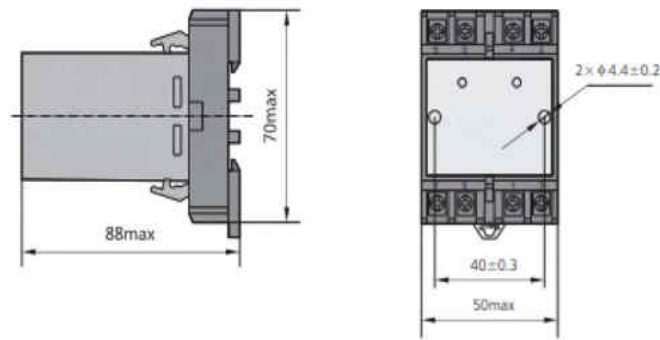


Figure B.3.2 : Dimensions et Montage de CHINT JYB-714C [3]

### B.3.2. Schéma de Raccordement Monophasé

Le principe de fonctionnement du relais de niveau est le suivant :

Lorsque la sonde reliée à la borne 5 détecte la présence de l'eau, le relais détecte un niveau haut et le contact passe de la borne 2 à la borne 4. Lorsque la sonde reliée à la borne 7 ne détecte pas la présence de l'eau, le relais détecte un niveau bas et le contact passe de la borne 4 à la borne 2.

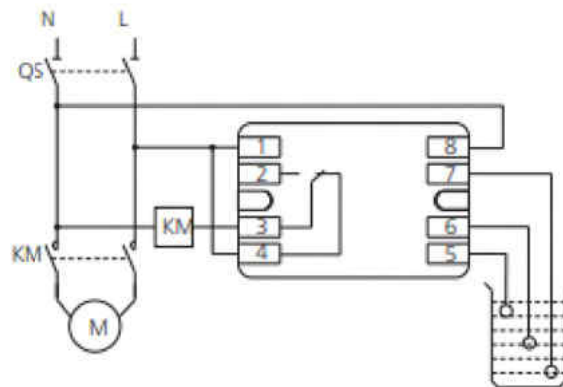


Figure B.3.2 : Schéma de raccordement monophasé de CHINT JYB-714C [3]

### B.3.3. Fiche Technique

Produit	Relais de Niveau CHINT
Capacité du contact	Ue/Ie : AC 15/220V/1.5A ; Ith :5A
Voltage d'opération	AC 50Hz/60Hz 36V,110V, 220V,380V
Courant maximal de l'électrode de control	50µA
Longueur du conducteur	Max 1Km
Température ambiante	-5°C +40°C
Puissance absorbée	< 3VA

Tableau B.3.1. Fiche Techique de CHINT JYB-714C [3]

## B.4. Alimentation Redressé Stablié HAS-120-24

Une alimentation stabilisée est une alimentation électrique qui fournit une tension continue (DC) stable et régulée, indépendamment des variations de la tension d'entrée, des variations de charge ou des perturbations électriques. Elle utilise des circuits de régulation pour maintenir la tension de sortie à une valeur fixe et constante, même lorsque la tension d'entrée varie ou lorsque la charge varie. Les circuits de régulation les plus couramment utilisés sont les régulateurs linéaires et les régulateurs à découpage.



Figure B.4.1 : Alimentation Redressé Stablié HAS-120-24

### B.4.1. Architecture Interne

Correction : L'architecture interne d'une alimentation redressée stabilisée peut varier en fonction de la conception et de la complexité de l'alimentation. Cependant, les blocs principaux que l'on retrouve dans toutes les alimentations redressées stabilisées sont :

- **Le transformateur** : il permet d'adapter la tension alternative (AC) du réseau électrique à la tension nécessaire pour l'alimentation.
- **Le redresseur** : il permet de transformer le courant alternatif (AC) en courant continu (DC).
- **Le condensateur de filtrage** : il permet de lisser la tension redressée et de réduire les ondulations résiduelles présentes sur la tension continue.
- **Le régulateur de tension** : il permet de maintenir la tension de sortie de l'alimentation à une valeur constante malgré les variations de la tension d'entrée ou les charges connectées à l'alimentation.

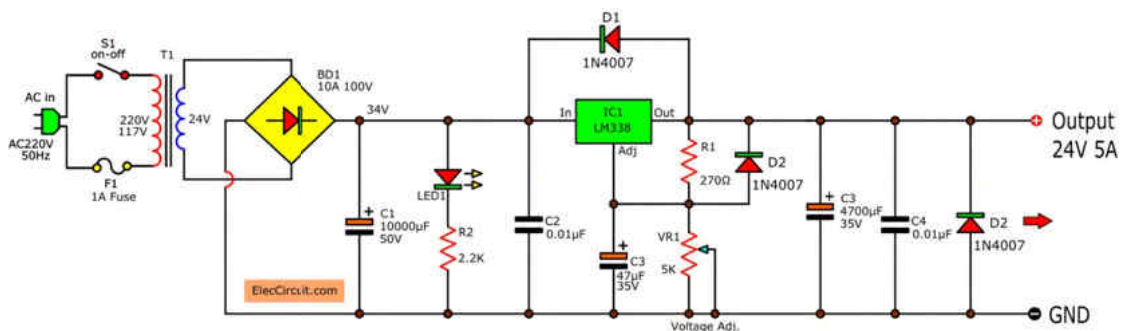


Figure B.4.2 : Exemple d'architecture d'une alimentation redressée stabilisée

## B.4.2. Caractéristiques Techniques de HAS-120-24

Produit	Alimentation Redressé Stabilité HAS-120-24
Tension et courant d'entrée	100-120VAC 3.3A, 200-240VAC 2.0A 50/60Hz
Tension et courant de sortie	24V DC 5A

Tableau B.4.1. Caractéristiques Techniques de HAS-120-24

## B.5. Disjoncteur Magnétothermique Bipolaire HB-63

Le disjoncteur magnétothermique est un dispositif de protection électrique qui permet de protéger les circuits contre les surcharges et les courts-circuits. Il est composé de deux éléments principaux : le dispositif de déclenchement magnétique et le dispositif de déclenchement thermique.

Le dispositif de déclenchement magnétique utilise un champ magnétique pour détecter les courants de surcharge et actionne un mécanisme de déclenchement pour couper le courant électrique.

Le dispositif de déclenchement thermique utilise une bimétallique pour détecter une surchauffe, qui se courbe et actionne également un mécanisme de déclenchement pour couper le courant électrique.



Figure B.5.1 : Symbole électrique du disjoncteur magnétothermique bipolaire

### B.5.1. Symbole Elèctrique

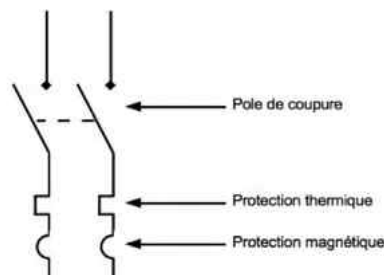


Figure B.5.2 : Symbole électrique du disjoncteur magnétothermique bipolaire

## B.6. Electrovanne

Une électrovanne est un dispositif électromécanique qui est utilisé pour contrôler le débit d'un fluide. Elle est composée d'un corps de vanne, d'une bobine électromagnétique, d'un noyau mobile et d'un ressort.

Lorsqu'une électrovanne est alimentée électriquement, le champ magnétique créé par la bobine attire le noyau mobile, ce qui ouvre la vanne et permet au fluide de s'écouler. Lorsque l'alimentation électrique est coupée, le ressort rappelle le noyau mobile à sa position initiale, ce qui ferme la vanne et arrête le débit. Il existe deux types d'électrovanne : **TOR** et **proportionnelle**.

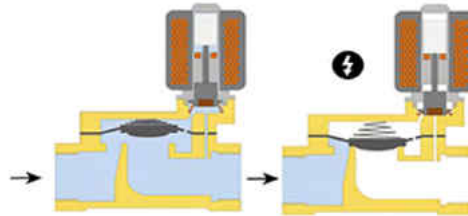


Figure B.6.1 : Principe de fonctionnement de l'électrovanne

### B.6.1. Symbole Pneumatique

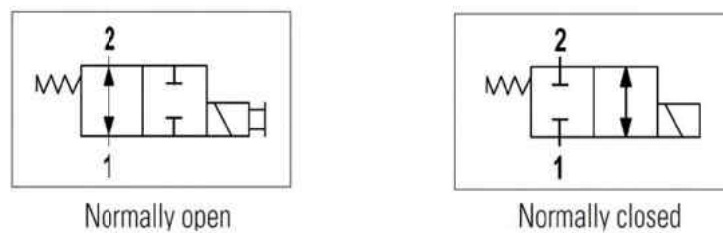


Figure B.6.2 : Symbole Pneumatique de l'électrovanne

## B.7. Pompe

Une pompe à eau électrique fonctionne à partir de l'énergie électrique. Elle transforme l'énergie électrique reçue en énergie mécanique servant à activer la pompe. L'énergie mécanique sera ensuite transmise au liquide où elle est convertie en énergie cinétique « débit » puis en énergie potentielle « pression ». C'est à partir de ces énergies que le fonctionnement de la pompe à eau électrique s'opère à travers des échanges et consommations d'énergies dans les circuits d'installation.



