

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Ecole Nationale Polytechnique



المدرسة الوطنية المتعددة التقنيات  
Ecole Nationale Polytechnique

Département d'Automatique

Mémoire de projet de fin d'études  
pour l'obtention du diplôme d'ingénieur d'état en automatique

# Réalisation, commande et supervision d'un prototype d'une station de pompage

**CHEKEBKEB Daoud**  
**BOURAS Kamel Eddine**

Sous la direction de **M. El Madjid BERKOUK Pr.**

Présenté et soutenu publiquement le (02/07/2019)

**Composition du jury :**

Président	<b>M. Hachemi CHEKIREB</b>	<b>Pr.</b>	<b>ENP</b>
Rapporteur	<b>M. El Madjid BERKOUK</b>	<b>Pr.</b>	<b>ENP</b>
Examineur	<b>M. Lahcen ABDELOUAL</b>	<b>MA.</b>	<b>ENP</b>

ENP 2019



REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Ecole Nationale Polytechnique



المدرسة الوطنية المتعددة التقنيات  
Ecole Nationale Polytechnique

Département d'Automatique

Mémoire de projet de fin d'études  
pour l'obtention du diplôme d'ingénieur d'état en automatique

# Réalisation, commande et supervision d'un prototype d'une station de pompage

**CHEKEBKEB Daoud**  
**BOURAS Kamel Eddine**

Sous la direction de **M. El Madjid BERKOUK Pr.**

Présenté et soutenu publiquement le (02/07/2019)

**Composition du jury :**

Président	<b>M. Hachemi CHEKIREB</b>	<b>Pr.</b>	<b>ENP</b>
Rapporteur	<b>M. El Madjid BERKOUK</b>	<b>Pr.</b>	<b>ENP</b>
Examineur	<b>M. Lahcen ABDELOUAL</b>	<b>MA.</b>	<b>ENP</b>

ENP 2019

## ملخص:

يستند العمل المقدم في هذه المذكرة أساساً على مراحل صنع نموذج لمحطة ضخ المياه واستعمال المتحكم المنطقي المبرمج S7-1200 لشركة SIEMENS للتحكم. كما استعملت واجهة الآلة البشرية SIMATIC Comfort TP900 لمراقبة هذه المحطة، بالإضافة إلى ضابط السرعة ALTIVAR 12 من Schneider Electric من أجل تغيير سرعة المضخة.

## الكلمات المفتاحية:

المتحكم المنطقي المبرمج، TIA PORTAL، SIEMENS، واجهة الآلة البشرية، ضابط السرعة، نموذج محطة ضخ، WinCC.

## ABSTRACT:

The presented work in this thesis concerns the construction of a pumping station prototype. We use the SIEMENS S7-1200 PLC for the control and supervision tasks, along with SIMATIC Comfort TP900 as a Human/Machine interface programmed by the TIA PORTAL V15 software.

We undertook a speed driver ALTIVAR 12 from Schneider Electric to vary the speed of the pump.

## Keywords:

API, TIA Portal V15, SIEMENS, HMI, pumping station prototype, Variable-frequency drive, WinCC.

## RÉSUMÉ :

Le travail présenté dans ce mémoire porte principalement sur les phases de construction d'un prototype d'une station de pompage et sur l'utilisation de l'automate SIEMENS S7-1200 pour la commande et la supervision, à l'aide d'une interface Homme-Machine SIMATIC Comfort TP900 programmés par le logiciel TIA PORTAL V15.

Nous avons entrepris la mise en oeuvre d'un variateur de vitesse ALTIVAR 12 de Schneider Electric pour faire varier la vitesse de la pompe.

## Mots-clés :

API, TIA Portal V15, SIEMENS, HMI, variateur de vitesse, prototype d'une station de pompage, WinCC.

## *Dédicace*

*Je dédie ce travail à mes chers parents Bayoub et Fatima  
et à ma grand mère Aicha.*

*À mes frères et soeurs que je chérie.*

*À toute ma famille.*

*À Merouane, Yacine, Kamel, Mancef, Nisso, Johnny,  
Nounou, Ishak, Abdellah, Amine, Nabil, Abdou et Rayane  
la seconde famille que j'ai choisi d'avoir.*

*Et bien sur à LPB, Lydia, Meriem, Lera et Toni  
et tous ceux qui m'ont encouragé et ont fait confiance à mes  
Capacités pendant mon parcours.*

*Daoud*

## *Dédicace*

*À mes chers parents Mohammed et Nanna  
pour tous leurs sacrifices, leur amour, et leurs prières tout au  
long de mes études.*

*À mes frères et ma soeur que je chérie.*

*À toute ma famille.*

*À mes chers amis : Mohammed (B), Salah, Smail(B),  
Daoud, Yahia, Smail(T), Nadir, Mounir, Toufik, Nassim,  
Brahim, Youcef, Seddik, Abdou, Rostom, Mohammed(M),  
Noureddine, Cherif, Abdeljalil et Housseem.*

*Je ne peux trouver les mots justes et sincères pour vous  
exprimer mon affection et mes pensées, vous êtes pour moi  
des frères, des amis sur qui je peux compter. en témoignage  
de l'amitié qui nous uni*

*je vous dédie ce travail et je vous souhaite une vie pleine  
de santé et de Bonheur.*

*Kamel Eddine BOURAS*

# *Remerciements*

On remercie Dieu le Tout Puissant de nous avoir donné le courage, la santé, et Nous a accordé son soutien durant les périodes les plus difficiles.

On remercie vivement notre encadreur, Monsieur **El madjid BERKOUK** de nous avoir aidé dans notre travail, et surtout d'avoir cru à nos compétences.

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

On remercie chaleureusement nos familles et tous nos amis pour leur soutien et leurs encouragements.

On souhaite aussi remercier tous les enseignants de l'Ecole Nationale Polytechnique d'Alger, et en particulier, nos professeurs d'Automatique qui nous ont encadrée auparavant et tous nos enseignants pour les connaissances qu'ils nous ont transmis, leur disponibilité et leurs efforts.

Que tous ceux qui ont contribué de près ou de loin à la réalisation de ce modeste travail trouvent ici l'expression de notre sincère gratitude.

# Table des matières

Liste des tableaux

Liste des figures

Liste des abréviations

Introduction générale 15

Plan de travail 16

**I Automates Programmables Industriels et Interfaces Homme-Machine 17**

I.1 Automates Programmables Industriels API . . . . . 18

I.1.1 Historique . . . . . 18

I.1.2 Définition générale . . . . . 18

I.1.3 Architecture des automates programmables industriels [1] . . . . . 19

I.1.4 Structure interne des automates programmables . . . . . 20

I.1.4.1 Le processeur . . . . . 20

I.1.4.2 Les modules d'entrées/sorties . . . . . 21

I.1.4.3 Les mémoires . . . . . 21

I.1.4.4 L'alimentation . . . . . 21

I.1.4.5 Liaisons de communication . . . . . 21

I.1.5 Fonctionnement cyclique d'un API . . . . . 22

I.1.6 Choix d'un automate programmable . . . . . 23

I.1.7 Langages de programmation [2] . . . . . 24

I.1.8 Présentation de la gamme SIMATIC S7-1200 de SIEMENS [3] . . . . . 25

I.1.8.1 La conception modulaire et flexible . . . . . 25

I.1.8.2 Communication industrielle . . . . . 26

I.1.8.3 La mémoire de l'API S7-1200 : . . . . . 26

I.1.8.4 Les modules d'entrées/sorties : . . . . . 26

I.1.8.5 Les boucles de régulation : . . . . . 26

I.2 Interfaces Homme-Machine IHM . . . . . 27

I.2.1 Définition générale . . . . . 27

I.2.2 Les différentes applications des IHM . . . . . 27

I.2.3 Présentation de la gamme SIMATIC HMI Comfort Panels [4] . . . . . 28

I.2.4	Les interfaces intégrées (réseaux industriels) . . . . .	29
I.3	Conclusion . . . . .	29
<b>II</b>	<b>Variateurs de Vitesse Industriels</b>	<b>30</b>
II.1	Historique . . . . .	31
II.2	Définition [5] . . . . .	31
II.3	Constitution . . . . .	32
II.3.1	Module de puissance . . . . .	32
II.3.2	Module de contrôle . . . . .	33
II.4	Quadrants de fonctionnement des variateurs [6] . . . . .	33
II.5	Les principales fonctions des variateurs de vitesse électroniques [7] . . . . .	34
II.6	Variateur de vitesse de Schneider Electric ATV12H037M2 . . . . .	35
II.6.1	Description du variateur ATV12H037M2 et ses applications . . . . .	35
II.6.2	Caractéristiques du variateur de vitesse Schneider ATV12H037M2 . . . . .	36
II.6.3	Schéma de câblage du variateur avec l'alimentation monophasée . . . . .	36
II.7	Programmation manuelle des variateurs . . . . .	38
II.7.1	Commande du variateur par 2 fils ou 3 fils . . . . .	38
II.7.1.1	La commande du variateur par 2 fils . . . . .	38
II.7.1.2	Commande du variateur par 3 fils . . . . .	39
II.7.2	Référence de vitesse . . . . .	40
II.7.3	Sortie analogique du variateur . . . . .	41
II.7.4	Configuration "Perte d'une phase" . . . . .	42
II.7.5	Le bruit du moteur . . . . .	42
II.7.6	Type de commande utilisé par le variateur [8] . . . . .	43
II.8	Conclusion . . . . .	44
<b>III</b>	<b>Modélisation de la station de pompage et simulation avec Matlab</b>	<b>45</b>
III.1	Introduction . . . . .	46
III.2	Modélisation du système . . . . .	46
III.3	Synthèse du régulateur . . . . .	49
III.3.1	Effet du gain sur la réponse du système . . . . .	50
III.4	Simulation du système en boucle fermée avec Matlab Simulink . . . . .	52
III.5	Conclusion . . . . .	54
<b>IV</b>	<b>Réalisation et gestion d'un prototype de station de pompage</b>	<b>55</b>
IV.1	Introduction . . . . .	56
IV.2	Définition du cahier des charges . . . . .	56
IV.2.1	Mode automatique (commande en boucle fermée) . . . . .	57
IV.2.2	Mode manuel (commande en boucle ouverte) . . . . .	58
IV.3	Présentation du prototype de la station de pompage . . . . .	59
IV.3.1	Description des éléments de l'installation . . . . .	59
IV.4	Gestion de la station à l'aide de l'IHM . . . . .	62
IV.4.1	Vue d'accueil . . . . .	63
IV.4.2	Vue de la station . . . . .	64
IV.4.3	Grafcet . . . . .	66
IV.4.4	Boucle de régulation PID . . . . .	66
IV.4.5	Fonctionnement en mode manuel . . . . .	68
IV.4.6	Alarmes et archivage . . . . .	69

IV.5 Les interfaces d'Entrées/Sorties et mesure des grandeurs . . . . .	71
IV.5.1 Interface d'entrée pour le débitmètre . . . . .	72
IV.5.2 Convertisseur courant tension . . . . .	73
IV.5.3 Les contacteurs et les relais . . . . .	74
IV.5.4 Un relais . . . . .	75
IV.6 Conclusion . . . . .	75
<b>Conclusion générale</b>	<b>76</b>
<b>Bibliographie</b>	<b>77</b>
<b>Annexe A : Programmation de l'API</b>	<b>78</b>
<b>Annexe B : Capteurs</b>	<b>91</b>
<b>Annexe C : Variateurs de vitesse Industriels</b>	<b>96</b>
<b>Annexe D : TIA PORTAL, logiciel de programmation des API/IHM SIEMENS</b>	<b>116</b>
<b>Annexe E : Logiciel de configuration des variateurs "Somove"</b>	<b>136</b>
<b>Annexe F : Travaux pratiques proposés pour exploiter la maquette</b>	<b>146</b>

# Liste des tableaux

III.1 Les paramètres et leurs valeurs numériques . . . . .	47
IV.1 les éléments dans le prototype de station de pompage . . . . .	60
IV.2 Disposition des éléments dans l'armoire électrique . . . . .	61
IV.3 suite du tableau "Disposition des éléments dans l'armoire électrique" .	62
IV.4 Les éléments de "Vue de la station" . . . . .	65
IV.5 Les éléments de "Vue de la station" . . . . .	66
IV.6 Les éléments de "Vue PID" . . . . .	67
IV.7 Les éléments de "Vue mode manuel" . . . . .	68
IV.8 Les éléments de "Vue mode manuel" . . . . .	70
IV.9 explication de la configuration du matériel . . . . .	120
IV.10expliquer de l'environnement de travail dans TIA Portal . . . . .	121

# Table des figures

I.1	Structure interne d'un API . . . . .	20
I.2	Grafcet du fonctionnement cyclique d'un API . . . . .	22
I.3	API SIMATIC S7-1200 de SIEMENS . . . . .	25
I.4	API SIMATIC S7-1200 avec ses différents module . . . . .	26
I.5	SIMATIC HMI TP900 COMFORT . . . . .	28
I.6	Interfaces de SIMATIC HMI TP900 COMFORT . . . . .	29
II.1	Structure générale d'un variateur de vitesse électronique . . . . .	32
II.2	Module de puissance d'un variateur de vitesse . . . . .	33
II.3	Quadrants de fonctionnement de la machine asynchrone . . . . .	34
II.4	Variateur de vitesse Schneider ATV12H037M2 . . . . .	35
II.5	Schéma de câblage du variateur ATV12H037M2 . . . . .	37
II.6	Câblage associé à la commande par 2 fils . . . . .	38
II.7	Le fonctionnement à deux fils . . . . .	38
II.8	Câblage associé à commande par 3 fils . . . . .	39
II.9	Le fonctionnement à deux fils . . . . .	39
II.10	Entrée analogique du variateur . . . . .	40
II.11	Sortie analogique du variateur AO1 . . . . .	41
II.12	Fonctionnement de la sortie analogique . . . . .	41
II.13	La courbe du couple en fonction de la fréquence . . . . .	43
III.1	Schéma du système modélisé . . . . .	46
III.2	Le diagramme de Bode de la fonction de transfert du système . . . . .	49
III.3	La réponse à un échelon . . . . .	49
III.4	La réponse à un échelon . . . . .	51
III.5	La réponse à un échelon . . . . .	51
III.6	La réponse à un échelon . . . . .	51
III.7	Schéma bloc du système en boucle fermée réalisé sur Simulink . . . . .	52
III.8	Réponse du système en boucle fermée . . . . .	53
III.9	La simulation sur simulink . . . . .	53
IV.1	Schéma du système modélisé . . . . .	57
IV.2	Grafcet du projet . . . . .	58
IV.3	Disposition des éléments dans le prototype de station de pompage . . . . .	59
IV.4	Figure Disposition des éléments dans l'armoire électrique . . . . .	61
IV.5	Les voyants sur l'armoire . . . . .	62

IV.6	vue d'accueil : intitulé du projet . . . . .	63
IV.7	Vue d'accueil : Vue de la station . . . . .	63
IV.8	Vue d'accueil : Menu . . . . .	64
IV.9	Vue de la station . . . . .	65
IV.10	Vue Grafset . . . . .	66
IV.11	Vue PID . . . . .	67
IV.12	Vue mode manuel . . . . .	68
IV.13	Vue alarmes et archivage . . . . .	69
IV.14	Archivage des alarmes . . . . .	70
IV.15	Archivage des mesures(débit, niveau de l'eau) . . . . .	71
IV.16	Schéma général de câblage . . . . .	72
IV.17	Filtre RC . . . . .	73
IV.18	Convertisseur courant tension . . . . .	74
IV.19	Contacteur de puissance . . . . .	74
IV.20	Relais Schneider Electric modèle RSB2A080BDS . . . . .	75
IV.21	Capteur de distance à ultrasons SICK UM18 . . . . .	92
IV.22	Débimètre Arduino . . . . .	94
IV.23	Code Arduino utilisé . . . . .	95
IV.24	Schéma de câblage de l'alimentation ATV12 . . . . .	97
IV.25	Schéma de câblage du variateur ATV12H037M2 . . . . .	98
IV.26	Entrée analogique du variateur . . . . .	99
IV.27	Fonctionnement du relais . . . . .	100
IV.28	Schéma de cablage du relais . . . . .	100
IV.29	Tableau de combinaison des entrées de vitesses présélectionnées . . . . .	101
IV.30	Comment introduire une vitesse présélectionnée . . . . .	102
IV.31	fonctionnement des Vitesses présélectionnées . . . . .	103
IV.32	La résistance de freinage VW3A7702 . . . . .	105
IV.33	La compatibilité de la résistance avec les variateurs . . . . .	105
IV.34	Le module de freinage VW3A7005 . . . . .	106
IV.35	Le Schéma de cablage du module de freinage . . . . .	107
IV.36	le module VW3A7005 n'existe plus . . . . .	107
IV.37	Ouverture d'un projet dans TIA Portal . . . . .	118
IV.38	Création d'un projet dans TIA Portal . . . . .	118
IV.39	Les étapes d'un projet d'automatisation dans TIA Portal . . . . .	119
IV.40	Configuration et ajout du matériel dans un projet TIA Portal . . . . .	119
IV.41	Choix des API et des CPU . . . . .	120
IV.42	Environnement de travail de TIA Portal . . . . .	121
IV.43	Ajout des modules et des éléments sur le rack . . . . .	122
IV.44	Configuration des modules E/S . . . . .	123
IV.45	Création des mnémoniques et édition d'adressage . . . . .	123
IV.46	Environnement de travail de TIA Portal . . . . .	124
IV.47	Simulation du projet dans TIA Portal . . . . .	125

IV.48	Interface du simulateur de TIA Portal . . . . .	125
IV.49	Visualisation des état lors de l'exécution du projet . . . . .	126
IV.50	Ajout des blocs dans TIA Portal . . . . .	127
IV.51	Configuration des langages de programmation des blocs de fonctions . . . . .	127
IV.52	Programmation d'un bloc de fonction . . . . .	128
IV.53	Insertion d'une instance de bloc de fonction dans le programme prin- cipale . . . . .	128
IV.54	Ajout d'une IHM dans TIA Portal . . . . .	129
IV.55	Choix d'une IHM dans un projet TIA Portal . . . . .	130
IV.56	Configuration d'une IHM dans TIA Portal . . . . .	130
IV.57	Configuration d'une IHM dans TIA Portal_2 . . . . .	131
IV.58	vue principale sur l'environnement de travail . . . . .	131
IV.59	Table des variables . . . . .	132
IV.60	Création des objets . . . . .	133
IV.61	Affectation des variables . . . . .	134
IV.62	Simulation du projet . . . . .	135

# Liste des abréviations

**AC** : Alternative Current

**AI** : Analog Input

**AO** : Analog Output

**API** : Automate Programmable Industriel

**CPU** : Processing Unit

**DB** : Data Bloc

**DC** : Direct Current

**DI** : Digital Input

**DO** : Digital Output

**EEPROM** : Electrically Erasable Programmable Read Only Memory

**E/S** : Entrée/Sortie

**FB** : Fonctionnel Bloc

**FC** : Fonction

**FM** : Function Module

**IHM** : Interface Homme-Machine

**INT** : Integer

**LED** : Light Emitting Diode

**LIST** : Le langage de liste d'instructions

**LOG** : LOGigramme

**OB** : Organisationnel Bloc

**PC** : Personal Computer

**PLC** : Programmable Logic Controller

**PWM** : Pulse Width Modulation

**PROFIBUS** : Process Field Bus

**PROM** : Programmable Read Only Memory

**RAM** : Random Access Memory

**ROM** : Read Only Memory

**SFB** : System Fonctionnel Bloc

**SFC** : System Fonction

**SIMATIC** : Siemens Automatic

**SM** : Signal Module

**S7** : Step 7

**TOR** : Tout Ou Rien

**TIA PORTAL** : Totally Integrated Automation Portal

# Introduction générale

Dans le but de former ses élèves ingénieurs automaticiens à la maîtrise de l'informatique industrielle, le Département d'Automatique de l'ENP a proposé le projet d'un prototype d'une station de pompage didactique, où la mise en marche et le développement de cette dernière permet aux élèves ingénieurs de se familiariser avec la gestion et la supervision des systèmes industriels à l'aide d'API/IHM.

Notre travail consiste à réaliser un prototype de station de pompage, englobant l'essentiel des systèmes que nous puissions trouver dans une station de pompage réelle, que ce soit du point de vue Hardware ou Software.

La communication entre la station et le pupitre de visualisation se fait via un réseau Profinet.

Ce travail a nécessité l'étude préalable du logiciel de programmation des API Siemens STEP7 et le logiciel de conception des interfaces homme-machine, WinCC flexible présents tous les deux sous un seul environnement commun et compact le TIA PORTAL V15.

# Plan de travail

Le premier chapitre présente une étude sur les automates programmables industriels et les interfaces Homme-Machine, en décrivant leur architecture, fonctionnement et rôle dans les projets d'automatisation. Ensuite on se focalise sur l'API SIMATIC S7-1200 et l'IHM Comfort TP900 utilisés dans notre projet, en abordant leurs fonctionnalités et caractéristiques en détail.

Le second chapitre sera consacré à la description des variateurs de vitesse en général, architecture et principe de fonctionnement, ensuite on s'intéresse au variateur de vitesse ATV12H037M2 de Schneider Electric et ses application.

On passe à la modélisation de la station de pompage dans le troisième chapitre, afin de synthétiser un régulateur adéquat pour commander notre système, ensuite on simule ce dernier dans Matlab Simulink avant de passer à l'implémentation.

Ces trois derniers chapitres constituent une préparation au projet qui nous a été soumis. Le dernier chapitre présente les étapes suivies pour la réalisation du prototype et l'étude de ses performances, dans lequel on a décrit la partie électrique, l'instrumentation ainsi que le cahier des charges traduit et programmé dans l'API/IHM l'aide du logiciel TIA Portal.

Chapitre I

Automates Programmables

Industriels et Interfaces

Homme-Machine

## **Introduction**

L'automate programmable industriel A.P.I est aujourd'hui le constituant le plus répandu pour réaliser des automatismes. On le trouve pratiquement dans tous les secteurs de l'industrie car il répond à des besoins d'adaptation et de flexibilité pour un grand nombre d'opérations. Cette émergence est due en grande partie, à la puissance de son environnement de développement et aux larges possibilités d'interconnexions.

Le système d'interface homme-machine (IHM) constitue l'interface entre l'opérateur et le processus. Le déroulement du processus est commandé par la CPU. L'opérateur peut visualiser le processus ou intervenir dans le processus en cours par le biais d'un pupitre opérateur.

## **I.1 Automates Programmables Industriels API**

### **I.1.1 Historique**

Les automates programmables industriels sont apparus à la fin des années soixante aux Etats Unis, à la demande de l'industrie automobile américaine (General Motors en leader), qui exigeait plus d'adaptabilité de ses systèmes de commande. Ce n'est qu'en 1971 qu'ils firent leur première apparition en France. Les années soixante-dix connaissent une explosion des besoins industriels dans le domaine de l'automatique, de la flexibilité et l'évolutivité des Systèmes Automatisés de production SAP [5].

### **I.1.2 Définition générale**

L'automate programmable industriel A.P.I ou Programmable Logic Controller PLC est un appareil électronique programmable. Il est défini suivant la norme française EN-61131-1, adapté à l'environnement industriel, et réalise des fonctions d'automatisme pour assurer la commande de préactionneurs et d'actionneurs à partir d'informations logiques, analogiques ou numériques. C'est aujourd'hui le constituant essentiel des automatismes. On le trouve non seulement dans tous les secteurs de l'industrie, mais

aussi dans les services et dans l'agriculture. La force principale d'un automate programmable industriel API réside dans sa grande capacité de communication avec l'environnement industriel. Outre son unité centrale et son alimentation, il est constitué essentiellement de modules d'entrées/sorties, qui lui servent d'interface de communication avec le processus industriel de conduite. Il a comme rôles principaux dans un processus :

- Assurer l'acquisition de l'information fournie par les capteurs.
- En faire le traitement.
- Élaborer la commande des actionneurs.
- Assurer également la communication pour l'échange d'informations avec l'environnement.

### **I.1.3 Architecture des automates programmables industriels**

#### **[1]**

Les automates sont organisés suivant l'architecture suivante :

- Un module d'unité centrale ou CPU, qui assure le traitement de l'information et la gestion de l'ensemble des unités. Ce module comporte un microprocesseur, des circuits périphériques de gestion des entrées/sorties, des mémoires RAM et EEPROM nécessaire pour stocker les programmes, les données, et les paramètres de configuration du système.
- Un module d'alimentation qui, à partir d'une tension 220V/50Hz ou dans certains cas de 24V fournit les tensions continues + /- 5V, +/-12V ou +/- -15V.
- Un ou plusieurs modules d'entrées 'Tout ou Rien' ou analogiques pour l'acquisition des informations provenant de la partie opérative (procédé à conduire).
- Un ou plusieurs modules de sorties 'Tout ou Rien' (TOR) ou analogiques pour transmettre à la partie opérative les signaux de commande. Il y a des modules qui intègrent en même temps des entrées et des sorties.

- Un ou plusieurs modules de communication comprenant :
  - ★ Interfaces série utilisant dans la plupart des cas comme support de communication, les liaisons RS-232 ou RS422/RS485.
  - ★ Interfaces pour assurer l'accès à un bus de terrain.
  - ★ Interface d'accès à un réseau Ethernet.

### I.1.4 Structure interne des automates programmables

La structure matérielle interne d'un API obéit au schéma donné sur la figure ci-dessous :

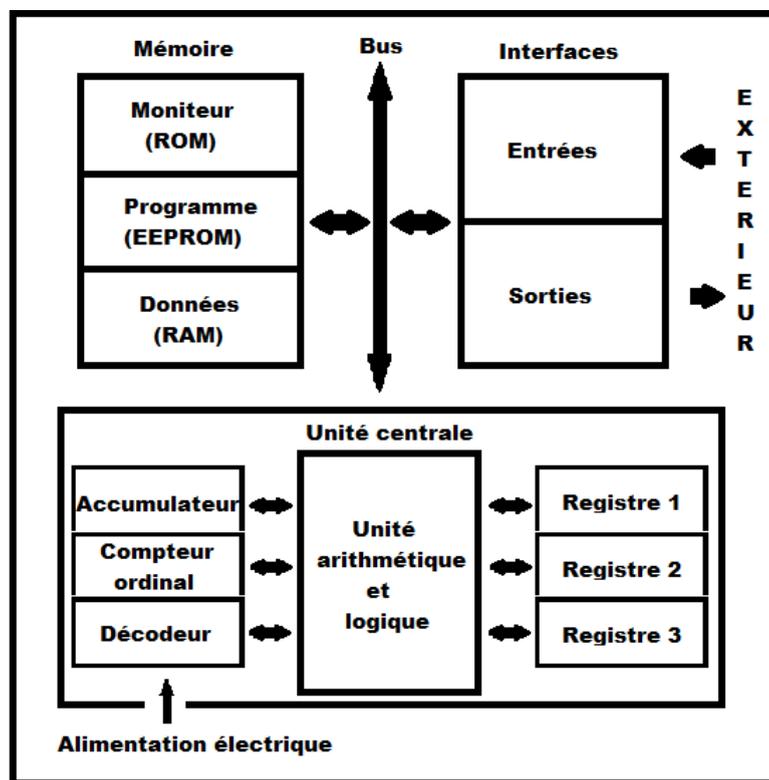


FIGURE I.1: structure interne d'un API [5]

Détaillons successivement chacun des composants qui apparaissent sur ce schéma.

#### I.1.4.1 Le processeur

Il Constitue le coeur de l'appareil dans l'unité centrale. En fait, un processeur se subdivise en une multitude de processeurs partiels plus petits, liés les uns aux autres.

#### **I.1.4.2 Les modules d'entrées/sorties**

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

- Modules TOR : l'information traitée ne peut prendre que deux états (vrai/faux, 0 ou 1 ...) C'est le type d'information délivrée par une cellule photoélectrique, un bouton poussoir ... etc.
- Modules TOR : l'information traitée ne peut prendre que deux états (vrai/faux, 0 ou 1 ...) C'est le type d'information délivrée par une cellule photoélectrique, un bouton poussoir ... etc.
- Modules spécialisés : l'information traitée est contenue dans des mots codés sous forme binaire ou bien hexadécimale. C'est le type d'information délivrée par un ordinateur ou un module intelligent.

