الجمهورية الجزائرية الديمقراطية الشعبية République Algérienne démocratique et populaire

وزارة التعليم السعسالي والبحث العلمسي Ministère de l'enseignement supérieur et de la recherche scientifique

> جامعة سعد دحلب البليدة Université SAAD DAHLAB de BLIDA

> > كلية التكفولوجيا Faculté de Technologie

قسم الآلية والإلكترو تقتي Département d'Automatique et Électrotechnique

Mémoire de Master

Filière : Automatique Spécialité : Automatique et Système

> Présenté par AIT SAID Tinhinane

Programmation et Supervision d'un Système de Pompage

Proposé par : Dr. CHENTIR Amina

Année Universitaire 2021-2022



Je remercie Allah, le tout puissant de m'avoir donné la volonté d'entamer et de terminer ce mémoire qui est le fruit d'efforts assemblés de plusieurs personnes.

Je veux tout d'abord adresser toute mes reconnaissances à ma chère enseignante et promotrice **Mme. CHENTIR AMINA**, pour tout ce que j'ai appris d'elle. Je la remercie aussi pour la qualité de son encadrement exceptionnel, sa patience, sa rigueur et sa disponibilité durant ma préparation de ce mémoire.

Mes remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à mon travail, en acceptant de l'examiner et de l'enrichir par leurs propositions.

Aussi à tous ceux qui de près ou de loin ont contribué à l'accomplissement de ce travail

Ait said Tinhinane.

Je dédie ce travail à mes chers parents, pour leur amour, leurs sacrifices et leurs prières qui m'ont rendue toujours plus forte tout au long de mon chemin de vie.

A mon adorable petit frère Taha et ma sœur Hiba, je dis merci pour tous leurs soutiens et leurs encouragements.

A tous mes amis sans exception, pour leur soutien et leurs encouragements durant ces années d'études.

A toute personne qui occupe une place dans mon cœur. Et à ma très chère grand-mère qui a toujours cru en moi. Qu'Allah les protège et les bénisse.

Tinhinane.

ملخص:

الغرض من مشروع نهاية الدراسات هذا هو تحقيق أتمتة نموذج عام لمحطة الضخ. للقيام بذلك ، تم اختيار وحدة تحكم منطقية صناعية قابلة للبرمجة (PLC) من نوع Siemens S7-1500 و هذا بسبب أفضل مزاياها مثل مرونتها الكبيرة وموثوقيتها وقدرتها على تلبية المتطلبات الحالية مثل التحكم والاتصال.

يتم تنفيذ برمجة هذه المحطة بواسطة برنامج TIA Portal. أثناء تصور هذه المحطة والتحكم فيها ، سنستخدم شاشة تعمل باللمس من نوع HMI (واجهة الإنسان والآلة) من نوع SAMKOON SK-102 للإشراف وأخيراً سننتقل إلى أرشفة البيانات بواسطة نظام إدارة قواعد البيانات العلائقية MySQL من خلال الأداة المجانية مفتوحة المصدر NodeRed التي تسمح بالاتصال بين واجهة برمجة التطبيقات وقاعدة البيانات.

كلمات المفاتيح :NodeRed *:*MySQL *:* SAMKOON *:*TIA Portal *:* API S7-1500 بمحطة الضخ. Résumé :

Le but de ce projet de fin d'études consiste à réaliser l'automatisation d'un modèle général d'une station de pompage. Pour ce faire, un automate programmable industriel (API) de type S7-1500 Siemens a été choisi et cela à cause de ses meilleurs avantages telles que sa grande souplesse, sa fiabilité et sa capacité à répondre aux exigences actuelles comme la commande et la communication.

La programmation de cette station est effectuée par le logiciel TIA Portal. Tandis que pour visualiser et commander cette station, on utilisera un écran tactile IHM (Human Machine Interface) de type SAMKOON SK-102 pour la supervision et enfin on va procéder à l'archivage des données par le système de gestion de bases de données relationnelles MySQL à travers l'outil open source gratuit NodeRed permettant la connexion entre l'API et la base de données .

Mots clés : API S7-1500; TIA Portal; SAMKOON; MySQL; NodeRed ; Station de pompage.

Abstract :

The aim of this final study project is to achieve the automation of a general model of a pumping station. For this purpose, a programmable logic controller (PLC) of the type S7-1500 Siemens was chosen because of its best advantages such as its high flexibility, reliability and its ability to meet current requirements such as control and communication.

Programming of this station is done by the TIA Portal software. For the visualization and monitoring of the station, a SAMKOON SK-102 HMI (Human Machine Interface) touch screen will be used. Finally, the data will be archived using the MySQL relational database management system through the free open source tool NodeRed allowing the connection between the API and the database. Keywords : API S7-1500 ; TIA Portal ; SAMKOON ; MySQL ; NodeRed ; pumping station.

Table des Matières

| Rem | erci | ements |
|-------|-------|---|
| Dédi | icace | es |
| Résu | ımés | 6 |
| Tabl | e de | es Matières |
| Liste | e des | s Figures et Tableaux |
| Liste | e des | s Abréviations |
| Intro | oduc | ction Générale1 |
| Cha | pitre | e 1 : Analyse Fonctionnelle de la Station de Pompage |
| 1.1 | Int | roduction |
| 1.2 | Pré | ésentation générale de la station de pompage |
| 1.3 | Or | ganigramme général de fonctionnement du système 4 |
| 1.4 | Mo | odes de fonctionnement du système 5 |
| 1.4 | 4.1 | Mode automatique |
| 1.4 | 4.2 | Mode local |
| 1.4 | 4.3 | Mode arrêt 6 |
| 1.5 | Eta | ats de fonctionnement des pompes 6 |
| 1.: | 5.1 | État disponible 6 |
| 1.: | 5.2 | État indisponible |
| 1.6 | Dé | fauts et alarmes |
| 1.0 | 5.1 | Définitions 7 |
| 1.0 | 5.2 | Types de défauts7 |
| 1.0 | 5.3 | Gestion des défauts |
| 1.0 | 5.4 | Logigramme des réactions automatiques en cas de défaut9 |
| 1.7 | Co | nclusion11 |
| Cha | pitro | e 2 : La Partie Opérative et la Partie Commande |
| 2.1 | [ntro | duction 12 |
| 2.2 | Less | systèmes automatisés 12 |
| 2.3 | Obje | ctifs d'un système automatisé12 |

| 2.4 Structure d'un système automatisé 12 | | |
|--|---|----|
| 2.5 Partie opérative (PO) 13 | | |
| 2.5.1 | Les capteurs | 13 |
| 2.5.2 | Les actionneurs | 17 |
| 2.6 Pa | rtie commande (PC) | 20 |
| 2.6.1 | Automate Programmable Industriel (API) | 20 |
| 2.6.2 | Les caractéristiques d'un API | 20 |
| 2.6.3 | Architecture d'un API | 21 |
| 2.6.4 | Programmation d'un API | 22 |
| 2.6.5 | Traitement du programme automate | 23 |
| 2.6.6 | Avantages de l'automate programmable industriel | 23 |
| 2.6.7 | Critères de choix d'un automate | 24 |
| 2.6.8 | Présentation de l'API S7-1500 de SIEMENS | 25 |
| 2.6.91 | Domaines d'application du S7 1500 | 27 |
| 2.6.10 | Présentation de la CPU 1515-2PN | 27 |
| 2.6.11 | Présentation des modules d'Entrées / Sorties | 30 |
| 2.7 Con | clusion | 31 |
| Chapitr | e 3 : Programmation et Supervision | |
| 3.1 Intro | duction | 32 |
| 3.2 Le lo | ogiciel de programmation TIA Portal | 32 |
| 3.2.1 | Les avantages du logiciel TIA Portal | 32 |
| 3.2.2 | Vues du TIA Portal | 33 |
| 3.2.3 | Adressage des E/S | 35 |
| 3.2.4 1 | Mémento de cadence | 35 |
| 3.2.5 | Adressage Ethernet de la CPU | 36 |
| 3.2.6] | Les variables API | 37 |
| 3.2.7 | Liaison avec l'automate | 37 |

| 3.2.8 Conception d'un programme avec TIA Portal V16 |
|---|
| 3.3 La supervision |
| 3.3.1 Avantages de la supervision |
| 3.3.2 Principe de fonctionnement d'une application de supervision |
| 3.3.3 IHM (Interface Homme/Machine) |
| 3.3.4 Présentation de l' IHM SAMKOON SK-102 HS 40 |
| 3.4 Création de Projet |
| 3.4.1 Création d'un nouveau projet 43 |
| 3.4.2 Ajout d'un automate |
| 3.4.3 Présentation de l'interface |
| 3.4.4 Configuration matérielle 46 |
| 3.4.5 Les adresses utilisées pour les entrées / sorties |
| 3.5 Programmation de l'automate |
| 3.5.1 Création d'une table de variables 47 |
| 3.5.2 Les blocs de programmation |
| 3.6 La supervision |
| 3.6.1 Création d'un projet et configuration77 |
| 3.6.2 Boîte de dialogue Paramètres de communication |
| 3.6.3 Programme IHM dans SAKTOOL |
| 3.7 Archivage des données par MySQL 81 |
| 3.7.1 Définition d'une base de données |
| 3.7.2 OPC UA 81 |
| 3.7.3 Node Red |
| 3.7.4 XAMPP |
| 3.7.5 Les étapes pour archiver les données 82 |
| 3.8 Conclusion |
| Conclusion Générale |

| Références Bibliographiques | |
|-----------------------------|--|
| | |
| A | |

Annexes

Liste des Figures et Tableaux

Liste des figures

| Figure 1.1 : Présentation de la station de pompage | 3 |
|--|----|
| Figure 1.2 : Les composants de la station | 4 |
| Figure 1.3 : Organigramme de mode de fonctionnement du système | 4 |
| Figure 1.4 : Logigramme des réactions automatique en cas de défaut | 10 |
| Figure 2.1 : Structure d'un système automatisé | 12 |
| Figure 2.2 : Schéma fonctionnel d'un capteur | 13 |
| Figure 2.3 : Poire de niveau | 13 |
| Figure 2.4 : Principe de fonctionnement de la poire | 14 |
| Figure 2.5 : Capteur de niveau | 15 |
| Figure 2.6 : Débitmètre électromagnétique | 16 |
| Figure 2.7 : Transmetteur de pression | 16 |
| Figure 2.8 : Electrovanne | 17 |
| Figure 2.9 : Rôle d'un vérin | 17 |
| Figure 2.10 : Vérin simple effet | 18 |
| Figure 2.11 : Vérin double effet | 18 |
| Figure 2.12 : Moteur asynchrone | 18 |
| Figure 2.13 : Principe de fonctionnement d'un moteur | 19 |
| Figure 2.14 : Pompe | 19 |
| Figure 2.15 : Un automate programmable industriel | 20 |
| Figure 2.16 : Structure externe | 21 |
| Figure 2.17 : Structure interne | 22 |
| Figure 2.18 : Fonctionnement cyclique d'un automate | 23 |
| Figure 2.19 : Automate S7-1500 | 25 |
| Figure 2.20 : Structure d'un automate S7-1500 | 26 |
| Figure 2.21 : CPU 1515-2PN | |
| Figure 2.22 : La CPU 1515-2PN | 28 |
| Figure 2.23 : Module d'entrée TOR | |
| Figure 2.24 : Module d'entrée analogique | 31 |
| Figure 3.1 : Vue du portal | 33 |
| Figure 3.2 : Vue de projet | 34 |
| Figure 3.3 : Adressage des E/S | 35 |
| Figure 3.4 : Mémento de cadence | 36 |
| Figure 3.5 : Adressage Ethernet de la CPU | |

| Figure 3.6 : Organisation pour la conception d'un projet sous TIA Portal38 |
|---|
| Figure 3.7 : IHM dans un processus automatisé |
| Figure 3.8 : Face avant de SK- 102 HS41 |
| Figure 3.9 : Schéma de connexion de l'écran tactile de la série SK42 |
| Figure 3.10 : Création d'un nouveau projet43 |
| Figure 3.11 : Ajout une Configuration43 |
| Figure 3.12 : Ajout un automate S7-150044 |
| Figure 3.13 : Ajout CPU S7-1515-2PN44 |
| Figure 3.14 : Présentation de l'interface45 |
| Figure 3.15 : Configuration matérielle46 |
| Figure 3.16 : Adresses utilisées par le différents modules de l'automate47 |
| Figure 3.17 : Table de variables |
| Figure 3.18 : Les différents blocs49 |
| Figure 3.19 : Les blocs de programme de l'automate49 |
| Figure 3.20 : Blocs d'organisation OB150 |
| Figure 3.21 : Permutation 3+1 |
| Figure 3.22 : Réservoir d'aspiration53 |
| Figure 3.23 : Réservoir de refoulement54 |
| Figure 3.24 : Pompe 1 |
| Figure 3.25 : Pompe 2 |
| Figure 3.26 : Pompe 3 |
| Figure 3.27 : Pompe de secours60 |
| Figure 3.28 : Fonction FC1 Software de la pompe61 |
| Figure 3.29 : Défaut de discordance |
| Figure 3.30 : Lancement de la commande temporisation réglable par l'opérateur62 |
| Figure 3.31 : Reset de la commande après l'écoulement de la temporisation62 |
| Figure 3.32 : Temporisation classe F63 |
| Figure 3.33 : Comptage de nombre démarrage pompe pendant une temporisation63 |
| Figure 3.34 : Reset de défaut automatique après l'écoulement de la tempo classe F64 |
| Figure 3.35 : Chaine de la disponibilité de la pompe64 |
| Figure 3.36 : Ordre de marche de la pompe64 |
| Figure 3.37 : Ordre d'arrêt65 |
| Figure 3.38 : Fonction FC2 Permutation 3+165 |
| Figure 3.39 : Commande arrêt des 4 pompes conditionnel |

| Figure 3.40 : Commande pompe 1 |
|---|
| Figure 3.41 : Commande pompe 267 |
| Figure 3.42 : Commande pompe 367 |
| Figure 3.43 : Commande pompe de secours prête |
| Figure 3.44 : Fonction FC3 Réservoir d'aspiration |
| Figure 3.45 : Défaut niveau très bas d'aspiration69 |
| Figure 3.46 : Défaut discordance vanne d'aspiration ouverture69 |
| Figure 3.47 : Défaut discordance vanne d'aspiration fermeture70 |
| Figure 3.48 : Défaut incohérence71 |
| Figure 3.49 : Disponibilité vanne pneumatique71 |
| Figure 3.50 : Ordre d'ouverture vers le distributeur bistable72 |
| Figure 3.51 : Ordre fermeture vers le distributeur bistable72 |
| Figure 3.52 : Fonction FC4 Réservoir de refoulement73 |
| Figure 3.53 : Alarme niveau très bas73 |
| Figure 3.54 : Défaut discordance ouverture vanne de refoulement74 |
| Figure 3.55 : Défaut discordance fermeture vanne de refoulement74 |
| Figure 3.56 : Incohérence vanne de refoulement75 |
| Figure 3.57 : Disponibilité75 |
| Figure 3.58 : Ordre d'ouverture de vanne de refoulement76 |
| Figure 3.59 : Ordre de fermeture de vanne de refoulement76 |
| Figure 3.60 : Blocs de données DB OPC MySQL77 |
| Figure 3.61 : Création d'un projet78 |
| Figure 3.62 : Boite de dialogue |
| Figure 3.63 : Process de système de IHM80 |
| Figure 3.64 : Commande du système de IHM80 |
| Figure 3.65 : Les alarmes du système de IHM81 |
| Figure 3.66 : Adresse de server OPC UA |
| Figure 3.67 : La fenêtre de OPC UA82 |
| Figure 3.68 : Adresse de server OPC UA83 |
| Figure 3.69 : Adresse de server OPC UA |
| Figure 3.70 : La fenêtre de Xampp |
| Figure 3.71 : Adresse de server OPC UA |
| Figure 3.66 : Archivage des données85 |

Liste des tableaux

| Tableau 2.1 : Les caractéristiques de la CPU 1515-2PN | 29 |
|---|----|
| Tableau 3.1 : Tableau de caractéristique de SK-102HS | 41 |

Liste des Abréviations

- **API**: Automate Programmable Industriel.
- **BP** : Bouton Poussoir.
- **CPU :** Central Processing Unit.
- DIN : Deutsches Institut für Normung.
- E/S: Entrées/ Sorties
- IHM : Interfaces Homme Machine.
- I/O : Input / Output.
- **IRT :** Real-time industrial communication.
- **LED**: Light-Emitting Diode.
- MPI : Multi Point Interface.
- **NB**: Niveau Bas.
- NC : Niveau Conditionnel.
- **NH :** Niveau Haut.
- NTB : Niveau Très Bas.
- **OPC UA :** OPen Connectivity Unified Architecture.
- **P-ASP** : Pompe d'ASPiration.
- PC : Partie Commande.
- PID : Proportionnelle, Intégrateur, Dérivateur.
- PO: Partie Opérative.
- **PROFIBUS** : PROcess FIeld BUS.
- **PROFINET :** PROcess Field NETwork.
- **P-SC :** Pompe de secours.
- SCADA : Supervisory Control and Data Acquisition
- TIA PORTAL : Totally Integrated Automation Portal.
- TOR : Tout Ou Rien.
- V-ASP : Vanne d'aspiration.
- **V-REF :** Vanne de refoulement.

Introduction Générale

L'automatique est une science qui traite de la modélisation, de l'analyse, de l'identification et de la commande des systèmes dynamiques. Elle inclut la cybernétique au sens étymologique du terme. Elle permet de commander un système en respectant un cahier des charges (rapidité, précision, stabilité...), permet aussi la programmation et la gestion d'information pour différents 'outils de production industriel.

L'automation industrielle est le recours à « une technique qui assure le fonctionnement d'une machine ou d'un groupe de machines sans intervention humaine », afin de réduire la charge d'activité du travailleur tout en améliorant la productivité et la qualité.

De nos jours l'automatisme industrielle est devenue indispensable dans tous les secteurs de productivité avec les lignes de conditionnement. En effet l'automatisation du processus facilite et augmente la productivité. Aucun domaine n'est resté à l'abri de cette technologie qui facilite les tâches aussi bien pour l'entreprise que pour le personnel.

Ainsi, les systèmes automatisés ont répondu à un besoin vif pour n'importe quel type d'entreprise, soit l'augmentation de la production qui est parmi les enjeux les plus primordiaux pour toutes les entreprises et touche pratiquement toutes les activités telles que la maintenance, la mécanique et l'ingénierie industrielle en général.

Dans le milieu industriel, on assiste de plus en plus à la multiplicité, à la sophistication des instruments et appareils d'automatisation d'une part, et d'autre part à l'exigence de la continuité du développement. Ces paramètres auront pour effets pour le service industriel, de renforcer sa fiabilité de production.

Une station de pompage est une installation comportant une ou plusieurs pompes pour l'extraction de l'eau d'un puits, d'une rivière, ou pour faciliter la circulation dans un réseau d'aqueduc, d'égout, ou de transport par conduites de gaz ou de pétrole.