Figure B.7.1 : La plaque signalitique de la pompe

# Annexe C

## Les Caracteristiques des Automates Programmables Utilisés

### C.1. Automate Zelio Logic

#### C.1.1. Les Caracteristiques Techniques de l'API Zelio Logic



Figure C.1 : Zelio SR2 B201BD

Références	SR2 B201BD
Horloge	Oui
Température ambiantes max	55 °C
Température ambiante min	-20 °C
Courant permanent max	250 V/AC – 8A
Possibilité d'extension	Non
Indice de protection	IP20
Tension de fonctionnement	24 V/DC
Nombre d'entrées max	12
Nombre de sortie relais max	8
Catégorie	API- Module de commande
Nombre d'entrées mixtes analogiques/numériques max	6
Nombre d'entrées numériques max	6
Affichage	Oui
Nombre de sorties max	8

Tableau C.1 : Caractéristiques techniques de l'automate Zelio logic [6]

## C.1.2. Les Caracteristiques Techniques du Câble de Raccordement



Figure C.2 : Câble SR2USB01

Catégorie	API- Câble
Modèle	Câble de connexion entre PC et module logique / Longueur : 3m
Référence	SR2 CBL01
Convient pour	Module logique ZELIO LOGIC Schneider Electric

Tableau C.2 : Caractéristiques techniques du câble SR2CBL01 [5]

## C.2. Automate LOGO!



Figure C.3 : LOGO! 0BA8

Références	LOGO! 0BA8 12/24RCE
	<b>Alimentation</b>
Tension d'entrée	12/24 V CC
Plage admissible	10,8 V CC à 28,8 V CC
Protection contre l'inversion de polarité	Oui
Consommation de courant	— 30 mA à 140 mA — 15 mA à 90 mA
— 12 V CC	
— 24 V CC	

Temps de maintien en cas de défaillance de la tension — 12 V CC — 24 V CC	— Typ. 2 ms — Typ. 5 ms
Perte de puissance — 12 V CC — 24 V CC	— 0,4 W à 1,7 W — 0,4 W à 2,2 W
Précision de l'horloge temps réel	Typ. $\pm 2$ s/jour
Séparation galvanique	Non
<b>Entrées TOR</b>	
Nombre	8
Séparation galvanique	Non
Nombre d'entrées rapides	4 (I3, I4, I5, I6)
Fréquence d'entrée — Entrée normale — Entrée rapide	— max. 4 Hz — max. 5 kHz
Tension continue maximale admissible	28,8 V
Tension d'entrée L+ — Signal 0 — Signal 1	— < 5 V CC — < 8,5 V CC
Courant d'entrée — Signal 0 — Signal 1	— < 0,88 mA (I3 à I6) — < 0,07 mA (I1, I2, I7, I8) — > 1,5 mA (I3 à I6) — > 0,12 mA (I1, I2, I7, I8)
Temps de retard — 0 à 1 — 1 à 0	— Typ. 1,5 ms < 1,0 ms (I3 à I6) — Typ. 1,5 ms < 1,0 ms (I3 à I6)
Longueur de câble (non blindé)	Max. 100 m
<b>Entrées Analogiques</b>	
Nombre	4 (I1=AI3, I2=AI4, I7=AI1, I8=AI2)
Plage	0 V CC à 10 V CC impédance d'entrée 72 k $\Omega$
Temps de cycle pour création des valeurs analogiques	300 ms
Longueur de câble (blindé et torsadé)	Max. 10 m
<b>Sorties TOR</b>	
Nombre	4
Type des sorties	Sorties à relais
Séparation galvanique	Oui

Tableau C.3 : Caractéristiques techniques de l'automate LOGO!

### C.3.Interface LOGO! TDE





Figure C.4 : Interface LOGO! TDE

### C.3.1. Caractéristiques Mécaniques

Clavier Ecran	Clavier à membranes à 10 touches Ecran graphique FSTN 160 x 96 (colonnes x lignes), rétro-éclairage à LED (blanc/ambre/rouge)
---------------	---

Tableau C.4 : Caractéristiques mécaniques de l'afficheur LOGO! TDE [13]

### C.3.2. Caractéristiques Techniques

<b>Alimentation</b>	
Tension d'entrée	<ul style="list-style-type: none"> <li>— 24 VAC</li> <li>— 12/24 V DC</li> </ul>
Plage admissible	<ul style="list-style-type: none"> <li>— 20,4 V AC à 26,4 V AC</li> <li>— 10,2 V CC à 28,8 V CC</li> </ul>
Fréquence de réseau admissible	47Hz à 63 Hz
Consommation (Ethernet et rétroéclairage blanc actif)	<ul style="list-style-type: none"> <li>— Typ. 145 mA</li> <li>— Typ. 70 mA</li> <li>— Typ. 75 mA</li> </ul>
<b>Indice de protection</b>	IP65
<b>Port de communication</b>	
Nombres d'interface Ethernet	Deux interfaces Ethernet
Distance de raccordement	Max. 30 m

Tableau C.5 : Caractéristiques techniques de l'afficheur LOGO! TDE [13]

# Annexe D

## Présentation de Logiciel KepserverEX

### D.1 Introduction

KEPServerEX de Kepware est une plateforme de connectivité industrielle largement utilisée qui permet la communication entre différents dispositifs industriels, équipements et applications logicielles. Il agit en tant que serveur de communication, permettant à différents dispositifs et applications d'échanger des données de manière transparente.

KEPServerEX prend en charge un large éventail de protocoles industriels, notamment OPC (OLE pour le contrôle des processus), Modbus, SNMP (Simple Network Management Protocol), DNP3 (Distributed Network Protocol) et bien d'autres. Ces protocoles sont couramment utilisés dans des industries telles que la fabrication, le pétrole et le gaz, les services publics, l'automatisation des bâtiments, etc.

### D.2 Interface de KepserverEX

Les sections principales de l'interface de KepserverEX sont :

- **Barre de Menu** : La barre de menu contient différentes options pour accéder aux fonctionnalités et aux paramètres de configuration de KEPServerEX, où nous trouverons les menus de : Fichier, Édition, Vue, Outils, Aide, Runtime.
- **Barre des boutons** : La barre des boutons regroupe des icônes qui fournissent un accès rapide aux fonctionnalités couramment utilisées. Ces boutons peuvent inclure des actions telles que la configuration, le démarrage ou l'arrêt des serveurs, l'ajout de canaux, etc.
- **Arborescence du projet** : L'arborescence du projet est une vue hiérarchique qui permet d'organiser les différents éléments du projet, tels que les canaux, les périphériques, les groupes d'éléments, etc. Elle offre une vue structurée et organisée de la configuration du projet et elle regroupe les éléments suivant :
  1. **Connectivity** : Cette section regroupe les canaux et les périphériques utilisés pour établir la connectivité entre KEPServerEX et les dispositifs ou logiciels externes.
  2. **Aliases** : Permet de créer des noms conviviaux et faciles à utiliser pour les tags ou les adresses mémoire utilisées dans votre configuration. Nous pouvez créer des alias pour simplifier la manipulation des données dans les applications ou systèmes clients.
  3. **Advanced Tags** : Cette section permet de configurer des options avancées pour les variables tel que : telles que des expressions mathématiques comme calcule le maximum , le minimum et la moyenne, et relier les variables entre eux.
  4. **Alarms and Events** : Cette section nous permet de configurer des alarmes et des événements pour surveiller les changements d'état ou les conditions spécifiques dans données. nous pouvez définir des seuils, des déclencheurs, des actions à effectuer en cas d'alarme, etc.

5. **DataLogger** : L'enregistreur de données permet de capturer et de stocker des données historiques provenant des périphériques.
  6. **EFM Exporter** : Où les données sur les flux et les tendances peuvent être saisies et coordonnées.
  7. **IDF for Splunk** : Où il est possible de configurer les flux de données pour la gestion et l'exploration des données.
  8. **IoT Gateway** : Facilite l'intégration de KEPServerEX avec les plateformes IoT (Internet of Things) et les solutions de gestion des périphériques connectés. Elle permet la collecte et la transmission des données vers les applications IoT.
  9. **Local Historian** : est un outil intégré de KEPServerEX qui permet de stocker et de gérer les données historiques localement, sans nécessiter une base de données externe. Il facilite l'accès aux données historiques pour l'analyse et la prise de décision.
  10. **Scheduler** : Le planificateur permet de configurer des tâches automatisées, telles que des sauvegardes planifiées, des exportations de données, des mises à jour de configuration, etc.
  11. **SNMP Agent** : Permet de superviser et de gérer KEPServerEX via le protocole SNMP (Simple Network Management Protocol). Il facilite l'intégration de KEPServerEX avec les systèmes de gestion réseau et de surveillance.
- **Vue détaillée** : La vue détaillée affiche les détails spécifiques de l'élément sélectionné dans l'arborescence du projet.
  - **Journal des événements** : est une fenêtre qui enregistre les informations et les événements importants liés à KEPServerEX. Il peut afficher des messages de diagnostic, des erreurs, des avertissements, des connexions établies ou perdues, etc.
  - **Barre d'état** : La barre d'état est située en bas de l'interface et affiche des informations sur l'état actuel du serveur KEPServerEX. Cela peut inclure des indications telles que la connectivité, l'état des canaux, la disponibilité des périphériques, etc. [11]