#### **I.1.4.3 Les mémoires**

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

- De stocker le système d'exploitation dans des ROM ou PROM.
- Le programme dans des EEPROM.
- Les données système lors du fonctionnement dans des RAM.

#### **I.1.4.4 L'alimentation**

Elle assure la distribution d'énergie aux différents modules. L'automate est alimenté généralement par le réseau monophasé 230V-50 Hz mais d'autres alimentations sont possibles (110V ... etc.).

#### **I.1.4.5 Liaisons de communication**

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

- avec l'extérieur par des borniers sur lesquels arrivent des câbles transportant le signal électrique.
- avec l'intérieur par des bus reliant divers éléments, afin de d'échanger des données, des états et des adresses [9].

### I.1.5 Fonctionnement cyclique d'un API

L'automate programmable reçoit les informations relatives à l'état du système et puis commande les pré-actionneurs suivant le programme inscrit dans sa mémoire. Généralement les automates programmables industriels ont un fonctionnement cyclique (Figure I.2). Le microprocesseur réalise toutes les fonctions logiques ET, OU, les fonctions de temporisation, de comptage, de calcul... Il est connecté aux autres éléments (mémoire et interface E/S) par des liaisons parallèles appelées ' BUS ' qui véhiculent les informations sous forme binaire.. Lorsque le fonctionnement est dit synchrone par rapport aux entrées et aux sorties, le cycle de traitement commence par la prise en compte des entrées qui sont figées en mémoire pour tout le cycle [5].

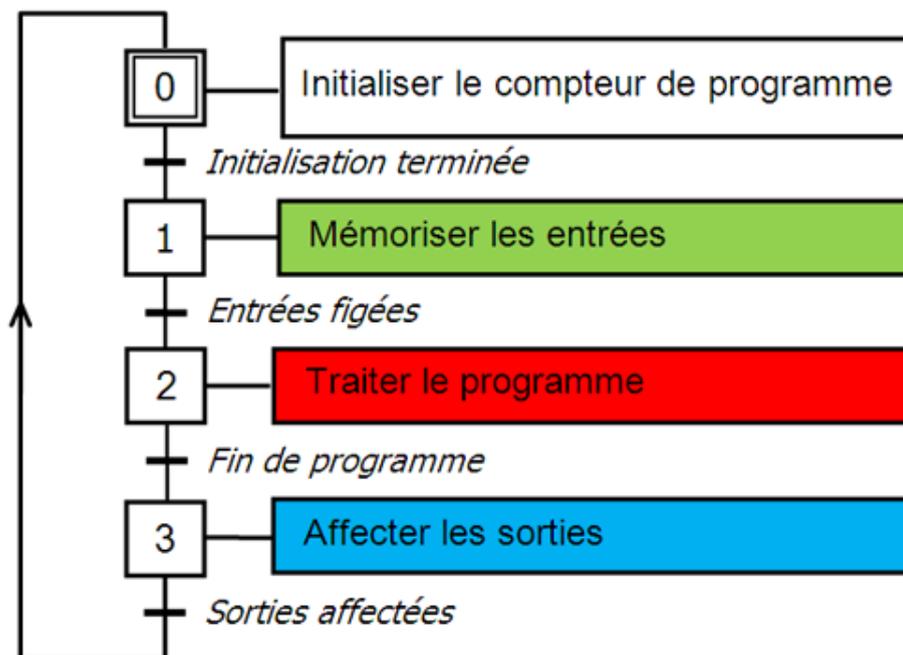


FIGURE I.2: Le Grafcet du fonctionnement cyclique d'un API [5]

Le processeur exécute alors le programme instruction par instruction en rangeant à chaque fois les résultats en mémoire. En fin de cycle les sorties sont affectées d'un

état binaire, par mise en communication avec les mémoires correspondantes. Dans ce cas, le temps de réponse à une variation d'état d'une entrée peut être compris entre un ou deux temps de cycle.

### **I.1.6 Choix d'un automate programmable**

Lorsqu'un ingénieur se lance dans un projet d'automatisation, il faut se rappeler que l'automate n'est qu'un outil utilisé pour effectuer les tâches nécessaires. La mise en œuvre de l'automatisation est donc une tâche primordiale, qui implique souvent de nombreuses méthodologies dans la phase de conception du système. Une fois les spécifications établies, le travail de sélection d'un automate approprié deviendra alors le plus important car cela déterminera la facilité avec laquelle le projet d'automatisation pourra se poursuivre.

Il existe aujourd'hui une vaste gamme de systèmes PLC disponibles, avec de nouveaux ajouts ou remplacements continuellement produits avec des caractéristiques améliorées d'un type ou d'un autre.

Les progrès technologiques sont rapidement adoptés par les fabricants afin d'améliorer la performance et le statut commercial de leurs produits. Cependant, quelle que soit la marque, la majorité des automates programmables de chaque gamme de taille sont très similaires en termes d'équipements de contrôle. Les différences significatives se situent au niveau des méthodes et des langages de programmation, ainsi qu'au niveau des normes de support et de sauvegarde du fabricant. Considérations dans le choix d'un automate approprié avec le vaste choix d'équipement maintenant disponible, l'ingénieur peut généralement obtenir des systèmes similaires auprès de plusieurs fabricants d'équipement d'origine (OEM). Le choix d'un automate programmable est généralement basé sur [5] :

- Nombre d'entrées / sorties : le nombre de cartes peut avoir une incidence sur le nombre de racks dès que le nombre d'entrées / sorties nécessaires devient élevé.

- Type de processeur : la taille mémoire, la vitesse de traitement et les fonctions spéciales offertes par le processeur permettront le choix dans la gamme souvent très étendue.
- Fonctions ou modules spéciaux : certaines cartes (commande d'axe, pesage ...) permettront de "soulager" le processeur et devront offrir les caractéristiques souhaitées (résolution, ...).
- Fonctions de communication : l'automate doit pouvoir communiquer avec les autres systèmes de commande (API, supervision ...) et offrir des possibilités de communication avec des standards normalisés (Profibus ...).

Tous ces critères dépendent les uns des autres, la taille de la mémoire étant directement liée à la quantité d'E/S ainsi qu'à la taille du programme. Plus la taille des E/S et de la mémoire augmente, plus le temps de traitement est long et plus le processeur central doit être puissant et rapide pour que les temps de balayage restent acceptables [5].

### I.1.7 Langages de programmation [2]

Dans l'environnement de programmation soit (Step7 ou TIA Portal), on trouve différents langages de programmation, parmi ces langages :

- **Le SCL (structured control langage)** : langage de programmation de haut niveau, on peut effectuer des opérations et complexes. La syntaxe utilisée est proche celle de PASCAL.
- **Le STL ou language list** : C'est un langage machine, avec des instructions de base, il permet de réduire le temps d'exécution.
- **Le Grafset ou SFC** : c'est un langage graphique permettant de montrer l'évolution séquentielle du programme.
- **Le Ladder** : Langage graphique, permet de programmer facilement l'automate, des blocs prédéfinis sont utilisés comme instructions.
- **Le FBD(function block diagram)** : aussi un langage graphique, dédié pour les opérations de calculs arithmétiques ou logiques.

### I.1.8 Présentation de la gamme SIMATIC S7-1200 de SIEMENS [3]

Destiné à des tâches d'automatisation simples mais d'une précision extrême, l'automate SIMATIC S7-1200 est modulaire et compact, polyvalent et parfaitement adaptée à une grande variété d'applications. Une conception modulaire et flexible, muni d'une interface de communication répondant aux exigences les plus sévères dans l'industrie et une large gamme de fonctions technologiques performantes et intégrées font de cet automate un composant à part entière d'une solution d'automatisation complète.



FIGURE I.3: API SIMATIC S7-1200 de SIEMENS

#### I.1.8.1 La conception modulaire et flexible

La famille des automates SIMATIC S7-1200 a été conçue pour offrir un maximum de souplesse lors de la configuration de machines personnalisées. Vous pouvez ainsi composer votre propre solution d'automatisation en fonction de vos besoins spécifiques, tout en sachant qu'il sera toujours possible d'étendre ultérieurement le système de manière simple et rapide.



FIGURE I.4: API SIMATIC S7-1200 avec ses différents module

### I.1.8.2 Communication industrielle

L'automate SIMATIC S7-1200 assure en tant que contrôleur d'E/S PROFINET la fonctionnalité intégrale de raccordement de stations d'E/S PROFINET. En outre, l'interface PROFINET intégrée garantit une communication optimale avec le système d'ingénierie intégré SIMATIC STEP 7 Basic pour la configuration et la programmation. Elle permet la programmation ainsi que la communication avec les pupitres SIMATIC HMI Basic Panels pour la visualisation, avec des automates supplémentaires pour la communication de CPU à CPU et avec des appareils.

### I.1.8.3 La mémoire de l'API S7-1200 :

L'automate présente jusqu'à 50 Ko de RAM – avec une part ajustable entre le programme utilisateur et les données utilisateur – et jusqu'à 2 Mo de mémoire de chargement intégrée et 2 Ko de mémoire de données rémanente.

### I.1.8.4 Les modules d'entrées/sorties :

Les CPU les plus performantes permettent de raccorder jusqu'à huit modules d'entrées/sorties et de disposer ainsi d'E/S TOR et analogiques supplémentaires.

### I.1.8.5 Les boucles de régulation :

L'automate SIMATIC S7-1200 supporte des boucles de régulation PID pour des applications simples de conduite des processus. Ces boucles de régulation peuvent

être aisément configurées grâce à un objet technologique de type régulateur PID dans le système d'ingénierie SIMATIC STEP 7 Basic. SIMATIC S7-1200 supporte en plus la fonction d'auto-ajustement PID, qui permet de calculer automatiquement des valeurs de réglage optimales pour la composante proportionnelle, intégrale ou différentielle.

## **I.2 Interfaces Homme-Machine IHM**

### **I.2.1 Définition générale**

Une Interface Homme-Machine (IHM) est une interface utilisateur permettant de connecter une personne à une machine, à un système ou à un appareil. En théorie, il est donc possible d'utiliser ce terme pour définir n'importe quel écran permettant à un utilisateur d'interagir avec un appareil. Cependant, il est généralement utilisé pour le contexte d'un processus industriel.

Les IHM peuvent prendre différentes formes. Il peut s'agir d'écrans directement intégrés aux machines, d'écrans d'ordinateur, de tablettes tactiles, et bien plus encore. Quelle que soit la forme, le but de ces interfaces reste toutefois le même : fournir des insights sur les performances et les progrès des machines.

### **I.2.2 Les différentes applications des IHM**

Dans un contexte industriel, les IHM peuvent être utilisées pour plusieurs raisons. Elles peuvent servir à afficher les données de façon visuelle, ou pour suivre les temps de production ou les tendances. Elles peuvent aussi servir à surveiller les indicateurs de performances, ou à surveiller les entrées et sorties des machines. Il s'agit là de quelques exemples, mais les IHM peuvent avoir de nombreux autres cas d'usage. L'opérateur d'une usine peut par exemple utiliser une IHM pour surveiller et contrôler la température d'un réservoir d'eau industriel, ou pour vérifier si les différentes pompes du complexe fonctionnent correctement. Ces interfaces servent donc de passerelles entre les opérateurs et les équipements industriels.

### I.2.3 Présentation de la gamme SIMATIC HMI Comfort Panels [4]

Tous les SIMATIC HMI Comfort Panels offrent en permanence la même fonctionnalité haut de gamme. Dotés d'écrans larges haute résolution de 4" à 22", au choix avec commande tactile ou par touches, ils s'adaptent de manière optimale à chaque application et peuvent être programmées à l'aide de **WINCC Comfort (TIA Portal)** qu'on va détailler dans le chapitre suivant.



FIGURE I.5: SIMATIC HMI TP900 COMFORT [4]

la gamme SIMATIC HMI Comfort Panels offre de nombreuses innovations :

- Afficheurs haute résolution flexibles.
- Nouvelle interface utilisateur graphique innovante.
- Fonctionnalité haut de gamme intégrée.
- Gestion efficace de l'énergie.
- Sécurité maximale des données en cas de panne de courant.
- Adapté aux environnements difficiles.

### I.2.4 Les interfaces intégrées (réseaux industriels)

Les SIMATIC HMI Comfort Panels peuvent être intégrés dans les réseaux PROFINET et PROFIBUS et disposent d'interfaces pour la connexion d'E/S USB. Ces appareils sont aussi équipés d'un commutateur Ethernet à 2 ports et à partir de 15" d'une interface PROFINET Gigabit supplémentaire.

Les câbles standard peuvent être utilisés pour le chargement de projets IHM à partir du PC via PROFINET / Ethernet ou USB, aucun câble spécial n'est nécessaire et les réglages de l'appareil sont effectués lors de la configuration.

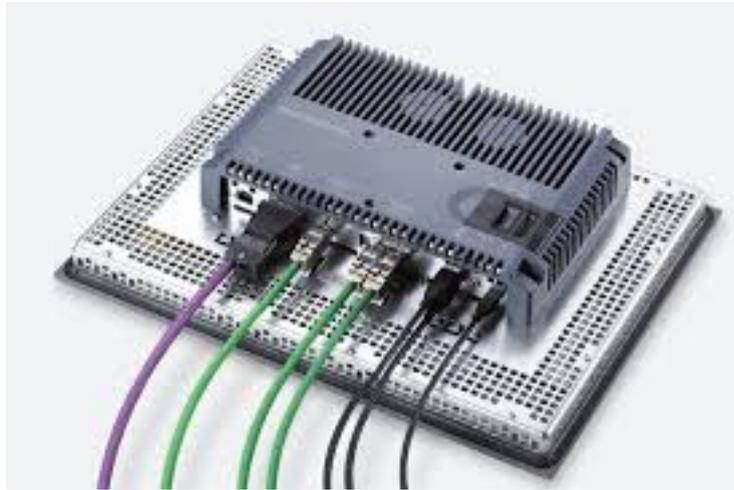


FIGURE I.6: les interfaces de SIMATIC HMI TP900 COMFORT [4]

## I.3 Conclusion

L'automate programmable est adaptable à un maximum d'application, d'un point de vue traitement, composants et langage. Son architecture complexe et modifiable selon la nécessité du projet ainsi que la maîtrise de son software sont considérés comme critères principaux pour mieux choisir son automate.

Dans ce chapitre l'étude de la gamme SIMATIC a orienté notre choix vers l'API et l'IHM qui conviennent le mieux à notre application.

# Chapitre II

## Variateurs de Vitesse Industriels

## Introduction

Ce chapitre traite les connaissances de base. Après une brève description des variateurs de vitesse électroniques et leurs usages, on explique les principes de fonctionnement et les différentes structures et composants de ces derniers. La fin de ce chapitre est consacrée au variateur de fréquences Altivar 12 qu'on a utilisé dans notre projet.

### II.1 Historique

Historiquement, le variateur électronique pour moteur à courant continu a été la première solution offerte. Les progrès de l'électronique de puissance et de la microélectronique ont permis la réalisation des convertisseurs de fréquence fiables et économiques. Les convertisseurs de fréquence modernes permettent l'alimentation des moteurs asynchrones standards avec des performances analogues aux meilleurs variateurs de vitesse à courant continu. Certains constructeurs proposent même des moteurs asynchrones avec des variateurs de vitesse électroniques incorporés dans une boîte à bornes adaptée ; cette solution est proposée pour des ensembles de puissance réduite (quelques kW).

### II.2 Définition [5]

Un variateur de vitesse est un convertisseur d'énergie permettant de moduler l'énergie électrique fournie au moteur. Les variateurs de vitesse sont constitués principalement d'un convertisseur statique et d'une électronique de commande. Les variateurs récents contiennent aussi un étage de correction du facteur de puissance afin de respecter les normes de compatibilité électromagnétique. L'électronique de commande réalise la régulation et l'asservissement de la machine à travers le convertisseur statique de sorte que l'utilisateur puisse commander directement une vitesse. Sa conception dépend essentiellement de la stratégie de commande choisie (commande vectorielle, commande scalaire, etc.).

## II.3 Constitution

Le variateur de vitesse est constitué d'un module de contrôle et de puissance.

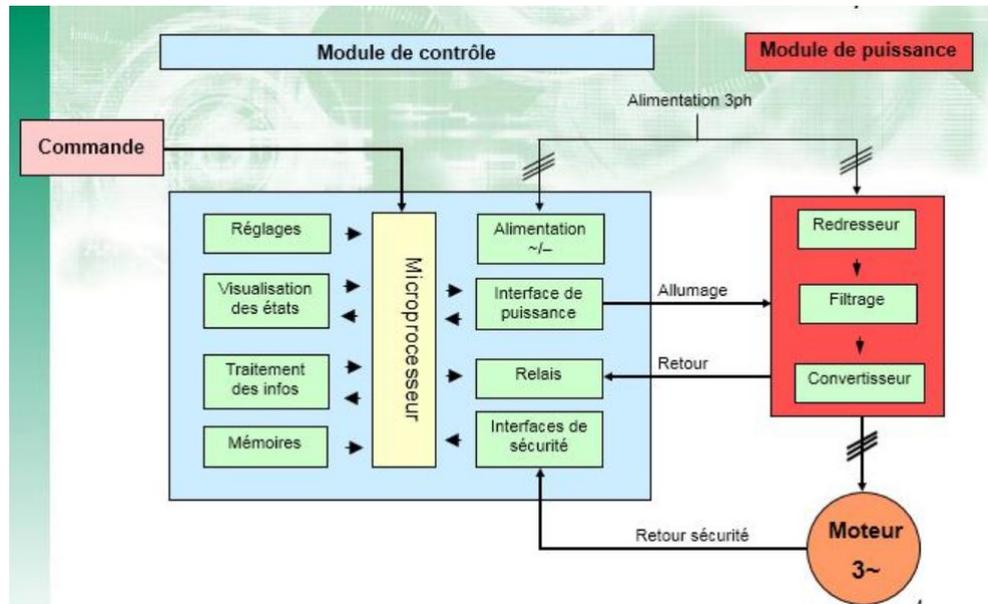


FIGURE II.1: Structure générale d'un variateur de vitesse électronique [7]

### II.3.1 Module de puissance

Le module de puissance du variateur est composé de 3 parties principales :

- **Le redresseur** qui, connecté directement au réseau, génère une tension continue à ondulation résiduelle (le signal n'est pas parfaitement continu). Le redresseur est composé de diodes de redressement.
- **Le circuit** intermédiaire de filtrage composé de condensateurs agissant principalement sur le "lissage" de la tension de sortie du redresseur (améliore la composante continue). Le circuit intermédiaire peut aussi servir de dissipateur d'énergie lorsque le moteur devient générateur.
- **L'onduleur** qui engendre le signal de puissance à tension et/ou fréquence variables, il est composé de transistors bipolaires de puissance (IGBT).

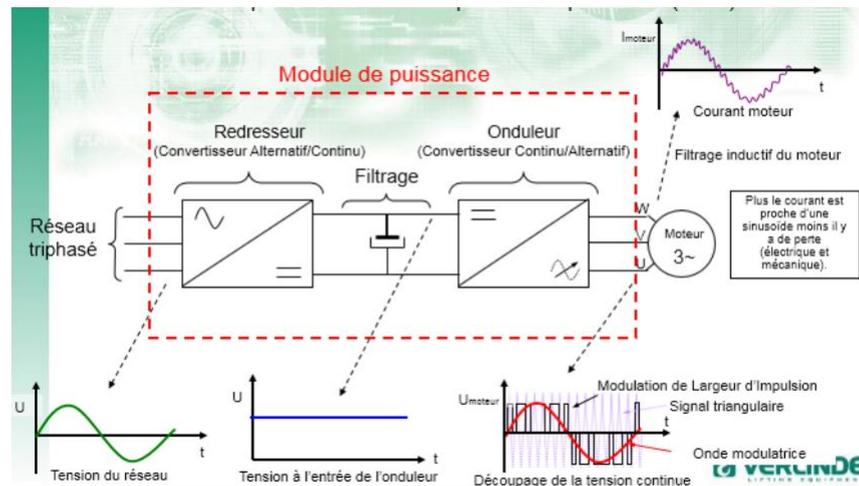


FIGURE II.2: Module de puissance d'un variateur de vitesse [7]

### II.3.2 Module de contrôle

Le module de contrôle est basé sur des circuits de hautes technologies tels que des ASIC (circuits numériques programmables) associés à un microprocesseur permettant un contrôle précis :

- Contrôle en fréquence, en tension, en courant.
- Contrôle en vitesse, en couple, en flux.
- La régulation de vitesse est réalisée par correction du type PID.

## II.4 Quadrants de fonctionnement des variateurs [6]

Deux paramètres définissent le fonctionnement des systèmes donc des moteurs. Ces deux paramètres sont le couple et la vitesse. Le couple dépend de la charge qui peut être entraînée ou entraînant. Le signe de la vitesse dépend du sens de rotation du moteur. 4 quadrants définissent les zones de fonctionnement.

Les variateurs de fréquence peuvent, selon les cas, être construits pour autoriser le fonctionnement dans un, deux ou quatre quadrants. Dans notre cas, seul le quadrant 1 est géré. Les pompes centrifugeuses, perceuses, ventilateurs et les aspirateurs utilisent ce genre de fonctionnement.

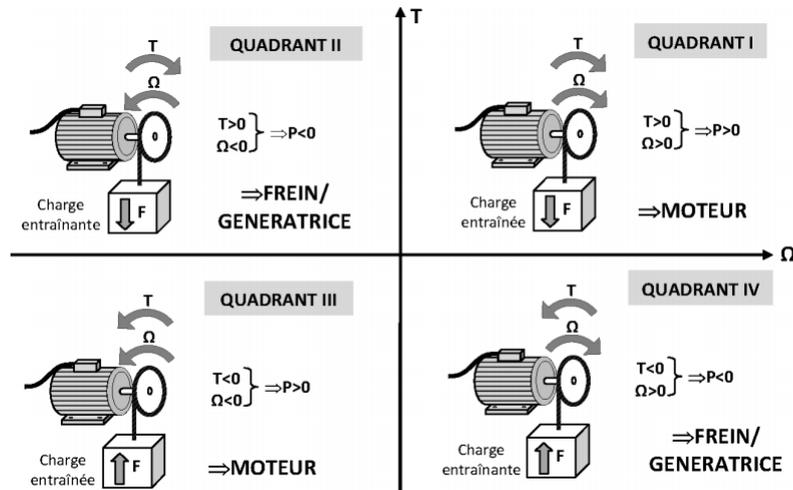


FIGURE II.3: Quadrants de fonctionnement de la machine asynchrone [6]

## II.5 Les principales fonctions des variateurs de vitesse électroniques [7]

Les variateurs de vitesses sont des systèmes qui convertissent les caractéristiques d'une alimentation en fonction d'une consigne donnée. Ils ont plusieurs fonctions parmi lesquelles :

- **Le démarrage** : le moteur passe de la vitesse nulle jusqu'à sa vitesse établie en un temps prédéfini et en évitant les pointes d'intensité.
- **La variation de vitesse** : modification de la fréquence de rotation du moteur par accélération ou décélération en un temps donné.
- **La régulation** : la fréquence de rotation du moteur est maintenue constante quelles que soient les fluctuations de la charge (dans certaines limites).
- **Le freinage** : le moteur passe d'une vitesse établie à une vitesse inférieure (ralentissement) ou à la vitesse nulle (arrêt) avec maintien en position possible.
- **L'inversion du sens de rotation** : permet de faire fonctionner le moteur dans les deux sens de rotation.
- **La récupération d'énergie** : permet lors d'un ralentissement ou d'un freinage des systèmes de transformer l'énergie mécanique en énergie électrique. Dans ce

cas, le moteur fonctionne en génératrice et l'énergie récupérée peut être soit dissipée dans des résistances, soit utilisée pour recharger des batteries ou encore réinjectée dans le réseau.

## II.6 Variateur de vitesse de Schneider Electric ATV12H037M2

### II.6.1 Description du variateur ATV12H037M2 et ses applications

Le ATV12H037M2 de Schneider Electric utilisé dans notre maquette est un Driver de moteur monophasé, asynchrone à vitesse variable avec une puissance de 0.37KW et un filtre CEM intégré. Ce Driver fonctionne avec une tension nominale allant de 200V à 240V et a des dimensions standard 143 mm de hauteur, 72 mm de largeur et 121.2mm de profondeur. Cette machine simple est utilisée avec des pompes et des ventilateurs centrifuges, On la trouve généralement dans l'industrie, l'automatisme et le contrôle de Process ainsi que le contrôle moteur.



FIGURE II.4: Variateur de vitesse Schneider ATV12H037M2 [9]

## **II.6.2 Caractéristiques du variateur de vitesse Schneider ATV12H037M2**

Ce modèle est doté des caractéristiques suivantes :

- Protocole de communication Port Modbus.
- RJ45 pour Modbus sur la face avant et 2 fils RS485 pour interface physique Modbus.
- dissipateur de chaleur (refroidisseur) sans ventilateur intégré.
- Fréquence d'alimentation de 50Hz/60Hz et fréquence de commutation nominale de 4 kHz.
- Poids du produit de 0.7 kg.
- IP20 sans plaque d'obturation sur la partie supérieure.
- Température de l'air ambiant pour une plage de fonctionnement de 40°C à 60°C avec un derating du courant de 2,2 %.
- 3 phase 200 à 230V en sortie ( entre phases )
- Fonctionne en position verticale

## **II.6.3 Schéma de câblage du variateur avec l'alimentation monophasée**

Le variateur de vitesse ATV12H037M2 est alimenté en monophasé avec une tension entre 200 et 240V

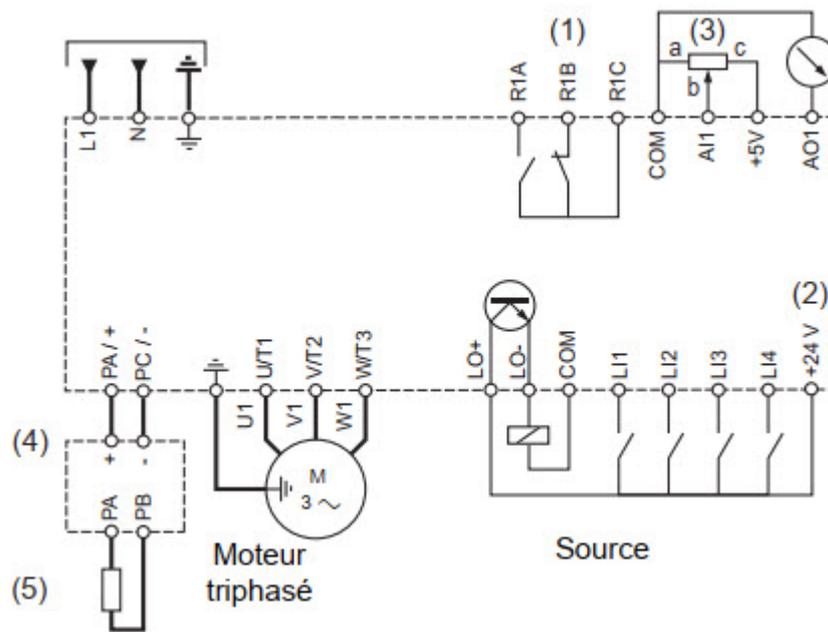


FIGURE II.5: Schéma de câblage du variateur ATV12H037M2

- (1) Contacts du relais R1, pour indiquer l'état actuel. (voir l'annexe C)
- (2) Alimentation interne du variateur qui délivre une tension continue de 24V, on peut l'utiliser comme une source comme on peut ramener une alimentation externe (+30V au maximum) pour les entrées logiques Lix, qui ont plusieurs fonctionnalités :
  - Le sens du moteur.
  - les vitesses présélectionnées.
- (3) 2.2 k $\Omega$  potentiomètre de référence. Il peut être remplacé par un potentiomètre 10 k $\Omega$  (maximum).
- (4) Module peut être utilisé au freinage.
- (5) Une résistance de freinage optionnelle.

## II.7 Programmation manuelle des variateurs

### II.7.1 Commande du variateur par 2 fils ou 3 fils

Schneider nous donne la possibilité de commander le variateur par deux méthodes :

#### II.7.1.1 La commande du variateur par 2 fils

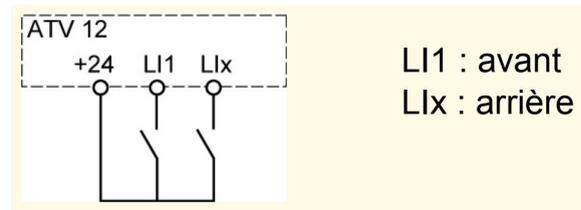


FIGURE II.6: Câblage associé à commande par 2 fils

Où le choix du sens est simple :

- **L'état 1 fermé** : sens avant
- **L'état 2 fermé** : sens arrière.