Dans notre travail, nous allons étudier et adapter un automate programmable pour le fonctionnement d'une station de pompage en général. Autrement dit, notre but est de faire une étude complète et détaillée de la station de pompage contrôlée par un API (Automate Programmable Industriel) de type **S7-1500** qui assurera alors de meilleurs avantages vus de sa

grande souplesse, sa fiabilité, sa capacité à répondre aux exigences actuelles comme la commande.

La programmation sera faite sous le logiciel **TIA PORTAL**, qui permet de simuler le fonctionnement des automates programmables et la supervision a été faite à l'aide d'un écran tactile **IHM** (Human Machine Interface) de SIEMENS de type **SAMKOON SK-102** pour visualiser et commander la station afin d'avoir un suivi en temps réel. Une dernière partie consiste à archiver l'état de la station dans une base de donnée crée grâce à MySQL en utilisant comme interface de liaison, la plateforme gratuite NodeRed.

Pour réaliser ce projet de fin d'étude, le travail présenté dans ce mémoire, sera organisé comme suit :

- Le premier chapitre sera dédié à la présentation du fonctionnement de la station de pompage. Ainsi, on va décrire les modes d'automatisation de cette station et les défauts qui peuvent survenir.
- Le deuxième chapitre sera consacré à présenter les différents éléments utilisés dans la station de pompage et expliquer également le rôle de chaque composant inclus dans la partie opérative et la partie commande du système.
- Dans le troisième chapitre, nous présenterons le logiciel de programmation TIA Portal et on donnera un aperçu sur l'automatisation ainsi que les étapes suivies pour la programmation de la station par automate programmable industriel, la supervision avec un IHM et l'archivage par MySQL dans une base de donnée en programmant une interface de liaison par NodeRed.
- Nous terminons notre mémoire par une conclusion générale suivie de quelques perspectives.

Chapitre 1 Analyse Fonctionnelle de la Station de Pompage

1.1 Introduction

Dans ce chapitre nous allons présenter le fonctionnement d'un modèle de station de pompage en général, en se basant sur des travaux précédents.

Cette étude fonctionnelle a pour but de décrire les modes d'automatisation de cette station ainsi que les défauts qui peuvent survenir.

1.2 Présentation générale de la station de pompage

Une station de pompage est une station servant à pomper l'eau ou plus généralement un fluide, tel que le pétrole par exemple. Elle peut être utilisée pour plusieurs applications telles que l'approvisionnement en eau des canaux, le drainage des terres basses, et l'élimination des eaux usées vers le site de transformation.

L'installation à étudier est présentée sur la Figure 1.1 et est composée :

- D'une station de pompage (4 pompes en mode de fonctionnement 3+1).
- D'un réservoir d'aspiration.
- Et d'un réservoir de refoulement.



L'alimentation de la station de pompage se fait depuis un réservoir d'une capacité X m³.

Figure 1.1 : Présentation de la station de pompage



Figure 1.2 : Les composants de la station de pompage

P-asp : Pompe d'aspiration.

P-SC : Pompe de secours.

1.3 Organigramme général de fonctionnement du système

En plus du mode arrêt, le système fonctionne selon deux modes : local et automatique, comme le montre la **Figure 1.2**.



Figure 1.2 : Organigramme de mode de fonctionnement du système

1.4 Modes de fonctionnement du système

Le système de pompage fonctionne en 3 modes :

- Mode automatique.
- Mode local (semi-distant).
- Mode arrêt.

1.4.1 Mode automatique

A l'activation du mode automatique le mode local sera désactivé automatiquement s'il est actif. Dans ce mode, la permutation des pompes fonctionnera selon le niveau du réservoir d'aspiration suivant l'ordre suivant :

- NB (niveau bas) : P1.
- NC (niveau conditionnel) : P1, P2.
- **NH** (niveau haut) : P1, P2, P3.

La 4 ^{-ème} pompe P-SC sera une pompe de secours. Son ordre de marche sera activé à l'indisponibilité d'une des 3 pompes fonctionnels, bien sûr selon le niveau d'aspiration.

Concernant les vannes pneumatiques d'aspiration et de refoulement, on a :

V-ASP : cette vanne s'ouvre automatiquement au démarrage de l'installation si le niveau d'aspiration est vérifié (niveau très bas NTB non vérifié).

Si le niveau NTB est vérifié la vanne se ferme automatiquement et le système sera en arrêt avec un déclenchement d'une alarme au niveau de l'IHM.

V-REF : cette vanne restera ouverte si le niveau NH n'est pas vérifiée sinon elle sera fermée automatiquement.

La phase de mesure des grandeurs physique sera assurée par une partie d'instrumentation composée par un transmetteur de pression et un débitmètre avec un collecteur software par impulsion.

1.4.2 Mode local

Pour travailler en ce mode, il faut que l'opérateur lance le mode local (C-LOCAL) présent dans la vue PROCESS sur l'IHM.

Lorsque le mode « Local » est choisi, les actionneurs seront commandés par les boutons poussoirs existants sur l'armoire de commande.

En principe ce mode n'est choisi que pour des opérations de test. Dans cette position toutes les actions sont commandées par l'opérateur depuis la boite BP locale. Seules restent actives, les fonctions propres qui touchent à la sécurité des personnes.

Dans ce mode, les autorisations liées aux temps de fonctionnement de chaque pompe ne sont pas valides. Les sécurités liées au niveau du réservoir (niveau bas anti marche à sec NTB) sont prises en charge (l'arrêt de pompes se fait directement par câblage).

1.4.3 Mode arrêt

La station sera mise en arrêt à l'action d'une :

- Désactivation du ON PROCESS dans l'IHM.
- L'absence du choix des modes de fonctionnement auto et local au même temps.
- Action opérateur sur l'arrêt d'urgence.
- Déclenchement d'une alarme à un classement dangereux (définie par l'ingénieur de process).

1.5 Etats de fonctionnement des pompes

Les pompes utilisées ne peuvent fonctionner que sous l'un des deux états suivants :

- Etat disponible.
- Etat indisponible

1.5.1 État disponible

Une pompe est en état disponible, si elle est prête à démarrer (ne présente aucun défaut mécanique ou électrique) ou déjà en fonctionnement.

1.5.2 État indisponible

Une pompe est en état indisponible si une des conditions suivantes est vraie :

- La pompe est en mode local.
- La pompe est en défauts (électrique/physique).
- Absence d'alimentation en tension.

1.6 Défauts et alarmes

Dans un système automatisé, le matériel utilisé est sujet à des défauts, ce qui peut engendrer le déclenchement d'alarmes.

1.6.1 Définitions

Défaut : On appelle défaut un événement dont l'origine est une anomalie physique.Alarme : On définit l'alarme comme une représentation visuelle du défaut.L'alarme est l'information issue du défaut. Elle est transmise au poste SCADA.

1.6.2 Types de défauts

Les défauts rencontrés dans un système automatisé sont classés comme suit :

Les défauts câblés

Ces défauts sont détectés directement par câblage depuis les armoires électriques. Ils correspondent à :

- Des dysfonctionnements liés à la sécurité des hommes : arrêts d'urgences, etc...
- > La protection des vannes : les défauts des départs actionneurs.

Ces défauts sont surveillés par le système en permanence et ceci quel que soit le mode de fonctionnement choisi par l'opérateur (position du commutateur), à cet effet une alarme est générée.

Les défauts générés par le programme

Ils peuvent être de plusieurs natures :

- Les défauts liés à une anomalie physique d'un ou plusieurs constituants de l'équipement. Par exemple : discordance entre la commande et le retour d'état du contacteur, etc.... Ces défauts sont calculés par le système uniquement lorsque l'équipement est commandé depuis l'automate.
- Les défauts système : Ils correspondent à des anomalies physiques liées aux organes du système (par exemple : défaut de la pile de l'automate).

1.6.3 Gestion des défauts

Les défauts électriques

Définition

Ce défaut câblé regroupe les dysfonctionnements liés à la chaîne de commande de l'actionneur : thermique, température moteur, etc. ...

Il est généré quel que soit le mode de marche de l'actionneur et est surveillé par le système même quand l'actionneur est à l'arrêt.

> Conséquence

En principe ce défaut est traité par les armoires électriques. Au niveau programme l'équipement concerné passe en position de sécurité (Arrêt des moteurs des vannes). Une alarme est immédiatement générée.

Disparition de défaut

A la disparition du défaut électrique, l'actionneur reste en position de sécurité tant que l'opérateurn'a pas effectué une procédure d'acquittement (acquittement depuis l'armoire ou bien depuis le poste SCADA).

Défaut de discordance

> Définition

Ce défaut a pour objet de détecter les incohérences qui peuvent se produire entre l'ordre émis par l'automate et le retour d'état réel d'un actionneur.

Ce défaut n'est généré que si l'équipement est commandé depuis l'automate.

Conséquence

Ce défaut provoque immédiatement le passage en position de sécurité du moteur. Une alarme est immédiatement générée.

L'actionneur reste en position de sécurité tant que l'opérateur n'a pas effectué une procédure d'acquittement.

Incohérences capteurs TOR

Définition

Les capteurs sont importants pour réaliser la gestion du fonctionnement des différents équipements (marche / arrêt de pompe par exemple) et il est donc nécessaire de contrôler en permanence que les états seuils de niveau sont cohérents les uns par rapport aux autres pour éviter les dysfonctionnements des équipements dus à une défaillance d'un capteur ou d'une rupture de fils.

Le défaut « incohérence capteurs de niveau » est donc généré si au moins deux mesures contradictoires sont vraies en même temps (par exemple si un niveau est détecté en même temps haut et bas).

> Conséquence

Une alarme est immédiatement émise.

Disparition de défaut

Après réparation des capteurs défectueux, le défaut est considéré comme disparu lorsqu'une procédure d'acquittement a été effectuée par l'opérateur.

Défaut entrées analogiques

Ces défauts sont générés sur les conditions suivantes :

- Signal analogique (valeur retransmise) en dessous de 3,8 mA (paramétrable).
- Signal analogique (valeur retransmise) au-dessus de 21 mA (paramétrable).

> Incohérences fins de course et limiteurs de couple

Les fins de courses sont nécessaires pour faire l'arrêt des moteurs, dans le cas d'ouverture (ou bien la fermeture) des vannes, le moteur doit tourner jusqu'à la détection d'ouverture (de fermeture respectivement) de la vanne à partir des fins de courses.

Donc, Il faut contrôler que les états des fins de course sont cohérents les uns par rapport aux autres

Le défaut « incohérence fin de course » est donc généré si deux mesures contradictoires sont vraies en même temps, et ceci dans le cas où on reçoit la détection d'ouverture et de fermeture de la vanne en même temps.

Idem, Le défaut « incohérence limiteur de couple » est donc généré si deux mesures contradictoires sont vraies en même temps, et ceci dans le cas où on reçoit la détection du limiteur de couple d'ouverture et de fermeture en même temps

1.6.4 Logigramme des réactions automatiques en cas de défaut

Disponibilité/Niveau :

Dans le mode automatique on a deux états :

> Etat normal :

- Les trois pompes disponibles (aucun défaut).
- Niveau suffisant d'aspiration et niveau conditionnel de refoulement.

Etat anormal :

- Deux pompes disponibles et une pompe en défaut (Anomalie détectée l'activation de la P-SC est déclenchée automatiquement).
- Niveau insuffisant d'aspiration ou niveau haut du refoulement (Arrêt de l'installation alarme IHM).

La **Figure 1.3** représente le logigramme qui regroupe les réactions automatiques en cas de défaut.



Figure 1.3 : Logigramme des réactions automatique en cas de défaut

1.7 Conclusion

Dans ce chapitre, on a présenté le fonctionnement et les règles d'automatisation de la station de pompage et les défauts.

Dans le chapitre qui suit, nous allons présenter la partie opérative et la partie commande du système.

Chapitre 2

Parties Opérative et Commande

2.1 Introduction

Dans ce chapitre, nous allons parler sur les systèmes automatisés et présenter les différents éléments utilisés dans la station de pompage avec leurs caractéristiques et les principes de fonctionnement de tous les capteurs et actionneurs de la partie opérative ainsi que la partie commande de cette dernière.

2.2 Les systèmes automatisés

Un système est dit automatisé s'il exécute toujours le même cycle de travail après avoir reçu les consignes d'un opérateur. Il effectue une tache répétitive [1].

2.3 Objectifs d'un système automatisé

- Produire une qualité constante.
- Fournir les quantités nécessaires.
- Augmenter la productivité.
- Améliorer les conditions de travail.

2.4 Structure d'un système automatisé

Un système automatisé est composé d'une partie commande (PC) et d'une partie opérative (OP) (Figure 2.1).





2.5 Partie opérative (PO)

Elle reçoit les ordres de la partie commande et adresse aussi des comptes rendus à cette dernière. La partie opérative regroupe l'ensemble des actionneurs et des capteurs, où les actionneurs exécutent les ordres reçus et les capteurs réagissent à l'état du système [1].

2.5.1 Les capteurs

Un capteur est un élément qui transforme une grandeur physique en une information exploitable vers un signal électrique (4...20 mA ou bien 0...10V). Les capteurs peuvent être des capteurs TOR, analogiques ou numériques (**Figure 2.2**).

- Détecteur avec contact : le capteur doit entrer en contact physique avec un phénomène pour le détecter.
- > Détecteur sans contact : le capteur détecte le phénomène à proximité [2].



Figure 2.2 : Schéma fonctionnel d'un capteur [2]

Parmi les différents capteurs rencontrés dans une station de pompage, on a :

La poire

C'est un dispositif suspendu au-dessus d'un plan d'eau au moyen d'un câble électrique (**Figure 2.3**). Il s'agit d'un capteur tout ou rien (TOR) qui adresse un compte rendu qui ne peut prendre que 2 valeurs (oui ou non, 0 ou 1, absence ou présence d'un phénomène).



Figure 2.3 : Poire de niveau [3]

Lorsque le niveau de l'eau augmente jusqu'à immerger le dispositif, sa capacité de flotter et sa forme de poire l'oblige à se retourner mettant ainsi deux fils en contact électrique (Figure2.4).

Les poires nécessaires sont :

- NTB : niveau très bas. C'est pour la sécurité.
- **NB** : niveau bas.
- NC : niveau conditionnel.
- NH : niveau haut. C'est pour éviter le débordement et la sécurité.



Figure 2.4 : Principe de fonctionnement de la poire [4]

> Dans le cas d'une vidange :

Si le flotteur est en haut, le contact est fermé et la pompe se met en marche.

Si le flotteur est en bas, le contact est ouvert et la pompe s'arrête.

> Dans le cas d'un remplissage :

Si le flotteur est en haut, le contact est ouvert et la pompe s'éteint.

Si le flotteur est en bas, le contact est fermé et la pompe se met en marche.

Capteur de niveau

Un capteur de niveau est un dispositif électronique qui permet de mesurer la hauteur du matériau, en général du liquide, dans un réservoir ou un autre récipient.

Ce capteur est utilisé pour assurer une surveillance de niveau de tout un système.

Il mesure le niveau de liquide dans une plage et pour la détection du niveau haut et bas de remplissage des cuves de stockage, des silos et réservoirs. Cette détection rapide rend le contrôle du niveau précis et fiable.

Généralement, ce type de détecteur de niveau fonctionne comme une alarme haute, pour signaler une condition de débordement, ou en tant qu'indicateur pour une condition d'alarme basse (**Figure 2.5**).



Figure 2.5 : Capteur de niveau [5]

Débitmètres électromagnétiques

Un débitmètre électromagnétique est un appareil destiné à mesurer le débit d'un fluide. Ces appareils sont généralement placés en ligne avec le tuyau qui transporte le fluide.

Le fonctionnement d'un débitmètre électromagnétique se base sur la loi de Faraday.

La tension induite est directement proportionnelle à la vitesse du fluide circulant dans le champ magnétique. La tension induite est transmise au transmetteur par le circuit d'électrodes. Le transmetteur convertit ensuite cette tension en vitesse d'écoulement quantifiable. Le débit volumique du fluide est calculé à l'aide de cette vitesse connue et de la superficie de la conduite (**Figure 2.6**).


Figure 2.6 : Débitmètre électromagnétique [6]

> Transmetteur de pression (PT)

Un transducteur de pression, parfois appelé transmetteur de pression, est un transducteur qui transforme la pression en un signal électrique analogique (**Figure 2.7**).



Figure 2.7 : Transmetteur de pression [7]

Electrovanne

Une électrovanne est un appareil de robinetterie qui ouvre ou ferme un circuit par l'envoi d'un courant électrique actionnant un champ électromagnétique créé par la bobine montée sur l'électrovanne (**Figure 2.8**).



Figure 2.8 : Electrovanne [8]

Les électrovannes sont utilisées pour convertir l'énergie électrique en une énergie mécanique.

2.5.2 Les actionneurs

Un actionneur transforme l'énergie (débit, pression) en énergie mécanique sous forme d'un mouvement permettant de provoquer un déplacement (translation, rotation). Parmi eux, on a :

Les vérins

Ils transforment l'énergie d'un fluide sous pressions en énergie mécanique. Ils peuvent accomplir différentes tâches comme le serrage, coupage, tirage (**Figure 2.9**).



Figure 2.9 : Rôle d'un vérin [1]

Leur classification tient compte de la nature du fluide, pneumatique ou hydraulique, et du mode d'action de la tige :

- Simple effet (air comprimé admis sur une seule face du piston tirer ou pousser) (Figure 2.10).
- Double effet (air comprimé admis sur les deux faces du piston tirer et pousser) (Figure 2.11).



Figure 2.10 : Vérin simple effet [1]



Figure 2.11 : Vérin double effet [1]

C'est l'air comprimé qui en pénétrant dans l'une des chambres exerce une force sur le piston ce qui permet le déplacement de la tige. L'air présent dans l'autre chambre est donc évacué du corps du vérin et le mouvement contraire est obtenu si nous inversons le sens de déplacement de l'air comprimé [1].

Moteur asynchrone triphasé

C'est un moteur qui se caractérise par le fait qu'il est constitué d'un stator « inducteur » qui se compose par des bobines alimentées en courant alternatif ce qui résulte un champ tournant qui provoque la rotation d'un rotor « induit » (**Figure 2.12**).

C'est un moteur dont la vitesse est proportionnelle à la fréquence du courant.



Figure 2.12 : Moteur asynchrone [9]

Les moteurs électriques sont les récepteurs les plus nombreux dans les industries et les installations tertiaires. Leur fonction est de convertir l'énergie électrique en énergie mécanique (**Figure 2.13**) [9].



Figure 2.13 : Principe de fonctionnement d'un moteur [1]

> Pompe

Une pompe est une machine destinée à faire circuler un fluide (en général un liquide) en l'aspirant, le refoulant ou en le comprimant (**Figure 2.14**) [10].

De façon générale, une pompe comporte :

- Un organe mobile : la roue ou le piston selon le type de pompe.
- Des organes fixes : à savoir, diffuseur, encore appelé stator et des canaux de retour.