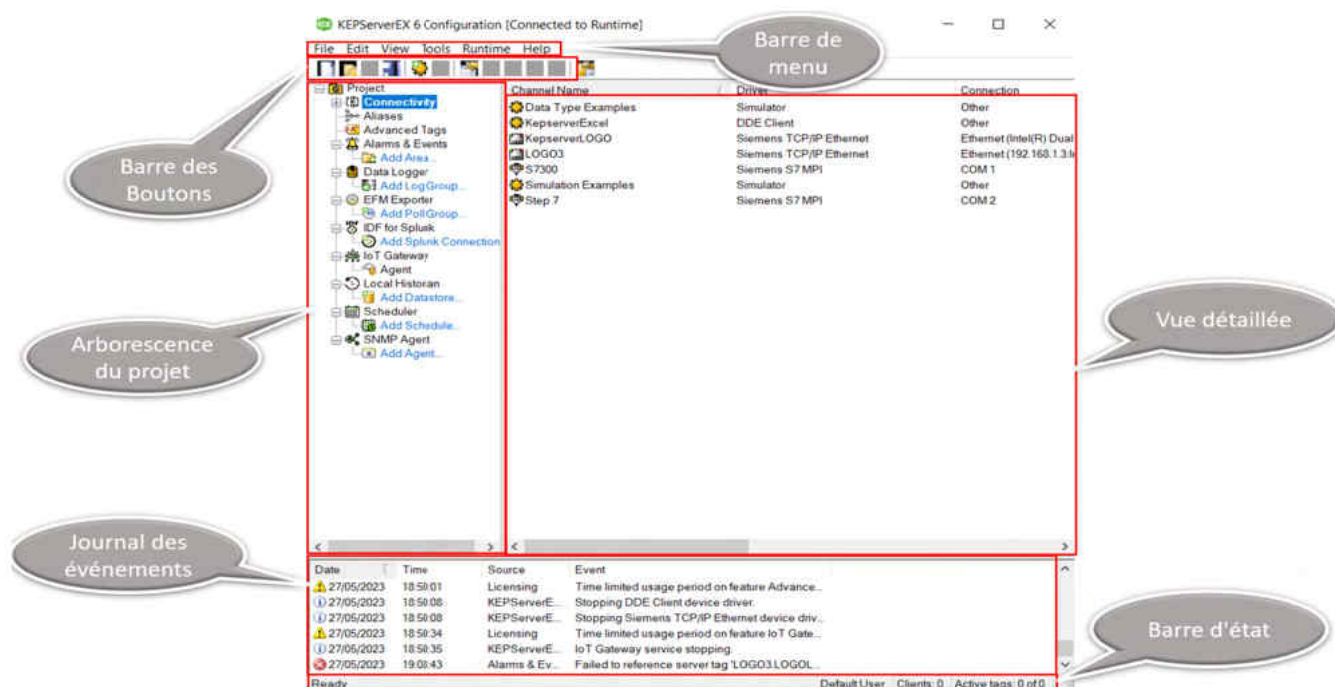


Figure D.1 : Interface de projet de KepserversEX

## D.3 Création des canneaux

Pour créer un canal dans KEPServerEX, nous suivons les étapes suivantes :

1. Nous cliquons avec le bouton droit de la souris sur **Connectivity** dans l'arborescence de projet, puis sélectionnons **New Channel**.
2. Dans la fenêtre de configuration du canal, choisissons le type de canal souhaité à partir de la liste déroulante des protocoles pris en charge.

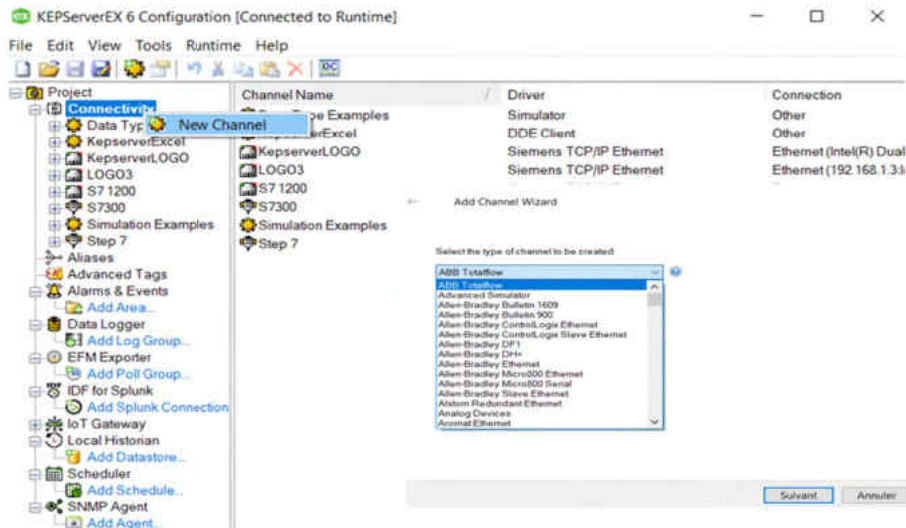


Figure D.2 : Insérer un nouveau canal

3. Complétons les paramètres spécifiques au canal sélectionné, tels que l'adresse IP, le numéro de port, les options de sécurité, les délais, etc. Les paramètres varient en fonction du canal choisi.

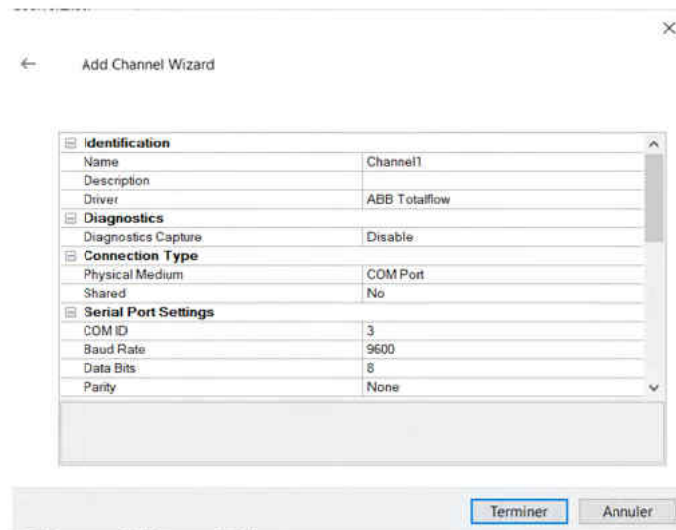


Figure D.3 : Paramètres du canal

### a. Configuration d'un canal Excel

Pour créer un canal Excel dans KEPServerEX, nous devons choisir le canal DDE dans la liste déroulante des types de canaux. DDE (Dynamic Data Exchange) est le protocole utilisé pour la communication avec les applications Microsoft Excel.

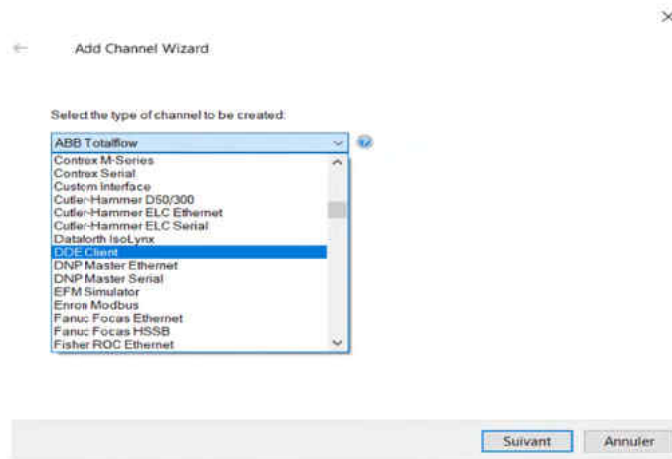


Figure D.4 : Canal DDE

Dans KEPServerEX, l'adresse des variables Excel est définie selon le format suivant :

**DDE Service Name — Topic Name!Item Name.** La signification de chaque partie de l'adresse est :

- **DDE Service Name** : Il s'agit du nom du service DDE utilisé pour communiquer avec Excel. Dans le cas d'Excel, le nom du service DDE est **Excel**.
- **Topic Name** : Le nom du topic fait référence à la page ou à la feuille d'Excel avec laquelle nous souhaitons établir une communication dans notre La page est nommée par **Systeme**.
- **Item Name** : Il s'agit du nom de la variable ou de la cellule spécifique que vous souhaitez utiliser dans KEPServerEX. nous pouvons faire référence à une cellule spécifique en utilisant sa notation (par exemple, A1, B2, etc.)