Si les deux états sont fermés au même temps, le variateur ne fonctionne pas, donc l'utilisateur doit choisir un seul sens à la fois, mais si l'un précède l'autre d'un petit moment, le variateur prend le premier. les deux états sont branchés à une alimentation de 24V, et on peut le commander manuellement ou à l'aide d'un automate programmable. La figure suivante explique le fonctionnement à deux fils :

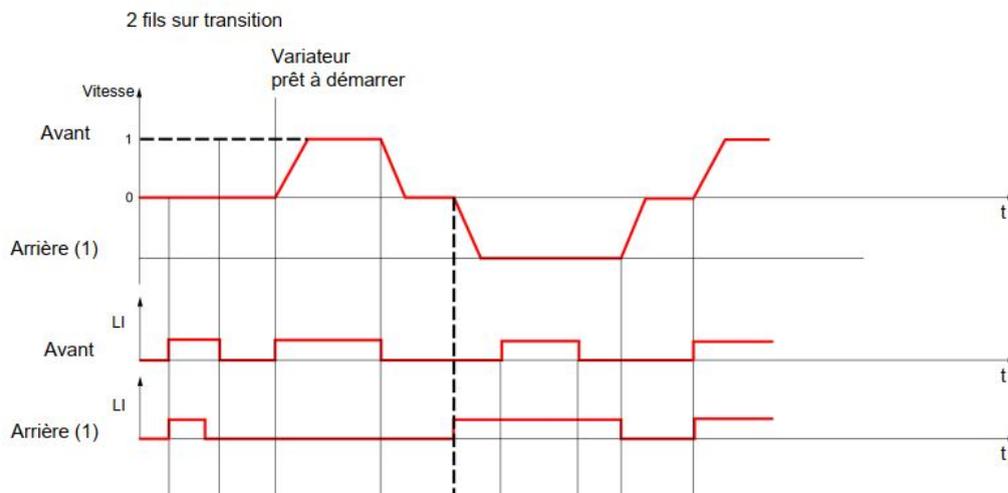


FIGURE II.7: Le fonctionnement à deux fils

### II.7.1.2 Commande du variateur par 3 fils

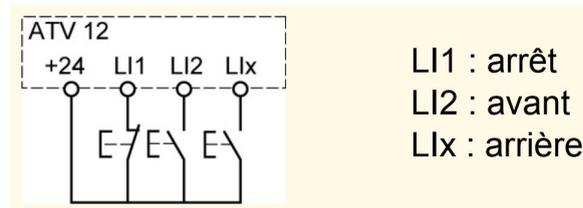


FIGURE II.8: Câblage associé à commande par 3 fils [9]

Dans cette configuration le choix du sens nécessite seulement des impulsions pour qu'un état considéré soit actif :

- **L'état 1** : donne le feu vert au variateur pour qu'il soit prêt.
- **L'état 2** : sens avant.
- **L'état 3** : sens arrière.

Pour faire une marche avant : on doit activer l'état 1 pour que le variateur soit prêt, une fois le variateur est prêt, on peut choisir un sens, dans notre cas, marche avant, donc on doit activer l'état 1 par une impulsion, si on garde l'état 1 actif, même si on active l'état 2, rien ne se passe jusqu'à ce qu'on désactive le premier état. La même chose pour la marche arrière. La figure ci-dessous montre le fonctionnement à 3 fils :

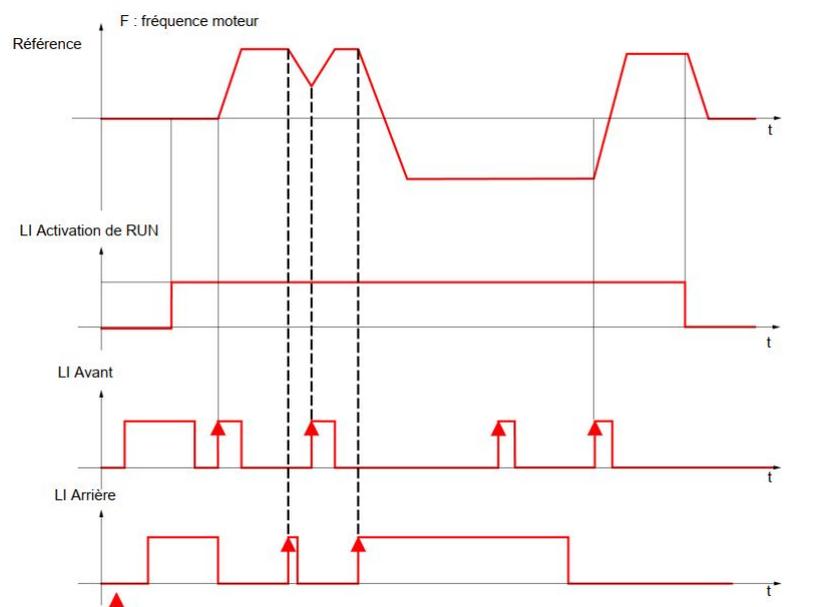


FIGURE II.9: Le fonctionnement à deux fils [9]

La programmation manuelle du variateur pour choisir l'un des modes :

$$\boxed{Conf \Rightarrow Full \Rightarrow I^-O^- \Rightarrow tcc \Rightarrow 2C \text{ ou } 3C} \quad (\text{II.1})$$

## II.7.2 Référence de vitesse

Le variateur suit la vitesse que l'utilisateur introduit via l'entrée analogique AI1 :

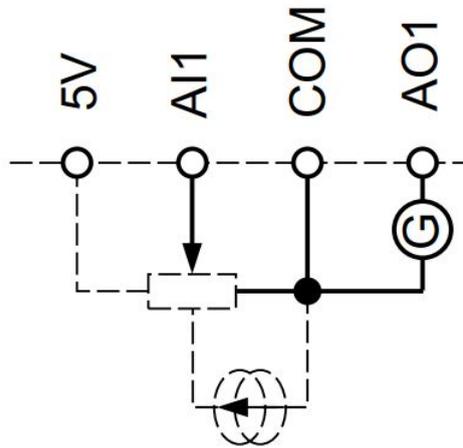


FIGURE II.10: L'entrée analogique du variateur [9]

L'utilisateur peut choisir le type de signal qui lui convient comme référence du variateur, on distingue 3 types :

- Tension : 0-5V (5U)
- Tension : 0-10V (10 U)
- Courant : x-y mA (0A) (**plage de x à y mA**).

Pour choisir l'un de ces types, on suit la configuration suivante :

$$\boxed{Conf \Rightarrow Full \Rightarrow I^-O^- \Rightarrow AII^- \Rightarrow AII t \Rightarrow 5U \text{ ou } 10U \text{ ou } 0A} \quad (\text{II.2})$$

Par exemple si on choisit 0A : courant de plage (x à y), on définit la plage par la configuration suivante :

- La valeur min :

$$\boxed{Conf \Rightarrow Full \Rightarrow I^-O^- \Rightarrow AII^- \Rightarrow CrLI \Rightarrow 0 \text{ ou } 4mA} \quad (II.3)$$

- La valeur max :

$$\boxed{Conf \Rightarrow Full \Rightarrow I^-O^- \Rightarrow AII^- \Rightarrow CrHI \Rightarrow 20mA} \quad (II.4)$$

### II.7.3 Sortie analogique du variateur

Le variateur possède une sortie analogique AO1 qu'on peut utiliser pour récupérer des informations à partir du variateur :

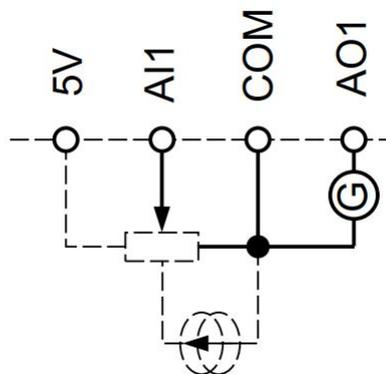


FIGURE II.11: La sortie analogique du variateur AO1 [9]

Ces informations sont listées dans le tableau ci-dessous :

Valeur possible / Fonction	
<b>nD</b>	Non
<b>DCr</b>	Courant moteur
<b>DFr</b>	Fréquence de sortie
<b>DrP</b>	Sortie rampe
<b>DPs</b>	Référence PID
<b>DPF</b>	Retour PID
<b>DPE</b>	Erreur PID
<b>DPr</b>	Puissance de sortie
<b>tHr</b>	État thermique du moteur
<b>tHd</b>	État thermique du variateur

FIGURE II.12: Fonctionnement de la sortie analogique [9]

Pour qu'on choisit l'une de ces fonctions, on suit la configuration suivante :

$$\boxed{Conf \Rightarrow Full \Rightarrow I^-O^- \Rightarrow AOI^- \Rightarrow AOI \Rightarrow La fonction dsire} \quad (II.5)$$

De même, la sortie possède 3 types différents, on choisit celle qui nous convient :

- Tension : 0-10V
- Courant : 0-20mA.
- Courant : 4-20mA.

On peut choisir l'un de ces types en suivant la configuration suivante :

$$\boxed{Conf \Rightarrow Full \Rightarrow I^-O^- \Rightarrow AOI^- \Rightarrow AOIt \Rightarrow 10U \text{ ou } 0A \text{ ou } 4A} \quad (II.6)$$

### II.7.4 Configuration "Perte d'une phase"

Le variateur est dédié pour contrôler les moteurs triphasés, mais en cas où la seule pompe disponible est monophasée, le variateur ne fonctionne pas, car il voit qu'une phase est perdue, ce qui nécessite le Shutdown du variateur pour des raisons de sécurité (électrocution d'une personne). Le variateur nous donne la possibilité d'ignorer cette protection, mais à condition de prendre la responsabilité en cas d'accidents.

Pour le faire, on suit la configuration matérielle suivante :

$$\boxed{Conf \Rightarrow Full \Rightarrow Flt^- \Rightarrow tHt^- \Rightarrow OpL \Rightarrow no} \quad (II.7)$$

### II.7.5 Le bruit du moteur

Le bruit broduit par le moteur pour les petites fréquences est grand, et parfois ça pose un problème pour l'ouvrier ou n'importe quelle personne prêt du moteur, ce qui nécessite un réglage dans le variateur qui permet de diminuer considérablement le bruit.

Pour appliquer ce réglage, on procède de la manière suivante :

$$\boxed{Conf \Rightarrow arrowFull \Rightarrow drC- \Rightarrow nrd \Rightarrow nO \text{ ou } YES} \quad (\text{II.8})$$

## II.7.6 Type de commande utilisé par le variateur [8]

Il existe plusieurs méthodes pour l'asservissement de vitesse des machines asynchrones, parmi ces méthodes, la plus utilisée par les variateur de fréquences :

**Le contrôle scalaire de tension :** plus précisément la loi :  
le rapport tension/fréquence reste constant.

On a l'expression du couple qui s'écrit sous la forme suivante :

$$\boxed{C_m = k\left(\frac{v}{f}\right)^2}$$

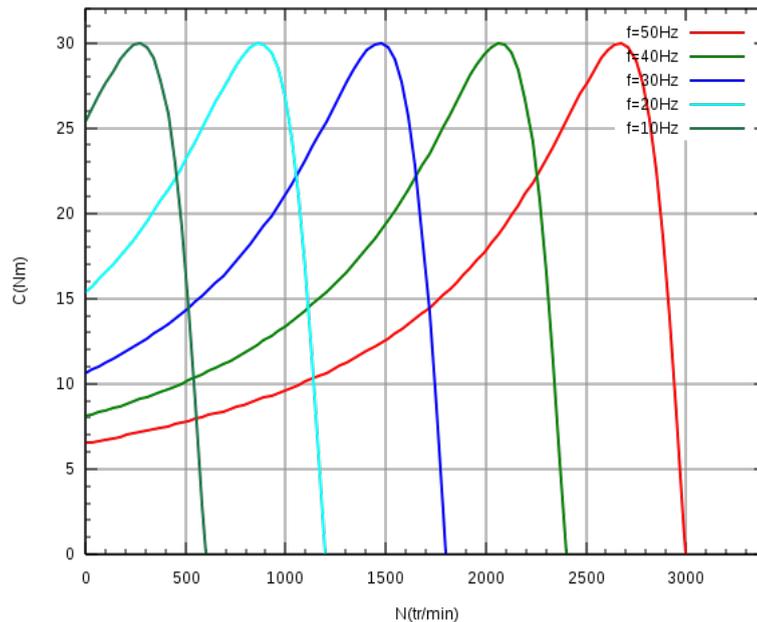


FIGURE II.13: La courbe du couple en fonction de la fréquence [5]

La commande U/f se base sur la mesure des grandeurs scalaires (valeurs d'amplitude en tension et en fréquence). C'est le système de commande de base des variateurs de fréquence. Afin de garder un flux constant dans le moteur et donc aussi une variation de vitesse à couple constant la tension et la fréquence varient proportionnellement jusqu'à la fréquence nominale du moteur (50 Hz). Lorsque la tension nominale est

atteinte, la tension ne sachant plus augmenter, il est toujours possible d'augmenter la fréquence, dans ce cas la variation se fait à puissance constante, le couple diminue avec la vitesse. Ce mode de fonctionnement est intéressant pour des charges à couple constant tels que les ascenseurs. En effet, le couple moteur "colle" mieux au profil du couple résistant, ce qui signifie que les consommations qui en découlent sont moindres. Donc pour garder un couple max quelque soit la vitesse du moteur, on doit garder le rapport tension fréquence constant :

La configuration de la commande f/u :

$$\boxed{Conf \Rightarrow Full \Rightarrow drC \Rightarrow Ctt \Rightarrow Std} \quad (II.9)$$

## II.8 Conclusion

Ce chapitre comporte une étude générale de la technologie des variateurs de vitesse et leur principe de fonctionnement. Il comporte aussi une présentation de la gamme des variateurs de fréquence ATV12H037M2, ses caractéristiques, fonctionnalités ainsi que ses modes d'utilisation.

## Chapitre III

# Modélisation de la station de pompage et simulation avec Matlab

### III.1 Introduction

Une étude théorique du système est nécessaire avant l'implémentation du régulateur, ceci nous donne des informations sur le comportement du système et nous permet de le commander et de réaliser de bonnes performances sans risquer de l'endommager. Dans ce chapitre on va aborder la modélisation de la station de pompe pour pouvoir synthétiser un régulateur adéquat, ensuite on procède à la simulation à l'aide du logiciel de simulation Matlab Simulink.

### III.2 Modélisation du système

La figure ci-dessous montre le schéma de la station de pompe qu'on doit modéliser, elle est composée de deux réservoirs l'un en dessus de l'autre. La pompe permet de pomper l'eau du réservoir A le refoulant dans le réservoir B, ces deux derniers sont liés par une vanne qui assure le retour du réservoir B au réservoir A. On s'intéresse à la commande du niveau d'eau dans le réservoir du dessus B.

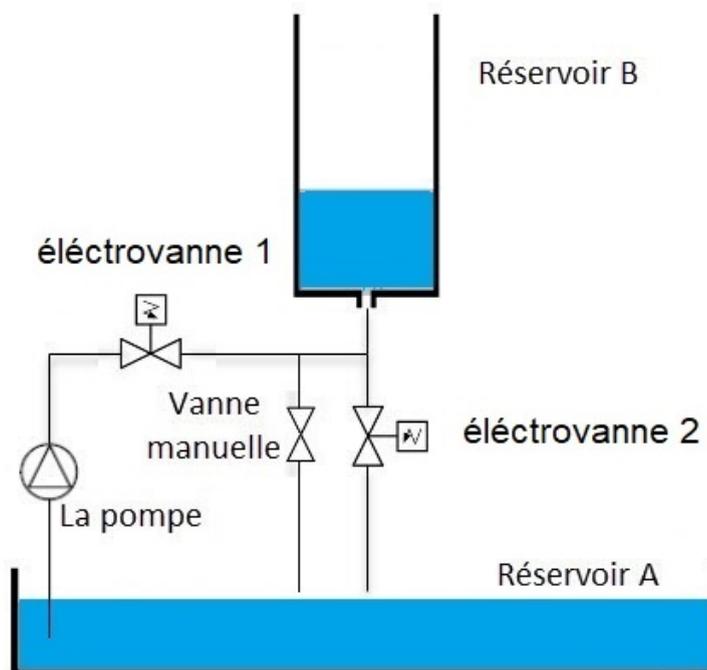


FIGURE III.1: Schéma du système modélisé

On définit le vecteur d'état :

$$\mathbf{X} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} h_1 \\ h_2 \end{pmatrix} \quad (\text{III.1})$$

Ce système dynamique est régi par les équations différentielles suivantes [10] :

$$\frac{dh_1}{dt} = (1 - \gamma) \frac{s}{S} v - \gamma \frac{s}{S} \sqrt{2\rho g h_1} \quad (\text{III.2})$$

$$\frac{dh_2}{dt} = \gamma \frac{s}{S} v + \gamma \frac{s}{S} \sqrt{2\rho g h_1} - \frac{s}{S} v \quad (\text{III.3})$$

avec :

Symbole	La signification
$h_1$	hauteur du réservoir B.
$h_2$	hauteur du réservoir A.
$s$	$s=4.91 \text{ cm}^2$ : la surface de l'ouverture du réservoir B.
$S$	$S=400 \text{ cm}^2$ : la surface de base du réservoir B.
$g$	$g=980 \text{ cm/s}^2$ : constante de la gravité.
$\rho$	$\rho = 1 \text{ g/cm}^3$ : masse volumique de l'eau.
$v$	$v$ : la tension de réglage ( $0 \Rightarrow 10v$ )
$\gamma$	$\gamma$ : représente le taux d'ouverture de la vanne manuelle.

TABLE III.1: Les paramètres et leurs valeurs numériques

On linéarise autour du point d'équilibre, pour trouver les matrices de la représentation d'état :

**La représentation d'état** d'un système est sous la forme :

$$\text{systeme : } \begin{cases} F = f(x, u) = Ax + Bu \\ y = h(x) \end{cases} \quad (\text{III.4})$$

On s'intéresse dans notre étude seulement à la dynamique du réservoir B qui est représentée par la variable  $h_1$ .

**Linéarisation du système :** Puisque le système est non-linéaire, on le linéariser autour du point d'équilibre :  $F(x, u) = 0$ .

on trouve que tous les points de fonctionnement sont des points d'équilibre, donc on choisit le point suivant :

$$\mathbf{P}_0 = \begin{pmatrix} h_1^0 \\ v^0 \end{pmatrix} = \begin{pmatrix} 20 \\ 7.5 \end{pmatrix}$$

$$A = \left. \frac{dF(x, u)}{dh_1} \right|_{P_0} = -\alpha = -2.42\gamma \quad (\text{III.5})$$

$$B = \left. \frac{dF(x, u)}{du} \right|_{P_0} = -\beta = 0.092\gamma \quad (\text{III.6})$$

$$C = \left. \frac{dy}{dx} \right|_{P_0} = 1 \quad (\text{III.7})$$

**La fonction de transfert :** Pour avoir la fonction de transfert depuis la représentation d'état, on applique cette formule :

$$G(s) = C[sI - A]^{-1}B \quad (\text{III.8})$$

On trouve la fonction de transfert suivante :

$$G(s) = \frac{0.092\gamma}{s + 2.42\gamma} \quad (\text{III.9})$$

On a choisi une valeur pour " $\gamma$ ", on a pris  $\gamma = 0.5$ , l'ouverture de la vanne manuelle, donc toute autre valeur de  $\gamma$  est considérée comme une perturbation en sortie, afin de mieux tester notre régulateur.

$$G(s)_{\gamma=0.5} = \frac{0.046}{s + 1.21} \quad (\text{III.10})$$

### III.3 Synthèse du régulateur

Dans la majorité des systèmes industriels, les régulateurs P.I sont les plus utilisés car ils sont plus adéquats à la nature assez rapide des systèmes, ainsi qu'à la présence des bruits de mesure. La fonction de transfert est de l'ordre 1, ce qui représente une dynamique simple.

Voici le diagramme de bode de la fonction de en BO du système :

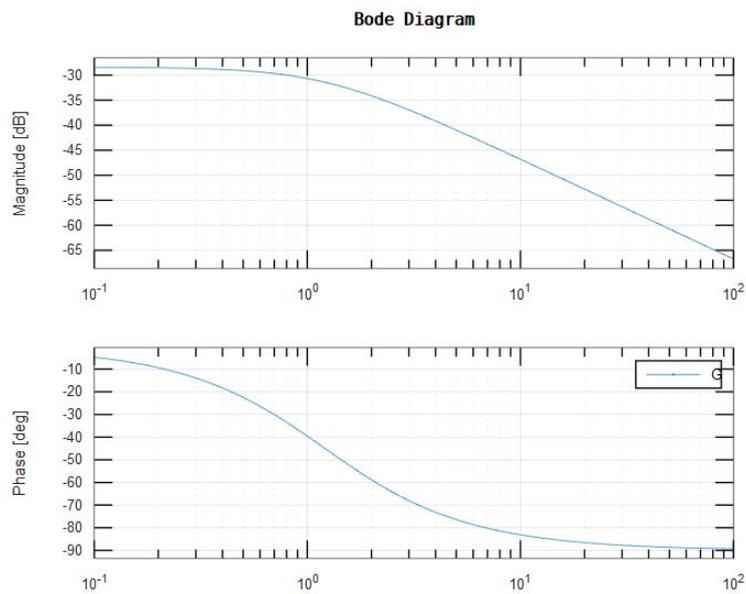


FIGURE III.2: Le diagramme de Bode de la fonction de transfert du système

**La boucle ouverte :** Le système en boucle ouverte est stable, car il possède un pôle négatif :  $P = -1.21$

La réponse à un échelon du système :

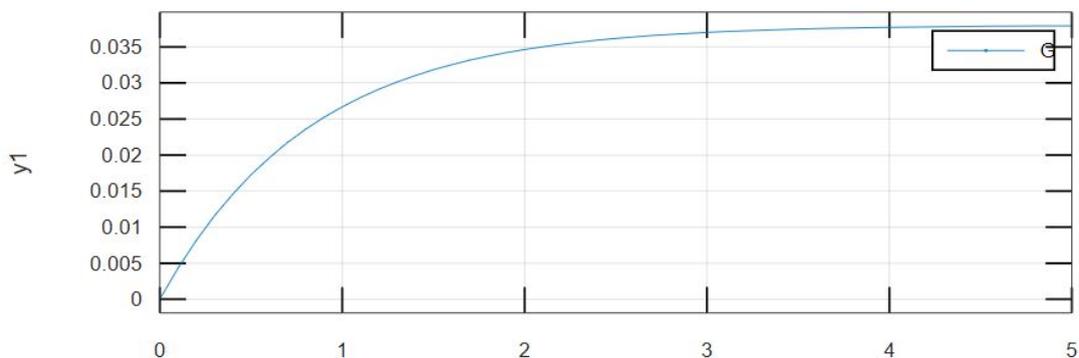


FIGURE III.3: La réponse à un échelon

On constate que l'erreur statique est grande, et cela est due au fait que la vanne manuelle est à moitié ouverte (grande perturbation en sortie) dans l'absence du feedback.

**La boucle fermée :** Pour améliorer les performances de notre système, on doit introduire un feedback qui nous aide à atteindre une référence donnée dans un temps minimal.

La fonction de transfert du système en boucle fermée est la suivante :

$$G_{BF}(s) = \frac{G_{BO}(s)}{1 + G_{BO}(s)} = \frac{0.046}{0.83s + 1.046} \quad (\text{III.11})$$

Afin de réaliser les performances souhaitées (erreur statique nulle, pas de dépassement), on utilise un régulateur PI de la forme :

$$R(s) = K_c \frac{1 + sT_i}{T_i s} \quad (\text{III.12})$$

Le choix des paramètres se fait comme suit : On compense la constante de temps, parce qu'elle est assez grande :  $T_i = 1.21$

### III.3.1 Effet du gain sur la réponse du système

Afin d'observer l'effet du gain sur la réponse du système en boucle fermée, on essaye plusieurs valeurs du gain  $K_c$  et on trace la vitesse de rotation de la pompe ainsi que le niveau du réservoir dans le même graphe pour chaque valeur du gain.

**Interprétation des résultats :** La pompe utilisée a un débit max limité :  $35L/min$  par conséquent on distingue 3 cas de figures :

**Cas 1 :** Si on choisit un gain élevé  $K_c = 25$ , la réponse est assez rapide mais la commande dépasse le seuil maximal, donc la pompe fonctionne en régime saturé du début jusqu'à la fin (voir figure III.4).

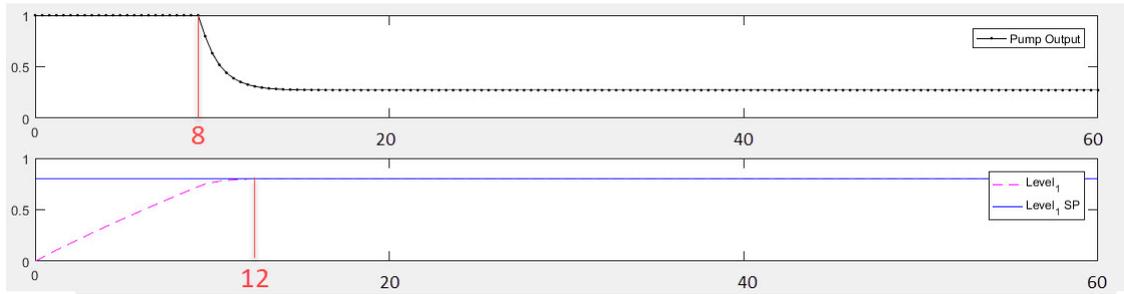


FIGURE III.4: La réponse à un échelon

**Cas 2 :** Pour un gain  $K_c$  assez petit  $K_c=8$ , la pompe n'atteint pas la saturation mais la réponse du système est très lente (voir figure III.5).

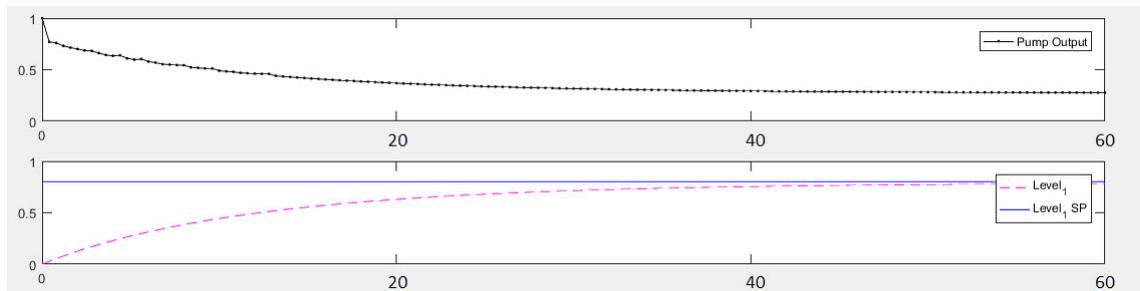


FIGURE III.5: La réponse à un échelon

**Cas 3 :** On trouve un compromis entre le temps de réponse et le fonctionnement en régime non saturé, la valeur du gain qui correspond est  $K_c= 15$  (voir figure III.6).

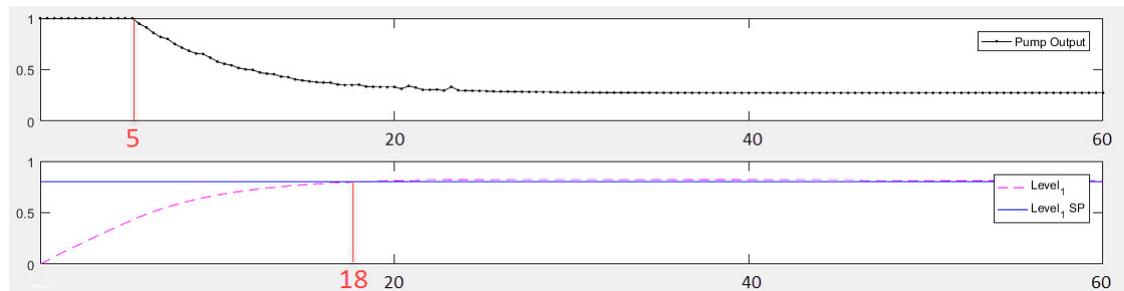


FIGURE III.6: La réponse à un échelon

Le régulateur s'écrit sous la forme :

$$R(s) = 15 \frac{1 + 1.21s}{1.21s} \quad (\text{III.13})$$

### III.4 Simulation du système en boucle fermée avec Matlab Simulink

Dans le but de tester le régulateur synthétisé dans la partie précédente, et avant de passer à l'implémentation, on simule le système avec Matlab Simulink.