Figure 2.14 : Pompe [11]

2.6 Partie commande (PC)

Elle reçoit les consignes de l'opérateur et les comptes rendus de la partie opérative.

La partie commande adresse des ordres à la partie opérative et des signaux à l'opérateur,

à partir :

- Du programme qu'elle contient.
- > Des informations reçues par les capteurs.
- Des consignes données par l'utilisateur ou par l'opérateur [1].

Généralement, la partie commande est constituée d'un ordinateur connecté à une interface « entrée/sortie ». Mais dans les systèmes industriels, il s'agit souvent d'un automate programmable.

2.6.1 Automate Programmable Industriel (API)

L'Automate Programmable Industriel (API) est un appareil électronique programmable, adapté à l'environnement industriel, qui réalise des fonctions d'automatisme pour assurer la commande de pré actionneurs et d'actionneurs à partir d'informations logique, analogique ou numérique et destiné à la conduite et la surveillance en temps réel de processus industriels [12].



Figure 2.15 : Un automate programmable industriel

2.6.2 Les caractéristiques d'un API

Il peut être directement connecté aux capteurs et pré-actionneurs grâce à ses entrées/sorties industrielles. Il est conçu pour fonctionner dans des ambiances industrielles sévères « température, vibrations, microcoupures de la tension d'alimentation, parasites, etc. ». Enfin sa programmation à partir de langages spécialement développés pour le traitement de fonctions d'automatisme fait en sorte que sa mise en œuvre et son exploitation ne nécessitent aucune connaissance en informatique [13].

2.6.3 Architecture d'un API

L'automate programmable industriel possède deux structures : externe et interne.

Structure externe

La structure externe d'un API est présentée sur la Figure 2.16 [13].



Figure 2.16 : Structure externe d'un API [13]

Avec :

- 1. Module d'alimentation.
- 2. Pile de sauvegarde.
- 3. Connexion au 24V cc.
- 4. Commutateur de mode (à clé).
- 5. LED de signalisation d'état et de défauts.
- 6. Carte mémoire.
- 7. Interface multipoint (MPI).
- 8. Connecteur frontal.
- 9. Volet en face avant.

> Structure interne

La structure interne d'un API est assez voisine de celle d'un système informatique simple. Les API comportent quatre parties principales :

- ✓ Une alimentation 230V AC, 50/60 Hz ; 24V (DC).
- ✓ Unité centrale (Processeur).
- ✓ Interfaces d'entrées / sorties.
- ✓ Mémoires.

Ces quatre parties (**Figure 2.17**) sont reliées entre elles par des bus (ensemble câble autorisant le passage de l'information entre ces 4 secteurs de l'API). Ces quatre parties réunies forment un ensemble compact appelé automate [14].



Figure 2.17 : Structure interne [14]

2.6.4 Programmation d'un API

La programmation d'un API consiste à traduire dans le langage spécialisé de l'automate, les équations de fonctionnement du système à automatiser.

Parmi les langages normalisés, on cite quelques-uns des plus connus et des plus utilisés:

- Langage LADDER ou schéma à contacts.
- Langage List d'instructions (Instruction List).
- Langage GRAFCET.
- Langage littéral structuré.

Généralement, les constructeurs d'API proposent des environnements logiciels graphiquespour la programmation.

2.6.5 Traitement du programme automate

Tous les automates fonctionnent selon le même mode opératoire montré ci-dessous (Figure 2.18) :



Figure 2.18 : Fonctionnement cyclique d'un automate [13]

Où :

- Traitement interne : L'automate y effectue des opérations de contrôle et met à jour certains paramètres systèmes « détection des passages en RUN / STOP, mises à jour des valeurs de l'horodateur ».
- Lecture des entrées : L'automate lit les entrées « de façon synchrone » et les recopie dans la mémoire image des entrées.
- Exécution du programme : L'automate exécute le programme instruction par instruction et écrit lessorties dans la mémoire image des sorties.
- Écriture des sorties : L'automate bascule les différentes sorties « de façon synchrone » aux positions définies dans la mémoire image des sorties.

Ces quatre opérations sont effectuées continuellement par l'automate en fonctionnement cyclique [13].

2.6.6 Avantages de l'automate programmable industriel

Les avantages de l'automate sont nombreux. Pour l'utilisateur, ils se situent à plusieurs niveaux :

* Logiciel

• Gain en temps.

- Simplicité des langages de programmation.
- Sauvegarde les programmes.

Industriel

- Economies.
- Sécurité.
- Possibilité de dialogue avec son entourage.
- Cahier des charges accessible à tous (souplesse d'application).

✤ Exploitation

- Intervention sur processus en cours.
- Suivi en temps réel de l'évolution de l'automatisme.
- Exploitation (fonctionnement) en conditions sévères.

2.6.7 Critères de choix d'un automate

Le choix d'un automate programmable est en premier lieu le choix d'une société ou d'un groupe et les contacts commerciaux et expériences vécues sont déjà un point de départ. Les grandes sociétés privilégieront deux fabricants pour faire jouer la concurrence et pouvoir se retourner en cas de perte de vitesse de l'une d'entre elles.

Le personnel de maintenance doit toutefois être formé sur ces matériels et une trop grande diversité des matériels peut avoir de graves répercussions.

Il faut ensuite quantifier les besoins [13]:

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

- Avoir les compétences et l'expérience nécessaire : pour programmer la gamme d'automate.
- Le cout : d'investissement, de fonctionnement, de maintenance de l'équipement.
- La qualité du service après-vente.

2.6.8 Présentation de l'API S7-1500 de SIEMENS

L'automate SIMATIC S7-1500 est un système de commande modulaire utilisé pour les moyennes et grandes performances. Il existe un éventail complet de modules pour une adaptation optimisée à la tâche d'automatisation aux conditions sur site (**Figure 2.19**).

SIMATIC S7-1500 est un perfectionnement des systèmes d'automatisation SIMATIC S7-300 et S7-400 avec les nouvelles performances suivantes :

- Performance système accrue.
- Fonctionnalité Motion Control intégrée.
- PROFINET IO IRT.
- Ecran intégré pour commande et diagnostic près de la machine.
- Innovation linguistiques STEP7 sous réserve de fonctions éprouvées.



Figure 2.19 : Automate S7-1500 [15]

Dans un automate S7-1500, on trouve :

- Une CPU.
- Un module d'interface pour PROFINET IO.
- Un module d'interface pour PROFIBUS DP.
- Un module de périphérie (Entrée TOR(DI)/Sortie TOR(DQ), entrée analogique (AI)/sortie analogique (AQ)).
- Une alimentation système (PS) tension d'entrée 24V à 230V.
- Et une alimentation externe (PM) tension d'entrée 120/230V.

Le cas échéant, des processus de communication et des modules fonctionnels sont ajoutés pour des tâches spéciales comme la commande de moteur pas à pas [15].

La Figure 2.20, présente un aperçu général de la structure de cet automate où :

- 1. Alimentation système.
- **2.** CPU.
- 3. Modules d'entrées/sorties pour les signaux numériques et analogiques.
- 4. Modules sont montés sur un profile-support avec un rail DIN symétrique intégré.



Figure 2.20 : Structure d'un automate S7-1500 [15]

Le programme S7 permet à l'automate programmable industriel (API) de contrôler et commander une machine ou un processus. Le système est programmé avec le logiciel STEP7 Professional V16 (TIA PORTAL).

2.6.9 Domaines d'application du S7 1500

Parmi les multiples domaines d'application du S7 1500, on peut citer :

- ✓ Industrie automobile.
- ✓ Construction de mécanique général.
- ✓ Construction de machines spéciales.
- ✓ Industrie de l'emballage.

2.6.10 Présentation de la CPU 1515-2PN

Le SIMATIC S7-1500 est doté d'une CPU 1515-2 PN, unité centrale avec 500Ko de mémoire de travail pour le programme et 3Mo pour les données, 1^{ère} interface : PROFINET IRT avec commutateur 2 ports, 2^{ème} interface : PROFINET RT, performance sur bit 30 NS, carte mémoire SIMATIC nécessaire.

Les CPU ont des performances différentes et exécutent le programme utilisateur. De plus, les modules suivants sont branchés sur l'alimentation système intégré via le bus de fond de panier (**Figure 2.21**).

Autres propriétés et fonctions de la CPU :

- ✓ Communication via Ethernet.
- ✓ Communication par PROFIBUS/PROFINET.
- ✓ Communication IHM pour les stations de contrôle/commande.
- ✓ Serveur Web.
- ✓ Fonction technologique intégré (Régulateur PID, Motion Control...etc.).
- ✓ Sécurité intégrée (Protection contre le piratage, Protection contre la copie ; Protection d'accès et Protection d'intégrité).



Figure 2.21 : CPU 1515-2PN [15]

La CPU contient le système d'exploitation et exécute le programme utilisateur qui se trouve sur la carte mémoire SIMATIC et il est traité dans la mémoire de travail de la CPU.

Les interfaces PROFINET se trouvant sur la CPU permettent la communication simultanée avec des appareils PROFINET, des contrôleurs PROFINET, des appareils IHM, des consoles de programmation, d'autres automates et d'autres systèmes.

La CPU 1515-2 PN prend en charge le fonctionnement en tant que contrôleur IO et périphérique I. [15]

La Figure 2.22 [15] présente un aperçu de la structure externe de la CPU 1515-2PN.



Figure 2.22 : La CPU 1515-2PN [15]

Où :

- 1. LED de signalisation pour l'état de fonctionnement et l'état de diagnostic actuels de la CPU.
- **2.** Ecran.
- **3.** Touches de fonction.

Le Tableau 2.1, résume donc les principales caractéristiques de la CPU 1215-2 PN :

| Informations générales | | | | | | |
|--------------------------------|--------------------------------|--|--|--|--|--|
| Désignation du type de produit | CPU 1515-2 PN | | | | | |
| Version du firmware | V2.8 | | | | | |
| Tension d'a | Tension d'alimentation | | | | | |
| Tension d'alimentation | 24 V CC | | | | | |
| Courant | d'entrée | | | | | |
| Consommation (valeur nominale) | 0,8 A | | | | | |
| Puissanc | e dissipée | | | | | |
| Puissance dissipée | 6,3 W | | | | | |
| Mémoire de travail | | | | | | |
| Intégré (pour programme) | 500 kbyte | | | | | |
| Intégré (pour données) | 3 Mbyte | | | | | |
| Sauve | egarde | | | | | |
| Sans maintenance | Oui | | | | | |
| Temps de tra | iitement CPU | | | | | |
| Pour opérations sur bits | 30 ns | | | | | |
| Program | nmation | | | | | |
| Langage de programmation | CONT, LOG, LIST, SCL, GRAPH | | | | | |
| Servei | ır Web | | | | | |
| HTTP | Oui | | | | | |
| HTTPS | Oui | | | | | |
| OPC | CUA | | | | | |
| Licence Runtime nécessaire | Oui | | | | | |
| Client OPC UA Oui | | | | | | |
| Server OPC UA | Oui, Data Access (Read, Write) | | | | | |

| Dimensions | | | | |
|------------|--------|--|--|--|
| Largeur | 70 mm | | | |
| Hauteur | 147 mm | | | |
| Profondeur | 129 mm | | | |
| Poids | | | | |
| Poids | 830 g | | | |

Tableau 2.1 : Les caractéristiques de la CPU 1515-2PN

2.6.11 Présentation des modules d'Entrées / Sorties

Les Figures 2.23 et 2.24, représentent les modules d'entrées / sorties TOR et analogiques respectivement.

> Les modules TOR

DI 16x24VDC / DQ 16x24VDC/ 0.5A BA

- Module d'entrées TOR DI16 x DC24V (Par groupes de 16, Retard à l'entrée 3,2 ms, Type d'entrée 3 (CEI 61131))
- Module de sorties TOR DQ16 x DC24V / 0,5A (Par groupes de 8, 4A par groupe).



Figure 2.23 : Module d'entrée TOR [15]

Les modules d'entrée analogique

AI 4xU/ I/RTD/TC / AQ 2xU/ I ST_1

- Module d'entrées analogiques AI4 x U/I/RTD/TC 16 bits (Groupes de 4, 2 voies avec mesure RTD, Tension de mode commun 10V, Diagnostic paramétrable, Alarmes de processus).
- Module de sorties analogiques AQ2 x U/I 16 bits (Groupes de 2, Diagnostic paramétrable, Valeur de remplacement paramétrable pour sortie).



Figure 2.24: Module d'entrée analogique [15]

2.7 Conclusion

Dans ce chapitre on a présenté les différents équipements de la partie opérative et la partie commande de notre système. Pour cela, on a défini le système automatisé et sa structure puis présenté aussi l'automate S7-1500, sa structure, ces composants et la CPU utilisés avec ces caractéristiques.

Dans le chapitre suivant on va présenter le logiciel TIA PORTAL avec lequel on a programmé l'API ainsi que l'IHM utilisé pour la supervision.

Chapitre 3 Programmation et Supervision

3.1 Introduction

Dans ce chapitre, nous allons présenter le logiciel de programmation TIA Portal et on va donner un aperçu sur la supervision du système avec IHM SAMKOON.

Nous allons expliquer les différentes étapes à suivre pour programmer et superviser le système pompage étudié.

3.2 Le logiciel de programmation TIA Portal

TIA Portal est le logiciel d'ingénierie de Siemens qui permet de programmer des automates de la gamme Siemens. La nouvelle version de Step7 est fournie dans le logiciel d'ingénierie de Siemens TIA Portal (Totally Integrated Automation Portal).

Autrement dit, TIA Portal est un logiciel tout en un qui permet de programmer des automates, panels et contrôleurs d'axes Siemens [16].

La plateforme « Totally Integrated Automation Portal » est le nouvel environnement de travail Siemens qui permet de mettre en œuvre des solutions d'automatisation avec un système d'ingénierie intègre comprenant les logiciels SIMATICStep7 et SIMATIC WinCC [16].

3.2.1 Les avantages du logiciel TIA Portal

Les avantages du TIA Portal sont résumés comme suit :

- **Programmation intuitive et rapide :** avec des éditeurs de programmation nouvellement développés SCL, CONT, LOG, LIST et GRAPH.
- Efficacité accrue grâce aux innovations linguistiques de STEP 7 : programmation symbolique uniforme, Boite à calculs, ajout de blocs durant le fonctionnement, et bien plus encore.
- Performance augmentée grâce à des fonctions intégrées : simulation avec PLCSIM, télémaintenance avec TeleService et diagnostic système cohérent.
- **Technologie flexible :** Fonctionnalité motion control évolutive et efficace pour les automates S7-1500 et S7-1200.
- Sécurité accrue avec Security Integrated : Protection du savoir-faire, protection contre la copie, protection d'accès et protection contre la falsification.
- Environnement de configuration commun avec pupitres IHM et entraînements dans l'environnement d'ingénierie TIA Portal [16].

3.2.2 Vues du TIA Portal

Lorsqu'on lance TIA Portal, l'environnement de travail se décompose de deux types de vues :

- Vue du portal : elle est axée sur les tâches à exécuter et sa prise en main est très rapide.
- Vue du projet : elle comporte une arborescence avec les différents éléments du projet, les éditeurs requis s'ouvrent en fonction des tâches à réaliser. Données, paramètres et éditeurs peuvent être visualisés dans une seule et même vue [16].

a) Vue du portal

Chaque portail permet de traiter une catégorie de tâche (action). La fenêtre affiche la liste des actions qui peuvent être réalisées pour la tâche sélectionnée (**Figure 3.1**) [17].

| M Siemens - série exercices 1 | | | _ ¤ X |
|-------------------------------|--|---|---|
| Dómarrer | Liste des actions | Mise en route | Totally Integrated Automation PORTAL |
| Appareils & | Ouvrir le projet existant Créer un projet Migror le projet Fermer le projet | Projet : "sérile exercices 1" cuvert avec succè Démarrer | s. Sélectionnez l'étape subvante : |
| En ligne 8 Diagnostic | Présentation de bienvenue Mise en route | Appareils & Stranger | Configurer un appareil Ecrire un programme API |
| Choix des portails | Longue de l'interface | Visualisation Visualisation Visualisation Visualisation | Configurer une vue IHM Ouvrir la vue du projet |
| Vue du projet | Projet ouvert : CNU sensUulien/Doo | uments\Exercices modules TIA\série exercic\se | irie exercices 1 |

Figure 3.1 : Vue du portal [17]

b) Vue du projet

L'élément « Projet » contient l'ensemble des éléments et des données nécessaires pour mettre en œuvre la solution d'automatisation souhaitée. La figure ci-dessous représente un exemple de vue du projet (**Figure 3.2**) [17].

| ijet till om öffshage insention Enligt | Quells Approximation pressure and pressure of the programme of the programe of the programme of | otally In Onglets de sélection des tâches |
|---|---|---|
| Selie seaticles 1 Acuter un appareil Acuter acuter annote Seater acuter acuter Acuter acuter acuter acuter acuter Acuter acuter acuter acuter acuter Acuter acut | | transmitter in transmitter intreamenter intransmitter intreamenter intreamenter |
| Nom Ditains Ditains | Internet Internet Général Variable Variable Variable Général | Alexandress and a second a |

Figure 3.2 : Vue de projet [17]

Où :

- La fenêtre de travail : permet de visualiser les objets sélectionnés dans le projet pour être traités. Il peut s'agir des composants matériels, des blocs de programme, des tablesdes variables, des interfaces homme machine (IHM).
- 2. La fenêtre d'inspection : permet de visualiser des informations complémentaires sur un objet sélectionné où sur les actions en cours d'exécution (propriété du matériel sélectionné, message d'erreur lors de la compilation des blocs de programme, ...).
- Les onglets de sélection de tâches : ont un contenu qui varie en fonction de l'objet sélectionné (configuration matérielle → bibliothèques des composants, bloc de programme →instructions de programmation).