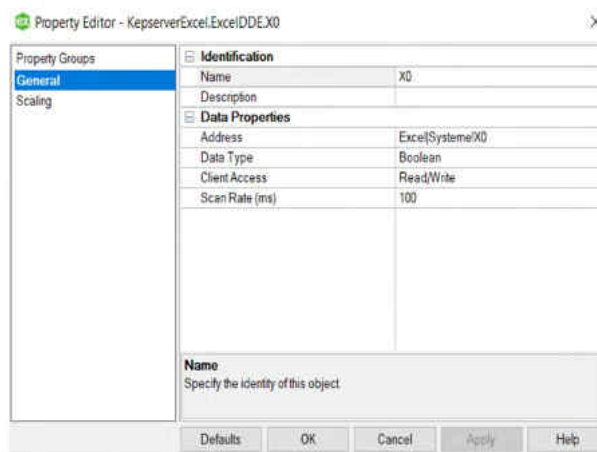


Figure D.5 : Déclaration variable excel dans KepserverEX

## b. Configuration d'un canal LOGO !

Pour créer un canal LOGO! dans KEPServerEX, nous devons choisir le canal **Siemens TCP/IP Ethernet** dans la liste déroulante des types de canaux et on choisit le type d'automate S7 1200, car il est compatible avec LOGO !.

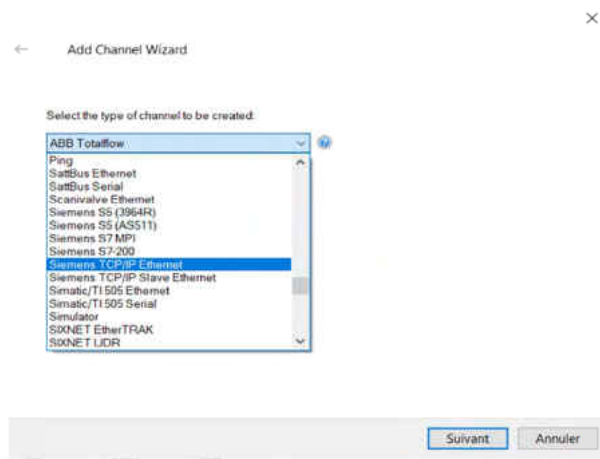


Figure D.6 : Canal LOGO!

La compatibilité entre les adresses de LOGO! et S7 1200 sont définie comme suit :

I	Address	Q	Address	M	Address
01	I0.0	Q1	Q0.0	M1	M0.0
02	I0.1	Q2	Q0.1	M2	M0.1
03	I0.2	Q3	Q0.2	M3	M0.2
04	I0.3	Q4	Q0.3	M4	M0.3
05	I0.4	Q5	Q0.4	M5	M0.4
06	I0.5	Q6	Q0.5	M6	M0.5
07	I0.6	Q7	Q0.6	M7	M0.6
08	I0.7	Q8	Q0.7	M8	M0.7
09	I1.0	Q9	Q1.0	M9	F1.0
10	I1.1	Q10	Q1.1	M10	F1.1
11	I1.2	Q11	Q1.2	M11	M1.2
12	I1.3	Q12	Q1.3	M12	M1.3
13	I1.4	Q13	Q1.4	M13	M1.4
14	I1.5	Q14	Q1.5	M14	M1.5
15	I1.6	Q15	Q1.6	M15	M1.6
16	I1.7	Q16	Q1.7	M16	M1.7
17	I2.0			M17	M2.0
18	I2.1			M18	M2.1
19	I2.2			M19	M2.2
20	I2.3			M20	M2.3
21	I2.4			M21	M2.4
22	I2.5			M22	M2.5
23	I2.6			M23	M2.6
24	I2.7			M24	M2.7
				M25	M3.0
				M26	M3.1
				M27	M3.2