La figure ci-dessous représente le schéma bloc du système réalisé avec l'outil de simulation Simulink.

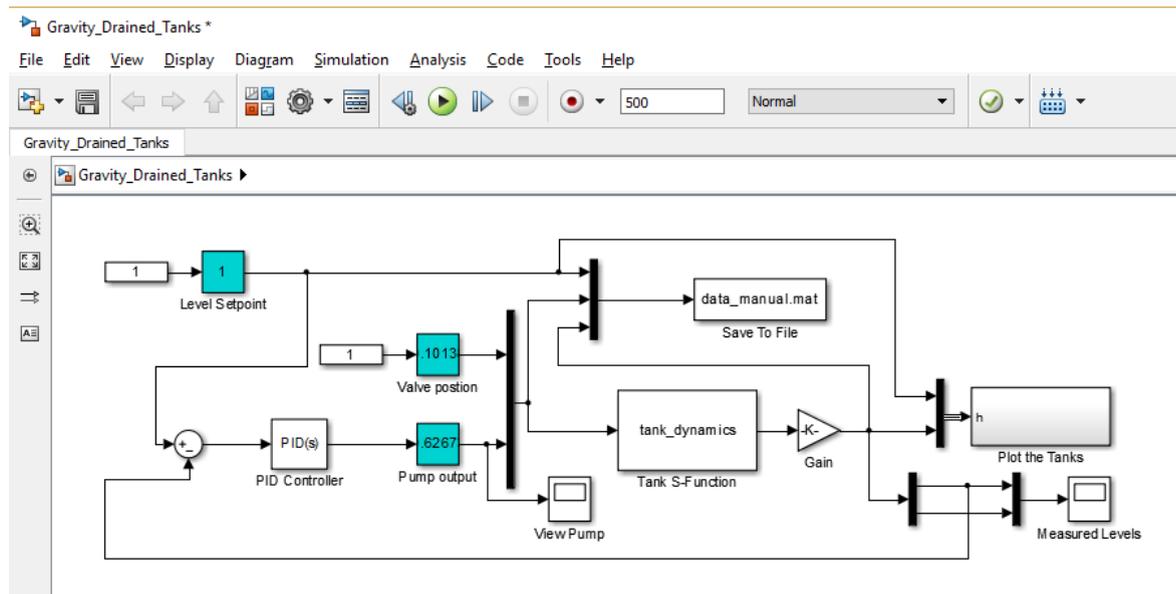


FIGURE III.7: Schéma bloc du système en boucle fermée réalisé sur Simulink

Des scripts Matlab s'exécutent en arrière plan comportants les équations qui régissent le système.

Avant de lancer la simulation, on choisit un temps de simulation assez long pour observer la réponse du système, on introduit aussi 3 paramètres : la référence de niveau, la valeur maximale de la commande qui correspond à la saturation de la pompe ainsi que l'ouverture de la vanne manuelle qui représente la perturbation.

Une fois la simulation lancée on observe la réponse du système en boucle fermée de deux manières différentes : D'abord on peut visualiser les courbes tracées en cliquant sur "Scoupe" :

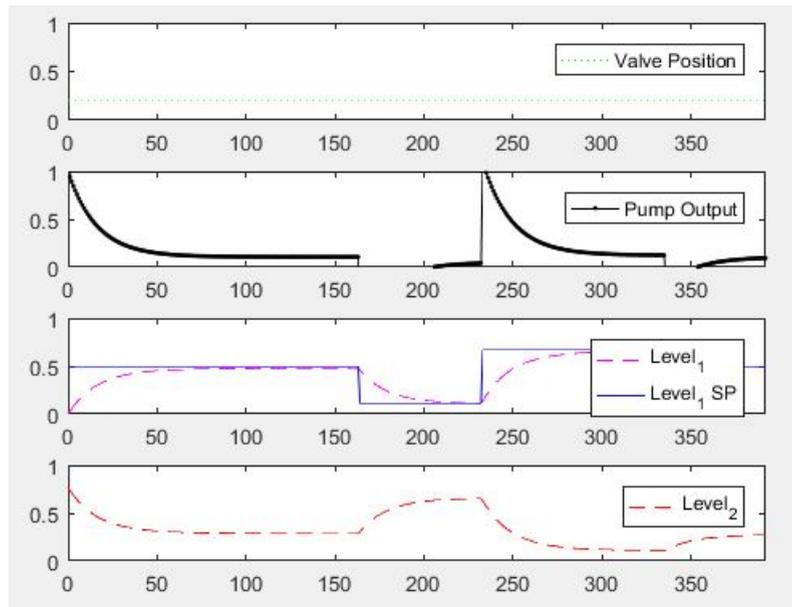


FIGURE III.8: Réponse du système en boucle fermée

Il est possible d'observer la réponse grâce l'animation réalisée qui montre le niveau dans les deux réservoir en bleu en train de varier en fonction du temps ainsi que la référence souhaité en rouge (Figure III.9).

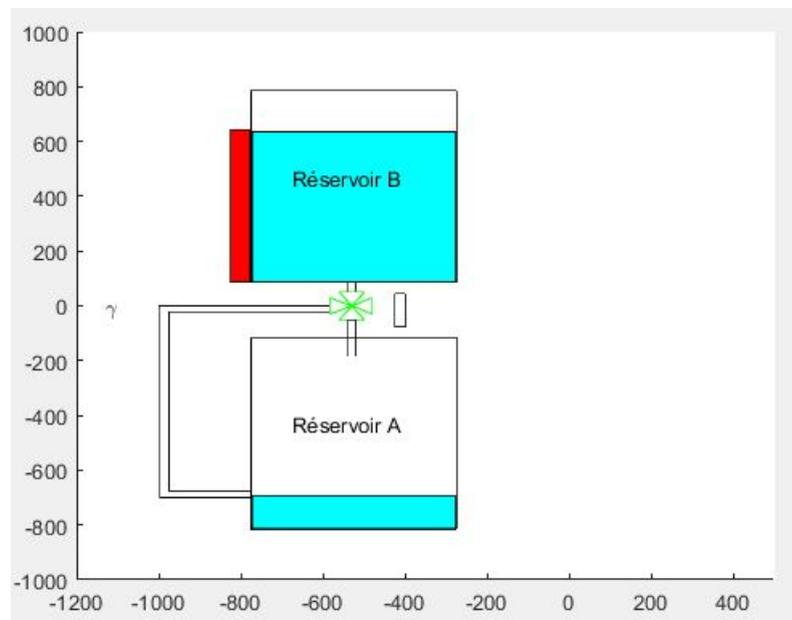


FIGURE III.9: La simulation sur simulink

## **III.5 Conclusion**

Les résultats obtenus répondent bien à l'objectif de l'asservissement, le régulateur ainsi choisi approche la consigne désirée dans un temps convenable pour de telles installations.

Dans le chapitre suivant, on va implémenter le régulateur calculé dans ce chapitre.

## Chapitre IV

# Réalisation et gestion d'un prototype de station de pompage

## IV.1 Introduction

Les stations de pompages hydrauliques sont primordiales dans les réseaux d'alimentation en eau potable, Elles fonctionnent grâce à un environnement constitué de pompes, débitmètres, électrovannes, capteurs de niveau, variateur de vitesse...etc. Ces accessoires permettent de collecter des informations sur le fonctionnement, et aident à mieux gérer cette station.

Dans ce chapitre, nous allons réaliser le prototype d'une station de pompage, ainsi que sa gestion effectuée à base d'un automate programmable SIMATIC S7-1200 de SIEMENS. Les étapes de réalisation de notre prototype s'articulent sur la :

- Détermination des éléments de l'application pour finaliser le cahier de charge.
- Détermination du type des éléments auxiliaire de l'installation tel que le débitmètre, capteur de niveau, les électrovannes ainsi que la tuyauterie.
- Écriture et implémentation des programme de l'API ainsi que celui l'IHM.
- Installation et assemblage de tous les éléments, fixation, plomberie et câblage de l'armoire et étalonnage des capteurs.
- Optimisation du fonctionnement et mise en service de l'installation.

## IV.2 Définition du cahier des charges

Le cahier des charges élaboré a pour but de commander le niveau d'eau dans le réservoir B, avec une marge d'erreur réglable. On a opté pour une commande PID dont les paramètres sont modifiables, afin de pouvoir observer les différentes réponses du système. Le cahier des charges a été défini comme suit :

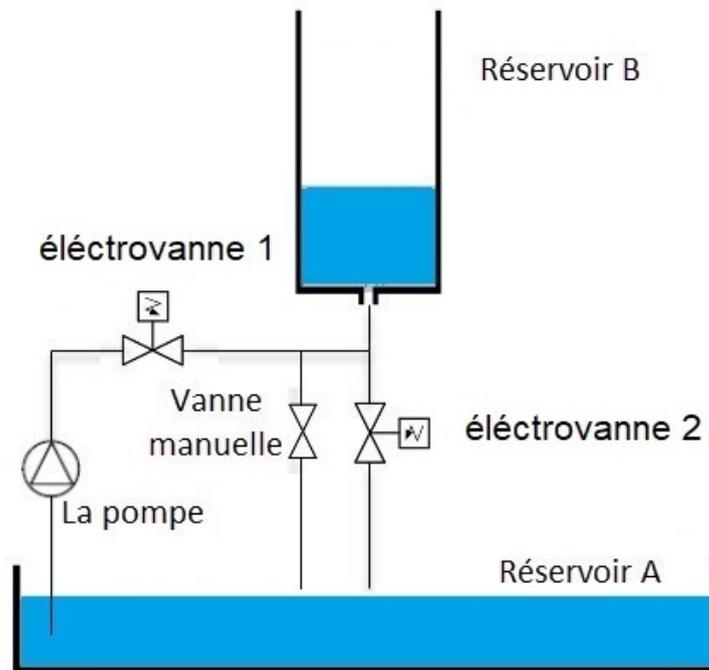


FIGURE IV.1: Schéma du système modélisé

La station étant à l'arrêt (état initiale), elle peut être commandée en 2 modes :

#### IV.2.1 Mode automatique (commande en boucle fermée)

En saisissant une référence de niveau d'eau sur l'écran IHM et en appuyant sur le bouton « marche » avec le bouton d'arrêt d'urgence désactivé, trois cas de figures se présentent :

- Si le niveau d'eau dans le réservoir B est inférieur à la consigne désirée, le moteur démarrera pour accomplir **la phase de remplissage** du réservoir (état 2) avec la première électrovanne EV1 ouverte et la deuxième EV2 fermée, jusqu'à ce que le niveau atteigne la référence désirée, après le système se met à l'arrêt (état 1). Si on constate qu'il n'y a pas de débit pendant cette phase, la pompe s'arrête pour éviter la marche à sec et aussi pour remédier à d'éventuels problèmes inhérents aux matériels.
- Si initialement le niveau dans le second réservoir était supérieur à la référence, on ouvre l'électrovanne 2 pour accomplir **la phase de vidange** (état 3) jusqu'à

ce que le niveau soit égale à la consigne, après le système se met à l'arrêt (état 1).

- Si dès le début le niveau d'eau dans le réservoir B est égale à la consigne ou bien dans la marge tolérée, le système reste à **l'arrêt (état 1)**.

Le voyant rouge V1 est allumé quand le système est à l'arrêt (état 1), et quand ce dernier se met à clignoter ça veut dire que le bouton d'arrêt d'urgence est appuyé, tandis que les deux voyants verts V2 et V3 indiquent respectivement l'activation de la phase de remplissage et la phase de vidange.

### IV.2.2 Mode manuel (commande en boucle ouverte)

Dans ce mode, on peut commander les deux électrovannes TOR ainsi que la pompe manuellement à l'aide de l'IHM..

Pour basculer vers ce mode, on va sur la vue "Mode manuel" sur l'IHM et on appuie sur le bouton "Switch", le fond d'écran gris devient vert, et là on pourra activer la pompe et faire varier le débit tant que le niveau d'eau maximal de sécurité n'est pas atteint, comme on peut ouvrir ou fermer les électrovannes.

Ce cahier des charges complet est représenté par le grafcet simplifié suivant :

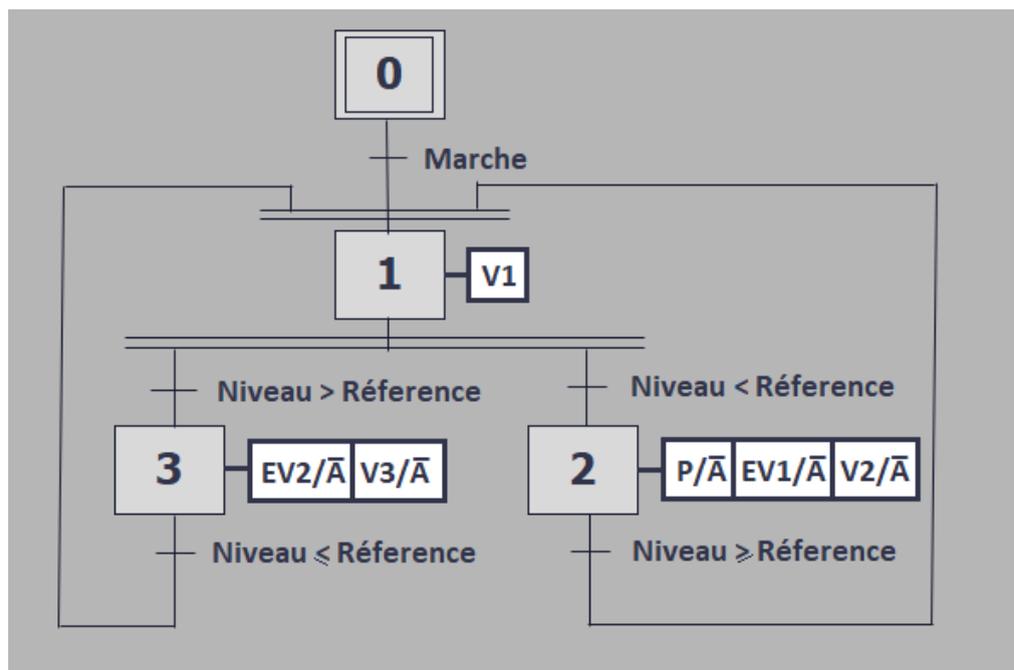


FIGURE IV.2: Grafcet du projet

## IV.3 Présentation du prototype de la station de pompage

### IV.3.1 Description des éléments de l'installation

Le prototype de notre installation a été élaboré suivant les phases d'un projet. Ces phases sont la conception, le développement et la construction. La conception de ce prototype a été réalisée sur le principe de fonctionnement des installations de pompage existantes, et son aspect final a été l'aboutissement de plusieurs idées afin de se rapprocher le plus du fonctionnement d'une véritable station de pompage.

La Figure IV.3 donne une vue générale du prototype réalisé :



FIGURE IV.3: Disposition des éléments dans le prototype de station de pompage

L'ensemble des éléments et les caractéristiques du prototype réalisé sont présentés comme suit :

Le numéro	La signification
1	Le prototype repose sur un <b>support métallique</b> 0.7m sur 0.7m et d'une hauteur de 1.50m avec deux parties ; une partie supérieure là où l'armoire électrique et le réservoir B sont fixés. une deuxième partie inférieure (base) sur laquelle on a posé la pompe centrifuge et le grand réservoir A.
2	<b>deux réservoirs</b> en verre transparent, avec des capacités de 40 litres et 20 litres respectivement.
3	<b>Une pompe centrifuge</b> XKm, couplé à un moteur asynchrone monophasé d'une puissance de 0.37 KW qui peut fournir un débit maximal de 35 L/min.
4	Un <b>débitmètre</b> .
5	Un <b>capteur de niveau</b> ultrasonique cylindrique SICK UM18 portée 250 mm avec sortie analogique 4-20mA alimenté en 24V.
6	<b>Deux électrovannes</b> TOR alimentées en 24V DC.
7	La <b>vanne manuelle</b> en plastique est utilisée pour vidanger le 2nd réservoir et au même temps simuler des perturbations externes de niveau.
8	<b>Armoire électrique</b> métallique (600 x 400 x 230mm).
9	<b>Interface Homme-Machine (IHMI)</b> SIMATIC TP900 Comfort 9" de SIEMENS 6AV2124-0JC01-0AX0.

TABLE IV.1: les éléments dans le prototype de station de pompage

La figure qui suit illustre la disposition des éléments dans l'armoire électrique :

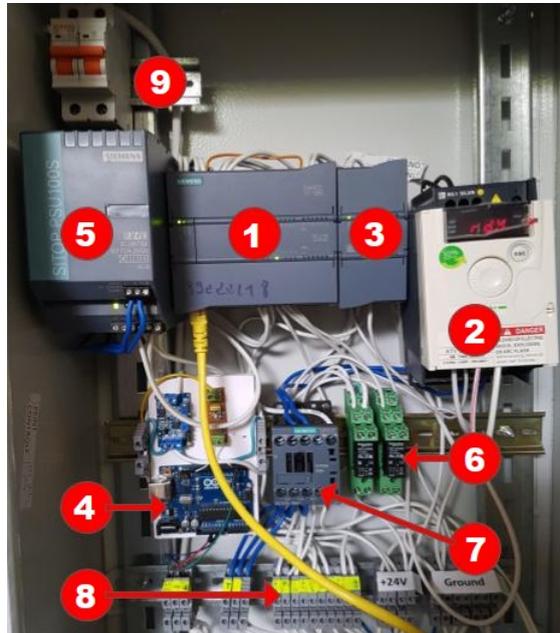


FIGURE IV.4: Figure Disposition des éléments dans l'armoire électrique

Le numéro	La signification
1	<b>Automate programmable industriel (API) SIMATIC S7-1200</b> de SIEMENS 6ES7214-1AG40-0XB0.
2	<b>Variateur de vitesse</b> ALTIVAR 12 de Schneider Electric ATV12H037M2.
3	<b>Module SM1232 sortie analogique</b> SIEMENS 6ES72324HD320XB0.
4	<b>Carte Arduino Uno</b> pour traiter le signal de sortie du débitmètre.
5	<b>Alimentation</b> à découpage SITOP PSU100S, SIEMENS, 24V cc, 5A.
6	<b>2 Relais</b> 24V DC, Schneider Electric RSB2A080BDS.
7	<b>Contacteur de puissance</b> , AC-3 : 12 A, 5,5 kW / 400 V 1 NO, 24 V CC 3 pôles, Taille S00 borne à vis.
8	<b>Borniers de raccordement</b> .
9	<b>Disjoncteur monophasé</b> Legrand 16A.

TABLE IV.2: Disposition des éléments dans l'armoire électrique

Sur la face de l'armoire électrique on trouve en dessous de l'IHM les éléments suivants :

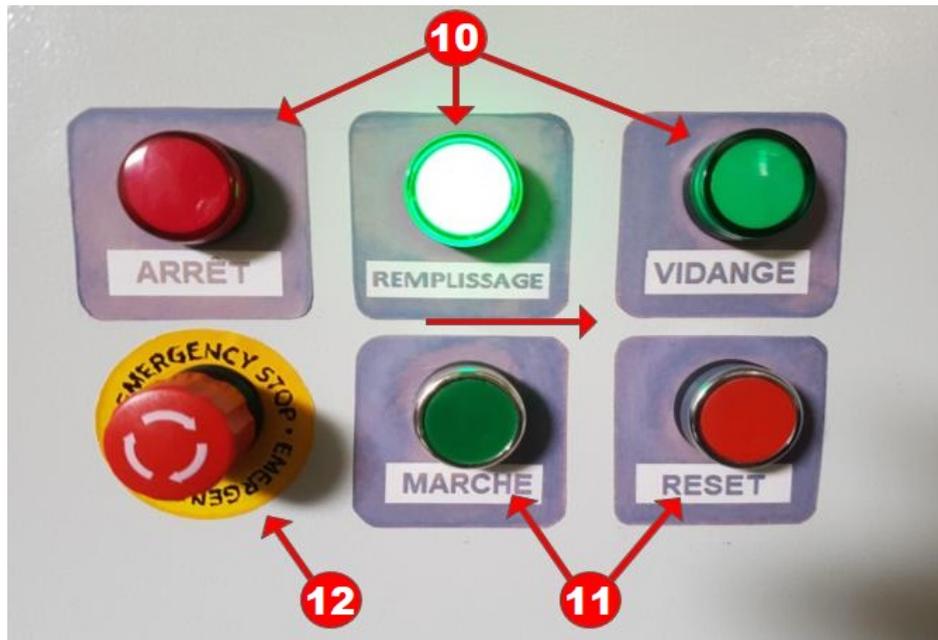


FIGURE IV.5: Les voyants sur l'armoire

Le numéro	La signification
10	3 <b>Voyants à led</b> Ø22.5 mm Rouge, Vert, 24VAC/DC.
11	2 <b>Boutons poussoirs</b> de commande.
12	<b>Bouton d'arrêt d'urgence</b> normalement fermé.

TABLE IV.3: suite du tableau "Disposition des éléments dans l'armoire électrique"

## IV.4 Gestion de la station à l'aide de l'IHM

L'interface homme-machine SIMATIC Comfort TP900 utilisée dans notre projet est attachée dans la face de l'armoire électrique, elle permet de visualiser les états du processus et ses grandeurs dans notre application, ainsi que les différentes courbes, tel que la mesure du débit, du niveau et la commande. Elle permet aussi de sélectionner le mode de fonctionnement de la station et aussi de changer les références et les paramètres du régulateur pour pouvoir observer les réponses via les courbes tracés.

#### IV.4.1 Vue d'accueil

C'est la première qui s'affiche, elle montre l'intitulé du projet, une photo réelle de la station ainsi qu'une petite description de cette dernière.



FIGURE IV.6: vue d'accueil : intitulé du projet

En appuyant sur le bouton "Commencer" en bas à droite on passe à la vue suivante.



FIGURE IV.7: Vue d'accueil : Vue de la station

En appuyant sur le bouton en haut à droite, le menu s'affiche et sur ce dernier on peut sélectionner la vue qu'on veut activer.

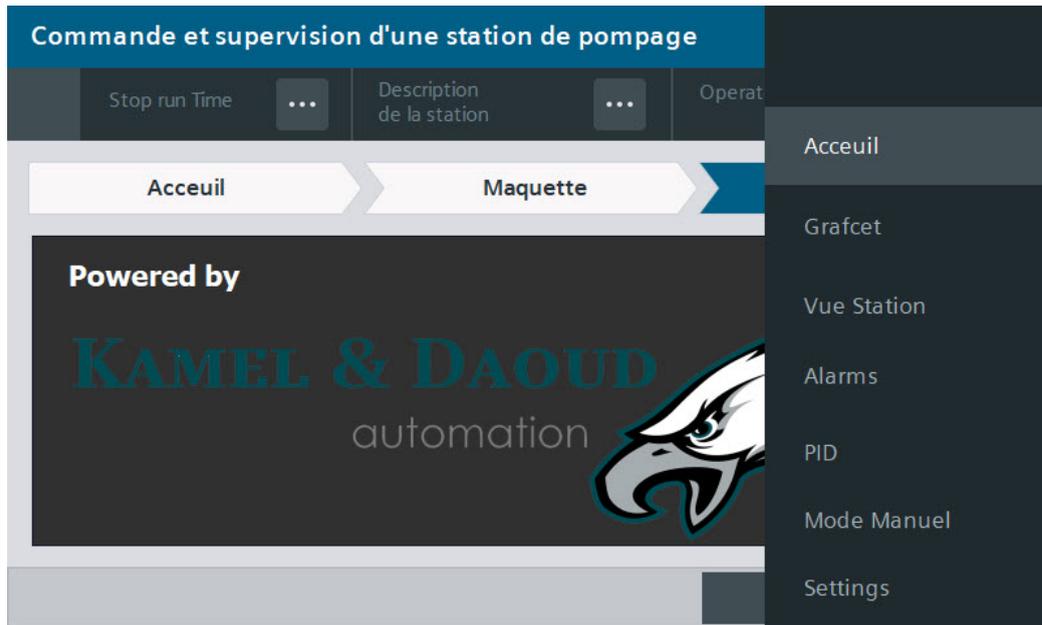


FIGURE IV.8: Vue d'accueil : Menu

## IV.4.2 Vue de la station

Cette deuxième vue permet de commander la station en boucle fermée (mode automatique) et d'observer la variation du débit et du niveau des deux réservoirs en temps réel, la figure ci-dessous montre la vue de la station.

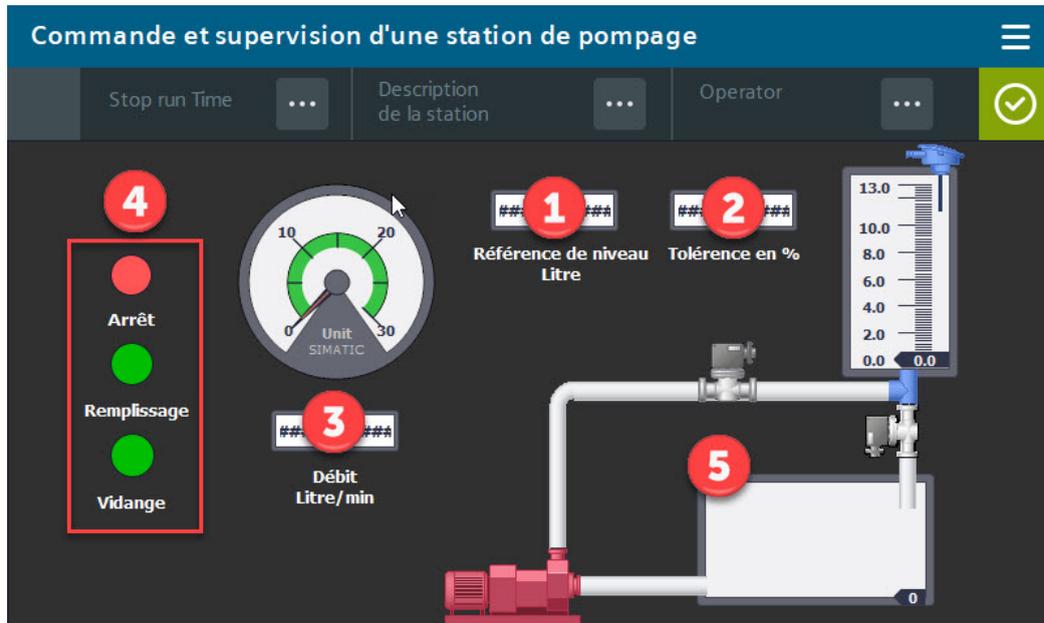


FIGURE IV.9: Vue de la station

Le numéro	La signification
1	Introduire la référence de niveau en litre.
2	Introduire la tolérance en% par rapport à la consigne
3	Affichage de la valeur du débit dans le circuit de remplissage.
4	Les voyants qui indiquent dans quel phase est le processus.
5	Une vue générale de la station qui affiche le niveau d'eau dans les deux réservoirs en temps réel ainsi l'état de la pompe.