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

3.2.3 Adressage des E/S

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

| | Choix de l'appareil | | Onglet « Vu | e des appareils |
|---|--|--|--|-----------------------|
| Siemens - Premier projet | | | 1 | |
| siet Edition Affichage Insertion | En ligne Outils accessoires Fenitre | Ade | | |
| 2 19 D Enregistrer le projet = 1 | | ID IE I Juisson en linne * | Totally | PORT |
| Navinateur du projet 34 | Premier projet + API 1 [CPU 314C-7 | PN/DPI | | |
| & no avails | | Www.teneloniawa | A Mus during | IN Vue des appareils |
| | A LAN A | the topologique | 100 vie un research | III vue des apparens |
| 300 - | N' N' | 1 140 Cal : 01 2 100% | | |
| | 1 2 | -4 5 | 6 7 8 | 9 10 11 |
| Apparent A Bissesso | Chāssis_0 | | | |
| * APL 1 (CPU 314C-2 PNUD | | | | |
| Y Configuration des a | 2 | | | |
| B Enligne & Diagnostic | LM. | | | |
| T Blocs de neonremme | Second Se | | | |
| B Elics de programme | | | | |
| Ajouter nouveau | | | | |
| Main [081] | | | | |
| Objets technologiqu | | the local data and the local dat | | |
| Sources externes | 21 | | | |
| Variables API | ×1 | C | | 1 |
| Types de données AFI | Vue d'ensemble des appareils | | | |
| Tables de visualisati | and loss of the | Internet and and a second second | 1 - Contraction of the contracti | law a se |
| informations surie p | Tr - Module | Chassis Empla_ Adressel Adre | tte Type | Nº de ret. : |
| Alarmes API | P3 307 50_1 | | PS 307 5A | EEST JOTTENDI GARD |
| Listes de testes | • ML1 | | CPU STAC-2 PRODP | 6557 314-6E/104-0400 |
| | historice Mride_1 | 0 2047 | A CONTRACT | |
| Vue détaillée | Interface PROFINET_1 | 0 1 12 2016 | Intensce PHOFINET | |
| | 0124/0016_1 | 0 25 136138 136. | .137 014/0016 | |
| Nom | AI5/A02_1 | 0 26 800609 800. | .803 AULAD2 | |
| | Comptage_1 | 0 27 816831 816. | .0J1 Comptage | |
| | Positionnement_1 | 0 28 832847 832 | | |
| | | 0 3 | A | |
| - | < II. | | | 3 |
| | | Proprié | 165 2110 0 2 | Diagnostic |
| 4 Vue du postali | ensem J. API 1 | 813 | | antion of axis STREA |
| a vue ou portan | and the second s | | Massatent : comige | Henori recasie Kirou- |
| | | | | |
| | | | | |
| | | | 6 9 PAGE 10 | |

Figure 3.3 : Adressage des E/S [17]

On sélectionne la CPU puis à l'aide de deux petites flèches (Figure 3.3), on fait apparaitre l'onglet « Vue d'ensembles des appareils ». Les adresses des entrées et sorties apparaissent. On peut alors les modifier en entrant une nouvelle valeur dans la case correspondante [17].

3.2.4 Mémento de cadence

Une fois la CPU déterminée, on peut définir le mémento de cadence. Pour cela, on sélectionne la CPU dans la fenêtre « Vue des appareils » et l'onglet « propriété » dans la fenêtre d'inspection, dans le menu « Général ». On choisi alors l'option « Mémento de cadence » (Figure 3.4), puis on coche la case « Mémento de cadence » en choisissant l'octet du mémento de cadence que l'on va utiliser [18].

| Stamons - Pro | mler projet fichagé (ns | ertion and | one Quitis | | | . То | tally integrate | d Automation |
|--|----------------------------|-------------|------------|------------|---------------|-------------|-----------------|---------------|
| Premier projet | P APL 1 [C | 10 3146-21 | | C= A 145 | 50 103 104 | 100 | | PORTA |
| de (Ap.s | | • 14 | | topologi | ans lats A | run du rési | 10 11 | des appareils |
| < | | | | | | 1ª inte | t & Diamon | |
| Général Diskibólie AlsiA02 Compage Posifécrimem. Max en rissie Cycle Monernitide Manernitide | | iento de ca | dence | 10 | nto de cadeni | in | N 1 2 Mindred | |

Figure 3.4 : Mémento de cadence [18]

3.2.5 Adressage Ethernet de la CPU

Dans les propriétés de la CPU, il est possible aussi de définir son adresse Ethernet. Un double clic sur l'icône Ethernet de la station fait apparaître la fenêtre d'inspection permettant de définir ses propriétés (**Figure 3.5**).

Pour établir une liaison entre la CPU et la console de programmation, il faut affecter aux deux appareils des adresses appartenant au même sous réseau. L'adresse utilisée est 192.168.0.2 de l'automate [19].



Figure 3.5 : Adressage Ethernet de la CPU [19]

3.2.6 Les variables API

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

- L'adresse absolue représente l'identificateur d'opérande (I, Q, M, ...) et son adresse et numéro de bit.
- Adresse symbolique correspond au nom que l'utilisateur a donné à la variable (ex : bouton marche).

Le lien entre les adresses symboliques et absolues se fait dans la tabledes variables API. Lors de la programmation, on peut choisir d'afficher les adresses absolues, symboliques ou encore les deux simultanément [19].

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

Autrement dit, lorsqu'on définit une variable API, il faut définir :

- Un nom : c'est l'adressage symbolique de la variable.
- Le type de donnée : BOOL, INT, ...
- L'adresse absolue : par exemple Q 1.5.

On peut également insérer un commentaire qui nous renseigne sur cette variable. Le commentaire peut être visible dans chaque réseau utilisant cette variable [17].

3.2.7 Liaison avec l'automate

Il faut maintenant charger la configuration programmée de l'automate dans celui-ci. Pour cela, il faut tout d'abord connecter l'automate au PC en utilisant l'interface SIMATIC S7 PC USB adapté, ensuite, après avoir sélectionner la vue « En ligne et diagnostique », on sélectionne alors les options suivantes :

- Mode : MPI
- Interface PG /PC : pc Adapter [17].

3.2.8 Conception d'un programme avec TIA Portal V16

La conception d'une solution d'automatisation se fait par deux alternatives, soit on commence par la programmation ou par la configuration matérielle.

La Figure 3.6 résume les différentes étapes d'organisation à suivre pour la conception d'un programme avec TIA Portal.



Figure 3.6 : Organisation pour la conception d'un projet sous TIA Portal

3.3 La supervision

La supervision du système, aussi appelée interface homme machine, permet d'effectuer des réglages, d'afficher des messages et de gérer les défauts.

La supervision est une technique de pilotage et de suivi informatique des procédés industriels automatisés qui permet à l'opérateur d'observer, d'analyser et d'intervenir sur le système d'une manière directe. Elle offre aussi la possibilité d'archiver les données pour pouvoir les analyser lors de l'établissement des diagnostics.

En informatique industrielle, la supervision des procédés peut être une application de surveillance, de contrôle-commande ou de diagnostic ou l'ensemble de ces dernières réunies. Elle se fait à travers un logiciel fonctionnant sur un ordinateur en communication, via un réseau local ou distant industriel, avec un ou plusieurs équipements.

3.3.1 Avantages de la supervision

Un système de supervision donne de l'aide à l'opérateur dans la conduite du processus, son but est de présenter à l'opérateur des résultats expliqués et interprétés et son avantage principal est :

- Surveiller le processus à distance.
- La détection des défauts.
- Le diagnostic et le traitement des alarmes.
- Traitement des données.

3.3.2 Principe de fonctionnement d'une application de supervision

La principale fonctionnalité d'une application de supervision est :

- Acquisition des données.
- Conduite de procédés.
- Représentation graphique d'informations.
- Traitement d'alarmes et gestion d'événements
- Archivage et calculs sur historiques.
- Suivi et traçabilité de la production

3.3.3 IHM (Interface Homme/Machine)

Une IHM constitue l'interface entre l'homme (opérateur) et le processus (machine/installation). Le contrôle du processus est assuré par le système d'automatisation. La figure ci-dessous montre l'utilité de L'IHM dans un processus automatisé.



Figure 3.7 : IHM dans un processus automatisé

Un système IHM se charge des tâches suivantes :

- **Représentation du processus :** Le processus est représenté sur le pupitre operateur lorsqu'un état du processus évolue l'affichage du pupitre operateur est mise à jour.
- **Commande du processus :** L'opérateur peut commander le processus via l'interfaceutilisateur graphique.
- Vue des alarmes : Lorsque surviennent des états critiques dans le processus, une alarme est immédiatement déclenchée, par exemple une valeur limite est franchie.
- Archivage de valeurs processus et d'alarmes : Les alarmes et valeurs processus peuvent être archivées par un système IHM. On peut, par exemple, commander la marche processus et accéder ultérieurement aux données de la production évoluée.
- Documentation de valeurs processus et d'alarme : les alarmes et valeurs processus peuvent être éditées par le système IHM sous forme de journal. On peut ainsi consulterles données de production à la fin du travail d'une équipe, par exemple.
- Gestion des paramètres de processus et de machine : Les paramètres de processus et de machine peuvent être enregistrés au sein du système IHM. Ces paramètres sont alors transférables en une seule opération sur l'automate pour démarrer la productiond'une variante du produit par exemple.

L'IHM utilisé dans notre projet est l'écran Touch Screen SAMKOON SK-102.

3.3.4 Présentation de l' IHM SAMKOON SK-102 HS

La série SK IHM de SAMKOON est un produit d'interface Homme-Machine de haute qualité. Il intègre entièrement une unité centrale, une unité d'entrée et de sortie, un écran, une mémoire et d'autres unités de module.

Il peut être largement appliqué dans les systèmes de contrôle industriels de tous les secteurs de l'industrie (Figure 3.8).

Avec une conception optimale via matériel et logiciel, il est conforme aux exigences de contrôlede la machine pour la précision et la précision du toucher, ainsi que pour les couleurs d'écran.

Sur la base de la version précédente, l'IHM de la série SK apporte de grandes améliorations avecun temps de démarrage plus rapide, vitesse de communication supérieure et une sensibilité plus élevée qu'auparavant.

Il fournit une interface série standard pouvant être connectée à d'autres périphériques. Principalement compatible avec la communication série, il peut communiquer avec les principaux API du marché, tels que SAMKOON, MITSUBISHI, SIEMENS, OMRON et MODICON.



Figure 3.8 : Face avant de SK- 102 HS [20]

a) Différents ports de connexion de l'alimentation de SK-102 HS

La figure ci-dessous (Figure 3.9) représente les différents ports de connexions et d'alimentation de SK-102 HS.



Figure 3.9 : Schéma de connexion de l'écran tactile de la série SK

- **1.** Un port de programmation (USB).
- **2.** Deux ports de programmation (COM).
- **3.** Un connecteur d'alimentation (24V DC).
- 4. Alimentation de 220V ou 380V alternative.
- 5. Un connecteur de connexion (Ethernet RG45).

b) Les caractéristiques techniques de SK-102HS

Le tableau suivant représente les caractéristiques techniques du SK-102HS.

| Tension d'alimentation | 24V DC (+/-15%) |
|---------------------------|---|
| Consommation d'énergie | 6W |
| Portes séries | COM1 et COM2 pour liaison série RS232/422/485 |
| Port USB-A | Support pour une vitesse de transmission de 10M/S |
| Port USB-B | Support pour flash disc, imprimante, souris et autres périphériques USB |
| Porte Ethernet | Réaliser la communication entre les IHM ou IHM et le dispositif de commande via Ethernet (S7-1200). |
| Mémoire | 128M FLASH + 128M DDR2 |
| Processeur | Processeur hautes performances de qualité industrielle : cortex A8, jusqu'à 600 MHz |
| Logiciel de programmation | SKTOOL 6.2 |

| łS |
|----|
| |

3.4 Création de Projet

Dans cette partie, on va présenter les différentes étapes suivies pour mener à bien la création de notre projet.

Dans l'arborescence de Menu démarrer, on clique sur TIA PORTAL V16 qui se trouve dans le dossier Siemens Automation.

3.4.1 Création d'un nouveau projet

La création d'un nouveau projet est réalisée, en suivant les étapes suivantes :

- On clique sur nouveau projet.
- On complète les données nécessaires.
- On clique sur le bouton créer (Figure 3.10).

| M Siemens | | | | _ • × |
|-------------------------|---------------------------|--------------------|--|---|
| | | | | Totally Integrated Automation PORTAL |
| Start 🧤 | | Create new project | | |
| Devices & | Open existing project | Project name: | Automatisation d'un système de pompage | |
| networks | Create new project | Path: Version: | V16 | |
| PLC 🔬 | | Author: | Snow | |
| programming | Migrate project | Comment: | | <u> </u> |
| Motion & technology | | | | ~ |
| Visualization | Welcome Tour | | | Create |
| Online & Diagnostics | | | | |
| | | | | |
| | Installed software | | | |
| | Melp | | | |
| | | | | |
| | 🕥 User interface language | | | |
| | | | | |
| Project view | | | | |

Figure 3.10 : Création d'un nouveau projet

3.4.2 Ajout d'un automate

Dans la liste, on sélectionne l'automate S7-1515-2PN (6ES7 515-2AM02-0AB0) puis on clique sur ajouter.

Les Figures 3.11, 3.12 et 3.13, présentent donc les différentes étapes à suivre.

| Siemens - C:\Users\ | Snow\Document | s\Automation\Automatisation d'un système de | pompage\Automa | tisation d'un systèm | e de pomp | age | _ # × |
|-------------------------|---------------|---|-----------------|------------------------|-----------|-------------------------------------|---|
| | | | | | | | Totally Integrated Automation PORTAL |
| Start | | | First steps | | | f di Di | |
| Devices & networks | * | Open existing project | Project: "Autom | atisation d'un syste | me de poi | mpage" was opened successfully. Pre | ease select the next step: |
| DLC. | | Create new project | | | | | |
| programming | - * | Migrate project | | | | | |
| Motion & technology | * | Close project | | | | | |
| | | | ⊢ | | Q Q | Configure a device | |
| Visualization | | Welcome Tour | | | - | | |
| Online & Diagnostics | 1 | 🥚 First steps | | | | Write PLC program | |
| bitgitterite | | | → | Motion & technology | - | Configure technology objects | |
| | | Installed software | → | | Ø | Configure an HMI screen | |
| | | Help | | | | | |
| | | | | | | | |
| | | 🚱 User interface language | Ļ | | | Open the project view | |
| | | | | | | | |



| Siemens - C:\Users\Snow\Documents\A | Automation/Automatisation d'un système de | pompage\Automatisation d | 'un système de pompage | | _ # × |
|---|--|--------------------------|------------------------|--|-------|
| | | | | Totally Integrated Automation POR | TAL |
| Start INFORMATION Start INFORMATION START INFORMATION START | Show all devices Add new device | Add new device | | Device: | |
| Visualization Diagnostics | Configure networks | HM PC systems | Device proxy | Atticle no.: Version: Description: Modules of the SIMATIC 57-1500 device family | = |
| | • Help | | | | |

Figure 3.12 : Ajout d'un automate S7-1500

| | tsAutomationAutomatisation d'un système de | e pompage\Automatisa | ition d'un système de pompage | | | | _ - × |
|--|--|-----------------------|--|---|--|--|----------------------|
| | | | | | | Totally Integrated A | Automation PORTAL |
| Start 🦓 | | Add new device _ | | | | | |
| Devices & antworks | Show all devices | Device name: PLC_1 | | | | | ^ |
| PLC programming Motion & technology | | Controllers | | ^ | Device: | | |
| Visualization | Configure networks | HM | ▶ □ CPU 1511C-1 PN ▶ □ CPU 1512C-1 PN ▶ □ CPU 1512C-1 PN ▶ □ CPU 1513-1 PN ▼ CPU 1515-2 PN ■ 6657 515-2AM01-0AB0 ■ 6657 515-2AM01-0AB0 | = | Article no.: Version: | CPU 1515-2 PN 66557 515-2AM02-0AB0 V2.8 | = |
| | Help | PC systems | CONTRACTORY CONTRACTORY | | Description: CPU with dig 3 MB data; 31 protection co motion contr isochronous interfaces: tr Open User C S7 routing. IF OPC UA: Serv companions 2 PROFINETIO performance Device, MRP, interface; firm | slay, wolf, memory 500 KB code and Dru bit operation time, i-stage noept, technology functions: ol, closed-loop control, counting ng, traing, Ruthime options; anaport protocol TCPIP, secure ommunication, 57 communication, forwarding, Web server, DMS client, et DA. Client DA. nethods, controller, supports KIIRK, upginde PROFINETV23.2, ports, I- MRD, isochronous mode; 2nd MRV 202 | |
| | | | CPU 1518F-4 PN/DP ODK | | interface: PR Device; firmv | OFINET IO controller, supports RT, I- vare V2.8 | ~ |

Figure 3.13 : Ajout du CPU S7-1515-2PN

3.4.3 Présentation de l'interface

La **Figure 3.14**, présente la vue obtenue de l'interface du projet réalisé et qui est subdivisée en plusieurs parties comme suit :



Figure 3.14 : Présentation de l'interface

> Navigateur de projet :

Dans cette partie, on retrouve la liste de tous les composants du projet. Plusieurs automates peuvent s'y retrouver lorsque le projet contient plusieurs automates en réseau. On peut également y retrouver les appareils de supervision ou les régulateurs présents dans l'installation. Pour chaque appareil du projet, On retrouvera tous les éléments nécessaires à la configuration, programmation ou mise en service.

> Partie centrale :

La partie centrale de la fenêtre permet d'afficher le contenu de menu sélectionné dans le navigateur de projet. On y retrouvera donc aussi bien les outils de programmation de l'automate que les menus de configuration des différents éléments du projet.

> Menu de droite :

Le menu à droite de l'écran varie également en fonction de l'élément sélectionné dans le navigateur de projet. On y retrouvera, par exemple, les blocs à intégrer lorsqu'un bloc de programme est sélectionné.

> Partie d'outils principale :

Cette partie d'outils contient, en plus des traditionnelles icônes présentes dans la plupart des logiciels, des icônes permettant de compiler et de charger les programmes dans l'automate connecté au PC.

3.4.4 Configuration matérielle

Une configuration matérielle est nécessaire pour :

- Les paramètres ou les adresses préréglé d'un module.
- Configurer les liaisons de communication.

Dans notre cas, nous avons ajouté les modules suivants (Figure 3.15) :

- DI 16x24VDC / DQ 16x24VDC / 0.5A BA
- AI $4xU/I/RTD/TC/AQ 2xU/I ST_1$

Pour les entrées et les sorties on a utilisé :

- Les entrées numériques : 16 entrées.
- Les sorties numériques : 8 sorties.



Figure 3.15 : Configuration matérielle

3.4.5 Les adresses utilisées pour les entrées / sorties

La prochaine étape consiste à connaître les adresses des différents modules à utiliser. Pour cela, après avoir sélectionné la vue « Configuration des appareils » dans le navigateur de projet, on clique sur la flèche permettant d'afficher les données de l'appareil, comme c'est montré sur la **Figure 3.16**.

| automatisation d'un système de pompage 🔸 PLC_1 [CPU 1515-2 PN] | | | | | | | | | īХ | | | | |
|--|---|--------|------------------------------|------|--------|-----------|-----------|---------------------|---------------------|----------|---|--|--|
| | | | | | 📇 Торо | ice view | | | | | | | |
| • | • | Device | Device overview | | | | | | | | | | |
| 1 | ^ | | Module | Rack | Slot | I address | Q address | Туре | Article no. | Firmware | | | |
| = | ≡ | | | 0 | 100 | | | | | | ^ | | |
| | | | | 0 | 0 | | | | | | | | |
| | | | PLC_1 | 0 | 1 | | | CPU 1515-2 PN | 6ES7 515-2AM02-0AB0 | V2.8 | | | |
| | | | Interface PROFINET_1 | 0 | 1 X1 | | | PROFINET interface | | | | | |
| | | | Interface PROFINET_2 | | 1 X2 | | | PROFINET interface | | | | | |
| | | | DI 16x24VDC / DQ 16x24VD | 0 | 2 | 01 | 01 | DI 16x24VDC / DQ | 6ES7 523-1BL00-0AA0 | V1.1 | | | |
| | - | | AI 4xU/I/RTD/TC / AQ 2xU/I S | 0 | 3 | 29 | 25 | AI 4xU/I/RTD/TC / A | 6ES7 534-7QE00-0AB0 | V1.0 | | | |
| | | | | 0 | 4 | | | | | | | | |
| | Ĥ | | | 0 | 5 | | | | | | | | |
| | | | | 0 | 6 | | | | | | | | |
| | | | | 0 | 7 | | | | | | | | |
| | | | | 0 | 8 | | | | | | | | |
| | | | | 0 | 9 | | | | | | | | |
| | | | | 0 | 10 | | | | | | | | |
| | | | | 0 | 11 | | | | | | | | |
| 1 | _ | | | 0 | 12 | | | | | | ~ | | |
| E 🤅 | | < | | | | | | | | 1 | > | | |

Figure 3.16 : Adresses utilisées par les différents modules de l'automate

3.5 Programmation de l'automate

On va maintenant passer à la programmation de notre automate.