Figure D.7 : Compatibilité des adresses

## D.4 Simulation du Projet

Pour simuler le KepserverEX

1. Allons dans le menu **Runtime** et sélectionnons **Reinitialize**.

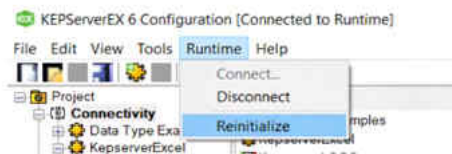


Figure D.8 : Réinitialisation du projet

2. Allons dans le menu **Tools** et sélectionnons **Launch OPC Quick Client**.

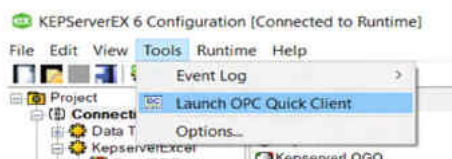


Figure D.10 : Launch OPC Quick client

3. Une fenêtre de simulateur s'affichera, où nous pourrons contrôler et superviser les variables dans KEPServerEX.

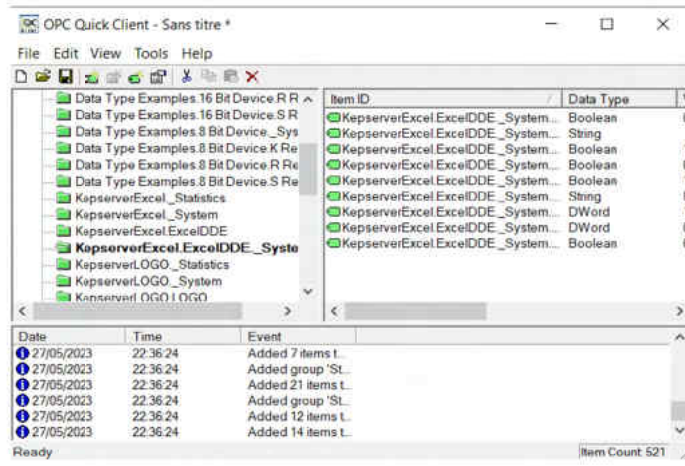


Figure D.9 : Simulateur KepserverEX

# Annexe E

## Les Programmes du Contrôle de La Station de Pompage

### E.1 Programmes du premiere cahier des charges

#### E.1.1 Programme du Zelio Soft

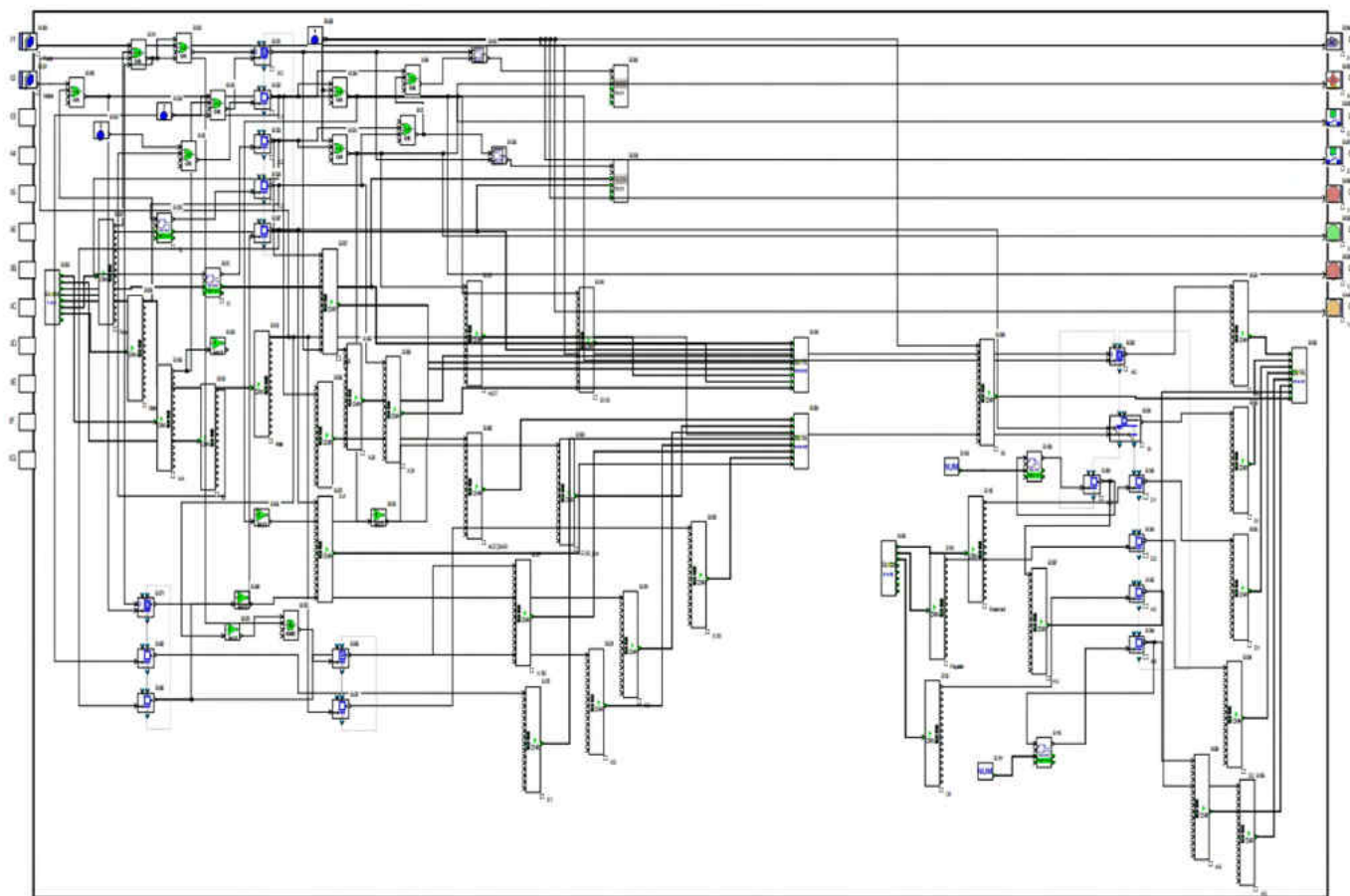


Figure E.1 : Programme des cahiers des charges Sur Zelio Logic

#### E.1.2 Programme LOGO!



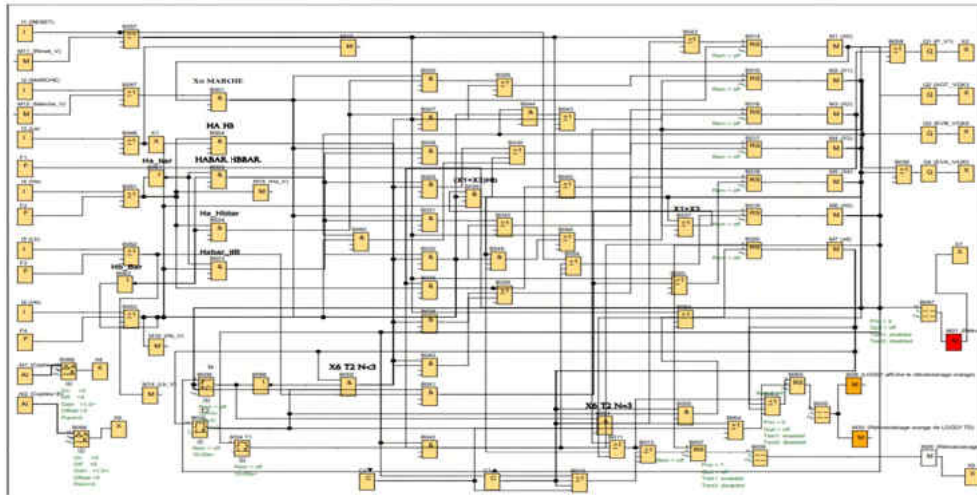


Figure E.2 : Programme du 1er cahier des charges pour la supervision sur LOGO!TDE

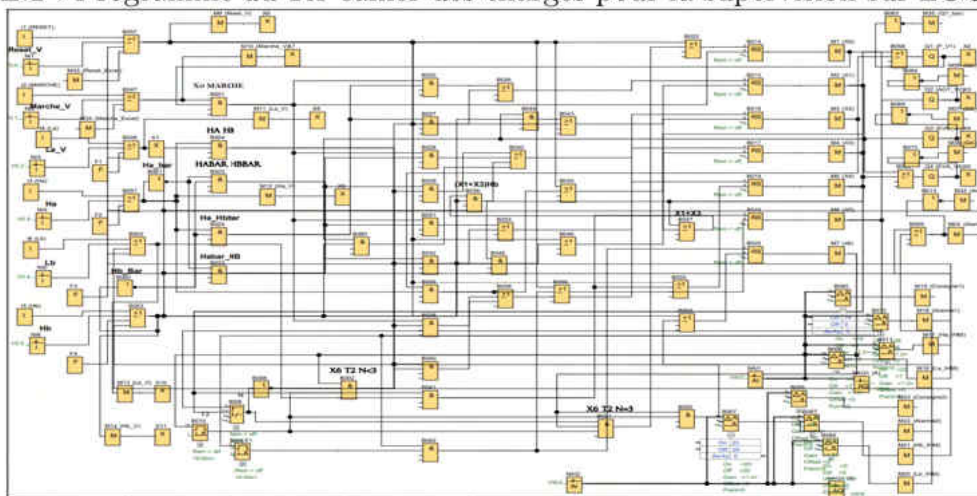


Figure E.3 : Programme du 1er cahier des charges pour la supervision sur Easybuilder Pro

## E.2 Programmes deuxieme cahier des charge

### E.2.1 Programme du Zelio Soft

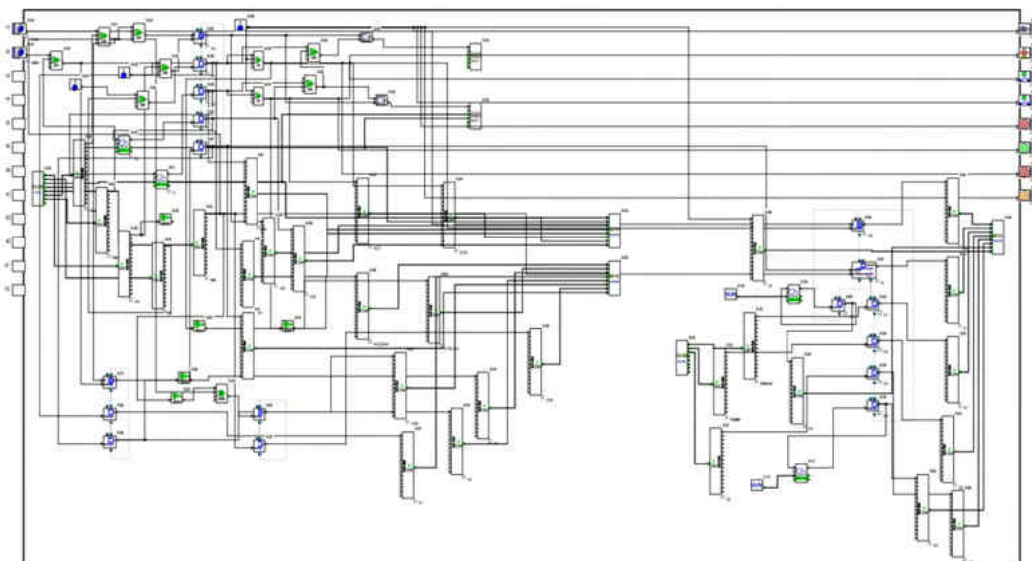


Figure E.4 : Programme deuxieme cahier des charges Sur Zelio Logic

## E.2.2 Programme du LOGO!

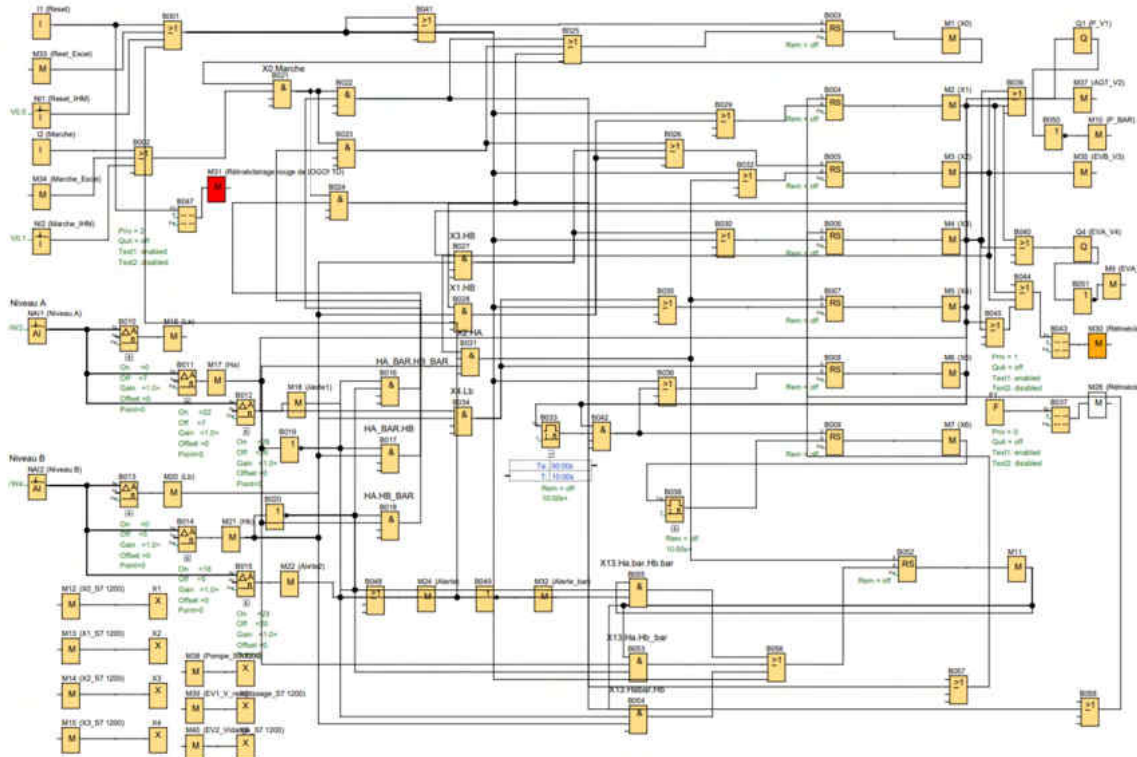


Figure E.5 : Programme deuxième cahier des charges Sur LOGO!