TABLE IV.4: Les éléments de "Vue de la station"

### IV.4.3 Grafcet

La vue Grafcet qu'on a programmé permet de savoir parmi tous les états, lequel est actif car il s'affiche en vert avec les actionneurs associés à cet état. La figure IV.10 montre la vue en question :

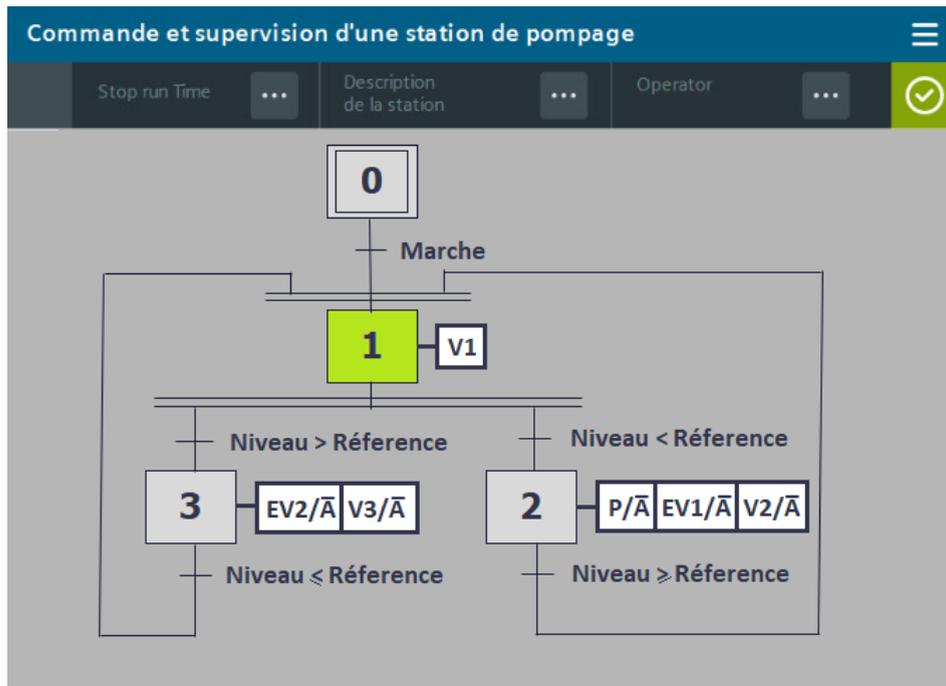


FIGURE IV.10: Vue Grafcet

Ce Grafcet non linéaire est composé de 4 étapes :

Le numéro	La signification
<b>état 0 :</b>	c'est l'état initiale dans lequel on peut faire la régulation en mode manuel.
<b>état 1 :</b>	dans cette étape le système se met à l'arrêt après la régulation.
<b>état 2 :</b>	correspond à la phase de remplissage.
<b>état 3 :</b>	correspond à la phase de vidange.

TABLE IV.5: Les éléments de "Vue de la station"

### IV.4.4 Boucle de régulation PID

Cette vue illustre la boucle de réglage du niveau et indique les différentes grandeurs calculées et introduites via l'IHM ainsi que les mesures réelle dans la station (niveau

et débit).

Une deuxième fonctionnalité très intéressante dans cette vue permet de changer les trois paramètres du régulateur et de visualiser les courbes réelles de la réponse du système, la référence ainsi que la commande calculée en temps réel.

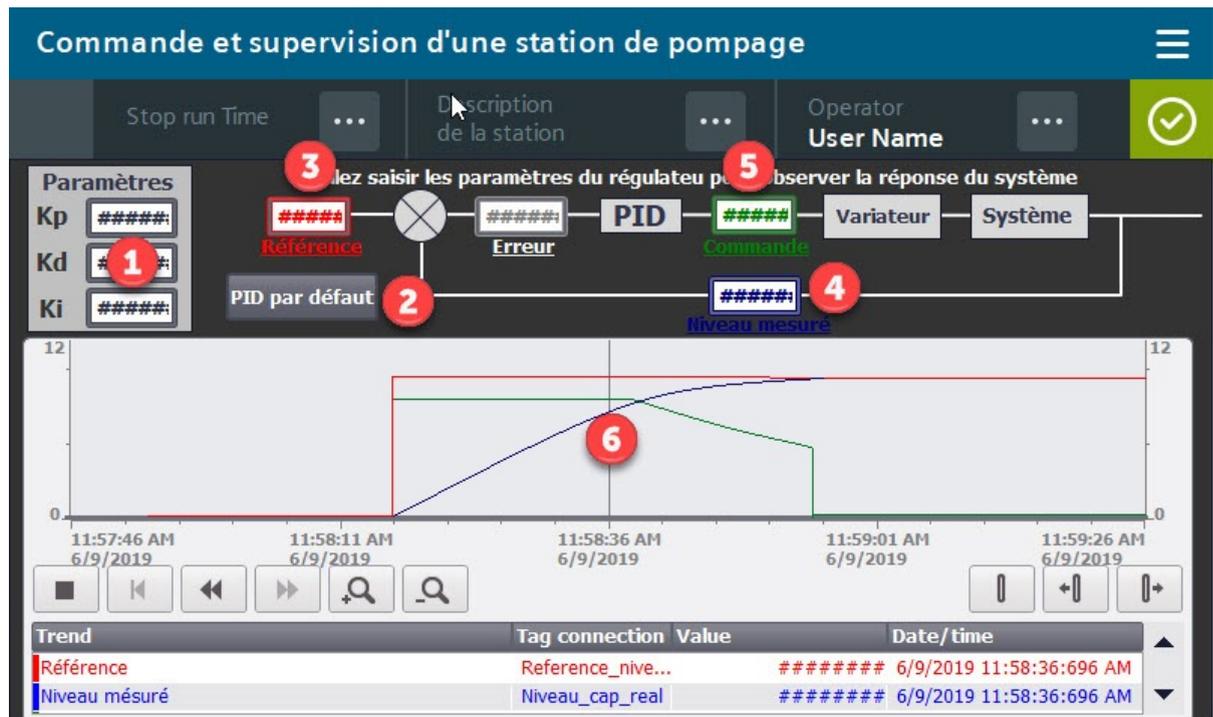


FIGURE IV.11: Vue PID

Le numéro	La signification
1	Les 3 champs qui correspondent à l'introduction des paramètres du PID.
2	en appuyant sur "PID par défaut" on initialise les paramètres du PID aux valeurs calculées lors du chapitre précédent.
3	Introduire la référence de niveau en litre.
4	Le niveau d'eau mesuré par le capteur.
5	Commande calculée par le régulateur qui est de type tension 0-10V.
6	Les courbes tracées en temps réel : en vert c'est la commande, en rouge c'est la référence désirée et en bleu c'est niveau mesuré par le capteur.

TABLE IV.6: Les éléments de "Vue PID"

### IV.4.5 Fonctionnement en mode manuel

Comme expliqué précédemment dans ce chapitre, cette station de pompage peut fonctionner en deux mode, et cette vue dans l'IHM permet de gérer la station en mode manuel, et ceci en commandant la pompe avec variation de débit ainsi que l'électrovanne de vidange. Il faut d'abord appuyer sur le bouton "Mode switch" pour basculer vers ce mode où le fond d'écran devient vert, sur cette même vue on peut visualiser la station comme dans la vue précédente.

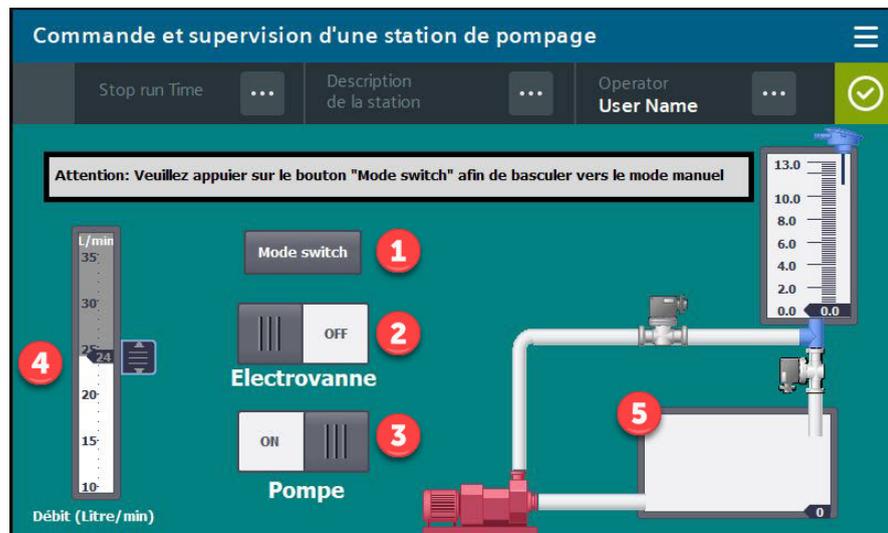


FIGURE IV.12: Vue mode manuel

Le numéro	La signification
1	Appuyez sur ce bouton pour basculer vers le mode manuel.
2	Allumage de l'électrovanne TOR.
3	Allumage de la pompe.
4	Variation du débit en sortie de la pompe entre 10 et 35 L/min.
5	Schéma de la station de pompage.

TABLE IV.7: Les éléments de "Vue mode manuel"

## IV.4.6 Alarmes et archivage

Toutes les vues précédentes sont conçues pour la commande et la supervision de la station de pompage, tandis que celle-ci est faite pour le diagnostic des défauts et le déclenchements des alarmes, on peut archive les 500 dernières alarmes.

**Archivage de variable sur modification :** Les données de la station (Niveau, états des actionneurs, alarmes) sont archivées dans la mémoire de l'automate, il suffit d'insérer une clé dans le port USB de l'IHM pour les récupérer sous la forme d'un fichier .CSV (Excel) qui est créé automatiquement.

Les valeurs du processus sont uniquement archivées en cas de modification. Pour la configuration, utiliser le mode de déclenchement "Sur modification".

On peut trouver davantage de modes de déclenchement d'archivage comme :

- Archivage cyclique.
- Archivage sur demande

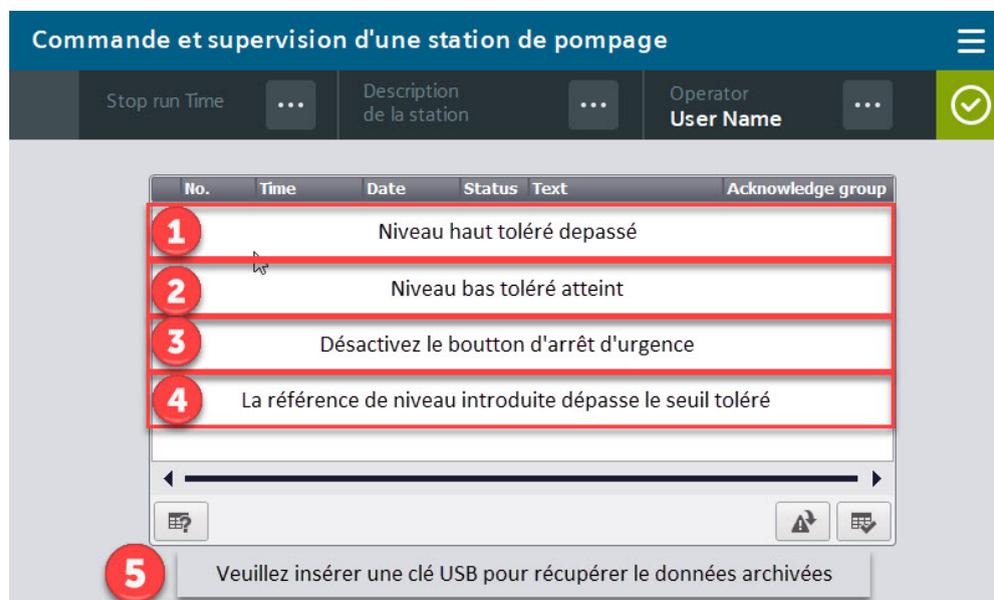


FIGURE IV.13: Vue alarmes et archivage

Le numéro	La signification
1	<b>Alarme 1</b> : déclenchée quand le niveau dans le réservoir B atteint le seuil maximal.
2	<b>Alarme 2</b> : déclenchée quand le niveau dans le réservoir B atteint le seuil minimal.
3	<b>Alarme 3</b> : déclenchée quand le bouton d'arrêt d'urgence est appuyé.
4	<b>Alarme 4</b> : déclenchée quand la référence introduite n'est pas dans la plage permise
5	<b>5</b> : le message qui indique l'envoi des données archivées vers la clé USB.

TABLE IV.8: Les éléments de "Vue mode manuel"

La figure IV.14 montre un archive des alarmes avec le temps de déclenchement et la date.

	A	B	C	D	E	N	O	P	Q
1	Time_ms	MsgProc	StateAfter	MsgClass	MsgNum	TimeString	MsgText		PLC
2	4,36E+10	7	1	33	6	18.06.2019 13:34:32	le bouton d'arrêt d'urgence est activé		HMI_Connection_5
3	4,36E+10	7	0	33	6	18.06.2019 13:34:34	le bouton d'arrêt d'urgence est activé		HMI_Connection_5
4	4,36E+10	7	1	33	6	18.06.2019 13:34:35	le bouton d'arrêt d'urgence est activé		HMI_Connection_5
5	4,36E+10	7	0	33	6	18.06.2019 13:34:36	le bouton d'arrêt d'urgence est activé		HMI_Connection_5
6	4,36E+10	7	1	33	4	18.06.2019 13:34:37	La référence de niveau introduite dépasse le seuil toléré		HMI_Connection_5
7	4,36E+10	7	1	33	6	18.06.2019 13:34:37	le bouton d'arrêt d'urgence est activé		HMI_Connection_5
8	4,36E+10	7	0	33	4	18.06.2019 13:34:37	La référence de niveau introduite dépasse le seuil toléré		HMI_Connection_5
9	4,36E+10	7	0	33	6	18.06.2019 13:34:38	le bouton d'arrêt d'urgence est activé		HMI_Connection_5
10	4,36E+10	7	1	33	6	18.06.2019 13:34:39	le bouton d'arrêt d'urgence est activé		HMI_Connection_5
11	4,36E+10	7	0	33	6	18.06.2019 13:34:41	le bouton d'arrêt d'urgence est activé		HMI_Connection_5

FIGURE IV.14: Archivage des alarmes

	A	B	C	D	E
40	Niveau_cap_real	18.06.2019 13:34:07	3,6	1	4,36E+10
41	Debit_eau_real	18.06.2019 13:34:52	7	1	4,36E+10
42	Debit_eau_real	18.06.2019 13:34:53	18	1	4,36E+10
43	Debit_eau_real	18.06.2019 13:34:54	29	1	4,36E+10
44	Debit_eau_real	18.06.2019 13:34:55	34	1	4,36E+10
45	Debit_eau_real	18.06.2019 13:34:57	33	1	4,36E+10
46	Niveau_cap_real	18.06.2019 13:35:01	3,7	1	4,36E+10
47	Niveau_cap_real	18.06.2019 13:35:02	3,9	1	4,36E+10
48	Niveau_cap_real	18.06.2019 13:35:03	4,2	1	4,36E+10
49	Debit_eau_real	18.06.2019 13:35:04	34	1	4,36E+10
50	Niveau_cap_real	18.06.2019 13:35:04	4,4	1	4,36E+10
51	Debit_eau_real	18.06.2019 13:35:05	33	1	4,36E+10
52	Niveau_cap_real	18.06.2019 13:35:05	4,7	1	4,36E+10
53	Niveau_cap_real	18.06.2019 13:35:06	4,9	1	4,36E+10
54	Niveau_cap_real	18.06.2019 13:35:07	5,2	1	4,36E+10
55	Niveau_cap_real	18.06.2019 13:35:08	5,4	1	4,36E+10
56	Niveau_cap_real	18.06.2019 13:35:09	5,7	1	4,36E+10
57	Niveau_cap_real	18.06.2019 13:35:10	5,9	1	4,36E+10
58	Niveau_cap_real	18.06.2019 13:35:11	6,2	1	4,36E+10
59	Niveau_cap_real	18.06.2019 13:35:12	6,5	1	4,36E+10
60	Niveau_cap_real	18.06.2019 13:35:13	6,7	1	4,36E+10
61	Niveau_cap_real	18.06.2019 13:35:14	7	1	4,36E+10
62	Niveau_cap_real	18.06.2019 13:35:15	7,2	1	4,36E+10

FIGURE IV.15: Archivage des mesures (débit, niveau de l'eau)

La figure IV.15 montre un exemple d'un fichier de mesures récupéré de l'IHM via une clé USB, prêt pour faire projet d'identification par exemple.

## IV.5 Les interfaces d'Entrées/Sorties et mesure des grandeurs

Les entrées/sorties TOR (Tout ou Rien) assurent l'intégration directe de l'automate dans son environnement industriel en réalisant la liaison entre le processeur et le processus. Elles ont toutes, de base, une double fonction :

- fonction d'interface pour la réception et la mise en forme de signaux provenant de l'extérieur (capteurs, boutons poussoirs, etc.) et pour l'émission de signaux vers l'extérieur (commande de pré-actionneurs, de voyants de signalisation, etc.).

La conception de ces interfaces avec un isolement galvanique ou un découplage opto-électronique assure la protection de l'automate contre les signaux parasites.

- fonction de communication pour l'échange des signaux avec l'unité centrale par l'intermédiaire du bus d'entrées/sorties. Dans cette partie on va parler des interfaces E/S utilisées, la figure IV.16 présente le schéma du câblage de toute la station.

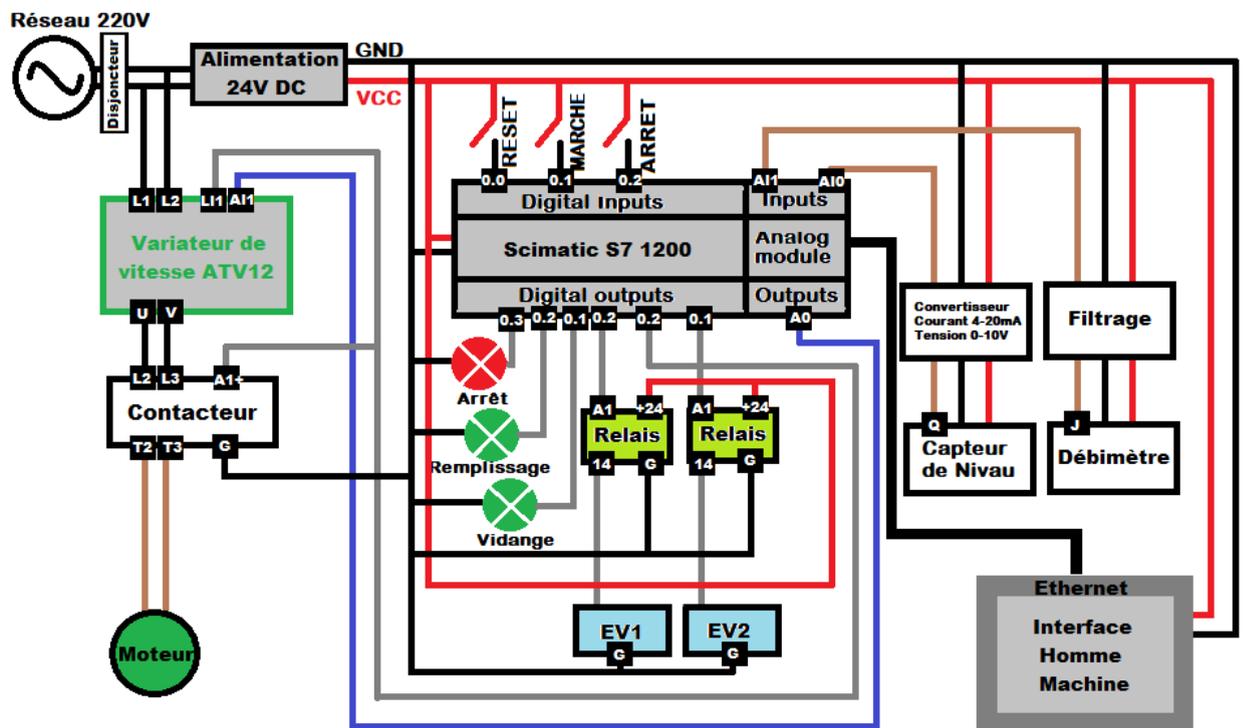


FIGURE IV.16: Schéma général de câblage

### IV.5.1 Interface d'entrée pour le débitmètre

Afin de mesurer le débit de l'écoulement de l'eau dans les conduites du circuit de remplissage, on a utilisé débitmètre adéquat qui possède un capteur à effet Hall sur lequel est attaché un aimant. Le capteur à Effet Hall qui équipe également ce débitmètre permet de compter combien de tour fait le moulinet, permettant ainsi mesurer combien de liquide est passé dans ce capteur, donc le signal de sortie est un train d'impulsions, un signal qui ne peut être exploité directement par l'automate, ce qui nécessite un conditionnement en amont.

Le traitement de ce signal se fait en deux phases : d'abord il passe par un microcontrôleur Arduino Uno via une entrée digitale, ce dernier est programmé à compter le nombre d'impulsions/seconde et le convertir en un signal PWM, donc la largeur des impulsions représente le débit mesuré par le capteur. Le programme Arduino est attaché dans l'annexe B.

La deuxième phase de conditionnement consiste à convertir le signal PWM en sortie de l'Arduino dont l'amplitude est de 5V, et dont la fréquence est de 490 Hz en un signal analogique 0-5V exploitable directement par l'automate. Donc on a utilisé un filtre passe-bas pour obtenir la composante continue du signal PWM. Les filtres passe-bas sont utilisés pour réduire l'amplitude des composantes de fréquences supérieures à la fréquence de coupure.

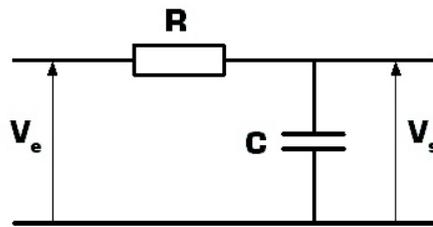


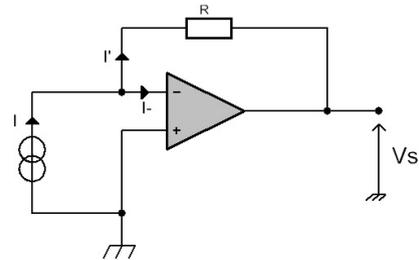
FIGURE IV.17: Filtre RC

### IV.5.2 Convertisseur courant tension

Le capteur ultrasonique SICK UM18 utilisé pour mesurer le niveau du réservoir délivre un signal analogique en sortie de type courant 4-20mA, tandis que l'API SIEMENS S7-1200 utilisé ne possède que deux entrées analogiques de type tension 0-10V, pour remédier à ce problème on a opté pour un circuit électronique à base d'amplificateurs opérationnels qui convertit le signal 4-20mA en signal de tension 0-10V.



(A) Module convertisseur  
courant tension



(B) Schéma électrique du  
convertisseur

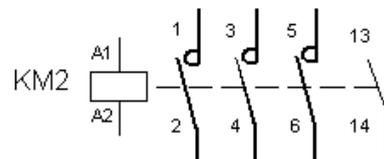
FIGURE IV.18: Convertisseur courant tension

### IV.5.3 Les contacteurs et les relais

Un **contacteur** est un appareil électrotechnique destiné à établir ou interrompre le passage du courant, à partir d'une commande électrique. Pour alimenter la pompe, on a utilisé un contacteur SIEMENS 3RH2140-1AP00, afin d'assurer la séparation entre le signal de commande et l'alimentation, donc aux bornes du contacteur on a branché notre pompe monophasée et de l'autre côté la sortie du variateur de vitesse, et la bobine du contacteur est commandée par la sortie digitale de l'automate S7-1200 qui assure la protection de la pompe contre la marche à sec comme expliqué dans notre cahier des charges précédemment.



(A) Contacteur SIE-  
MENS



(B) Schéma électrique du  
contacteur

FIGURE IV.19: Contacteur de puissance

## IV.5.4 Un relais

électromécanique est un organe électrique permettant de dissocier la partie puissance de la partie commande : il permet l'ouverture et la fermeture d'un circuit électrique par un second circuit complètement isolé (isolation galvanique) et pouvant avoir des propriétés différentes. On a utilisé deux relais 24V DC, Schneider Electric RSB2A080BDS afin de commander les deux électrovannes et les voyants en LED par l'API.



FIGURE IV.20: Relais Schneider Electric modèle RSB2A080BDS

## IV.6 Conclusion

Dans ce chapitre on a décrit tous les éléments qui constituent la station de pompage, et on a expliqué le rôle des interfaces E/S utilisées pour adapter les puissances et protéger l'instrumentation.

L'objectif de la réalisation du prototype a été atteint avec succès, en commençant par l'élaboration du cahier des charges et en passant par toutes les étapes de construction, câblage et étalonnage des capteurs, jusqu'au développement du programme principale et de gestion et de supervision.

# Conclusion générale

Notre contribution s'est portée sur la réalisation du prototype de la station de pompage, l'élaboration du cahier de charge, du programme principale et celui de la supervision.

La gestion de cette station est assurée par un automate Siemens S7-1200, qui forme une unité de traitement et de commande d'une grande flexibilité, qu'on a pu programmé grâce au logiciel de programmation et de supervision TIA Portal. La supervision de la station de pompage est assurée à travers une interface HMI SIMATIC TP900 COMFORT.

Ce projet nous a permis de concevoir de nouvelles compétences dans la programmation et la réalisation des interfaces HMI ainsi on a développé nos connaissances dans l'instrumentation et le câblage électrique ainsi que les réseaux industriels. On a appris à résoudre divers problèmes, à gérer le temps, à travailler en équipe, développer un esprit d'analyse et la mise en application des connaissances acquises.

En fin, nous espérons que notre travail sera utile aux étudiants en automatique, afin d'appliquer leurs connaissances théoriques et apprendre à maîtriser la programmation et la supervision sous TIA Portal grâce aux énoncés de TP qui ont été proposés dans l'annexe.

# Bibliographie

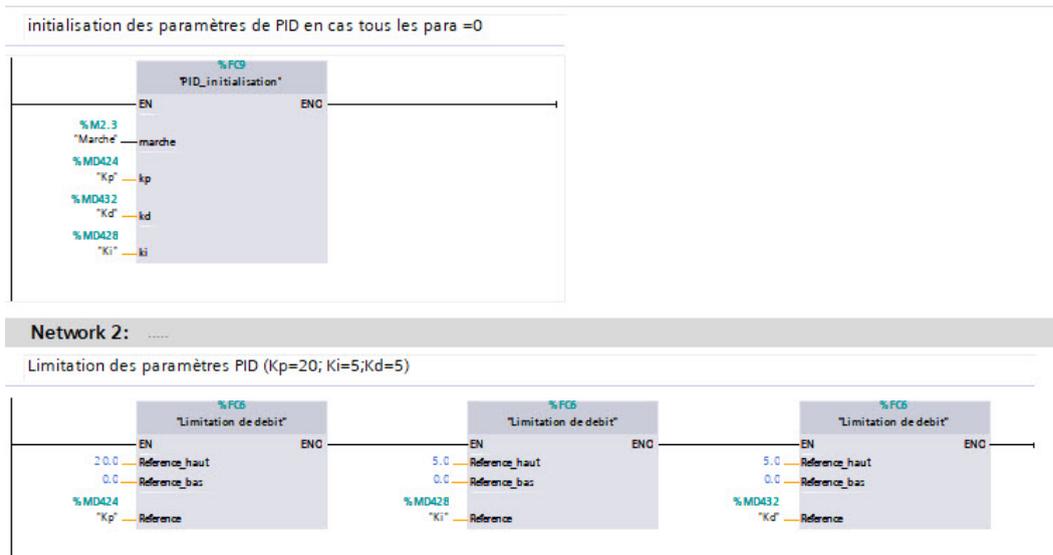
- [1] G. MICHEL. Les api architecture et application des automates programmables industriels, edition dunod. 1987.
- [2] Automationsense. <https://www.automation-sense.com/blog/automatisme/langage-de-programmation-siemens-lad-stl-fbd-scl-sfc.html>.
- [3] SIEMENS. Automate modulaire compact simatic s7-1200 manuel.
- [4] SIEMENS. Simatic hmi devices, comfort panels manuel.
- [5] Wikipedia. [https://fr.wikipedia.org/wiki/Automate\\_programmable\\_industriel#Constitution](https://fr.wikipedia.org/wiki/Automate_programmable_industriel#Constitution).
- [6] BONNASSIEUX Y LAVABRE M LOUIS J, MULTON B. Convertisseurs des fréquences pour la variation de vitesse des mas.
- [7] Schneider Electric. Chapitre 5, départs moteur.
- [8] Guillaume DUPAS. <https://www.tecnipass.com/cours-materiels-contrrole-parametrage.variateur?page=5>.
- [9] Schneider Electric. Altivar 12 variable speed drives for asynchronous motors user manual. Mai 2010.
- [10] Karl H. Johansson. Anders Hansson. Teaching multivariable control using the quadruple-tank process iee. February 1999.
- [11] Sick UM18. Capteurs à ultrasons, fiche technique en ligne.
- [12] SIEMENS. Initiation à la programmation du simatic s7-1200 avec tia portal vx. 2016.