3.5.1 Création d'une table de variables

La première étape consiste à créer la table des variables qui sert à donner un nom symbolique aux variables utilisées.

Elle contient toutes les variables d'entrées/sorties (TOR, analogiques et mémoires), ainsi que les variables intermédiaires qui vont nous permettre de gérer notre processus technique.

Dans tous programme il faut définir la liste des variables qui vont être utilisées lors de la programmation pour cela la table des variables est créé.

L'utilisation des nomes appropriés rend le programme plus compréhensible est plus facile à manipuler. Ce type d'adressage est appelé « relatif ».

Apres le nom on définit le type de donné de la variable, puis l'adresse. On remplit la table des variables en respectant notre cahier de charge, pour les entrées et les sorties.

La figure suivante (**Figure 3.17**) représente partie de la table de variables utilisés dans notre programme, la table entière se trouve dans l'annexe.

| tomatisation d'un système de pompage 🔸 PLC_1 [CPU 1515-2 PN] 🔸 PLC tags 🕨 Table de variables standard [246] 🛛 🗕 🖬 🖬 🗙 | | | | | | | | | | | | |
|---|------|-----------------------------|-----------|---------|--------|---------|----------|--------------|----------|---------|---|--|
| 📹 Tags 🛛 🗉 User constants 🖉 System constants | | | | | | | | | nstants | | | |
| 🖆 🔮 🖶 🐨 🛍 | | | | | | | | | | | | |
| Table de variables standard | | | | | | | | | | | | |
| | Name | | Data type | Address | Retain | Acces | Writa | Visibl | Supervis | Comment | | |
| 1 | | imp d'acquitement | Bool 🔳 | %M0.0 | | | | | | | ^ | |
| 2 | | Tag_1 | Bool | %M0.2 | | | ~ | | | | ≡ | |
| 3 | | Tag_2 | Bool | %M0.3 | | | ~ | | | | | |
| 4 | | Tag_3 | Bool | %M0.4 | | | ~ | ~ | | | | |
| 5 | | bS pompe 1 | Bool | %M0.1 | | | | | | | | |
| 6 | | temps réglable HMI | Time | %MD2 | | | | \checkmark | | | | |
| 7 | | nombre démarrage maximal ré | Int | %MW6 | | | | | | | | |
| 8 | | System_Byte | Byte | %MB1 | | | ~ | \checkmark | | | | |
| 9 | | FirstScan | Bool | %M1.0 | | | ~ | ~ | | | | |
| 10 | | DiagStatusUpdate | Bool | %M1.1 | | | ~ | | | | | |
| 11 | | Alwa ys TRUE | Bool | %M1.2 | | | ~ | | | | | |
| 12 | - | AlwaysFALSE | Bool | %M1.3 | | | | | | | | |
| 13 | -00 | Clock_Byte | Byte | %MBO | | | ~ | | | | | |
| 14 | -00 | Clock_10Hz | Bool | %M0.0 | | | | | | | | |
| 15 | - | Clock_5Hz | Bool | %M0.1 | | | | | | | ~ | |
| | < | | | | | | | | | > | | |

Figure 3.17 : Table de variables

Les variables peuvent être de type :

> Entrées

Pour savoir l'état et le déroulement de procès l'automate récolte des informations venantes de l'installation et cela via des entrées automate qui sont connecté aux déférents capteurs et boutons de l'installation pour ensuit les traiter et générer la commande.

> Sorties

Apres traitement des données d'entré et pour commander l'installation, l'automate doit générer et envoyer des signaux par ces sortie. Les sorties automate sont connecté au différentes vannes et actionneurs de l'installation.

> Mémento

Zone de mémoire dans la mémoire système d'une CPU. Il est possible d'y accéder en écriture et en lecture (par bit, octet, mot et double mot). La zone des mémentos permet à l'utilisateur d'enregistrer des résultats intermédiaires.

3.5.2 Les blocs de programmation

L'automate met à disposition différents types de blocs qui contiennent le programme et les données correspondantes.

Selon les exigences et la complexité du processus, il est possible de structurer le programme en différents blocs : OB, FB, FC, DB (Figure 3.18).



Figure 3.18 : Les différents blocs

Le dossier Blocs contient les blocs que l'on doit charger dans la CPU pour réaliser la tâche d'automatisation. Il englobe :

- Les blocs de code (OB, FB, FC) qui contiennent les programmes qu'on doit chargerdans la CPU.
- Les blocs de données (DB d'instance et DB globaux) qui contiennent les programmes du programme.

La Figure 3.19 présente les différents blocs utilisés dans notre travail.



Figure 3.19 : Les blocs de programme de l'automate

Avec :

a) Blocs d'organisation (OB)

Les OB sont appelés par le système d'exploitation, on distingue plusieurs types :

- Ceux qui gèrent le traitement de programmes de programmes cycliques
- Ceux qui sont déclenchés par un événement.
- Ceux qui gèrent le comportement à la mise en route de l'automate programmable eten fin, Le bloc OB1 est généré automatiquement lors de la création d'un projet. C'est le programme cyclique appelé par le système d'exploitation.

Les blocs d'organisation permettent de structurer votre programme. Ils servent d'interface entre le système d'exploitation et le programme utilisateur. Les OB sont déclenchés sur événement.

La CPU exécute un OB à la suite d'un événement, tel qu'une alarme de diagnostic ou un intervalle de temps. Certains OB ont des événements déclencheurs et un comportement prédéfini. A l'état MARCHE, les OB de cycle de programme s'exécutent au niveau de priorité le plus bas et peuvent être interrompus par tous les autres types d'événements [19].

Blocs d'organisation

La figure suivante (Figure 3.20) représente tous les réseaux dans OB1.

On a utilisé le bloc d'organisation OB1 qui est appelé par le système d'exploitation, il fait appel aux autres blocs qui constituent le programme, lorsqu'on appelle un bloc fonctionnel dans l'OB1 un bloc de donnée associé sera créé automatiquement.



Figure 3.20 : Blocs d'organisation OB1
Réseau 1 : Permutation 3+1

La figure suivante (**Figure 3.21**) représente le bloc de permutation, il assure la permutation en cas de défaut de l'une des trois pompes selon le niveau du réservoir d'aspiration par la pompe de secours.





Figure 3.21 : Permutation 3+1

Réservoir d'aspiration

La figure suivante (Figure 3.22) représente le bloc de réservoir d'aspiration et les défauts qui peut survenir.

| 1 | | I | %FC3 | | |
|---|-----|---------------------|--------------------------|-----|----|
| | | | "Réservoir d'aspiration" | | |
| | | | | | |
| | | 0 | EN | ENC | |
| | | %M14.7 | | | |
| | | "NTB ASP" | - NTB | | |
| | | %M8.5 "NB 458" | NB | | |
| | | 11070 | | | |
| | | "NCASP" | NC | | |
| | | V 140 7 | | | |
| | | "NH ASP" | — NH | | |
| | | %M15.0 | | | |
| | | "vanne | Vanna | | |
| | | aspiration | aspiration | | |
| | | ou verte | ouverte | | |
| | | | | | |
| | | %M15.1 | | | |
| | | "vanne | vanne | | |
| | | aspiration | aspiration | | |
| | | lemee | term ee | | |
| | | %M15.2 | Barrad March | | |
| | | reset min | Keset Hmi | | |
| | | "reset hardware" | - Reset hardware | | |
| | | %M15.4 | m ede auto | | |
| | | "m ode a u to" | V-ASP | | |
| | | %M15.5 | m ode local | | |
| | | "mode local" | · V-ASP | | |
| 1 | | | | | |
| | L | %M15.6 | | | |
| | L | "cas 1 on V-ASP" | cas 1 on | | |
| | L | N 1 1 5 7 | | | |
| | L | "cas 2 on V ASP" | cas 2 on | | |
| | L | N M C C | | | |
| | I 1 | "cas 3 on V-ASP" | no Ezer | | |
| | I 1 | SM16-1 | | | |
| | | "cas 1 off V-asp" | cas1 off | | |
| | | %M16.2 | | | |
| | | "cas 2 off V asp" — | cas 2 off | | |
| | | %M16.3 | | | |
| | | | Cas 3 Off | | |
| | L | %M16.5 | | | |
| | L | "defaut niveau | defautniveau | | |
| | L | aspiration" | tres bas aspiration | | |
| | I 1 | · · · · · | | | |
| | L | %M16.6 | | | |
| | L | "défaut | défaut | | |
| | L | redondance | redon dan ce | | |
| | L | contradictoire | contradictoire | | |
| | L | %M16.7 | | | |
| | 1 | "BS ASPINITB" 🔛 | BS ASP NTB | | |
| | 1 | %M17.0 | | | |
| | 1 | "BS ASP INC" | BS ASP INC | | |
| | 1 | %M17.1 | | | |
| | 1 | "défaut | defaut | | |
| | 1 | di scordan ce | discordan ce | | |
| | 1 | aspiration | aspiration | | |
| | 1 | | | | I. |



Figure 3.22 : Réservoir d'aspiration

> Réservoir de refoulement

La figure suivante (Figure 3.23) représente le bloc de réservoir d'aspiration et les défauts qui peut survenir.

| | %FC4 "Réservoir de refoulem en t" | |
|------------------------------------|--------------------------------------|----|
| | -EN E | NC |
| %MB.0 "NTB REF" | NTB | |
| %M8.1 | NB | |
| %M8.2 | - NB | |
| "NCREF" | -NC | |
| "NH REF" | -NH | |
| "cas 1 on V-REF" | cas 1 on | |
| "cas 2 on V REF" | cas 2 on | |
| %M21.2 "cas 1 off V REF" — | - cas 1 off | |
| %M19.6 "cas 2 off V REF" | - cas 2 off | |
| %M19.7 "cas 3 off V REF" | - cas 3 off | |
| %M20.0 | - | |
| Gass on vier | - cas 3 on | |
| "ordre d'ou verture v | | |
| ref" | OUVRIR V-REF | |
| %M20.2 | | |
| ferm etu re v ref" | FERMER V REF | |
| %M20.3 "v ref ou verte" — | V REFOUVERTE | |
| %M20.4 "v ref ferm ée" | V REF FERMEE | |
| %M20.5 | | |
| "défaut discordance | Défaut discordance | |
| ou vertu re V REF" | ou vertu re V ref | |
| %M20.6 | défaut aiuaau | |
| ntb REF" | _ très bas ref | |
| %M20.7 "défaut | défaut discordance | |
| discordan ce ferm etu re v ref" | ferm etu re v ref | |
| %M21.0 | | |
| "défaut in cohérence v ref" | défaut in cohérence v | |
| %M21.1 | | |
| "dispov ref" — | - dispo | |

Figure 3.23 : Réservoir de refoulement

> Pompe 1

La figure suivante (**Figure 3.24**) représente le bloc de pompe 1. Elle montre les niveaux de fonctionnement et les défauts qui peuvent survenir.

| - | | |
|--------------------------|--------------------------------|------|
| | %FC1 "Software de la Pompe" | |
| | sutware de la Pulipe | 5115 |
| | - EN | ENC |
| %M10.2 | | |
| marcheP1" | retour de marche | |
| %M21.3 | | |
| "arret d'urgen ce" 🗕 | Arret d'urgen ce | |
| %M21.4 | | |
| defaut | défaut | |
| nardware i Fi | nardware 1 | |
| %M21.5 "DH 2 P1" | défaut hardware 2 | |
| SM01.6 | diferre | |
| "DH 3 P1" | hardware 3 | |
| %M15.5 | | |
| "m ode local" | - m ode local | |
| %M15.4 | mode | |
| "mode auto" | automatique | |
| %M15.3 | | |
| reset nardware" | - reset hardware | |
| "reset hmi" | reset h mi | |
| %M21.7 | car 1 (N) | |
| "Cas 1 on P1" | process | |
| %M22.0 | cas 2 ON | |
| "cas 2 on P1" | process | |
| - | | |
| %M22.2 | | |
| "cas 3 on process P1" | cas 3 ON | |
| | | |
| "cas 1 off P1" | cas 1 OFF _ process | |
| | · | |
| "cas 2 off | cas 2 OFF | |
| process P1" | process | |
| 961022.5 | | |
| "cas 3 off | cas 3 OFF | |
| process P1" | process | |
| %M22.6 | | |
| Tordre de | ordre de | |
| marcheP1" | marche | |
| %M22.7 | d'acquitement | |
| "e fp P 1" | Fp | |
| %M23.0 | | |
| "bascule DD p1" | - bascule DD | |
| %M23.1 | | |
| "com m an de | commande | |
| tem po P1" | _ de tem po | |
| %M23.2 | | |
| "tem ps ecoule P1" | tem ps écou lé | |
| %M23.3 | défaut classe | |
| "Defaut classe | fnb dem atteint | |
| i nomrer I | | |
| | | |



Figure 3.24 : Pompe 1

> Pompe 2

La figure suivante (**Figure 3.25**) représente le bloc de pompe 2. Elle montre les niveaux de fonctionnement et les défauts qui peuvent survenir.

| | | | _ |
|-----------------------|------------------------|-----|---|
| 1 | %FC1 | | |
| | "Software de la Pompe" | | |
| | EN | ENC | |
| %M10.3 | | | |
| "retouir de | retour de | | |
| marche P2" | _ marche | | |
| %M21.3 | | | |
| "arret d'urgen ce" | Arret d'urgen ce | | |
| %M23.5 | défaut | | |
| "DH 1 P2" | hardware 1 | | |
| %M23.6 | défaut | | |
| "DH 2 P2" | hardware 2 | | |
| %M23.7 | défaut | | |
| "DH3 P2" | hardware 3 | | |
| %M15.5 | | | |
| "mode local" — | - mode local | | |
| %M15.4 | mode | | |
| "mode au to" | automatique | | |
| %M15.3 | | | |
| "reset hardware" | reset hardware | | |
| %M15.2 | | | |
| "reset h m i" — | – reset h m i | | |
| %M24.0 | cas 1 ON | | |
| 0610112 | process | | |
| %M24.1 | cas 2 ON | | |
| case of the | | | |
| %M24.2 | Cat 3 (N) | | |
| "Cas3 on p2" | process | | |
| %M24.3 | cas 1 OFF | | |
| "cas 1 off p2" - | process | | |
| %M24.4 | cas 2 OFF | | |
| "cas 2 off p2" | process | | |
| %M24.5 | cas 3 OFF | | |
| cass or p2 | proces | | |
| %M24.6 | | | |
| marche P2" | ordre de _ marche | | |
| | etat | | |
| %M24.7 | d'acquitement Fo | | |
| etat p F 2 - | | | |
| "bascule DD P2" | basculeDD | | |
| 5LM25_1 | com manda | | |
| "cm p P2" | _ de tem po | | |
| %M25.3 | | | |
| "tem ps écoul é P2" 🗕 | tem ps écoul é | | |
| • | | | 1 |

| %M25.2 "Défaut nombre démarrage P2" | défaut classe f nb dem atteint | |
|---|--------------------------------------|--|
| %M25.4 "défaut discordance P2" | Défaut discordance | |
| %M9.7 "dispo P2" — | – disponibilitė | |

Figure 3.25 : Pompe 2

> Pompe 3

La figure suivante (**Figure 3.26**) représente le bloc de pompe 3. Elle montre les niveaux de fonctionnement et les défauts qui peuvent survenir.

| ENC |
|-----|
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |



Figure 3.26 : Pompe 3

> Pompe de secours

La figure suivante (Figure 3.27) représente le bloc de pompe de secours. Elle procède au remplacement des pompes qui ont des défauts.



Figure 3.27 : Pompe de secours

b) Blocs fonctionnels (FB)

Les blocs fonctionnels sont des blocs de code qui traitent une fonction spécifique et qui mémorisent durablement les paramètres d'entrées, de sorties et d'entrée/sortie dans des blocs de données d'instance afin qu'il soit possible d'y accéder même après le traitement de blocs.

C'est pourquoi ils sont également appelés "Blocs avec mémoire". Ils peuvent aussi travailler avec des variables temporaires. Cependant ces dernières ne sont pas enregistrées dans le DB d'instance mais disponibles uniquement tout le temps d'un cycle [19].

c) Fonctions (FC)

La FC contient des routines pour les fonctions fréquemment utilisées. Elle est sans mémoire et sauvegarde ses variables temporaires dans la pile de données locales cependant elle peut faire appel à des blocs de données globaux pour la sauvegarde de ses données, les SFC sont utilisées pour des fonction spéciales, intégrées dans la CPU S7, elleest appelée à partir du programme [19].

> Fonction FC1 Software de la pompe

La figure suivante (Figure 3.28) représente tous les réseaux dans FC1.

La FC contient des routines pour les fonctions fréquemment utilisées. Elle est sans mémoireet sauvegarde ses variables temporaires dans la pile de données locales. Cependant elle peutfaire appel à des blocs de données globaux pour la sauvegarde de ses données

| • | Network 1: | Défaut de discordance |
|-----|-------------|---|
| ۲., | Network 2: | lancement de la commande tempo réglable par l'opérateur |
| ۲. | Network 3: | reset de la commande après l'écoulement de la tempo |
| • | Network 4: | temporisation Classe F |
| ۰. | Network 5: | comptage de nombre démarrage pompe pendant une tempo Définie |
| • | Network 6: | Reset de défaut automatique après l'écoulement de la tempo classe F définie |
| • | Network 7: | chaine du disponibilité de la pompe |
| • | Network 8: | ordre de marche de la pompe |
| ۰. | Network 9: | ordre d'arret |
| • | Network 10: | • |

Figure 3.28 : Fonction FC1 Software de la pompe

Défaut de discordance

La figure suivante (Figure 3.28) représente le défaut de discordance.

Ce défaut s'active à cause d'une contradiction entre l'ordre de marche et le retour de marche de la pompe.