# Annex F

## Schéma Electrique

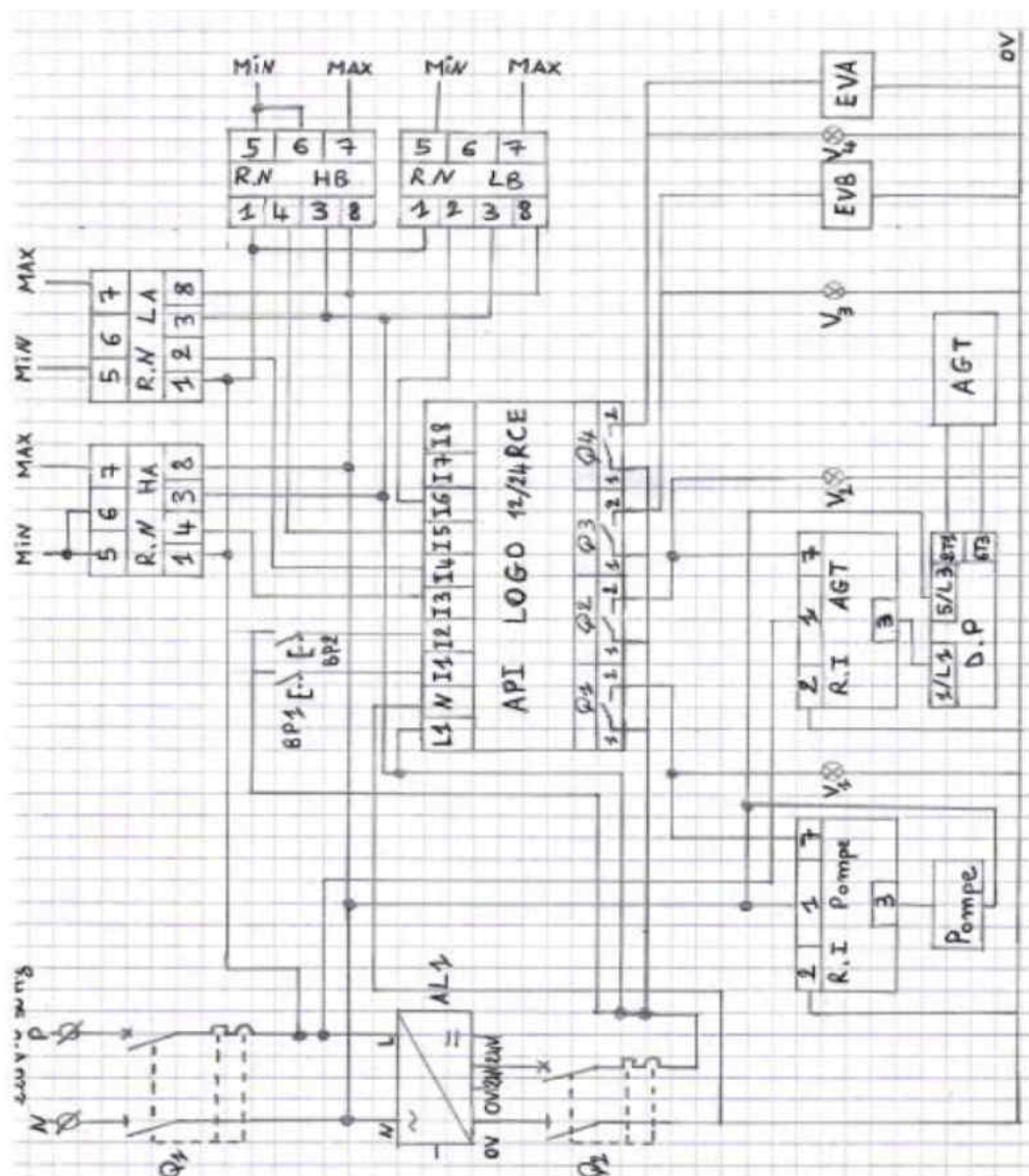


Figure F.1 : Schémas de raccordement LOGO!

# Annexe G

## Guide d'Etude de Mode de Marche et d'Arrêt GEMMA

### G.1 Guide d'Etude de Mode de Marche et d'Arrêt GEMMA

Le GEMMA est un outil graphique d'aide complémentaire au GRAFCET, développé par l'Agence pour le Développement de la Productique Appliquée ADEPA [1], en avril 1981. Il permet de définir les modes de marche et d'arrêt du système automatisé en décrivant tous les états du système, ainsi que les transitions entre ces états. De plus, il permet de détailler les procédures à mettre en œuvre après l'analyse d'une défaillance ou d'un défaut de fonctionnement.

Le GEMMA est rempli progressivement lors de la conception du système.

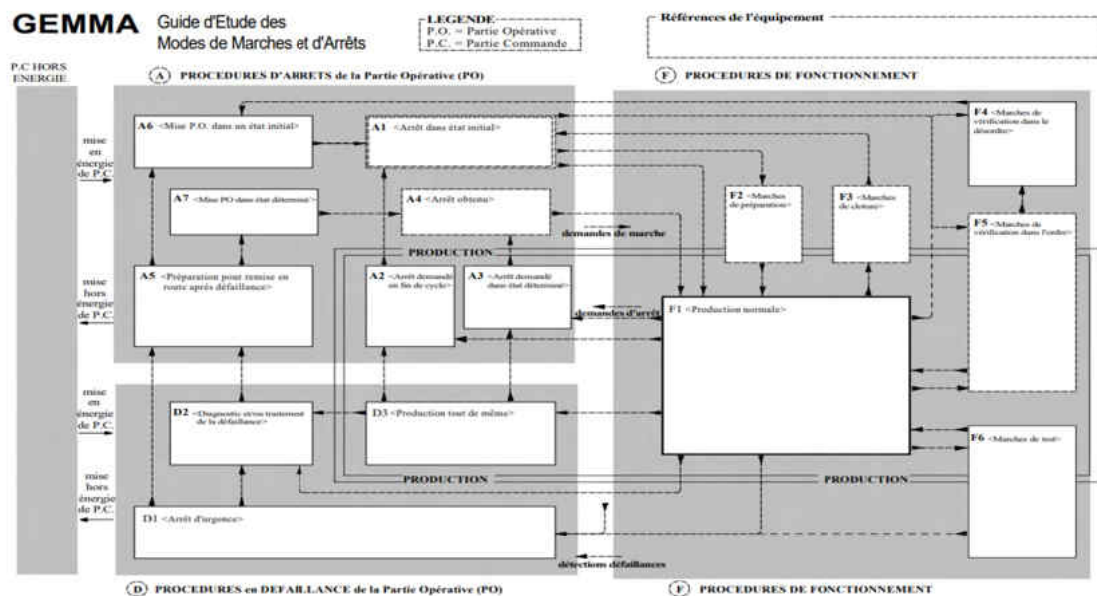


Figure G.2 : Feuille de GEMMA

### G.2 Structure de GEMMA

La feuille de GEMMA est organisée en deux parties :

**a. Partie commande hors énergie :** Cette partie située à l'extrême gauche correspond à l'état inopérant de la partie commande. Dans cette partie il n'y a pas de modes traités par la partie commande, seules les actions externes peuvent se réaliser par exemple : Enclencher et couper le disjoncteur principal.

**b. Partie commande en énergie :** Cette partie permet de définir les différents modes de marche et d'arrêt du système automatisé et les conditions de transition d'un mode à un autre. Chacun de ces modes correspond à un **rectangle état** disposé sur le graphisme selon une structure précise. Cette partie regroupe trois familles des états de modes de marches et d'arrêts qui sont :

— **Famille F Procédures de fonctionnement :** définissent les états de fonctionnement du système en fonctionnement normal et lors des essais de vérification. Dans cette famille nous avons 6 procédures de fonctionnement qui sont :

1. **F1 < Production normale > :** Dans cet état, le système fonctionne normalement selon le cahier des charges.
2. **F2 < Marche de préparation > :** Cet état est utilisé pour les systèmes nécessitant une préparation préalable à la production normale, tels que le préchauffage de l'outillage ou le remplissage.
3. **F3 < Marche de clôture > :** Cet état est utilisé pour les systèmes nécessitant d'être positionnés dans un état spécifique avant l'arrêt, par exemple le nettoyage des équipements ou le retour des vérins à leur état initial.
4. **F4 < Marche de vérification dans le désordre > :** Cet état permet de vérifier les différents actionneurs du système automatisé sans suivre l'ordre de la production normale.
5. **F5 < Marche de vérification dans l'ordre > :** Cet état permet de vérifier les différents actionneurs du système automatisé en respectant l'ordre de la production normale.
6. **F6 < Marche de test > :** Cet état permet d'effectuer des opérations de réglage sur la machine, que ce soit pendant la production ou en dehors de la production.

— **Famille A Procédures d'arrêt et de remise en route de la partie opérative :** Les procédures d'arrêt définissent les états de d'arrêt du système. Dans cette famille nous avons 7 procédures de fonctionnement qui sont :

1. **A1 < Arrêt dans l'état initial > :** C'est l'état repos du système. Il correspond en général à la situation initiale du grafset.
2. **A2 < Arrêt demandé en fin de cycle > :** Cet état permet au système de passer à un arrêt à la fin d'un cycle de production. Le système poursuit sa production et s'arrête une fois que le cycle en cours est terminé.
3. **A3 < Arrêt demandé dans un état déterminé > :** Cet état permet d'amener le système à un arrêt à une position autre que la fin du cycle. Il permet par exemple d'arrêter le système dans un état permettant une intervention sur un système.
4. **A4 < Arrêt obtenu > :** Le système est en arrêt en une autre position que la fin de cycle.
5. **A5 < Préparation pour remise en route après défaillance > :** Cet état toutes les opérations de remise en route (dégagements, nettoyages, etc.) après une défaillance.