# Annexe A : Programmation de l'API

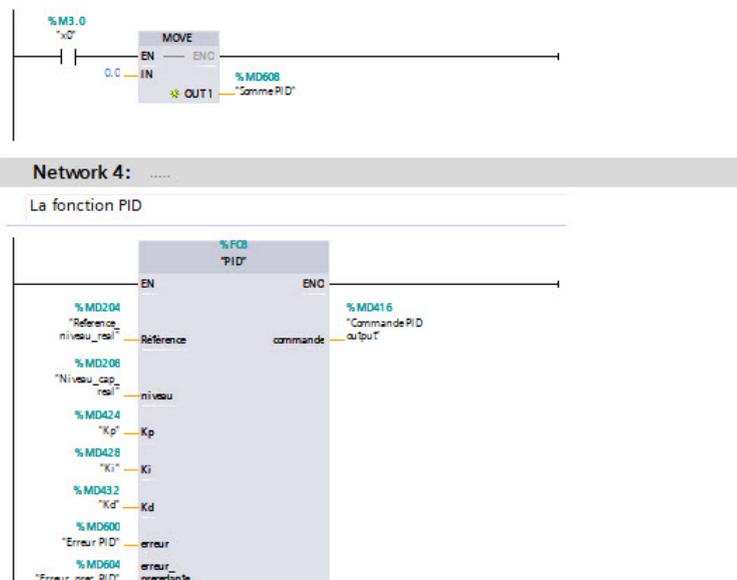
## Introduction

Dans cette première annexe on présente le programme implémenté dans la CPU s7-1200 de SIEMENS, avec des petites explications de la logique suivie.

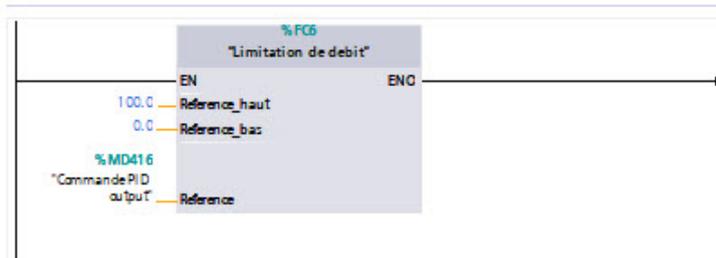
## Le programme principal (Main)



Les paramètres du PID sont initialisés automatiquement pour des valeurs idéales, et sont aussi limités par des seils à ne pas dépasser pour l'utilisateur.

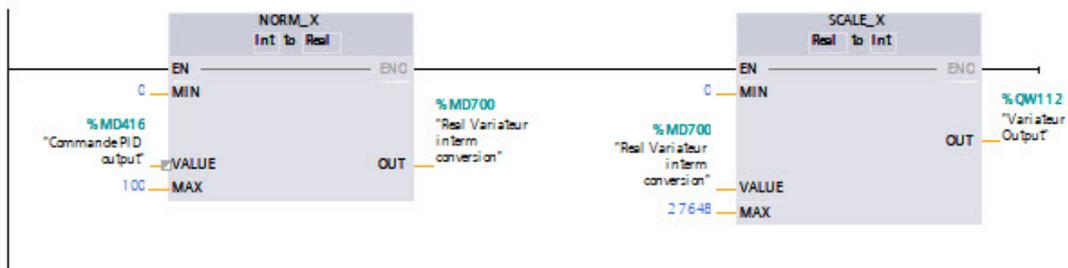


La somme du PID (l'action intégrale) est initialisée à chaque fois le bouton RESET est appuyé.



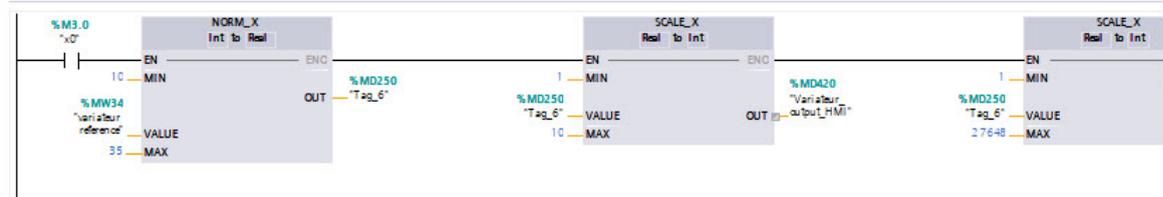
**Network 6:** .....

Adapter la commande en sortie analogique



La commande calculée doit être transférée comme une sortie analogique (en 16bit), donc une valeur entre (0 et 27648).

La référence en cas du mode manuel...



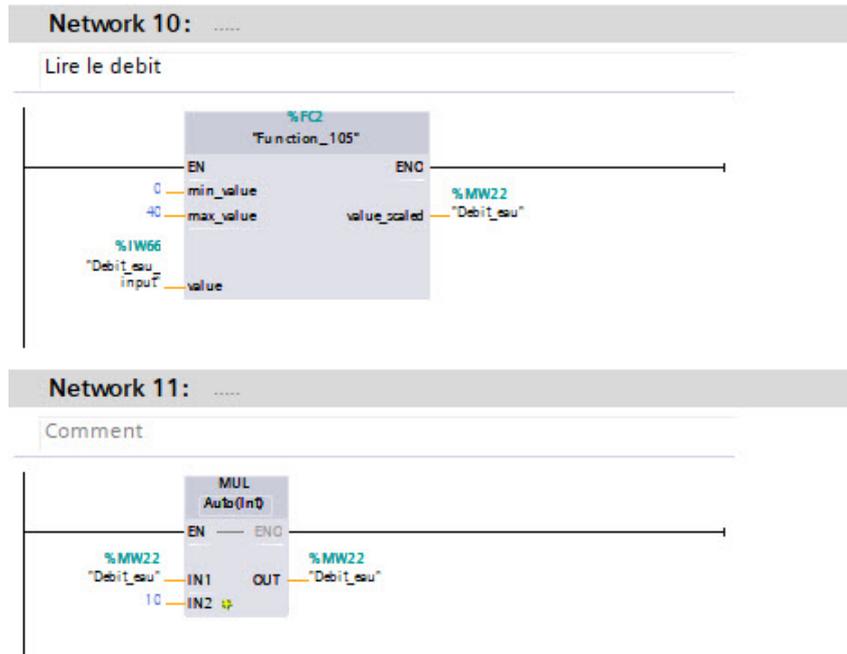
**Network 8:** .....

**Network 9:** .....

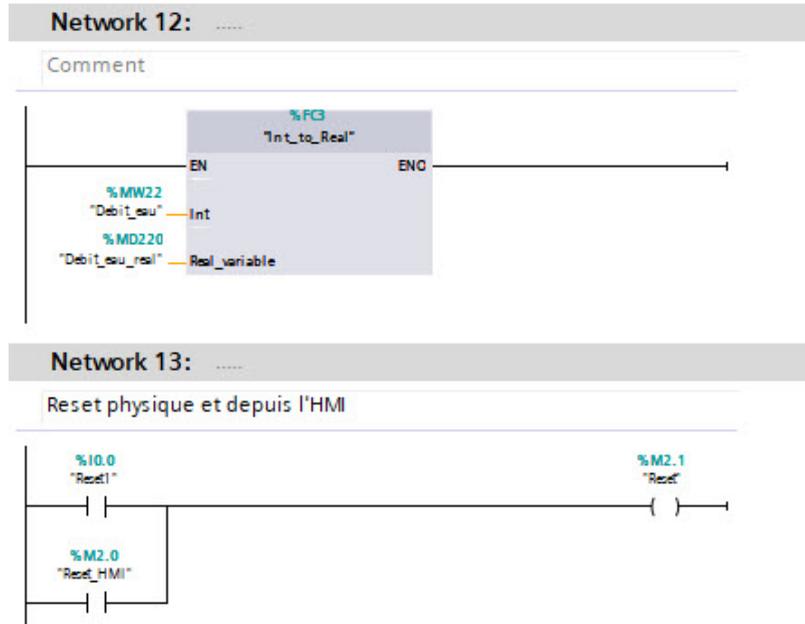
Mise a 1 du bit M26.0 pour les alarmes



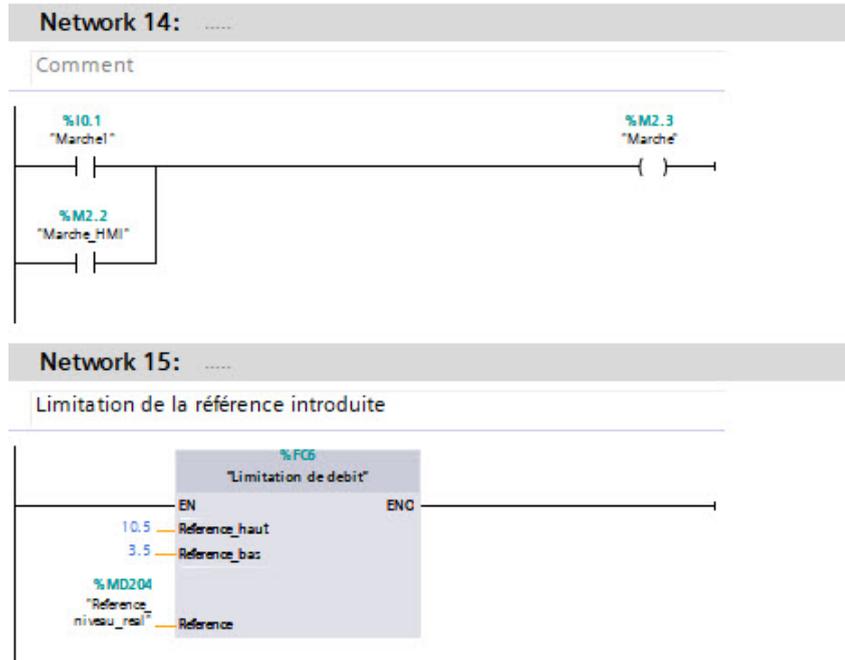
Le réseau 7 montre le fonctionnement du mode manuel qui est relié à l'HMI, avec une mise à 1 d'un bit qui indique que le bouton d'arrêt d'urgence est appuyé.



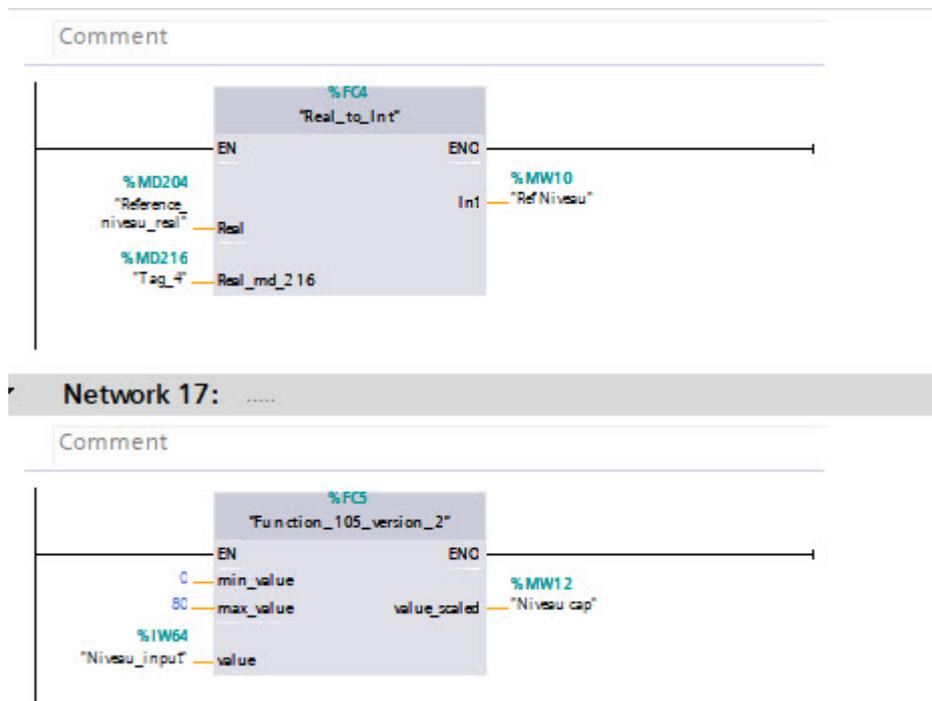
Le réseau 10 montre la fonction qui récupère les données du débitmètre qui fonctionne avec le microcontrôleur Arduino, et le réseau 11 est une mise à l'échelle (scale).

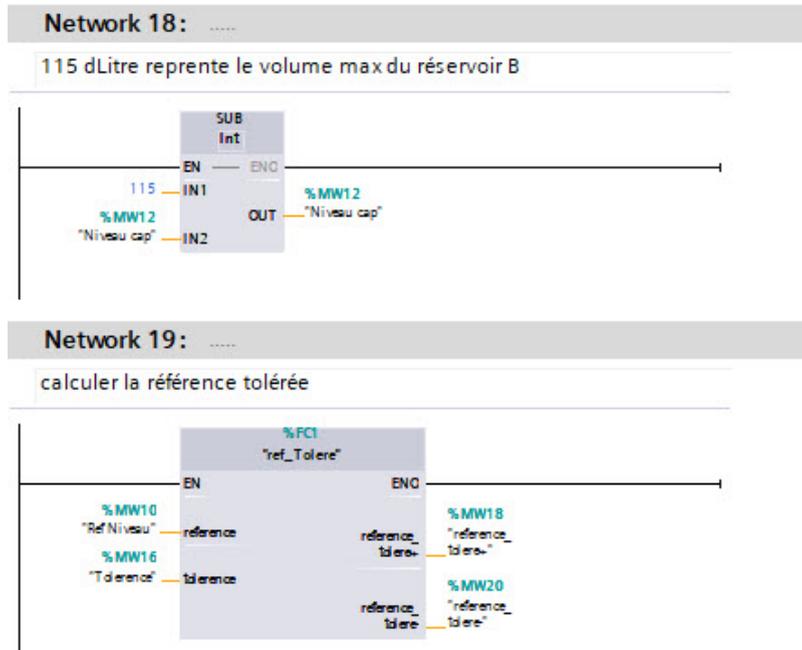


Le calcul est fait en dLitre, donc, on a fait une conversion de toutes les variables vers le type "int" pour le calcul, et une fois le calcul est fait, on les reconvertit en valeur "réel" pour l'affichage sur l'IHM.

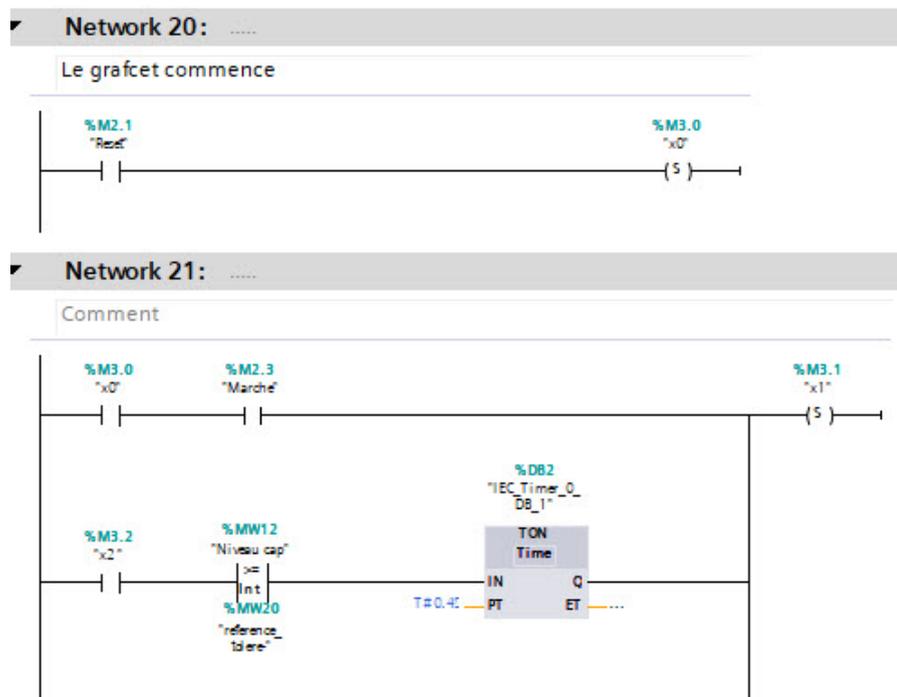


Le réseau 15 concerne la limitation de la référence de niveau introduite par l'utilisateur, ces valeurs sont choisies par rapport aux dimensions du réservoir.

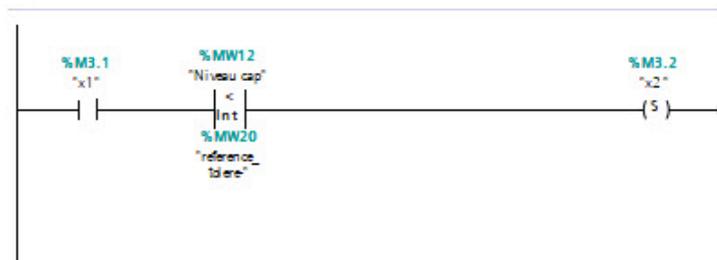




Le réseau 18 montre le calcul du niveau de l'eau dans le réservoir B en dLitre.  
 Le réseau 19 montre la fonction chargée du calcul de la référence tolérée (plage de tolérance).

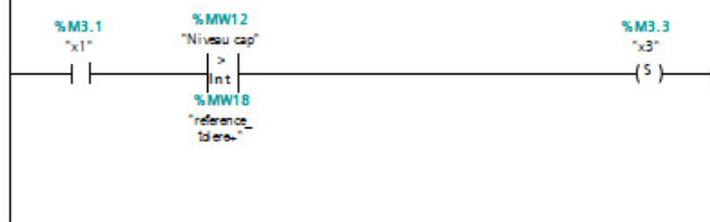


Le réseau montre les équations du grafcet.



**Network 23:** .....

Comment

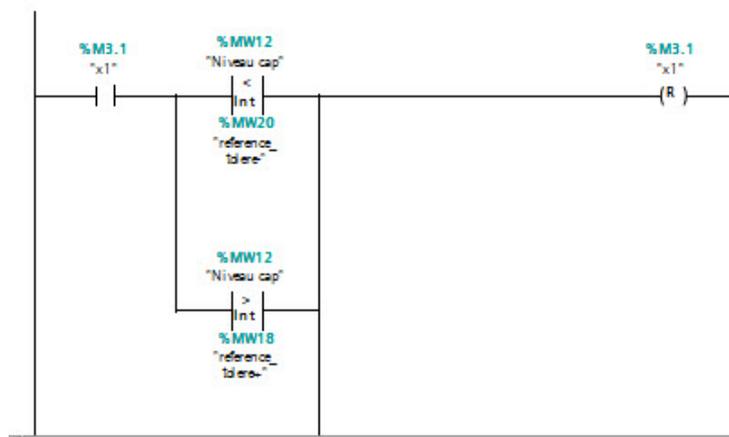


Comment

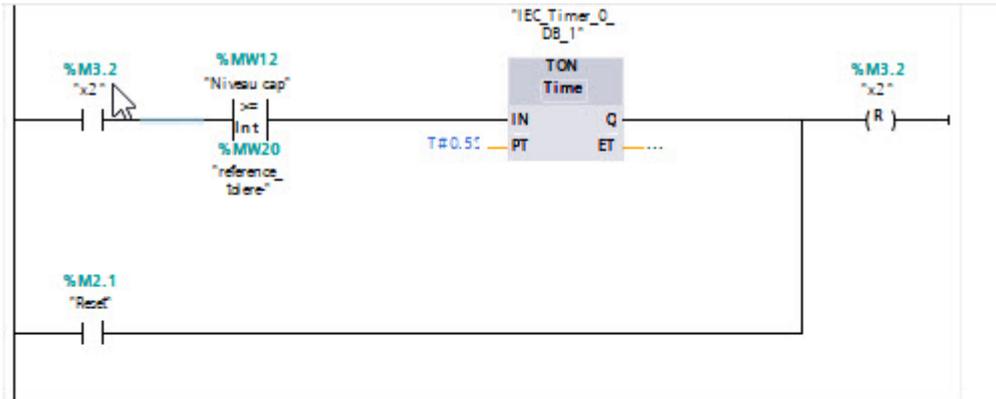


**Network 25:** .....

Comment

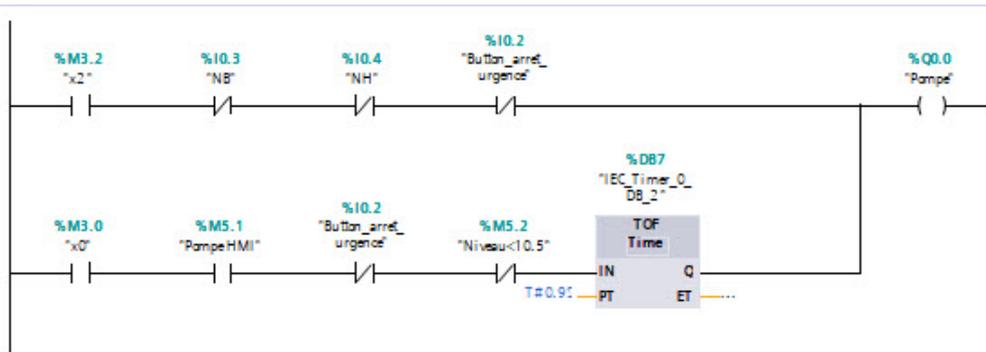
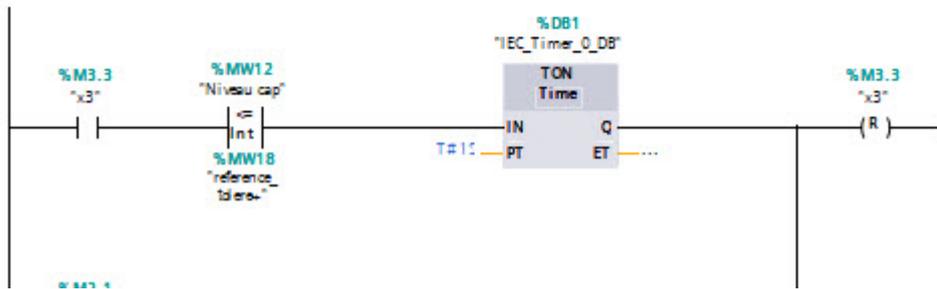


Toujours les équations du Grafcet.



Network 27: .....

Comment

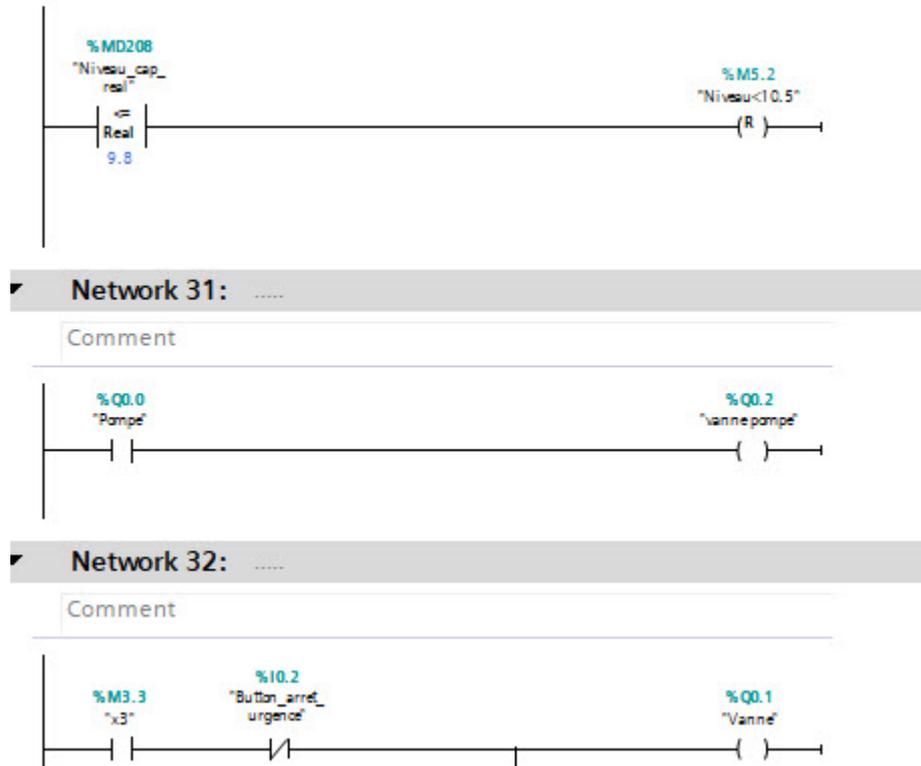


Network 29: .....

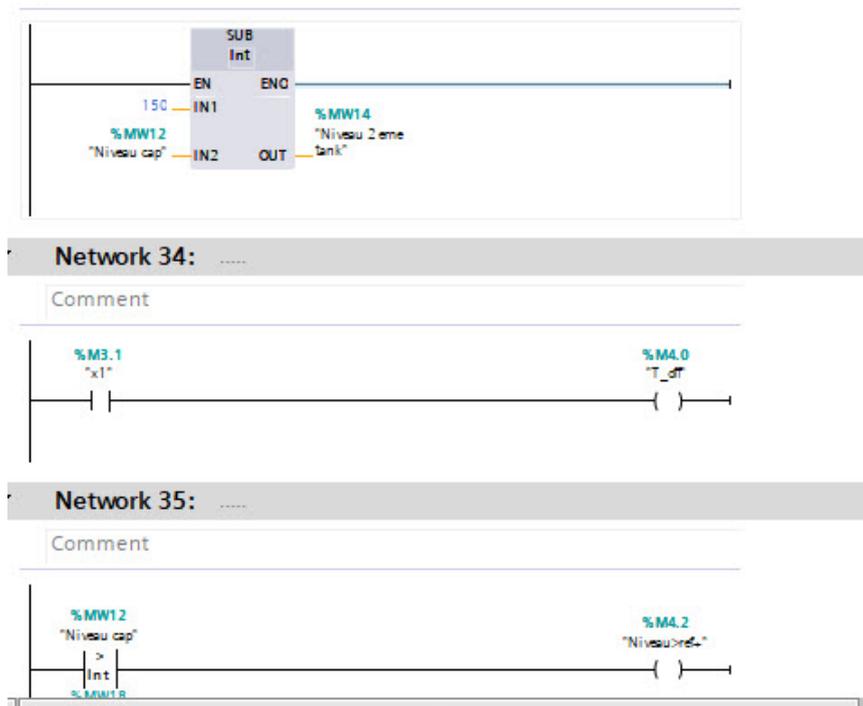
Comment



le réseau 28 montre les conditions pour le fonctionnement de la pompe.



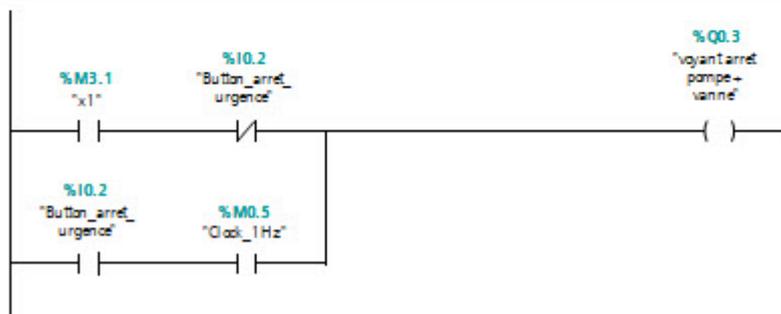
Le réseau 31 représente la commande de l'électrovanne (anti-retour).  
 Le réseau 32 représente la commande de l'électrovanne de vidange.