Figure 3.29 : Défaut de discordance

> La commande temporisation

La figure 3.30 représente lancement de la commande temporisation réglable par l'opérateur.

| Network 2: | lancement de la commande tempo réglable par l'opérateur |
|------------------------|---|
| Comment | |
| #"retour de marche" | #"commande de tempo" { S } |
| | |

Figure 3.30 : Lancement de la commande temporisation réglable par l'opérateur

Reset de la commande de temporisation

La figure 3.31 représente le reset de la commande après l'écoulement de la temporisation.

| ▼ N | letwork 3: reset de la commande après l'écoulement de la temp | 00 |
|-----|---|-------------------------|
| Co | omment | |
| #" | "temps écoulé" | #"commande de tempo" |
| | | (R) |

Figure 3.31 : Reset de la commande après l'écoulement de la temporisation

> Temporisation classe F

La **figure 3.32** représente la temporisation de temps écoulé (la durée du démarrage de la pompe).

| • | Network 4: temp | porisation Classe F | | |
|---|--|--------------------------------|-----------------|--|
| _ | | | | |
| | | %DB2 "IEC_Timer_0_ DB_1" | | |
| | #"commande de tempo" | TON Tíme | #"temps écoulé" | |
| | <mark>%MD2</mark> "temps réglable HMI" — | ET — T#Oms | | |

Figure 3.32 : Temporisation classe F

Comptage de nombre démarrage pompe pendant une temporisation définie

La **figure 3.33** représente le comptage du nombre de démarrage de la pompe pendant une temporisation définie.



Figure 3.33 : Comptage de nombre démarrage pompe pendant une temporisation définie

Reset de défaut automatique

La **figure 3.34** représente le reset de défaut automatique après l'écoulement de la temporisation classe F définie.

| Network 6: | Reset de défaut automatique après l'écoulement de la tempo classe F définie |
|--------------------------------|---|
| Comment | |
| #"temps écou | #"défaut classe f nb dem atteint" (R) |

Figure 3.34 : Reset de défaut automatique après l'écoulement de la tempo classe F définie

Chaine de la disponibilité de la pompe

La **figure 3.35** représente la chaine de la disponibilité de la pompe. Cette chaine nous informe que la pompe est prête à démarrer (l'absence des défauts est vérifiée).



Figure 3.35 : Chaine de la disponibilité de la pompe

> Ordre de marche de la pompe

La figure 3.36 représente ordre de marche de la pompe (démarrage de la pompe).



Figure 3.36 : Ordre de marche de la pompe

> Ordre d'arrêt

La figure 3.37 représente ordre d'arrêt de la pompe (Arrêt de la pompe).



Figure 3.37 : Ordre d'arrêt

Fonction FC2 Permutation 3+1

La figure suivante (Figure 3.38) représente tous les réseaux dans FC2.



Figure 3.38 : Fonction FC2 Permutation 3+1

> Commande arrêt des 4 pompes conditionnel

La **figure 3.39** représente la commande d'arrêt des 4 pompes conditionnel au niveau haut dans le réservoir de refoulement.

| Network 1: Commande arret 4 pompes conditionnel | |
|---|---|
| Comment | |
| #*NH REF* | #"Commande arret 4 pompes conditionnel" () |

Figure 3.39 : Commande arrêt des 4 pompes conditionnel

Commande pompe 1

La **figure 3.40** représente la commande de la pompe 1 au niveau bas dans le réservoir d'aspiration et la commande pour arrêter les pompes 2 et 3.



Figure 3.40 : Commande pompe 1

Commande pompe 2

La **figure 3.41** représente la commande de la pompe 2 au niveau conditionnel dans le réservoir d'aspiration et la commande pour arrêter la pompe 3.



Figure 3.41 : Commande pompe 2

Commande pompe 3

La **figure 3.42** représente la commande de la pompe 3 au niveau haut dans le réservoir d'aspiration.



Figure 3.42 : Commande pompe 3

Commande pompe de secours prête

La **figure 3.43** représente la commande de la pompe de secours en cas un des 3 pompes est en défaut.



Figure 3.43 : Commande pompe de secours prête

> Fonction FC3 Réservoir d'aspiration

La figure suivante (Figure 3.44) représente tous les réseaux dans FC3.



> Défaut niveau très bas d'aspiration

La figure 3.45 représente le défaut niveau très bas d'aspiration.

Le rôle de ce défaut est évité le danger d'aspiration d'air des pompes.



Figure 3.45 : Défaut niveau très bas d'aspiration

> Défaut discordance cas vanne d'aspiration en état ouverture

La figure 3.46 représente le défaut discordance en cas vanne d'aspiration est en état d'ouverture.

Ce défaut est lorsqu'il y a un ordre d'ouverture de la vanne d'aspiration et il n'y a pas ouverture de la vanne donc il affiche une alarme dans IHM, ou bien il n'y a pas ordre pour l'ouverture de la vanne mais la vanne est indiquée qu'elle est à position d'ouverture.



Figure 3.46 : Défaut discordance vanne d'aspiration ouverture

> Défaut discordance cas vanne d'aspiration en état fermeture

La **figure 3.47** représente le défaut discordance en cas vanne d'aspiration est en état de fermeture.

Ce défaut est lorsqu'il y a un ordre de fermeture de la vanne d'aspiration et il n'y a pas fermeture de la vanne donc il affiche une alarme dans IHM, ou bien il n'y a pas ordre pour fermeture de la vanne mais la vanne est indiquée qu'elle est à position d'ouverture.



Figure 3.47 : Défaut discordance vanne d'aspiration fermeture

Défaut incohérence

La figure 3.48 représente le défaut d'incohérence.

Ce défaut informe que on a deux informations de postions contradictoires (vanne d'aspiration ouverte et fermé en même temps ou bien pas d'ouverture et fermeture en même temps) dans ce cas on excite le déclenchement d'une alarme dans IHM.



Figure 3.48 : Défaut incohérence

> Disponibilité vanne pneumatique de réservoir d'aspiration

La **figure 3.49** représente la chaine de la disponibilité de la vanne pneumatique de réservoir d'aspiration. Cette chaine nous informe que la vanne est prête pour l'indication de la position (l'absence des défauts est vérifiée).

| Network 5: dis | ponibilité vanne pre | eumatique | | |
|--|--|-----------------------------|-----------------------------|----------------|
| Comment | | | | |
| #"défaut niveau très bas aspiration" | #"défaut incohérence aspiration" | #"discordance ouverture" | #"discordance fermeture" | #disponibilité |

Figure 3.49 : Disponibilité vanne pneumatique

> Ordre d'ouverture vers le distributeur bistable

La **figure 3.50** représente ordre d'ouverture de la vanne pneumatique. (Impulsion excite l'alimentation de l'air vers l'ouverture).



Figure 3.50 : Ordre d'ouverture vers le distributeur bistable

> Ordre fermeture vers le distributeur bistable

La **figure 3.51** représente ordre de fermeture de la vanne pneumatique. (Impulsion excite l'alimentation de l'air vers fermeture).



Figure 3.51 : Ordre fermeture vers le distributeur bistable

> Fonction FC4 Réservoir de refoulement

La figure suivante (Figure 3.52) représente tous les réseaux dans FC4 pour le réservoir de refoulement.

| • | Network 1: | alarme ntb Ref |
|---|------------|--------------------------|
| • | Network 2: | discordance d'ouverture |
| • | Network 3: | discordance de fermeture |
| • | Network 4: | incohérence v ref |
| • | Network 5: | disponibilité |
| • | Network 6: | ordre d'ouverture |
| • | Network 7: | ordre de fermeture |
| • | Network 8: | |
| | | |

Figure 3.52 : Fonction FC4 Réservoir de refoulement

Alarme niveau très bas

La **figure 3.53** représente l'alarme niveau très bas pour éviter le danger d'aspiration d'air des pompes qu'il t'afficher dans IHM.



Figure 3.53 : Alarme niveau très bas

> Défaut discordance ouverture vanne de refoulement

La **figure 3.54** représente le défaut discordance en cas vanne de refoulement est en état d'ouverture.

Ce défaut est lorsqu'il y a un ordre d'ouverture de la vanne de refoulement et il n'y a pas ouverture de la vanne donc il affiche une alarme dans IHM, ou bien il n'y a pas ordre pour l'ouverture de la vanne mais la vanne est indiquée qu'elle est à position d'ouverture.



Figure 3.54 : Défaut discordance ouverture vanne de refoulement

> Défaut discordance fermeture vanne de refoulement

La **figure 3.47** représente le défaut discordance en cas vanne de refoulement est en état de fermeture.

Ce défaut est lorsqu'il y a un ordre de fermeture de la vanne de refoulement et il n'y a pas fermeture de la vanne donc il affiche une alarme dans IHM, ou bien il n'y a pas ordre pour fermeture de la vanne mais la vanne est indiquée qu'elle est à position d'ouverture.



Figure 3.55 : Défaut discordance fermeture vanne de refoulement

Incohérence vanne de refoulement

La figure 3.56 représente le défaut d'incohérence.

Ce défaut informe que on a deux informations de postions contradictoires (vanne de refoulement ouverte et fermé en même temps ou bien pas d'ouverture et fermeture en même temps) dans ce cas on excite le déclenchement d'une alarme dans IHM.



Figure 3.56 : Incohérence vanne de refoulement

> Disponibilité de la vanne pneumatique de refoulement

La **figure 3.57** représente la chaine de la disponibilité de la vanne pneumatique de réservoir de refoulement. Cette chaine nous informe que la vanne est prête pour l'indication de la position (l'absence des défauts est vérifiée).



Figure 3.57 : Disponibilité

Ordre d'ouverture de vanne de refoulement

La **figure 3.58** représente ordre d'ouverture de la vanne pneumatique de réservoir de refoulement. (Impulsion excite l'alimentation de l'air vers l'ouverture).



Figure 3.58 : Ordre d'ouverture de vanne de refoulement

> Ordre de fermeture de vanne de refoulement

La **figure 3.59** représente ordre de fermeture de la vanne pneumatique de réservoir de refoulement. (Impulsion excite l'alimentation de l'air vers fermeture).



Figure 3.59 : Ordre de fermeture de vanne de refoulement

d) Blocs de données (DB)

Ces blocs de données (**Figure 3.60**) servent uniquement à stocker des informations et des données mais pas d'instruction comme les blocs de code, Les données utilisateurs stockés seront utilisées par la suite par d'autres blocs [19].

| | Project tree | | | au | ıtom | atis | ation d'un système d | le pompage | ▶ PLC_ | 1 [CPU 1515-2 | PN] 🕨 Pro | gram blocks | • OPC | mysql [DB4 |] | _ • | ∎× |
|------|--------------|---------------------|---|----|------|------------|----------------------|---------------|--------|---------------|-----------|-------------------|---------|------------|----------|----------|----|
| | Devices |] | | | | | | | | | | | | | | | |
| | ĒŇ | | 1 | Ť | k 🛒 | ı, | n 🛃 📰 🧐 Keepa | actual values | 🔒 Sn | apshot 🔤 🖷 | Copysnap | shots to start va | lues 🔣 | - B- | | | |
| 2 | | | | | OP | Cmy | ysql | | | | | | | | | | |
| Ξ. | 📑 Add r | iew device | ^ | | | Nam | e | Data type | | Start value | Retain | Accessible f | Writa | Visible in | Setpoint | Supervis | Co |
| am a | nh Devic | es & networks | | 1 | -00 | ▼ 5 | Static | | | | | | | | | | |
| 160 | 👻 🗖 PLC_1 | [CPU 1515-2 PN] | | 2 | | • | pression REF | Real | | 0.0 | | | | | | | |
| Ē. | 🕎 De | evice configuration | = | 3 | - | • | débit Refoulement | Real | | 0.0 | | | | | | | |
| Ĕ | 🖁 OI | nline & diagnostics | | 4 | | • | niveau ASP | Real | | 0.0 | | ~ | | | | | |
| | 🕨 🝺 So | oftware units | | 5 | -00 | • | niveau REF | Real | | 0.0 | | | | | | | |
| | 🔻 🔙 Pr | ogram blocks | | | | | | | | | | | | | | | |
| | | Add new block | | | | | | | | | | | | | | | |
| | | Dynamique et A | | | | | | | | | | | | | | | |
| | | Process [OB1] | | | | | | | | | | | | | | | |
| | | Permutation 3+ | | | | | | | | | | | | | | | |
| | | Réservoir d'aspir | | | | | | | | | | | | | | | |
| | | Réservoir de ref | | | | | | | | | | | | | | | |
| | | Software de la P | | | | | | | | | | | | | | | |
| | | Al instrumentati | | | | | | | | | | | | | | | |
| | | OPC mysql [DB4] | | | | | | | | | | | | | | | |
| | • 🔤 | System blocks | | | | | | | | | | | | | | | |
| | 🕨 🙀 Te | chnology objects | ~ | | | | | | | | | | | | | | |
| | < | | > | | | | | | | | | | | | | | |
| | ✓ Details vi | ew | | 1 | | | | | | | | | | | | | |
| | | | | | < | | | | | II | 11 | | | | | | > |

Figure 3.60 : Blocs de données DB OPC MySQL

3.6 La supervision

Créé pour la série IHM de SAMKOON SK à partir de Shenzhen SAMKOON Technologie, SKTOOL est un logiciel d'édition de configuration de SAMKOON IHM, SKTOOL prend en charge les pilotes de communication PLC des principaux fabricants mondiaux.

Le logiciel de configuration SKTOOL est un système de développement avec un environnement de développement intégré et de nombreuses fonctions avancées, il est facile à apprendre et à appliquer, ainsi les ingénieurs peuvent facilement développer leurs propres projets pour la configuration utilisée dans SKTOOL.

3.6.1 Création d'un projet et configuration

Cette opération est nécessaire créer un nouveau projet. On sélectionne Nouveau projet dans le menu Fichier ou sur le bouton Nouveau projet. La figure ci-dessous représente la boîte de dialogue Nouveau projet dans laquelle l'utilisateur décide du nom, du chemin d'accès, du modèle et du mode d'affichage.

| roject Properties | | |
|-------------------|-------------------------------------|--------|
| Project Name: | | |
| Path: | C:\Program Files\SKTOOL6.2\Project\ | Browse |
| Model: | 5K-102HS | • |
| Show Model: | forizontal | Ŧ |
| Iodel Parameters | | |
| Size:1 | 0.2 inch | |
| Resolution:1 | 024x600 Pixels (VGA) | |
| Color:2 | 62,144 Colors TFT LCD | |
| RAM:1 | 2M | |
| Power supply:D | C24V(+/-15%) | |
| COM1:R | S232/RS422/RS485 | |
| COM2:R | S232/RS422/RS485 | |
| USB:2 | Ports B-type/A-Type | |
| Ethernet:R | 345 | |

Figure 3.61 : Création d'un projet IHM

Le modèle spécifique est déterminé par le modèle IHM utilisé. Dans notre banc d'essai, on utilise SK-102HS, puis on clique sur Suivant pour définir le port de communication.

3.6.2 Boîte de dialogue Paramètres de communication

La boîte de dialogue Paramètres de communication sert à définir les paramètres de communication du port de communication de l'IHM.

Seuls des paramètres de communication corrects peuvent assurer une communication normale entre l'automate et les écrans tactiles, Les paramètres de communication spécifiques dépendent du modèle d'automate à connecter.

Les paramètres le plus importants dans La boîte de dialogue Paramètres (**Figure 3.62**) de communication sont :

- 1. Nom de la connexion : nom du port de communication.
- **2.** Service de périphérique : sélection de la marque de l'automate et du type de CPU.
- 3. Connexion Interface : sélection de COM port ou Ethernet port
- **4. Intervalle d'adresse continue de l'automate** : définit l'espace adresse consécutive de l'automate où la valeur par défaut est recommandée.

| Link ID: | 1 | | |
|--------------------|--------------------|--------|--------|
| Link Name: | Link 1 | |] |
| Link Interface: | Ethernet | • | 1 |
| Device Service: | SIEMENS | ic S7- | 1 |
| PLC Continuous Add | Iress Space: 256 👻 | | , |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | Next | Cancel |

Figure 3.62 : Boite de dialogue

La figure ci-dessus représente la boîte de dialogue des paramètres du port de communication dans laquelle les utilisateurs choisissent les services de périphérique correspondants enfonction du type de périphérique de connexion. Pour utiliser une connexion Ethernet, on choisit l'option « Connexion directe (Ethernet) » dans le « Type de connexion ». On clique sur « Suivant » dans la boîte de dialogue Paramètres de communication pour afficher une boîte de dialogue Nouvel écran.

3.6.3 Programme IHM dans SAKTOOL

Pour programmer l'IHM, nous avons utilisé plusieurs éléments pour commander et visualiser l'état de la machine trieuse. Notre programme IHM contient les figures suivantes :

- La **Figure 3.63** représente la page de process de système.
- La Figure 3.64 représente la page de commande de système des pompes et des électrovannes.
- La Figure 3.65 représente la page d'alarme qui prévient le système.



Figure 3.63 : Process du système de IHM.



Figure 3.64 : Commandes du système de IHM

| RESET | EFFACER LISTE | EFFACER ALARME |
|---------|-----------------|------------------|
| Time | Date | Mess age |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| ACCUEIL | GESTION ALARMES | OPTIONS SETTINGS |

Figure 3.65 : Les alarmes du système de IHM

3.7 Archivage des données par MySQL

3.7.1 Définition d'une base de données

Une base de données sert essentiellement à stocker des informations (comme des noms, prénoms, adresses, numéros de téléphone, commandes ou autre) pour ensuite être capable de les traiter, les filtrer, les trier, etc. afin d'en extraire des statistiques, de récupérer les données d'un individu à partir de son nom, etc.

3.7.2 OPC UA

OPC UA (Open Protocol Communication Unified Architecture) est un protocole de communication machine à machine pour l'automatisation industrielle.

3.7.3 Node Red

Node Red est un outil open source de programmation permettant de connecter ensemble des périphériques matériels, des API et des services en ligne de manière nouvelle et intéressante.

3.7.4 XAMPP

XAMPP est un ensemble de logiciels permettant de mettre en place un serveur Web local. Il offre une bonne facilité d'utilisation et d'installation et un l'environnement de développement PHP le plus populaire.

3.7.5 Les étapes pour archiver les données

Pour l'archivage des données au niveau de base de données on va utiliser le Node Red comme un lien entre le Plc et la base de données et le OPC UA comme un protocole de communication entre le Plc et le Node Red.

a) Copier l'adresse de server OPC UA

On va copier l'adresse de serve opc ua pour la mettre dans opc ua pour connecter.