6. **A6 < Mise PO dans état initial >** : Cet état permet de remettre manuellement ou automatiquement la partie opérative dans une position permettant un redémarrage à partir de l'état initial.

7. **A7 < Mise PO dans état déterminé >** : Cet état permet de remettre la partie opérative dans une position permettant un redémarrage à partir d'une position autre que l'état initial.

— **Famille D Procédures de défaillance de la partie opérative** : définissent les états auxquels la partie opérative doit être amenée en cas de défaillance. Dans cette famille nous avons 3 procédures de fonctionnement qui sont :

1. **D1 < Marche ou arrêt en vue d'assurer la sécurité >** : Cet état est activé lors d'un arrêt d'urgence, et il prévoit non seulement les arrêts, mais aussi les cycles de dégagement, ainsi que les procédures et précautions nécessaires pour prévenir ou limiter les conséquences résultant de la défaillance.

2. **D2 < Diagnostic et ou traitement de défaillance >** : Cet état permet à la maintenance maintenance de diagnostiquer l'origine de la défaillance et de planifier le traitement approprié qui permettra le redémarrage du système après la résolution de la défaillance.

3. **D3 < roduction tout de même >** : Parfois, il est nécessaire de poursuivre la production malgré la défaillance du système. Cela peut se traduire par une **production dégradée**, une **production forcée** ou une production nécessitant l'intervention d'opérateurs non prévus dans le cadre de la production normale. [2]

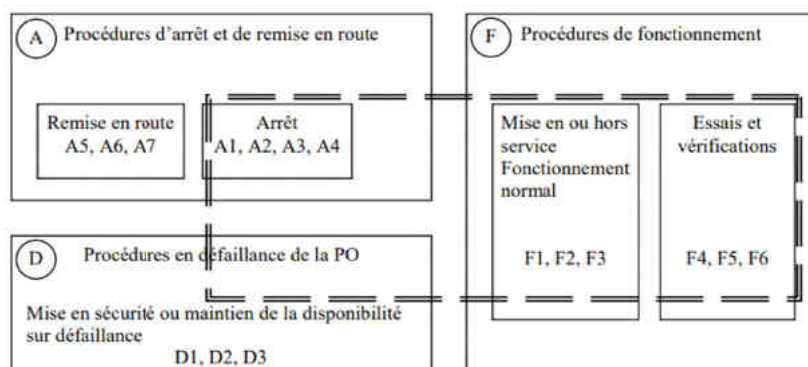


Figure G.2 : Familles et sous familles des procédures [2]

En effet important de noter qu'il n'est pas obligatoire d'utiliser toutes les procédures dans le GEMMA. La sélection et l'utilisation des procédures dépendent des besoins spécifiques du système ainsi que des objectifs de fonctionnement. Chaque système peut nécessiter l'adaptation et la personnalisation des procédures du GEMMA pour répondre aux exigences spécifiques.

# Annexe H

## Support de TP

Cette annexe a pour objectif de permettre de se familiariser avec la programmation des PLCs LOGO! et Zelio Logic à travers l'utilisation de leurs propres logiciels : LOGO!Soft Comfort et Zelio Soft. Pour cela elle présente deux catégories d'exercices : les exercices sur la logique combinatoire et les exercices sur la logique séquentielle. Ces exercices permettent de manipuler différents blocs fonctionnels et d'apprendre à concevoir des programmes qui répondent à des besoins spécifiques.

### H.1. Partie 1 : Logique Combinatoire

Dans cette partie, nous allons apprendre à utiliser les fonctions logiques de base telles que l'ET, le OU et le NON pour réaliser des opérations de contrôle simples. Nous allons également voir comment utiliser les timers et les compteurs pour réaliser des tâches plus complexes telles que la temporisation et le comptage ainsi leurs configuration sur les logiciels LOGO!Soft Comfort et Zelio Soft.

#### Exercice 1

Cet exercice a pour but de permettre aux apprenants de comprendre comment réaliser des opérations logiques de base telles que l'ET, le OU et le NON pour contrôler les sorties d'un automate à partir de deux boutons poussoirs : **Reset** et **Marche**. Les deux boutons poussoirs sont utilisés comme entrées de l'automate, tandis que les sorties contrôlées peuvent être des relais, des voyants ou d'autres dispositifs.


- $Q_1 = \overline{Reset}$
- $Q_2 = Reset \cdot Marche$
- $Q_3 = Marche$
- $Q_4 = Reset + Marche$

#### Exercice 2

Dans cet exercice, nous allons manipuler deux types de timers : le premier est un timer retardé à l'enclenchement (retardé au travail), et le deuxième est un timer retardé au déclenchement (retardé au repos).

- $Q_1 = Reset \cdot T_1$
- $Q_2 = Marche \cdot T_2$

#### 1. Programmation sur LOGO!Soft Comfort

Pour le timer retardé à l'enclenchement, nous utilisons le bloc  tel que **Trig** est l'entrée d'activation du timer. Pour le configurer, il suffit de double-cliquer sur ce bloc et de saisir les paramètres du timer comme illustré ci-dessous :

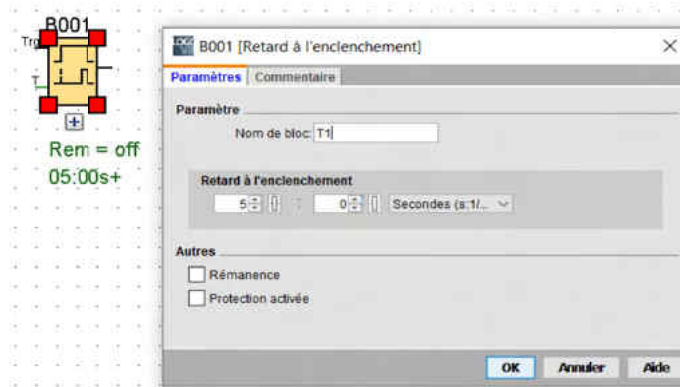



Figure H.1 : Paramètres du timer retardé à l'enclenchement

Pour le timer retardé au déclenchement, nous utilisons le bloc  tel que **Trig** est l'entrée d'activation du timer et **R** est l'entrée pour réinitialiser le timer à 0. Pour le configurer, il suffit de double-cliquer sur ce bloc et de saisir les paramètres du timer comme illustré ci-dessous :

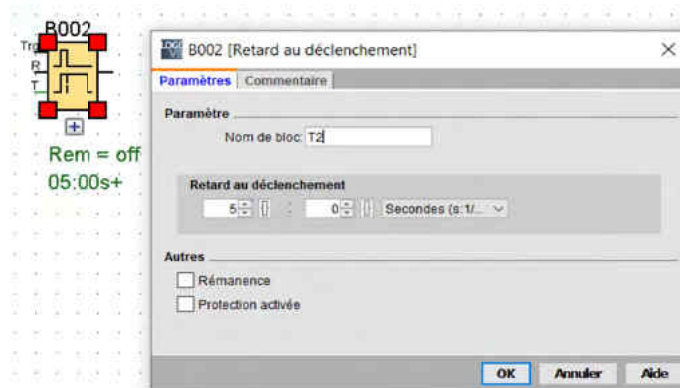



Figure H.2 : Paramètres du timer retardé au déclenchement

## 2. Programmation Zelio Soft

Dans le cas de la programmation avec Zelio Soft, on utilise le bloc fonctionnel  en sélectionnant l'un des timers disponibles. La bobine **T** sert à activer le timer, tandis que la bobine **R** permet de le désactiver. Pour configurer le timer, il suffit de double-cliquer sur la bobine **T** et de choisir le type de timer souhaité. Dans le cas d'un timer retardé à l'enclenchement, on sélectionne la **fonction A**, et dans le cas du timer retardé au déclenchement, on choisit la **fonction C**. Après avoir sélectionné la fonction de timer, il faut saisir la valeur de temporisation souhaitée dans la fenêtre de configuration qui s'ouvre comme illustré ci-dessous :

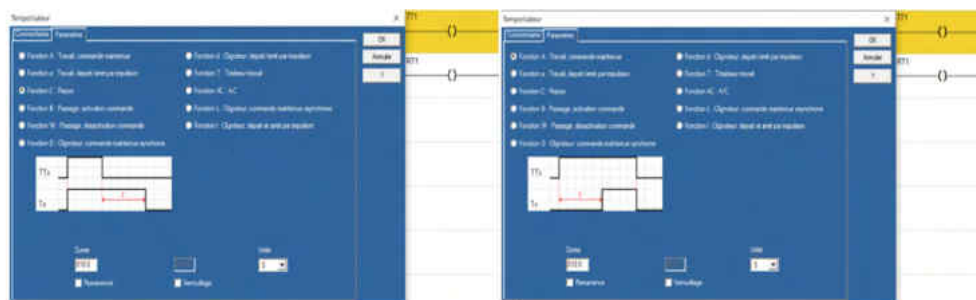


Figure H.3 : Paramètres du timer retardé à l'enclenchement



### Exercice 3

Dans cet exercice, nous allons manipuler les compteurs et les décompteurs.