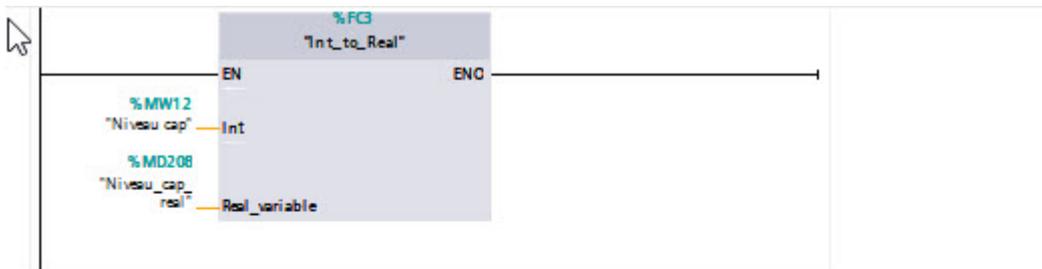


**Network 37: .....**

Comment

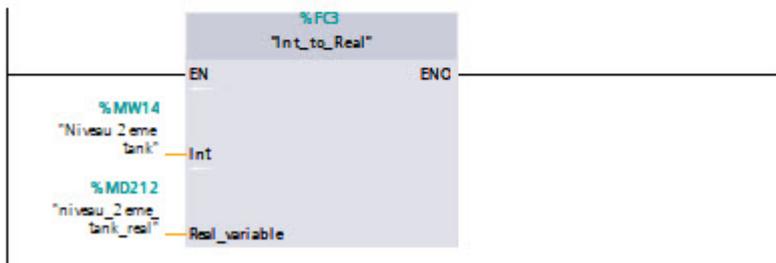


Le réseau 37 montre les conditions sur le voyant qui indique le fonctionnement de la pompe.

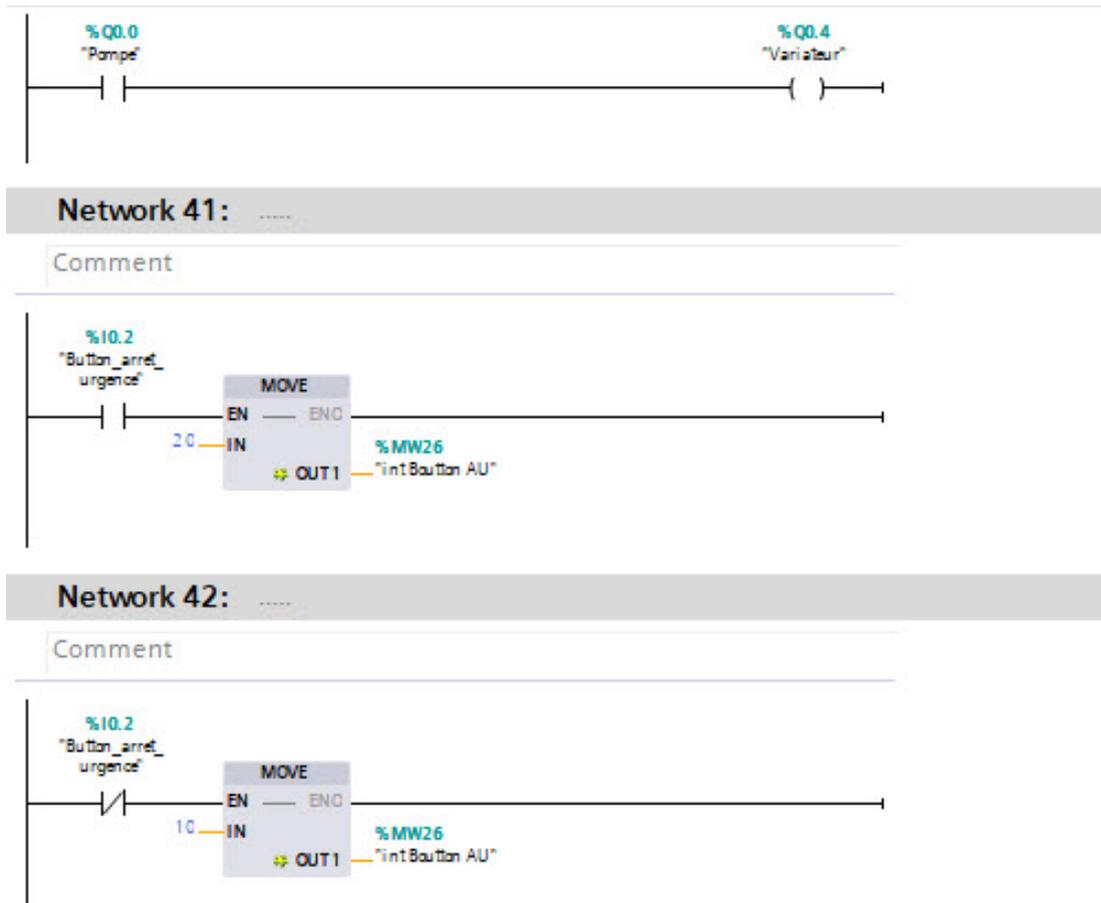


**Network 39: .....**

Comment



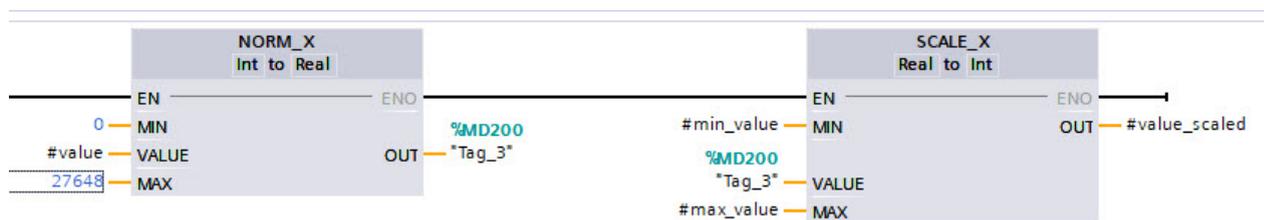
Reconversion de la variable du type "Int" vers "Real". (réseau 38 et 39).



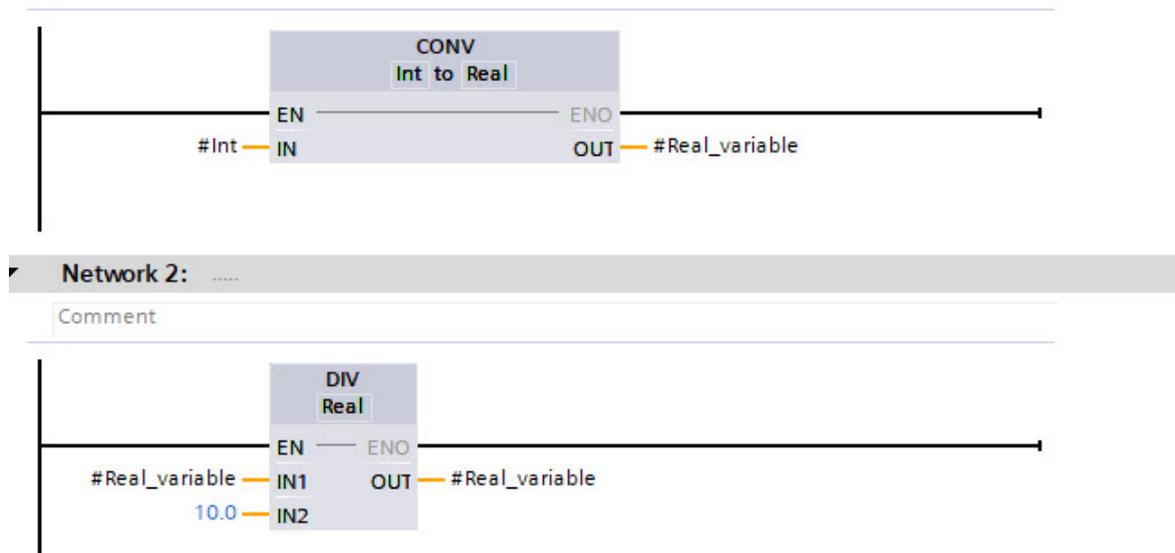
Les réseaux 41 et 42 concernent la détection du bouton d'arrêt d'urgence comme une erreur par l'HMI.

## Les fonctions utilisées dans le main

On a créé des fonctions pour les tâches répétitives, ainsi que pour les instructions programmées en SCL (script).



Cette fonction effectue la mise à l'échelle des entrées analogiques et les transforme en grandeurs physiques.



La mise à l'échelle des grandeurs calculées pour les afficher sur les vues de l'HMI.

Limitation de debit			
	Name	Data type	Default value
1	Input		
2	Reference_haut	Real	
3	Reference_bas	Real	
4	Output		
5	<Add new>		

```

IF... CASE... FOR... WHILE... (*...*) REGION
OF... TO DO.. DO...

1 IF #Reference < #Reference_bas THEN
2   #Reference := #Reference_bas;
3 ELSIF #Reference > #Reference_haut THEN
4   #Reference := #Reference_haut;
5   ;
6 END_IF;
7
    
```

La limitation de la référence introduite par l'utilisateur par une plage permise selon les contraintes imposées par le système.

On a aussi utiliser cette fonction pour d'autres limitations comme la limitation des paramètres du PID.

PID				
	Name	Data type	Default value	Comment
1	Input			
2	Référence	Real		
3	niveau	Real		
4	Kp	Real		
5	Ki	Real		

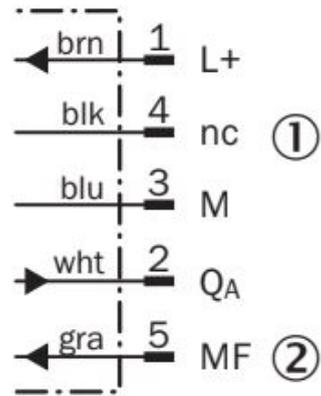
IF...	CASE... OF...	FOR... TO DO..	WHILE... DO...	(*...*)	REGION
<pre> 1 #erreur := #Référence - #niveau; 2 #somme := #somme + #erreur; 3 #delta := #erreur - #erreur_precedante; 4 (* 5 #commande := #Kp * #erreur + #Ki * #somme + #Kd * #delta; 6 #sum := #somme * 0.1; 7 *) 8 #commande := #Kp * #erreur + #Ki * #somme * #Temps_d + #Kd * #delta / #Temps 9 #erreur_precedante := #erreur; </pre>					

**La fonction PID :** Il existe un bloc prédéfini dans TIA PORTAL pour le calcul de la commande par PID, mais on a créé une fonction similaire nous même pour pouvoir varier les paramètres du régulateur via l'IHM.

## Annexe B : Capteurs



(A) Capteur de distance cylindrique



- ① Non affecté
- ② Entrée multifonction

(B) Schéma de câblage du capteur

FIGURE IV.21: Capteur de distance à ultrasons SICK UM18

## 1 Capteur de distance [11]

Afin de mesurer le niveau d'eau dans le réservoir, on a utilisé le capteur de distance à ultrasons "SICK UM18". Mesure fiable, quelles que soient la couleur du matériau, la transparence, la brillance et les lumières parasites.

Schéma de câblage du capteur :

fil marron (L+) : Vcc +24V

fil blanc (QA) : sortie analogique du type 4-20 mA

fil bleu (M) : GND

## Performance

<b>Portée de travail, portée limite</b>	30 mm ... 250 mm, 350 mm
<b>Objet à mesurer</b>	Objets naturels
<b>Résolution</b>	$\geq 0,08$ mm
<b>Répétabilité</b>	$\pm 0,15$ % <sup>1)</sup>
<b>Précision</b>	$\leq 2$ % <sup>1)</sup>
<b>Temps de réponse</b>	32 ms
<b>Durée de sortie</b>	8 ms
<b>Fréquence des ultrasons (typique)</b>	320 kHz
<b>Plage d'enregistrement (courante)</b>	Voir les schémas

## Interfaces

<b>Sortie analogique</b>	1 x 4 mA ... 20 mA ( $\leq 500 \Omega$ ) <sup>1)</sup>
<b>Résolution sortie analogique</b>	12 bit
<b>Entrée multifonction (MF)</b>	1 x

<sup>1)</sup> Pour 4 mA ... 20 mA et  $U_V \leq 20$  V charge max.  $\leq 100 \Omega$ .

## Mécanique/électronique

<b>Tension d'alimentation <math>U_V</math></b>	CC 10 V ... 30 V <sup>1)</sup>
<b>Puissance absorbée</b>	$\leq 1,2$ W <sup>2)</sup>
<b>Durée d'initialisation</b>	< 300 ms
<b>Forme</b>	Cylindrique

## 2 Débitmètre

Le débitmètre utilisé dans le prototype est dédié aux applications Arduino et non pas à l'industrie, on l'a utilisé pour des raisons budgétaires.



FIGURE IV.22: Débitmètre Arduino

Schéma de câblage du débitmètre :

fil rouge : Vcc +5V

fil noir : GND

fil jaune : sortie numérique du débitmètre.

La sortie du débitmètre délivre un train d'impulsions, donc la fréquence du signal de sortie est linéaire par rapport au débit de l'écoulement. Ce signal ne peut être exploité directement par l'automate, il nécessite un traitement, et pour cela on a élaboré un code Arduino qui compte le nombre d'impulsions du débitmètre, et le transforme en un signal PWM.

```

int pwmOut = 3; // The output to the filter to PLC
volatile int flow_frequency; // Measures flow meter pulses
unsigned int l_hour; // Calculated litres/hour
unsigned int l_min; // Calculated litres/min
unsigned char flowmeter = 2; // Flow Meter Pin number
unsigned int debit;
unsigned long currentTime;
unsigned long cloopTime;

void flow () // Interruot function
{
    flow_frequency++;
}

void setup()
{
    pinMode(flowmeter, INPUT);
    Serial.begin(9600);
    attachInterrupt(0, flow, RISING); // Setup Interrupt

    sei(); // Enable interrupts
    currentTime = millis();
    cloopTime = currentTime;
}

void loop ()
{
    currentTime = millis();
    // Every second, calculate and print litres/hour
    if(currentTime >= (cloopTime + 1000))
    {
        cloopTime = currentTime; // Updates cloopTime
        // Pulse frequency (Hz) = 7.5Q, Q is flow rate in L/min. (Results in +/- 3% range)
        l_hour = (flow_frequency * 60 / 7.5); // (Pulse frequency x 60 min) / 7.5Q = flow rate in L/hour
        l_min=l_hour;
        flow_frequency = 0; // Reset Counter
        Serial.print(l_min, DEC); // Print litres/hour
        Serial.println(" L/min");
        debit=map(l_min,0,140,0,255);
        analogWrite(pwmOut, debit); // to be filtred by RC filter
    }
}

```

FIGURE IV.23: Code Arduino utilisé

# Annexe C : Variateurs de vitesse Industriels

## Variateur de vitesse de Schneider Electric ATV12

### Schéma de câblage de l'alimentation monophasée

Il y a 3 types d'alimentations dans la gamme ATV12 de Schneider Electric :

- **ATV12....F1** alimentation monophasée entre 100 et 120V
- **ATV12....M2** alimentation monophasée entre 200 et 240V
- **ATV12....M3** alimentation triphasée entre 200 et 240V

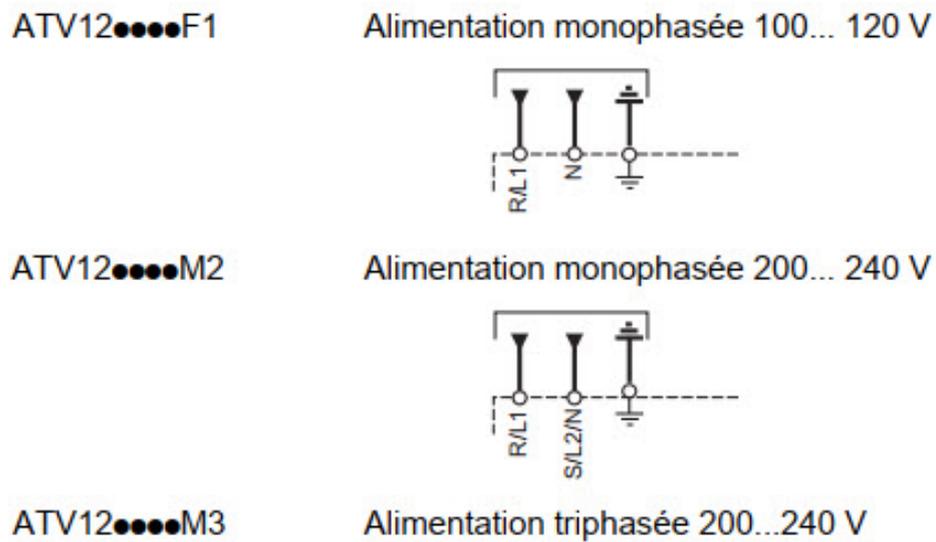


FIGURE IV.24: Schéma de câblage de l'alimentation ATV12

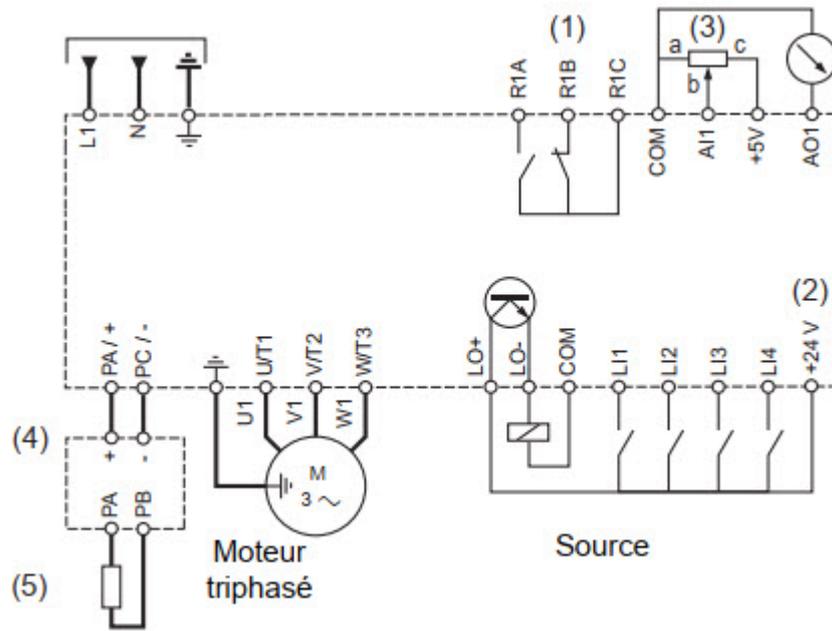


FIGURE IV.25: Schéma de câblage du variateur ATV12H037M2

- (1) Contacts de relais R1, pour indiquer l'état actuel du variateur. (voir....)
- (2) : Alimentation interne du variateur qui donne 24V, on peut l'utiliser comme une source ou une alimentation externe (+30V au maximum) pour les entrées logiques Lix, qui ont des différentes fonctionnalités :
  - Le sens du moteur.
  - les vitesses présélectionnées.
- (3) : 2.2 k $\Omega$  potentiomètre de référence. Il peut être remplacé par un potentiomètre 10 k $\Omega$  (maximum).
- (4) : Module peut être utilisé au freinage.
- (5) : Une résistance de freinage optionnelle.

## Programmation manuelle des variateurs

### Référence de vitesse

Le variateur suit la vitesse que l'utilisateur introduit via l'entrée analogique AI1 :

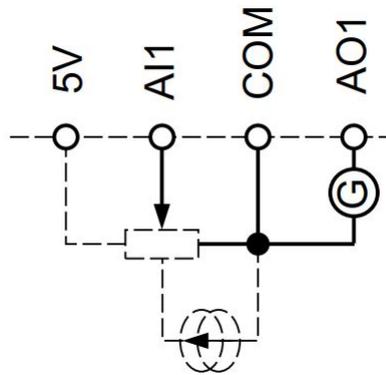


FIGURE IV.26: L'entrée analogique du variateur [9]

L'utilisateur peut choisir le type d'information qui lui convient comme référence du variateur, on distingue 3 types :

- Tension : 0-5V (5U)
- Tension : 0-10V (10 U)
- Courant : x-y mA (0A) (**plage de x à y mA**).

Pour choisir l'un de ces types, on suit la configuration suivante :

$$\boxed{Conf \Rightarrow Full \Rightarrow I^-O^- \Rightarrow AII^- \Rightarrow AII^t \Rightarrow 5U \text{ ou } 10U \text{ ou } 0A} \quad (IV.1)$$

Par exemple si on choisit 0A : courant de plage (x à y), on définit la plage par la configuration suivante :

- La valeur min :

$$\boxed{Conf \Rightarrow Full \Rightarrow I^-O^- \Rightarrow AII^- \Rightarrow CrLI \Rightarrow 0 \text{ ou } 4mA} \quad (IV.2)$$

- La valeur max :

$$\boxed{Conf \Rightarrow Full \Rightarrow I^-O^- \Rightarrow AII^- \Rightarrow CrHI \Rightarrow 20mA} \quad (IV.3)$$

## Sortie relais

Le variateur donne une sortie relais qui fonctionne en cas de défaut, le contact du relais R1 s'ouvre et le code d'erreur affiche sur l'écran.

La sortie relais peut être utilisée pour la protection en cas d'erreurs.

Le contact du relais réagit aux erreurs suivantes ;

Valeur possible / Fonction	
<b>nD</b>	Non affecté
<b>FLt</b>	Aucune erreur détectée
<b>rUn</b>	Marche du variateur
<b>FtA</b>	Seuil de fréquence atteint
<b>FLA</b>	HSP atteint
<b>CtA</b>	Seuil I atteint
<b>SrA</b>	Référence de fréquence atteinte
<b>tSA</b>	Therm. moteur atteint
<b>ULA</b>	Alarme de sous-charge
<b>OLA</b>	Alarme de surcharge
<b>AP I</b>	A11 AI. 4-20

FIGURE IV.27: Fonctionnement du relais

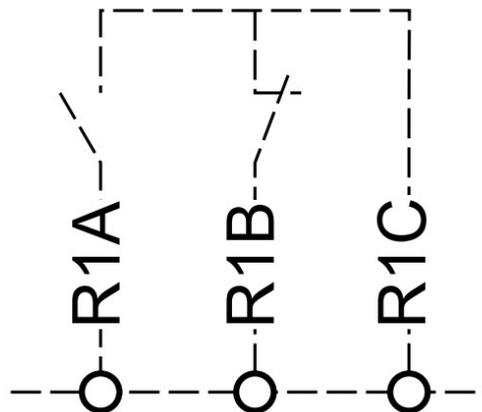


FIGURE IV.28: Schéma de câblage du relais

**Le Schéma de câblage** Le relais du variateur contient deux contacts :

- **R1A** : Contact du relais Normalement Ouvert (NO)
- **R1B** : Contact du relais Normalement Fermé (NC).

- **R1C** : Borne commune du relais.

## Les vitesses présélectionnées

Le variateur peut fonctionner sur des vitesses présélectionnées dans le cas où on a besoin des vitesses bien définies pour un certain processus.

Le variateur ATV12 nous donne la possibilité de choisir jusqu'à 8 vitesses, et chacune de ces vitesses a un code associé pour l'activer.

### Remarques :

- Les références de vitesse via la commande logique sont prioritaires sur les références analogiques.
- Dans le cas où les vitesses présélectionnées sont utilisées, on ne peut pas utiliser la configuration 3 fils, seulement 2 fils.

On a la possibilité de sélectionner soit 2, 4, ou bien 8 vitesses.

L'activation de ces vitesses se fait par le code binaire montré dans la figure ci-dessous :

8 vitesses LI (PS8)	4 vitesses LI (PS4)	2 vitesses LI (PS2)	Référence fréq.
0	0	0	Référence
0	0	1	SP2
0	1	0	SP3
0	1	1	SP4
1	0	0	SP5
1	0	1	SP6
1	1	0	SP7
1	1	1	SP8

FIGURE IV.29: Tableau de combinaison des entrées de vitesses présélectionnées

La programmation manuelle sur le variateur pour choisir l'un des modes :

$$\boxed{Conf \Rightarrow Full \Rightarrow FUn \Rightarrow Pss^- \Rightarrow Ps2 \text{ ou } Ps4 \text{ ou } Ps8} \quad (IV.4)$$

**Exemple d'application :**

Dans cette exemple, on essaye d'introduire 4 vitesses présélectionnées : La première vitesse est toujours réservée à la référence donnée par le potentiomètre, donc ce qui nous donne la possibilité d'introduire que 3 vitesses en cas on a choisi "Ps4". On doit aussi définir les entrées logiques qu'on va utiliser pour le contrôle de ces vitesses : Par exemple on choisit : LI2 et LI3, on laisse toujours LI1 pour le contrôle du sens :

$$\boxed{Conf \Rightarrow Full \Rightarrow FUN \Rightarrow Pss^- \Rightarrow Ps2 \Rightarrow L2H} \quad (IV.5)$$

$$\boxed{Conf \Rightarrow Full \Rightarrow FUN \Rightarrow Pss^- \Rightarrow Ps4 \Rightarrow L3H} \quad (IV.6)$$

Et pour configurer une de ces vitesses :

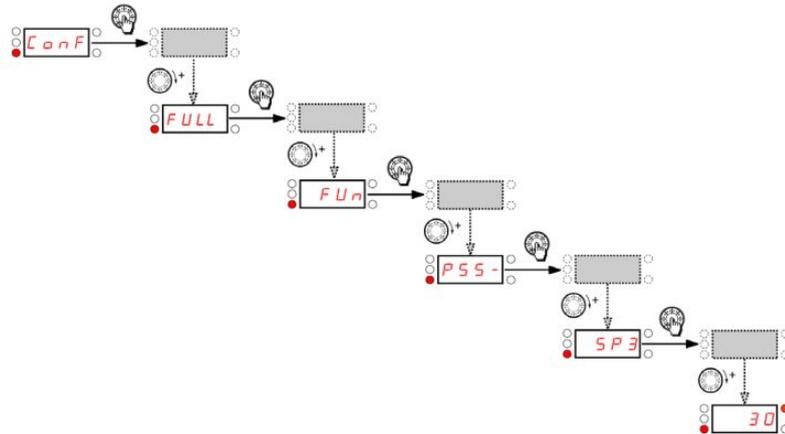


FIGURE IV.30: Comment introduire une vitesse présélectionnée

**Une simulation du fonctionnement de ces 4 vitesses :**

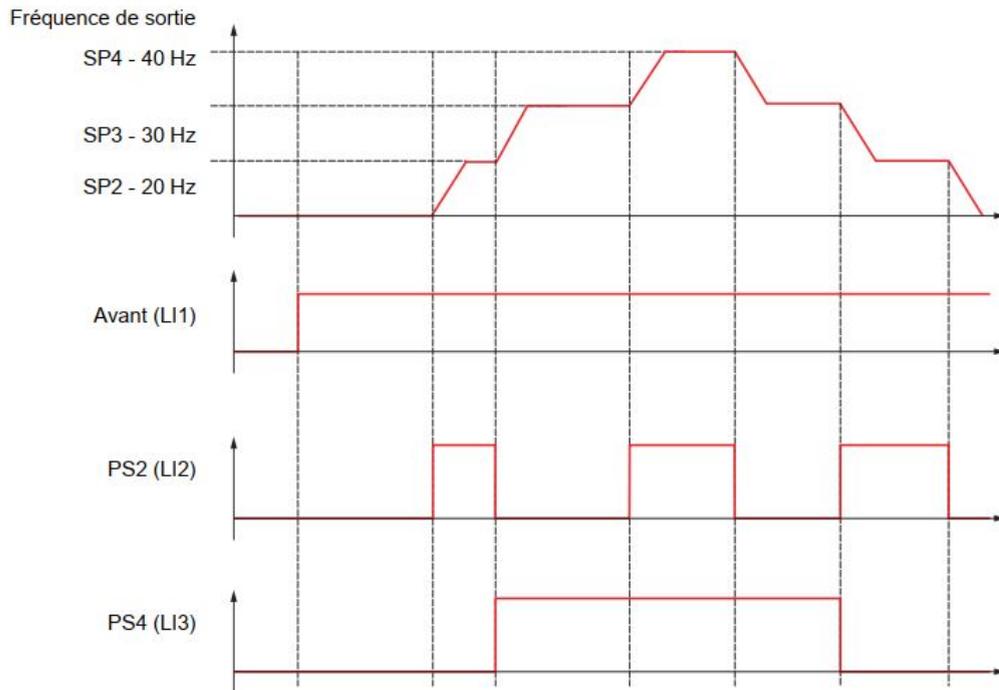


FIGURE IV.31: fonctionnement des Vitesses présélectionnées [9]

## Protection thermique

Le variateur peut protéger le moteur contre les surcharges ou les courts-circuit dont la fonction est d'interrompre le courant électrique. Le variateur mesure le courant absorbé par le moteur et le compare avec le courant nominal inséré par l'utilisateur, en cas d'une surcharge ou de court-circuit, le moteur absorbe plus de courant, ce qui nécessite une intervention pour couper arrêter le moteur. La programmation manuelle sur le variateur pour choisir l'un des modes :

$$\boxed{Conf \Rightarrow Full \Rightarrow Flt^- \Rightarrow itH \Rightarrow \frac{Irth}{In}} \quad (IV.7)$$

où  $Irth$  représente le courant thermique du moteur. et  $In$  représente le courant nominal du variateur (4.9 à 5.9A) selon la tension (200 à 240V) respectivement.