Figure 3.66 : Adresse de server OPC UA

b) Open OPC UA

Pour vérifier les valeurs de base de données on va à OPC MySQL.

| Search | Attributes and References | F | | | | | |
|-------------------------------|---------------------------|---------------------------|---|---------------------------------|-------------|------------------|-----------|
| IEC_Timer_0_DE^ | 6 | | | \leftrightarrow \rightarrow | Filters 🔝 🔛 | Browse Direction | Forward v |
| IEC_TIMEr_0_DE IEC_TIMEr_0_DE | Attribute | Value | 4 | Referen | сеТуре | Target | |
| IEC_Timer_0_DE | ► Nodeld | ns=3;s="OPC mysql"."dr" | | HasType | eDefinition | BaseDataVaria | bleType |
| IEC_Timer_0_DE | NodeClass | Variable | | | | | |
| IEC_Timer_0_DE | BrowseName | 3:dr | | | | | |
| IEC_Timer_0_DE | DisplayName | dr | | | | | |
| ► 🗑 IEC Timer 0 DE | Description | | | | | | |
| ► 🗑 IEC Timer 0 DE | WriteMask | NONE (0) | | | | | |
| ► 🗑 IEC Timer 0 DE | UserWriteMask | NONE (0) | | | | | |
| E IEC Timer 0 DE | ► Value | 0.0 | | | | | |
| ▶ III Icon | ► DataType | Float | n | | | | |
| V R OPC mysel | ValueRank | ValueRank Scalar | | | | | |
| t or c mysqi | ArrayDimensions | null | | | | | |
| | AccessLevel | CurrentRead, CurrentWrite | | | | | |
| na na | UserAccessLevel | CurrentRead, CurrentWrite | | | | | |
| ► 😆 nr | MinimumSamplingInterval | -1.0 | | | | | |
| ► 💽 pr | Historizing | false | | | | | |
| DataBlocksInstance | | | | | | | |
| DeviceManual | | | | | | | |
| DeviceRevision | | | | | | | |
| EngineeringRevisior | | | | | | | |
| HardwareRevision | | | | | | | |

Figure 3.67 : La fenêtre de OPC UA

| au | utomatisation d'un système de pompage 🔸 PLC_1 [CPU 1515-2 PN] 🔸 Program blocks 🔸 OPC mysql [DB4] 🦳 🗖 🖷 | | | | | | | | | |
|----|--|------------------------|-----------|-------------|---------------|--------|--------------|---------|------------|-------|
| | | | | | | | | | | |
| 1 | 学 🐳 🔩 🛃 🧮 📽 Keep actual values 🔒 Snapshot 🦄 🍕 Copy snapshots to start values 🛃 🚱 🎽 | | | | | | | | | |
| | OPC mysql (snapshot created: 9/17/2022 2:49:52 AM) | | | | | | | | | |
| | | Name | Data type | Start value | Monitor value | Retain | Accessible f | Writa | Visible in | Setpo |
| 1 | | ▼ Static | | | | | | | | |
| 2 | | pr | Real | 0.0 | 5.0 | | | | | |
| 3 | - | dr | Real | 0.0 | 2.0 | | | | | |
| 4 | | na | Real | 0.0 | 1.0 | | | | | |
| 5 | - | nr | Real | 0.0 | 3.0 | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |

c) Modifier les valeurs au niveau bloc de donnée

Figure 3.68 : Adresse de server OPC UA

d) Connecter au Node Red

Ce bloc interface est construit autour d'un « NODE-RED » qui peut être installé dans dans un pc ordinateur ou bien dans un Smart phone. Dans cette platforme on peut faire toutes les opération et traitement des données reçus de manière automatique à partir de notre API.

Il comprend une réseau de nœuds (node) pour gérer la réception des données à partir de l'API et exécuter leurs organisations sous forme de blocs. Les résultats obtenus sont alors envoyés vers la base de données pour y être archivées.



Figure 3.69 : Adresse de server OPC UA

e) Open Xampp

| 😫 XAMPI | P Control Par | nel v3.3.0 [Co | ompiled: Apr 6th 20 | 21] | | | _ | \Box \times |
|--|--|--|---|---|-------|--------|------|-----------------|
| 8 | XAI | MPP Con | trol Panel v3 | .3.0 | | | | Je Config |
| Modules Service | Module | PID(s) | Port(s) | Actions | | | | Netstat |
| × | Apache | 9656 14736 | 4433, 8080 | Stop | Admin | Config | Logs | Shell |
| × | MySQL | 8488 | 3306 | Stop | Admin | Config | Logs | Explorer |
| × | FileZilla | | | Start | Admin | Config | Logs | Services |
| | Mercury | | | Start | Admin | Config | Logs | 😡 Help |
| × | Tomcat | | | Start | Admin | Config | Logs | Quit |
| 22:46:39 22:46:40 22:46:40 22:46:53 22:46:53 22:46:53 22:46:54 22:46:55 | [Apache] [Apache] [mysql] [Mysql] [Apache] [Mysql] [mysql] | Attemptir Status ch Attemptir Status ch Attemptir Status ch Attemptir Status ch | ing to stop Apache nange detected: s ng to stop MySQL hange detected: s ng to start Apachi hange detected: n ng to start MySQL hange detected: r | e (PID: 2424 topped app topped e app unning L app unning | 1) | | | ^ |

Figure 3.70 : La fenêtre de Xampp

f) Création de base de données nommé opc

| phpMuAdmin | 😑 🗊 Serveur : 127.0.0.1 » 👩 Base de données : opc |
|---|--|
| <u>Ω 5 0 0 8 </u> € | M Structure 🔄 SQL 🔍 Rechercher 💿 Requête 🚍 Exporter 🖾 Importer 🥜 Opérations 🖭 Privilèges 🚷 Procédures stockées |
| Récentes Préférées | Filtres |
| Nouvelle base de données information_schema mysql | Contenant le mot : |
| - <u>opc</u> Nouvelle table | Table _ Action Lignes @ Type Interclassement Taille Perte |
| | □ opc |
| ⊕- phpmyadmin ⊕- test | Tout cocher Avec la sélection : |
| | Imprimer 🛺 Dictionnaire de données |

Figure 3.71 : Adresse de server OPC UA

g) Archivage de donnée

On remarque que les valeurs qu'on a modifier au niveau plc sont archivé la figure suivant montre cette modification.

| php <mark>MyAdmin</mark> | ← 📑 Serveur : | 127.0.0.1 » (|) Base de donr | iées : | opc » | 🕞 Tal | ole : opc | | | | |
|---|---------------|---------------|----------------|--------|-------|---------|-------------------|------|--------|-----------|-------|
| <u>Ω 5 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 </u> | Parcourir | M Structu | ure 📄 SQL | . 9 | Rec | cherche | er 📑 Insérer | Exp | porter | Himporter | PI PI |
| Récentes Préférées | | | | | | | | | | | |
| 60 | ←T→ | | ▼ p | r dr | na | nr | timestamp | id | | | |
| Nouvelle base de données | 🔲 🥜 Éditer | 🛃 Copier 🧯 | Supprimer | 0 | 0 | 0 0 | 2022-9-19 1:51:57 | 2785 | | | |
| terent information_schema terent mysql | 🗌 🥜 Éditer | 👫 Copier 🌾 | Supprimer | 0 | 0 | 0 0 | 2022-9-19 1:52:2 | 2786 | | | |
| - opc | 🗌 🥜 Éditer | 📑 Copier 🧯 | Supprimer | 0 | 0 | 0 0 | 2022-9-19 1:52:7 | 2787 | | | |
| + opc | 🗌 🥜 Éditer | 🚮 Copier 🧯 | Supprimer | 0 | 0 | 0 0 | 2022-9-19 1:52:12 | 2788 | | | |
| + performance_schema | 🗌 🥜 Éditer | 🛃 i Copier (| Supprimer | 0 | 0 | 0 0 | 2022-9-19 1:52:17 | 2789 | | | |
| - test | 🗋 🥜 Éditer | 📑 Copier (| Supprimer | 0 | 0 | 0 0 | 2022-9-19 1:52:22 | 2790 | | | |
| | 🗋 🥜 Éditer | Copier 🌾 | Supprimer | 5 | 2 | 0 0 | 2022-9-19 1:52:27 | 2791 | | | |
| | 🗌 🥜 Éditer | 🛃 Copier 🧯 | Supprimer | 5 | 2 | 0 0 | 2022-9-19 1:52:32 | 2792 | | | |
| | 🔲 🥜 Éditer | 🛃 Copier (| Supprimer | 5 | 2 | 1 3 | 2022-9-19 1:52:37 | 2793 | | | |
| | 🗋 🥔 Éditer | 🛃 Copier (| Supprimer | 5 | 2 | 1 3 | 2022-9-19 1:52:42 | 2794 | | | |
| | 🗌 🥜 Éditer | 🛃 Copier 侯 | Supprimer | 5 | 2 | 1 3 | 2022-9-19 1:52:47 | 2795 | | | |
| | 🗌 🥜 Éditer | 📑 Copier (| Supprimer | 5 | 2 | 1 3 | 2022-9-19 1:52:52 | 2796 | | | |
| 00:02 50 | 🗌 🥜 Éditer | Copier 🌾 | Supprimer | 5 | 2 | 1 3 | 2022-9-19 1:52:57 | 2797 | | | |

Figure 3.66 : Archivage des données

3.8 Conclusion

Dans ce chapitre nous avons présenté l'automatisation et la supervision du système et les étapes principales impliquées pour réaliser notre objectif.

Où nous avons présenté les logiciels de programmation TIA Protal et le SAMKOON utilisés.

Dans le côté de programmation, on a plusieurs parties : la programmation de l'API par le logiciel TIA Portal, la programmation de l'IHM, la programmation de l'interface de liaison sous NodeRed et la programmation de la base de donnée MySQL pour l'archivage des données.
Conclusion Générale

Le développement massif des techniques de l'automatisme a permis le passage de la machine câblée à celui des systèmes automatisés de production, qui permettent d'avoir une meilleure qualité des produits en plus de la sécurité et de la flexibilité des processus, mais cela entraîne un accroissement des besoins. Cela explique que les systèmes câblés deviennent trop volumineux et trop rigides pour de telles applications, et que l'on se tourne donc vers des solutions utilisant les techniques de traitement de l'information par processeurs programmables ou les automates programmables industriels occupent une place de choix.

Pour expliquer l'intérêt et l'importance de l'utilisation de l'API dans le domaine de l'industrie (passage de la logique câblée à logique programmer) d'une part et d'autre part, on a choisi comme projet de fin d'études, de réaliser la commande d'une station de pompage.

Pour cela, on a passé en revue les automates programmables industriels de la gamme SIEMENS, leurs caractéristiques, critères de choix, leurs avantages, ainsi que les langages de programmation utilisés. La communication et le transfert d'information via un réseau, rendront un système automatisé plus simple et plus performant par l a diminution du câblage.

Ce travail présentant l'automatisation d'une station de pompage est composée de deux réservoirs qui sont les réservoirs d'aspiration et de refoulement. Le système est donc commandé par un automate programmable industriel de type S7-1500 pour contrôler les différents capteurs et actionneurs nécessaires.

Nous avons aussi supervisé notre système avec un écran tactile de type SAMKOON, et afin de prévoir aussi les défaillances que peut présenter le système, on va les afficher en tant qu'alarmes sur l'écran de supervision.

Nous avons ensuite utilisé l'outil open source NodeRed pour gérer la liaison entre notre API et une base de données avec MySQL que nous avons créé afin de permettre le suivi permanent de la situation de la station de pompage à distance en consultant les données archivées.

En conclusion, je voudrai mentionner que ce travail m' a été très bénéfique car il m' a permis d'enrichir mes connaissances théoriques acquises durant mes études et m'a été d'un grand apport dans la compréhension du processus global du fonctionnement de la station de pompage.

Références Bibliographiques

- [1] M.L. Fas, «Actionneurs», cours en 3^{ème} année Licence, Automatique, Département d'électronique, Université Saad Dahlab de Blida 1, 2017, <u>https://elearning.univ-blida.dz</u>.
- [2] M.L. Fas, « Capteurs et Chaines de Mesures », cours en 3^{ème} année Licence, Automatique, Département d'électronique, Université Saad Dahlab de Blida 1,2017, <u>https://elearning.univ-blida.dz</u>.
- [3] « Poire de niveau », Dans guide directindustry [En ligne]. <u>https://www.directindustry.fr/prod/flygt/product-113401-1101505.html</u> (consulté le 6/9/2022).
- [4] « Principe de fonctionnement des poires », [En ligne]. <u>https://www.seatec.fr/blog/2021/01/15/tout-savoir-sur-les-interrupteurs-flotteurs-pour-pompes.</u>
- [5] « Capteur de niveau », Dans guide.directindustry, [En ligne]. https://guide.directindustry.com/fr/bien-choisir-un-capteur-de-niveau/
- [6] « Débitmètre électromagnétique », <u>https://www.tecfluid.fr/fr/débitmètre-</u> <u>électromagnétique-série-flomid-0</u>
- [7] « Débitmètre électromagnétique », Dans guide.directindustry, [En ligne]. Available : <u>https://www.directindustry.fr/fabricant-industriel/transmetteur-pression-64758.html</u>
- [8] « Electrovanne », [En ligne]. <u>https://euro-makers.com/fr/accessoires-pneumatique/3326-pne-ev-24v5p.html</u>
- [9] P. Berger, « Les moteurs électriques à courant alternatif » , 2017, Moteur Asynchrone, <u>http://philippe.berger2.free.fr/automatique/cours/moteurs/moteurs.html</u>
- [10] A. Chetti, « Pompes et station de pompage », Cours à l'Université Elbayadh, Algérie, 2019, [En ligne] <u>https://www.cu-elbayadh.dz/ar/wp-content/uploads/2020/03/pompes.pdf</u>
- [11] « Pompe », <u>https://www.labonnepompe.com/245-pompe-a-eau-calpeda-nm-65-</u> centrifuge-a-brides-triphase-380v.html
- [12] A. Gonzaga, « Les automates programmable industriels », Geea.org, 2019, [En ligne] <u>https://www.geea.org/IMG/pdf/LES_AUTOMATES_PROGRAMMABLES_INDUSTRI</u> <u>ELS_pour_GEEA.pdf</u>
- [13] A. Gonzaga, « Les automates programmables industriels », 2004, [En ligne] <u>https://sitelec.org/download_page.php?filename=cours/automates_programmables_industr_iels.pdf</u>
- [14] W. Bolton, « Automate programmable industriel , Ed. Dunod, 2015.
- [15] « Manuel SIMATIC S7-1500 », CPU 1515-2 PN (6ES7515-2AM00-0AB0), [En ligne] https://cache.industry.siemens.com/dl/files/167/81162167/att_92038/v1/s71500cpu_1515

2 pn manual en-US en-US.pdf

- [16] Siemens AG, «SIMATIC STEP 7 dans le portal Totally Integrated Automation Portal», 2013, [En ligne] <u>https://documents.pub/document/simatic-step-7-dans-le-ortail-totallyintegrated-automation-portal-.html?page=1</u>
- [17] SIEMENS, « SIEMENS, manuel produit TIA PORTAL V13 », documentation et manuel d'utilisation de TIA PORTAL V13, 2013, [En ligne] <u>https://cache.industry.siemens.com/dl/files/417/109054417/att_854716/v1/STEP_7_Basic_V13_1_frFR_fr-FR.pdf</u>
- [18] «Initiation à la programmation du SIMATIC S7-1200 avec TIA Portal VX », [En ligne] <u>https://sti-monge.fr/maintenancesystemes/wp-ntent/uploads/2013/02/Initiation-1-TIA-Portal-MS1.pdf</u>
- [19] « Système d'informations du SIEMENS »,2017,https://support.industry.siemens.com/cs/start?lc=fr-FR
- [20] « SK- 102 HS », <u>https://p-avt.ru/catalog/paneli-operatora-samkoon/sk-102as.html</u>

Annexes

1. La table de variables

| PLC tags | | | | | | | | | | |
|----------|------|-----------------------------|------------------------|-----------|-----------|--------|---------|-----------------|--------------|----------|
| | N | ame | Tag table | Data type | Address | Retain | Acces | Writa | Visibl | Supervis |
| 1 | - | imp d'acquitement | Table de variabl 💌 | Bool 🔳 | %M0.0 | | | | | ^ |
| 2 | - | Tag_1 | Table de variables s | Bool | %M0.2 | | | | | |
| 3 | -00 | Tag_2 | Table de variables s | Bool | %M0.3 | | | | | = |
| 4 | -00 | Tag_3 | Table de variables s | Bool | %M0.4 | | | | | |
| 5 | - | bS pompe 1 | Table de variables s | Bool | %M0.1 | | | ~ | ~ | |
| 6 | -00 | temps réglable HMI | Table de variables s | Time | %MD2 | | | | ~ | |
| 7 | - | nombre démarrage maximal ré | Table de variables s | Int | %MW6 | | | | ~ | |
| 8 | - | System_Byte | Table de variables s | Byte | %MB1 | | | | ~ | |
| 9 | -00 | FirstScan | Table de variables s | Bool | %M1.0 | | | | ~ | |
| 10 | - | DiagStatusUpdate | Table de variables s | Bool | %M1.1 | | | ~ | ~ | |
| 11 | - | AlwaysTRUE | Table de variables s | Bool | %M1.2 | | | | ~ | |
| 12 | - | AlwaysFALSE | Table de variables s | Bool | %M1.3 | | | \checkmark | ~ | |
| 13 | - | Clock_Byte | Table de variables s | Byte | %MB0 | | | ~ | ~ | |
| 14 | -00 | Clock_10Hz | Table de variables s | Bool | %M0.0 | | | | ~ | |
| 15 | - | Clock_5Hz | Table de variables s | Bool | %M0.1 | | | \checkmark | ~ | |
| 16 | - | Clock_2.5Hz | Table de variables s | Bool | %M0.2 | | | ~ | ~ | |
| 17 | - | Clock_2Hz | Table de variables s | Bool | %M0.3 | | | $\mathbf{\sim}$ | ~ | |
| 18 | - | Clock_1.25Hz | Table de variables s | Bool | %M0.4 | | | | | |
| 19 | - | Clock_1Hz | Table de variables s | Bool | %M0.5 | | | | | |
| 20 | -00 | Clock_0.625Hz | Table de variables s | Bool | %M0.6 | | | | | |
| 21 | - | Clock_0.5Hz | Table de variables s | Bool | %M0.7 | | | | ~ | |
| 22 | -00 | System_Byte(1) | Table de variables s | Byte | %MB1 | | | | | |
| 23 | - | NTB REF | Table de variables s | Bool | %M8.0 | | | | | |
| 24 | -00 | NB REF | Table de variables s | Bool | %M8.1 | | | | | ~ |
| | | | | | | | - | _ | | |
| 25 | -00 | NC REF | Table de variables s | Bool | %M8.2 | | | | | |
| 26 | -00 | NH REF | Table de variables s | Bool | %M8.3 | | | | | |
| 27 | - | NTASP | Table de variables s | Bool | %M8.4 | | | | | |
| 28 | - | NBASP | Table de variables s | Bool | %M8.5 | | | | | |
| 29 | -570 | NC ASP | Table de variables s | Bool | %M8.6 | | | | | |
| 30 | -671 | NH ASP | Table de variables s | Bool | %M8.7 | | | | | |
| 31 | | arret 4 pompes cnd | Table de variables s | Bool | %M9.0 | | | | | L |
| 22 | | commande 1 nomne | Table de variables s | Bool | % MO 1 | | | | | |
| 22 | | commande 2 pomper | Table de variables s | Bool | %MQ 2 | | | | | |
| 24 | | commande de 2 pompes | Table de variables s | Bool | /01VI3.2 | | | | | |
| 25 | | commande de 5 pompes | Table de variables s | Bool | % MO 4 | | | | | |
| 26 | | commanda of sociurs | Table de variables s | Bool | % MO E | | | | | |
| 27 | | dispo P1 | Table de variables s | Bool | %NO 6 | | | | | |
| 20 | | dispo P2 | Table de variables s | BOOI | %NO 7 | | | | | |
| 20 | | dispo P2 | Table de variables s | BOOI | %N10.0 | | | | | |
| 39 | | dispo P3 | Table de variables s | BOOI | %//10.0 | | | | | |
| 40 | | dispo P4 | Table de variables s | BOOI | %MIU.1 | | | | | |
| 41 | | retour de marche P1 | lable de variables s | Bool | %M10.2 | | | | | |
| 42 | 1 | retour de marche P2 | Table de variables s | BOOI | %M10.3 | | | | | |
| 43 | | retour de marche P3 | table de variables s | 8001 | %M10.4 | | | | | |
| 44 | - | retour de marche PSC | lable de variables s | Bool | %M10.5 | | | | | |
| 45 | | commande on pompe secours . | . Table de variables s | Bool | %M10.6 | | | | | |
| 46 | | commande arret pompe 2 | Table de variables s | Bool | %M10.7 | | | | | |
| 47 | | commande arret pompe 3 | Table de variables s | Bool | %M11.0 | | | | | |
| 48 | - | commande arret pompe 1 | Table de variables s | Bool | %M11.1 | | | | | |
| 10 | - | | Table de la 1990 | Deal | N 1 1 1 2 | | | | - | |
| 49 | 1 | commande on pompe 1 | Table de variables s | Bool | %M11.2 | | | | | |
| 50 | | commande on pompe 2 | lable de variables s | Bool | %M11.3 | | | | | |
| 51 | | commande on pompe 3 | Table de variables s | Bool | %M11.4 | | | | | |
| 52 | - | commandeon pompe secours | Table de variables s | Bool | %M11.5 | | | | | |
| 53 | - | commande arret pompe secour | Table de variables s | Bool | %M11.6 | | | | | |
| 54 | - | CMPO 1 NB | Table de variables s | Bool | %M11.7 | | | | | |
| 55 | - | CMPF 1 NB | Table de variables s | Bool | %M12.0 | | | | | |
| 56 | - | CMPO 1 NC | Table de variables s | Bool | %M12.1 | | | | | |
| 57 | - | CMPF 1 NC | Table de variables s | Bool | %M12.2 | | | | | |
| 58 | - | CMPO 1 NH | Table de variables s | Bool | %M12.3 | | | | | |
| 59 | - | CMPF 1 NH | Table de variables s | Bool | %M12.4 | | | ~ | \checkmark | |
| 60 | - | CMPO 2 NB | Table de variables s | Bool | %M12.5 | | | | | |