- $Q_1$  est activé si le compteur  $C_1 \geq 4$  et s'éteint lorsque  $C_1 \geq 8$ . (La valeur de lancement est 0)
- $Q_2$  est activé si le compteur  $C_2 < 5$ . (La valeur de lancement est 9)

#### 1. Programmation sur LOGO!Soft Comfort

Pour programmer le compteur avec LOGO!Soft Comfort nous utilisons le bloc  tel que :

- **R** : l'entrée pour réinitialiser le timer à la valeur de lancement.
- **Cnt** : l'entrée de comptage/décomptage.
- **Dir** : Cette entrée définit le sens de bloc en comptage ou décomptage. Lorsqu'elle est désactivé ( à 0 ), le bloc est en état de comptage. Si elle est activé ( à 1 ), le bloc est en état de décomptage.

Pour configurer le compteur configurer, il suffit de double-cliquer sur ce bloc et de saisir les paramètres souhaités comme illustré ci-dessous :

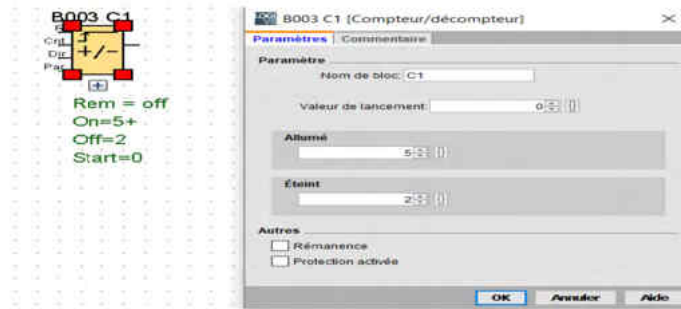



Figure H.4 : Paramètres du compteur/décompteur

#### 2. Programmation Zelio Soft

Lors de la programmation avec Zelio Soft, on utilise le bloc fonctionnel  en sélectionnant l'un des compteurs disponibles. La bobine **C** sert à activer le compteur, tandis que la bobine **R** permet de le désactiver. La bobine **D** définit l'état du compteur : si elle est désactivée, le compteur est en état de comptage, tandis que s'il est activé, le compteur est en état de décomptage.

Pour configurer le compteur, il suffit de double-cliquer sur la bobine **C** et de saisir la valeur de lancement dans la fenêtre de configuration qui s'ouvre. Cette valeur correspond à la valeur initiale du compteur, qui sera incrémentée ou décrémentée en fonction de son état de comptage ou de décomptage comme illustré ci-dessous :

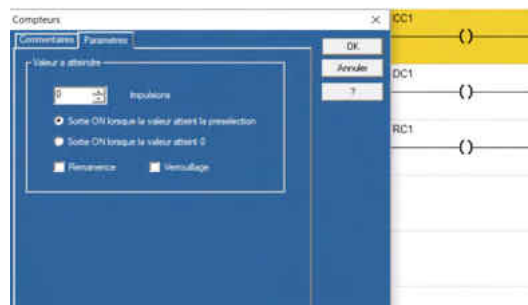


Figure H.5 : Paramètres du timer retardé à l'enclenchement

## H.2. Partie 2 : Logique Séquentielle

Dans cette partie, nous allons manipuler les mêmes blocs fonctionnels que dans la partie sur la logique combinatoire, mais les tâches à accomplir seront plus complexes. Les entrées seront cette fois-ci les boutons poussoirs **Marche** et **Reset** ainsi que les touches  $F_1, F_2, F_3$  et  $F_4$  pour le module LOGO! TDE (respectivement  $Z_1, Z_2, Z_3$  et  $Z_4$  pour le module Zelio)

### Exercice 1

Le système étant à l'arrêt, pour le redémarrer, il faut appuyer sur le bouton **Reset**. Le système sera alors à l'étape 0, où la sortie  $Q_1$  est activée.

- Lorsqu'on appuie sur le bouton **Marche** et la touche  $F_1$  simultanément, le système passe à l'étape 1, où la sortie  $Q_2$  est activée.
- Lorsqu'on appuie sur la touche  $F_2$ , le système passe à l'étape 2, où la sortie  $Q_3$  est activée.
- Lorsqu'on appuie sur la touche  $F_3$ , le système passe à l'étape 3, où la sortie  $Q_4$  est activée.
- Enfin, lorsqu'on appuie sur la touche  $F_4$ , le système revient à l'étape 0.

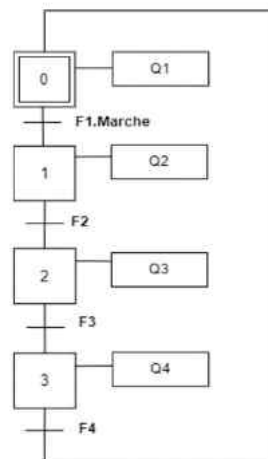


Figure H.6 :Grafcet exercice 1

### Exercice 2

Le système étant à l'arrêt, pour le redémarrer, il faut appuyer sur le bouton **Reset**. Le système sera alors à l'étape 0.

- Pour passer de l'étape 0 à l'étape 1, il faut appuyer simultanément sur le bouton **Marche** et la touche  $F_1$ . La sortie  $Q_1$  sera alors activée.
- Pour passer de l'étape 1 à l'étape 2, il suffit d'appuyer sur la touche  $F_2$ . Le timer sera alors activé pour une durée de 10 secondes, à la fin desquelles le système passe à l'étape 3 et la sortie  $Q_2$  sera activée.
- Pour revenir à l'étape 0 à partir de l'étape 3, il suffit d'appuyer sur la touche  $F_3$ .

- Pour passer de l'étape 0 à l'étape 4, il faut appuyer simultanément sur le bouton **Marche** et la touche  $F_2$ . La sortie  $Q_4$  sera alors activée.
- Pour passer de l'étape 4 à l'étape 5, il faut appuyer sur la touche  $F_4$ . Le timer 2 sera alors activé pour une durée de 10 secondes, à la fin desquelles le système reviendra automatiquement à l'étape 0.

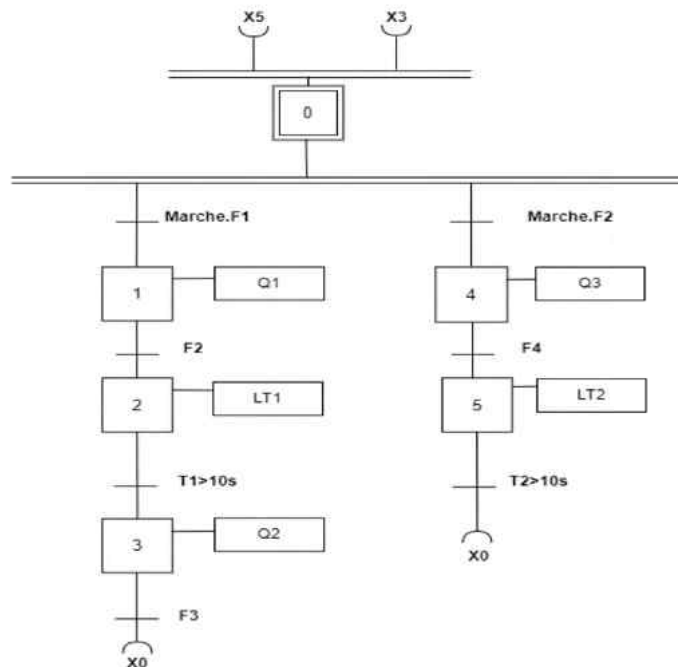


Figure H.7 :Grafcet exercice 2

### Exercice 3

Le système étant à l'arrêt, pour le redémarrer, il faut appuyer sur le bouton **Reset**. Le système sera alors à l'étape 0. La sortie  $Q_1$  sera activée et le compteur sera remis à 0 ( $N=0$ ).

- Lorsqu'on appuie sur le bouton **Marche**, le système passe à l'étape 1, où la sortie  $Q_2$  est activée.
- Lorsqu'on appuie sur la touche  $F_1$ , le système passe à l'étape 2, où la sortie  $Q_3$  est activée et le compteur s'incrémentera.
- Lorsqu'on appuie sur la touche  $F_2$ , le système passe à l'étape 3, où la sortie  $Q_4$  est activée.
- Lorsqu'on appuie sur la touche  $F_3$ , deux scénarios sont possibles. Si la valeur du compteur est égale à 3 ( $N=3$ ), le système reviendra à l'étape 0. Dans le cas contraire, il reviendra à l'étape 2.

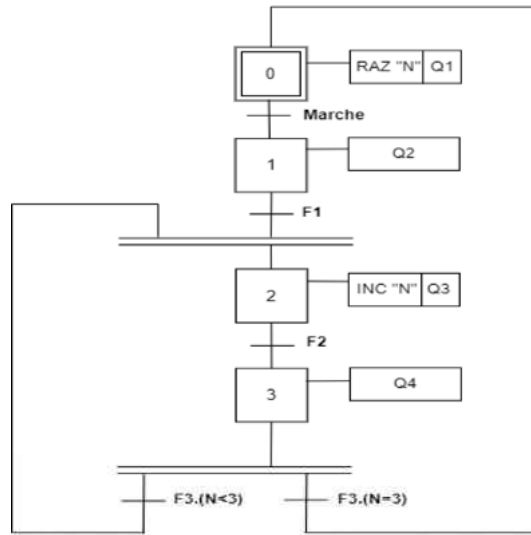


Figure H.8 :Grafcet exercice 3