| 61 | -00 | CMPF 2NB | Table de variables s | Bool | %M12.6 | | | | | [|
|-----|------|-----------------------------------|-----------------------|------|------------|----------|-----------------|-----------------|--------------|---|
| 62 | -00 | CMPO 2 NC | Table de variables s | Bool | %M12.7 | Ä | | | | |
| 63 | | CMPF 2 NC | Table de variables s | Bool | %M13.0 | Ä | | | | |
| 64 | -470 | CMPO 2 NH | Table de variables s | Bool | %M13.1 | Ä | | | | |
| 65 | | CMPE 2 NH | Table de variables s. | Bool | %M13.2 | H | | | | |
| 66 | - | CMPO 3 NB | Table de variables s. | Bool | %M13.3 | H | | | | |
| 67 | | | Table de variables s | Bool | %M12.4 | - | | | | |
| 69 | | CMPO 3 NC | Table de variables s | Bool | %M12.5 | | | | | |
| 60 | | | Table de variables s | Bool | 76IVIT3.5 | | | | | |
| 70 | | CMPC 3 NU | Table de variables s | Bool | %N/13.0 | | | | | |
| 70 | | | Table de variables s | Bool | 76IVI15.7 | | | | | |
| 71 | | | Table de variables s | BOOI | %1/14.0 | | | | | |
| 72 | | CMPOSCINB | lable de variables s | BOOL | %M14.1 | | | | | |
| /3 | | CMPF SC N | Table de variables s | BOOL | %M14.2 | | | | | |
| 74 | | CMPOSENC | lable de variables s | BOOL | %M14.3 | | | | | |
| 75 | | CMPF SC NC | Table de variables s | Bool | %M14.4 | | | | | |
| 76 | | CMPO SC NH | Table de variables s | Bool | %M14.5 | 8 | | | | |
| 77 | | CMPF SC NH | Table de variables s | Bool | %M14.6 | 9 | | | | |
| 78 | | NTB ASP | Table de variables s | Bool | %M14.7 | | | | | |
| 79 | | vanne aspiration ouverte | Table de variables s | Bool | %M15.0 | | | | | |
| 80 | | vanne aspiration fermée | Table de variables s | Bool | %M15.1 | | | | | |
| 81 | | reset hmi | Table de variables s | Bool | %M15.2 | | | ~ | | |
| 82 | | reset hardware | Table de variables s | Bool | %M15.3 | | | | | |
| 83 | | mode auto | Table de variables s | Bool | %M15.4 | | | | | |
| 84 | | mode local | Table de variables s | Bool | %M15.5 | | | | | |
| | | | | | | _ | - | - | - | |
| 85 | | cas 1 on V-ASP | Table de variables s | Bool | %M15.6 | | | | | |
| 86 | | cas 2 on V ASP | Table de variables s | Bool | %M15.7 | | | | | |
| 87 | | cas 3 on V-ASP | Table de variables s | Bool | %M16.0 | | | | \checkmark | |
| 88 | | cas 1 off V-asp | Table de variables s | Bool | %M16.1 | | $\mathbf{\sim}$ | $\mathbf{\sim}$ | | |
| 89 | -00 | cas 2 off V asp | Table de variables s | Bool | %M16.2 | | $\mathbf{\sim}$ | | \checkmark | |
| 90 | | cas 3 off V ASP | Table de variables s | Bool | %M16.3 | | | | | |
| 91 | | niveau très bas aspiration alar | Table de variables s | Bool | %M16.4 | | | | | |
| 92 | | défaut niveau très bas aspiration | Table de variables s | Bool | %M16.5 | | | | | |
| 93 | | défaut redondance contradicto | Table de variables s | Bool | %M16.6 | | | | | |
| 94 | | BS ASP NTB | Table de variables s | Bool | %M16.7 | | | ~ | | |
| 95 | | BS ASP INC | Table de variables s | Bool | %M17.0 | | \checkmark | | | |
| 96 | | défaut discordance aspiration | Table de variables s | Bool | %M17.1 | | | | | |
| 97 | | défaut incohérence aspiration | Table de variables s | Bool | %M17.2 | | | | | |
| 98 | | etat d'acq ntb reset | Table de variables s | Bool | %M17.3 | | | | | |
| 99 | | etat d'acq nc | Table de variables s | Bool | %M17.4 | | | | | |
| 100 | | ordre d'ouverture vanne aspirat. | Table de variables s | Bool | %M17.5 | | | | | |
| 101 | | ordre de fermeture vanne d'asp. | Table de variables s | Bool | %M17.6 | | | | | |
| 102 | -00 | etat fb discordance ouverture | Table de variables s | Bool | %M17.7 | Ā | | | | |
| 103 | - | detat fp discordance fermeture | Table de variables s | Bool | %M18.0 | | | | | |
| 104 | | BS dis ouverture | Table de variables s | Bool | %M18.1 | Ä | | | | |
| 105 | - | bd disc fermeture | Table de variables s | Bool | %M18.2 | | | | | |
| 106 | - | défaut discordance ouvertureV | Table de variables s | Bool | %M18.3 | Ä | | | | |
| 107 | -671 | défaut discordance fermeture | Table de variables s | Bool | %M18.4 | Ä | | | | |
| 108 | -571 | dispo V-ASP | Table de variables s. | Bool | %M18.5 | Ä | | | | |
| | - | | | 2001 | | | | | | |
| 109 | - | BS inclasp | Table de variables s | Bool | %M18.6 | | | | | |
| 110 | - | etat d'acq definc | Table de variables s | Bool | %M18.7 | | | | | |
| 111 | - | bs1 | Table de variables s | Bool | %M19.0 | | | | | |
| 112 | - | bs2 | Table de variables s | Bool | %M19.1 | | | | | |
| 113 | - | bs 3 | Table de variables s | Bool | %M19.2 | | | | | |
| 114 | - | bs4 | Table de variables s | Bool | %M19.3 | | | | | |
| 115 | -00 | cas 1 on V-REF | Table de variables s | Bool | %M19.4 | N | | | | |
| 116 | -00 | cas 2 on V REF | Table de variables s | Bool | %M19.5 | N | | | | |
| 117 | -671 | cas 2 off V REF | Table de variables s | Bool | %M19.6 | | | | | |
| 118 | -67 | cas 3 off V REF | Table de variables s | Bool | %M19.7 | Ŭ. | | | | |
| 119 | - | Cas 3 on v ref | Table de variables s. | Bool | %M20.0 | | | | | |
| 120 | | ordre d'ouverture v ref | Table de variables s | Bool | %M20.1 | | | | | |
| 120 | | orare a ouverture viter | table de valiables 5 | 5001 | 101012-0.1 | | | | | |

Annexes

| 12 | 1 | | ordre de fermeture v ref | Table de variables s. | . Bool | %M20.2 | | | ~ | | |
|------|------------------|--|---------------------------------|-----------------------|--------|------------|----------------------|-----------------|--------------|---------|--|
| 123 | 2 | | v ref ouverte | Table de variables s. | . Bool | %M20.3 | | | | | |
| 123 | 3 | -00 | v ref fermée | Table de variables s. | . Bool | %M20.4 | | | | | |
| 124 | 4 | | défaut discordance ouverture V. | Table de variables s. | . Bool | %M20.5 | Ä | | | | |
| 12 | 5 | - | Défaut niveau ntb REF | Table de variables s | Bool | %M20.6 | Ä | | | | |
| 120 | 6 | - | défaut discordance fermeture v | Table de variables s | Bool | %M20.7 | Ä | | | | |
| 12 | 7. | - | défaut incohérence v ref | Table de variables s | Bool | %M21.0 | Ä | | | | |
| 12 | , g | | dispo v ref | Table de variables s | Bool | %M21.1 | | | | | |
| 120 | 0 | | cas 1 off / PEE | Table de variables s. | Rool | %M21.2 | | | | | |
| 12: | 2 | | | Table de variables s. | . BOOI | 70IVI21.2 | | | | | |
| 130 | | | arret d'urgence | Table de variables s. | . BOOI | %IVI21.3 | | | | | |
| 13 | 1 | | defaut hardware 1 P1 | lable de variables s. | BOOL | %M21.4 | | | | | |
| 13: | 2 | | DH 2 P1 | Table de variables s. | Bool | %M21.5 | <u> </u> | | | | |
| 13 | 3 | | DH 3 P1 | Table de variables s. | . Bool | %M21.6 | | | | | |
| 134 | 4 | | Cas 1 on P1 | Table de variables s. | . Bool | %M21.7 | | | | | |
| 13 | 5 · | | cas 2 on P1 | Table de variables s. | . Bool | %M22.0 | | ~ | ~ | | |
| 13 | 6 | | cas 3 on process | Table de variables s. | . Bool | %M22.1 | | | | | |
| 13 | 7 | -00 | cas 3 on process P1 | Table de variables s. | . Bool | %M22.2 | | \checkmark | \checkmark | | |
| 13 | 8 | -00 | cas 1 off P1 | Table de variables s. | . Bool | %M22.3 | | | | | |
| 139 | 9 | | cas 2 off process P1 | Table de variables s. | . Bool | %M22.4 | | | | | |
| 140 | 0 | | cas 3 off process P1 | Table de variables s. | . Bool | %M22.5 | | | | | |
| 14 | 1 | | ordre de marche P1 | Table de variables s. | . Bool | %M22.6 | | | | | |
| 143 | 2 | -00 | e fp P1 | Table de variables s. | . Bool | %M22.7 | Ä | | | | |
| 14 | 3 | -671 | bascule DD p1 | Table de variables s | Bool | %M23.0 | Ä | | | | |
| 14 | 1 | | commande tempo P1 | Table de variables s | Bool | %M23.1 | | | | | |
| 14 | + | | commande tempor r | lable de vallables s. | . 6001 | /01/12.5.1 | | <u> </u> | | | |
| 145 | | <i>—</i> | tomas ésoulé P1 | Table de veriables e | Peel | 9/ MOD 0 | | | | | |
| 143 | | - | Different elegene fragment Pt | Table de variables s | Bool | %IVI25.2 | | | | | |
| 146 | > - | 1 | Defaut classe f nomre P1 | lable de variables s | BOOL | %M23.3 | | M | | | |
| 147 | ′ ◄ | 1 | Défaut discordance P1 | Table de variables s | Bool | %M23.4 | | | | | |
| 148 | 3 - | 1 | DH 1 P2 | Table de variables s | Bool | %M23.5 | | | | | |
| 149 | • | 1 | DH 2 P2 | Table de variables s | Bool | %M23.6 | | | | | |
| 150 |) - | 1 | DH3 P2 | Table de variables s | Bool | %M23.7 | | | \checkmark | | |
| 151 | | • | CAS 1 on P2 | Table de variables s | Bool | %M24.0 | | \checkmark | \checkmark | | |
| 152 | 2 - | • | cas 2 on P2 | Table de variables s | Bool | %M24.1 | | < | ~ | | |
| 153 | 3 4 | 1 | Cas 3 on p2 | Table de variables s | Bool | %M24.2 | | | | | |
| 154 | L 4 | • | cas 1 off p2 | Table de variables s | Bool | %M24.3 | | ~ | | | |
| 155 | ; 4 | a | cas 2 off p2 | Table de variables s | Bool | %M24.4 | $\overline{\square}$ | | | | |
| 156 | ; - | ണ | cas 3 off p2 | Table de variables s. | Bool | %M24.5 | | | | | |
| 157 | | - - | ordre de marche P2 | Table de variables s | Bool | %M24.6 | | | | | |
| 159 | | | etatin P.2 | Table de variables s | Bool | %M24.7 | | | | | |
| 150 | | | basevia DD P2 | Table de variables s | Bool | % M2E 0 | 8 | | | | |
| 105 | | | bascule DD 12 | Table de variables s | Bool | %M25.0 | | | | | |
| 160 | , . | - - | cmp P2 | lable de variables s | BOOL | %M25.1 | | | | | |
| 161 | - | • | Defaut nombre demarrage P2 | lable de variables s | Bool | %M25.2 | | | | | |
| 162 | 2 4 | 1 | temps écoulé P2 | Table de variables s | Bool | %M25.3 | | | | | |
| 163 | 3 4 | 1 | défaut discordance P2 | Table de variables s | Bool | %M25.4 | | | | | |
| 164 | + + | 1 | DH 2 P3 | Table de variables s | Bool | %M25.5 | | ~ | ~ | | |
| 165 | 5 - | 11 | DH 1 P3 | Table de variables s | Bool | %M25.6 | | \checkmark | | | |
| 166 | 5 4 | 1 | DH 3 P3 | Table de variables s | Bool | %M25.7 | | | \checkmark | | |
| 167 | 7 - - | 1 | cas 1 on P3 | Table de variables s | Bool | %M26.0 | | \checkmark | \checkmark | | |
| 168 | 3 - | 1 | cas 2 on P3 | Table de variables s | Bool | %M26.1 | | \checkmark | | | |
| | | | | | | | _ | _ | _ | _ | |
| 169 | | 1 | cas 3 on P3 | Table de variables s | Bool | %M26.2 | | \checkmark | \checkmark | | |
| 170 | • | - | Cas 1 off p3 | Table de variables s | Bool | %M26.3 | | $\mathbf{\sim}$ | \checkmark | | |
| 171 | • | - | cas 2 off p3 | Table de variables s | Bool | %M26.4 | | \checkmark | \checkmark | | |
| 172 | 2 - | - | cas 3 off p3 | Table de variables s | Bool | %M26.5 | | | ~ | | |
| 173 | 3 - | - | ordre de marche P3 | Table de variables s | Bool | %M26.6 | | | | | |
| 174 | 1 | - | etat fp P3 | Table de variables s | Bool | %M26.7 | | | | | |
| 175 | 5 - | - | bascule dd p3 | Table de variables s | Bool | %M27.0 | | | | | |
| 176 | 5 - | - | commande tempo p3 | Table de variables s | Bool | %M27.1 | | | | | |
| 177 | 7 | (| temps écoulé P3 | Table de variables s | Bool | %M27.2 | | | | | |
| 178 | 3 | a | défaut nomre démmarage p3 | Table de variables s | Bool | %M27.3 | | | | | |
| 179 | , | - - | défaut discordance n3 | Table de variables s | Bool | %M27.4 | ă | | | | |
| 180 | | | DH 1 PSC | Table de variables s. | Bool | %M27.5 | | | | | |
| 1.04 | | | DH 2 PSC | Table de variables 5 | Bool | %M27.6 | | | | | |
| 101 | | - | | Table de variables S | Real | 0/M07 7 | | | | | |
| 182 | - | | UN S FSC | Table de variables s | Bool | 701V127.7 | | | | | |
| 183 | | Image: Construction of the second seco | cas i on psc | able de variables s | 5001 | 761/128.0 | | | | | |
| 184 | + • | <□ | cas 2 on psc | lable de variables s | ROOL | %M28.1 | | | | | |
| 185 | • | 1 | cas 3 on psc | lable de variables s | Bool | %M28.2 | | | | | |

Annexes

| 186 | -00 | cas 1 off psc | Table de variables s | Bool | %M28.3 | | | |
|-----|-----|------------------------------|----------------------|------|--------|--------------|----------|--|
| 187 | - | cas 2 off psc | Table de variables s | Bool | %M28.4 | ~ | ~ | |
| 188 | - | cas3 off psc | Table de variables s | Bool | %M28.5 | | | |
| 189 | - | ordre de marche psc | Table de variables s | Bool | %M28.6 | | | |
| 190 | -00 | etat fp psc | Table de variables s | Bool | %M28.7 | | | |
| 191 | -00 | bascule dd psc | Table de variables s | Bool | %M29.0 | | | |
| 192 | -00 | commande tempo psc | Table de variables s | Bool | %M29.1 | \checkmark | | |
| 193 | -00 | temps écoulé psc | Table de variables s | Bool | %M29.2 | \checkmark | | |
| 194 | | défaut démmarage atteint psc | Table de variables s | Bool | %M29.3 | | | |
| 195 | -00 | défaut discordance psc | Table de variables s | Bool | %M29.4 | | | |