La plage du réglage varie de 0.2 à 1.5  $In$  du variateur.

## Modes de freinage

Le moteur électrique tourne dans son fonctionnement en utilisant l'énergie électrique, mais pour l'arrêter, il faut utiliser l'une de ces méthodes :

### Arrêt en roue libre

où il n'existe pas une contrainte sur le temps de décélération du moteur, donc on laisse le moteur s'arrêter sur sa propre inertie, ce qui fait que le temps d'arrêt dépend de la charge et peut prendre beaucoup de temps pour s'arrêter.

### arrêt par injection de courant continu [9]

L'injection du courant dans les trois phases provoque la décélération du moteur d'une manière plus rapide, mais toujours le temps de décélération dépend de la charge.

Le variateur donne la possibilité de choisir l'un des types suivants dans l'injection du courant DC :

- **Fonction inactive** où le variateur n'injecte aucun courant dans les phases.
- **Injection limitée** Où le courant DC injecté se fait par des durées limitées.
- **Injection continue** Où le courant DC injecté se fait d'une manière continue.

On peut choisir l'un de ces types respectivement en suivant la configuration suivante :

$$\boxed{\text{Conf} \Rightarrow \text{Full} \Rightarrow \text{FUN-} \Rightarrow \text{AdC-} \Rightarrow \text{AdC} \Rightarrow \text{nO} \text{ ou } \text{YES} \text{ ou } \text{Ct}} \quad (\text{IV.8})$$

## arrêt sur module de freinage

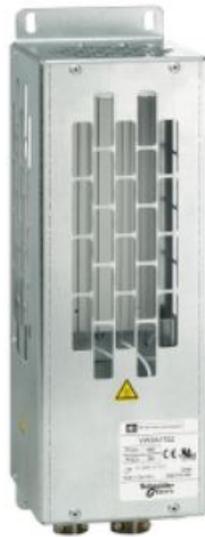


FIGURE IV.32: La résistance de freinage VW3A7702

Dance ce mode, le moteur fonctionne comme génératrice et le variateur dissipe cette énergie dans la résistance (module de freinage).

Pour Altivar12, Schneider recommande la résistance : *VW3A7702* qui est compatible aussi avec d'autre série de variateurs :

### Compatibilité de gamme

Altivar 31C  
Altivar 312 Solar  
Altivar 32  
Altivar 12  
Altivar 71  
Altivar Easy 310  
Altivar 61  
Altivar 312

FIGURE IV.33: La compatibilité de la résistance avec les variateurs

La résistance a comme caractéristique :

- **Valeu ohmique** 60 Ohm.
- **Puissance moyenne** 115W à 41°C et 100W à 50°C.
- **Poids** 2.4 Kg.
- **Température de fonctionnement** 0 à 50°C.

**Le module VW3A7005** La résistance ne peut pas être branchée directement avec le variateur, donc il faut utiliser un autre module : VW3A7005

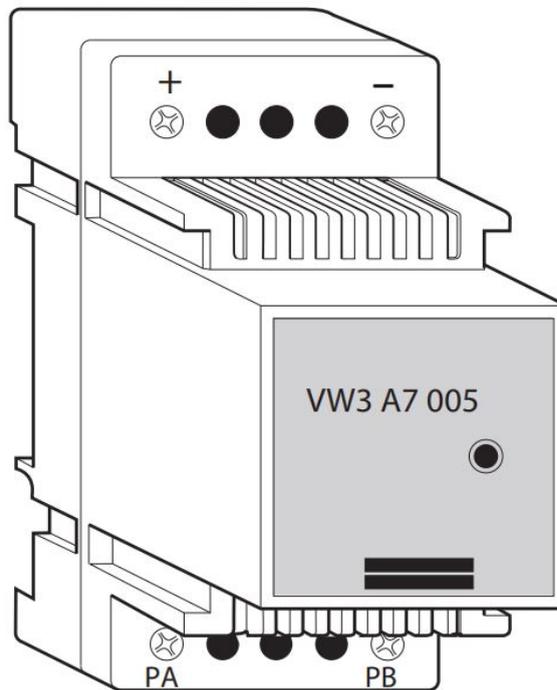


FIGURE IV.34: Le module de freinage VW3A7005

Le module est compatible seulement avec Altivar 12.

Ce module permet de connecter la résistance VW3A7702 avec le variateur pour qu'elle soit exploitable. Le Schéma de câblage des deux avec le variateur est comme suit :

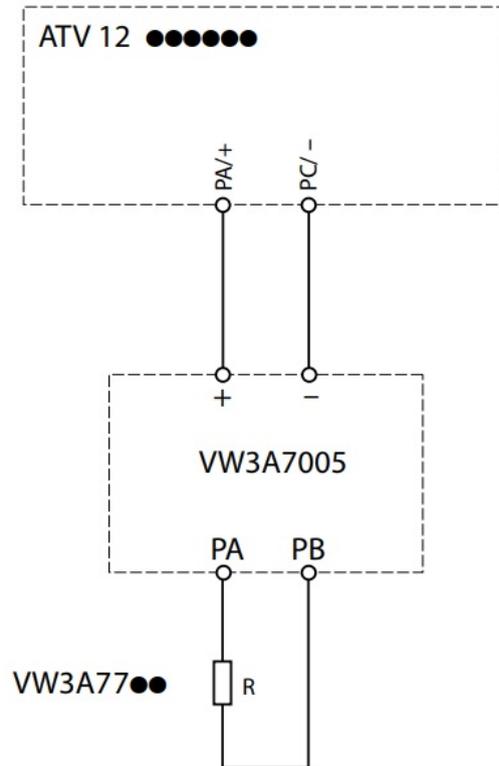


FIGURE IV.35: Le Schéma de câblage du module de freinage

**Remarque :** Schneider Electric ne produit plus le module VW3A7005, donc, il faut les chercher ailleurs.

Il n'est plus possible de connecter une résistance de freinage sur un variateur ATV12.

Le module qui permettait cette connexion de référence VW3A7705 n'est plus commercialisé.

FIGURE IV.36: le module VW3A7005 n'existe plus

## Le bruit du moteur

Le bruit broduit par le moteur pour les petites fréquences est grand, et parfois ça pose un problème pour l'ouvrier ou n'importe quelle personne pret du moteur, ce qui nécessite un réglage dans le variateur qui permet de diminuer considérablement le bruit.

Pour appliquer ce réglage, on procède de la manière suivante :

$$\boxed{Conf \Rightarrow arrowFull \Rightarrow drC- \Rightarrow nrd \Rightarrow nO \text{ ou } YES} \quad (IV.9)$$

## Limitation du courant

On peut limiter le courant livrer par le variateur pour des raisons de surcharges ou des raisons de securités, et le choix doit d'être fait selon le couant du variateur  $I_n$ . Pour appliquer ce réglage, on procède de la manière suivante :

$$\boxed{Conf \Rightarrow Full \Rightarrow FUN- \Rightarrow CLI- \Rightarrow CL1 \Rightarrow \frac{courant}{I_n}} \quad (IV.10)$$

La plage de réglage est de 0.25 à 1.5  $I_n$ .

## Diagnostic et Dépannage [9]

Le manuel fourni par Schneider Electric, donne une bonne interprétation sur les codes d'erreurs, il faut juste de bien lire les instructions, et une petite réflexion :

### Defauts impossible a supprimer directement

Les erreurs qui apparaissent dans la figure ci-dessous doivent être trités avant de mettre hors tension le variateur, par ce que le variateur ne fonctionne que si la panne est réglée :

Code	Nom	Causes possibles	Remède
<b>C r F 1</b>	Bus DC précharge	<ul style="list-style-type: none"> <li>Défaut de contrôle du relais de chargement ou résistance de chargement endommagée</li> </ul>	<ul style="list-style-type: none"> <li>Éteignez puis rallumez le variateur</li> <li>Vérifiez les connexions</li> <li>Vérifiez la stabilité de l'alimentation principale</li> <li>Contactez votre représentant local Schneider Electric</li> </ul>
<b>I n F 1</b>	Erreur calibre	<ul style="list-style-type: none"> <li>La carte de puissance n'est pas la même que la carte stockée.</li> </ul>	<ul style="list-style-type: none"> <li>Contactez votre représentant local Schneider Electric</li> </ul>
<b>I n F 2</b>	Puiss. incompatible	<ul style="list-style-type: none"> <li>La carte de puissance est incompatible avec la carte de contrôle</li> </ul>	<ul style="list-style-type: none"> <li>Contactez votre représentant local Schneider Electric</li> </ul>
<b>I n F 3</b>	Déf. liaison série int.	<ul style="list-style-type: none"> <li>Interruption de communication entre les cartes internes</li> </ul>	<ul style="list-style-type: none"> <li>Contactez votre représentant local Schneider Electric</li> </ul>
<b>I n F 4</b>	Déf. interne zone fab	<ul style="list-style-type: none"> <li>Données internes incohérentes</li> </ul>	<ul style="list-style-type: none"> <li>Contactez votre représentant local Schneider Electric</li> </ul>
<b>I n F 9</b>	Déf. interne - mesure I	<ul style="list-style-type: none"> <li>La mesure du courant n'est pas correcte à cause du circuit matériel.</li> </ul>	<ul style="list-style-type: none"> <li>Contactez votre représentant local Schneider Electric</li> </ul>
<b>- - - -</b>	Problème au niveau du firmware de l'application	<ul style="list-style-type: none"> <li>Mise à jour invalide du firmware de l'application en utilisant l'outil Multi-Loader</li> </ul>	<ul style="list-style-type: none"> <li>Mettez à nouveau à jour le firmware de l'application du produit</li> </ul>
<b>I n F b</b>	Déf. int. capteur temp	<ul style="list-style-type: none"> <li>Le capteur de température du variateur ne fonctionne pas correctement.</li> <li>Le variateur est en court-circuit, ou</li> </ul>	<ul style="list-style-type: none"> <li>Contactez votre représentant local Schneider Electric</li> </ul>
<b>I n F E</b>	Défaut CPU	<ul style="list-style-type: none"> <li>Défaillance du microprocesseur interne</li> </ul>	<ul style="list-style-type: none"> <li>Éteignez puis rallumez le variateur</li> <li>Contactez votre représentant local Schneider Electric</li> </ul>

Code	Nom	Causes possibles	Remède
<b>OCF</b>	Surintensité	<ul style="list-style-type: none"> <li>Les paramètres du menu <b>Contrôle moteur d r C</b> - page 57 ne sont pas corrects</li> <li>Inertie ou charge trop élevée</li> <li>Verrouillage mécanique</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez les paramètres</li> <li>Vérifiez la taille du moteur / du variateur / la charge</li> <li>Vérifiez l'état du mécanisme</li> <li>Connectez des inductances moteur</li> <li>Réduisez la <b>Fréquence découp. S F r</b> page 59.</li> <li>Vérifiez la connexion à la terre du variateur, le câble moteur et l'isolation du moteur.</li> </ul>
<b>SCF1</b>	Court-circuit mot.	<ul style="list-style-type: none"> <li>Court-circuit ou mise à la terre au niveau de la sortie du variateur</li> <li>Défaut de terre pendant la marche</li> <li>Commutation des moteur pendant la marche</li> <li>Important courant de fuite à la terre si plusieurs moteurs sont connectés en parallèle</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez les câbles connectant le variateur au moteur et l'isolation du moteur</li> <li>Connectez les inductances moteur</li> </ul>
<b>SCF3</b>	Court-circuit terre		
<b>SCF4</b>	Court-circuit IGBT	<ul style="list-style-type: none"> <li>Court-circuit de composant de puissance interne détecté lors de la mise sous tension</li> </ul>	<ul style="list-style-type: none"> <li>Contactez votre représentant local Schneider Electric</li> </ul>
<b>SDF</b>	Survitesse	<ul style="list-style-type: none"> <li>Instabilité</li> <li>Survitesse due à l'inertie de l'application</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez le moteur</li> <li>La survitesse est supérieure de 10 % à la <b>Fréquence maxi. t F r</b> page 57, réglez ce paramètre si nécessaire.</li> <li>Ajoutez une résistance de freinage.</li> <li>Vérifiez la taille du moteur / du variateur / la charge</li> <li>Vérifiez les paramètres de la boucle de vitesse (gain et stabilité)</li> </ul>
<b>t n F</b>	Autoréglage	<ul style="list-style-type: none"> <li>Le moteur n'est pas connecté au variateur</li> <li>Perte d'une phase moteur</li> <li>Moteur spécial</li> <li>Le moteur tourne (entraîné par la charge, par exemple)</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez que le moteur et le variateur sont compatibles</li> <li>Vérifiez que le moteur est présent pendant l'autoréglage</li> <li>Si un contacteur de sortie est utilisé, fermez-le pendant l'autoréglage</li> <li>Vérifiez que le moteur est complètement arrêté</li> </ul>

Les messages d'erreurs qui apparaissent dans la figure ci-dessous peuvent être supprimés en redemarrant le variateur :

Code	Nom	Causes possibles	Remède
<b>LFF1</b>	Défaut perte courant AI	Détection si : <ul style="list-style-type: none"> <li>L'entrée analogique AI1 est configurée comme le courant</li> <li>Valeur mini AI1 <math>C_{rL1}</math> page 52 est supérieur à 3 mA</li> <li>L'entrée analogique de courant est inférieure à 2 mA</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez les connexions au bornier</li> </ul>
<b>ObF</b>	Freinage excessif	<ul style="list-style-type: none"> <li>Freinage trop brutal ou charge entraînant trop forte</li> </ul>	<ul style="list-style-type: none"> <li>Augmentez le temps de décélération</li> <li>Installez un module avec une résistance de freinage, si nécessaire</li> <li>Vérifiez la tension d'alimentation secteur pour vous assurer qu'elle est sous le maximum acceptable (20 % au-dessus de l'alimentation secteur maximum en fonctionnement)</li> </ul>
<b>DHF</b>	Surchauffe var.	<ul style="list-style-type: none"> <li>Température trop élevée du variateur</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez la charge du moteur, la ventilation du variateur et la température ambiante. Laissez le temps au variateur de refroidir avant de le redémarrer. Voir Conditions de montage et de température page 13.</li> </ul>
<b>OLC</b>	Surcharge process	<ul style="list-style-type: none"> <li>Surcharge du process</li> </ul>	<ul style="list-style-type: none"> <li>Assurez-vous que le process et les paramètres du variateur sont cohérents</li> </ul>
<b>OLF</b>	Surcharge moteur	<ul style="list-style-type: none"> <li>Déclenché par un courant moteur excessif</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez la protection thermique du moteur et la charge du moteur.</li> </ul>
<b>DPF1</b>	Perte 1 phase mot.	<ul style="list-style-type: none"> <li>Perte d'une phase à la sortie du variateur</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez les connexions allant du variateur au moteur</li> <li>En cas d'utilisation d'un contacteur en aval, vérifiez la connexion du câble et du contacteur</li> </ul>
<b>DPF2</b>	Perte 3 phases mot.	<ul style="list-style-type: none"> <li>Moteur non connecté</li> <li>Puissance trop basse du moteur, inférieure à 6 % du courant nominal du variateur</li> <li>Contacteur aval ouvert</li> <li>Instabilité momentanée du courant du moteur</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez les connexions entre le variateur et le moteur</li> <li>Test sur un moteur à faible puissance ou sans moteur : en mode réglage usine, la détection de perte de phase du moteur est active <b>Perte phase moteur DPL</b> page 94 = <b>YES</b>. Pour vérifier le variateur dans un environnement de test ou de maintenance, sans avoir à utiliser un moteur de même puissance que le variateur, désactivez la détection de perte de phase du moteur <b>Perte phase moteur DPL = NO</b></li> <li>Vérifiez et optimisez les paramètres suivants : <b>Compensation RI (loi U/F) UFr</b> page 58, <b>Tension nom. mot UnS</b> page 57 et <b>Courant nom. mot. nCr</b> page 57 et effectuez un <b>Auto-réglage tUn</b> page 60.</li> </ul>
<b>OSF</b>	Surtension réseau	<ul style="list-style-type: none"> <li>Tension réseau trop élevée : <ul style="list-style-type: none"> <li>Uniquement à la mise sous tension du variateur, l'alimentation est supérieure de 10 % au niveau de tension maximal acceptable</li> <li>Mise sous tension sans ordre de marche 20 % au-dessus de l'alimentation réseau maximum</li> </ul> </li> <li>Alimentation réseau perturbée</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez la tension réseau</li> </ul>

Code	Nom	Causes possibles	Remède
<b>P H F</b>	Perte phase réseau	<ul style="list-style-type: none"> <li>Le variateur est mal alimenté ou un fusible a sauté</li> <li>Défaillance d'une phase</li> <li>ATV12 triphasé utilisé sur une alimentation secteur monophasée</li> <li>Charge déséquilibrée</li> <li>Cette protection ne fonctionne que si le variateur est en charge</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez le raccordement de puissance et les fusibles.</li> <li>Utilisez une alimentation secteur triphasée.</li> <li>Désactivez le défaut en sélectionnant <b>Perte phase réseau I P L</b> page <b>94</b> = <b>n 0</b>.</li> </ul>
<b>S C F 5</b>	Court circuit charge	<ul style="list-style-type: none"> <li>Court-circuit au niveau de la sortie du variateur</li> <li>Détection de court-circuit sur ordre de fonctionnement ou ordre d'injection DC si le paramètre <b>Tests IGBT S t r t</b> page <b>95</b> est réglé sur <b>Y E 5</b></li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez les câbles entre le variateur et le moteur et l'isolation du moteur</li> </ul>
<b>S L F 1</b>	Déf. com. Modbus net.	<ul style="list-style-type: none"> <li>Interruption des communications sur le réseau Modbus</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez les connexions du bus de communication.</li> <li>Vérifiez le time-out (paramètre <b>Time out Modbus t t 0</b> page <b>97</b>)</li> <li>Consulter le guide d'exploitation de Modbus.</li> </ul>
<b>S L F 2</b>	Déf. com SoMove	<ul style="list-style-type: none"> <li>Interruption de communication avec SoMove</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez le câble de connexion SoMove.</li> <li>Vérifiez le time-out</li> </ul>
<b>S L F 3</b>	Déf. com. HMI	<ul style="list-style-type: none"> <li>Interruption de communication avec le terminal externe</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez la connexion aux bornes.</li> </ul>
<b>S P I F</b>	Faute détecté Retour PI	<ul style="list-style-type: none"> <li>Retour PID inférieur à la limite basse</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez le retour de la fonction PID.</li> <li>Vérifiez le seuil et la temporisation de la supervision du retour PID page <b>76</b>.</li> </ul>
<b>L U L F</b>	Déf. surcharge process	<ul style="list-style-type: none"> <li>Sous-charge du process</li> <li>Courant du moteur inférieur au paramètre <b>S.couple Fréq.nulle L U L</b> page <b>55</b> pendant une période réglée par le paramètre <b>Tps.Dét.Souscharge U L t</b> page <b>55</b> pour protéger l'application.</li> </ul>	<ul style="list-style-type: none"> <li>Assurez-vous que le process et les paramètres du variateur sont en phase</li> </ul>
<b>t J F</b>	Surchauffe IGBT	<ul style="list-style-type: none"> <li>Surchauffe du variateur</li> <li>La température interne de l'IGBT est trop élevée par rapport à la température ambiante et à la charge</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez la taille de la charge / du moteur / du variateur.</li> <li>Réduisez la <b>Fréquence découp. S F r</b> page <b>59</b>.</li> <li>Laissez le temps au variateur de refroidir avant de le redémarrer.</li> </ul>

## Defauts disparaissent dès l'élimination des causes

Les erreurs qui apparaissent dans la figure ci-dessous sont supprimés directement après l'élimination de leurs causes :

Code	Nom	Causes possibles	Remède
<b>CF F</b>	Config. incorrecte	<ul style="list-style-type: none"> <li>Bloc contrôle remplacé par un bloc contrôle configuré sur un variateur ayant une puissance nominale différente</li> <li>La configuration actuelle des paramètres par l'utilisateur n'est pas cohérente.</li> </ul>	<ul style="list-style-type: none"> <li>Faites un réglage usine ou récupérez la configuration sauvegardée, si elle est valide.</li> <li>Si le défaut persiste après avoir fait un réglage usine, contactez votre représentant local Schneider Electric.</li> </ul>
<b>CF I</b> (1)	Config. invalide	<ul style="list-style-type: none"> <li>Configuration invalide</li> <li>La configuration chargée sur le variateur à l'aide du bus ou du réseau de communication est incohérente. Le chargement de la configuration a été interrompu ou n'est pas totalement terminé.</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez la configuration chargée précédemment.</li> <li>Chargez une configuration compatible.</li> </ul>
<b>CF I2</b>	Téléchargement config. invalide	<ul style="list-style-type: none"> <li>Interruption du téléchargement avec Loader ou SoMove</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez la connexion avec un Loader ou SoMove.</li> <li>Pour réinitialiser le défaut, redémarrez le téléchargement ou refaites un réglage usine.</li> </ul>
<b>US F</b>	Sous-tension	<ul style="list-style-type: none"> <li>Alimentation secteur insuffisante</li> <li>Baisse de tension passagère</li> </ul>	<ul style="list-style-type: none"> <li>Vérifiez la tension et les paramètres du menu <a href="#">Gestion sous-tension US b</a> - page <a href="#">95</a>.</li> </ul>

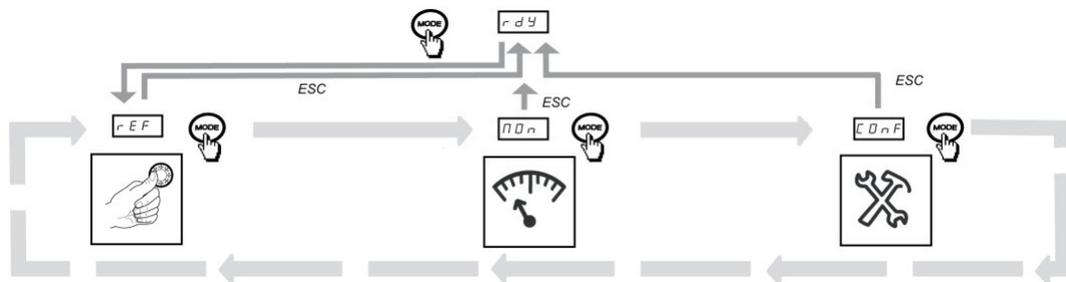
## Codes d'erreurs en cas de détection d'un défaut qui s'affiche sur le terminal déporté

Code	Nom	Description
<b>In I E</b>	Lors de son initialisation propre	<ul style="list-style-type: none"> <li>Initialisation du microcontrôleur</li> <li>Recherche de configuration des communications</li> </ul>
<b>CO N. E</b> (1)	Erreur de communication	<ul style="list-style-type: none"> <li>Il s'agit d'une erreur de time-out de 50 ms.</li> <li>Ce message s'affiche après 220 tentatives.</li> </ul>
<b>A - I 7</b> (1)	Alarme touche	<ul style="list-style-type: none"> <li>Une touche a été enfoncée en continu pendant plus de 10 secondes.</li> <li>Interrupteur membrane déconnecté.</li> <li>Le clavier émet une alarme lorsqu'une touche est enfoncée.</li> </ul>
<b>c L r</b> (1)	Confirmer remise à zéro de défaut	<ul style="list-style-type: none"> <li>Ce message s'affiche si la touche STOP est enfoncée alors qu'il y a un défaut du clavier.</li> </ul>
<b>d E U. E</b> (1)	Incompatibilité variateur / console	<ul style="list-style-type: none"> <li>Le type (marque) de variateur n'est pas compatible avec le type de console (marque)</li> </ul>
<b>r O N. E</b> (1)	Anomalie ROM	<ul style="list-style-type: none"> <li>Anomalie de la ROM du clavier détectée par le calcul du checksum.</li> </ul>
<b>r A N. E</b> (1)	Anomalie RAM	<ul style="list-style-type: none"> <li>Anomalie de la RAM du clavier détectée.</li> </ul>
<b>C P U. E</b> (1)	Autre défaut	<ul style="list-style-type: none"> <li>Un autre défaut détecté.</li> </ul>

## Configuration générale du variateur

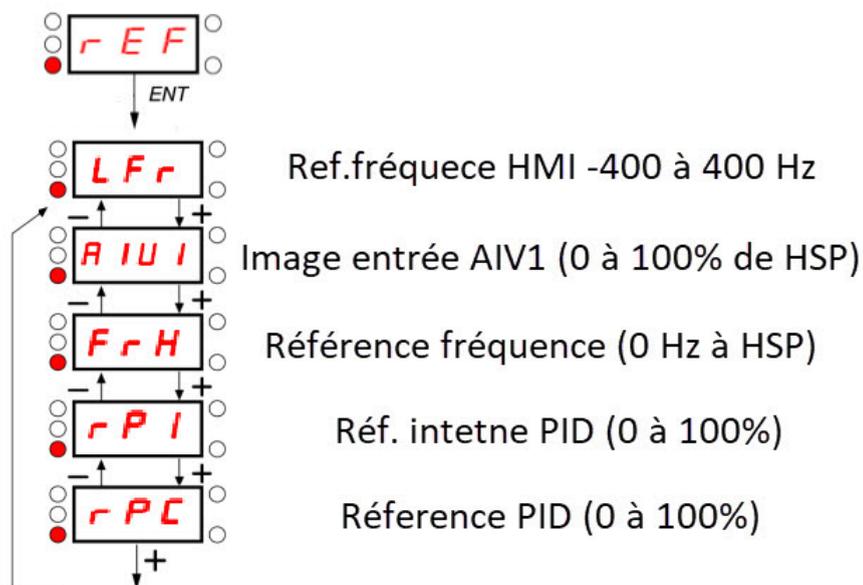
Il existe beaucoup de paramètres à configurer dans le variateur, mais on peut se contenter de configurer seulement les paramètres essentiels.

Quand le variateur est bien câblé, on se trouve avec le message ready dans l'HMI, puis on peut changer dans les paramètres en choisissant l'une de ces rubriques :



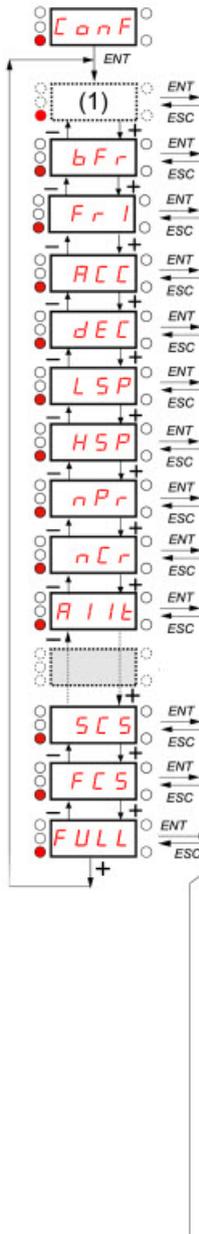
### Configuration reF

Pour la configuration de tout ce qui est en relation avec la référence, on choisit le mode *reF*, et on trouve les sous-rubriques suivantes :



### Configuration reF

Pour la configuration de tout ce qui est en relation avec les paramètres, on choisit le mode *reF*, et on trouve les sous-rubriques suivantes :



**Standard fréq. moteur (50 ou 60 Hz)**

Canal réf.1 (Bornier, Comm HMI, Modbus, Bouton de navigation)

Accélération (0.0s à 999.9s)

Décélération (0.0s à 999.9s)

Petite vitesse (0 Hz à HSP)

grande vitesse (LSP à tFr (Hz))

Puissance nominal. mot (NCV -5 à NCV +2)

Courant nominal. mot (0.25 à 1.5 In(variat))

Types d'E Analog (Tension ou courant)

Sauvegarde de confi actuelle (nO ou StrI)

Rappel de confi actuel (nO ou StrI)

**Macro conf (3 conf préprog proposées)**

2-3 fils + Confi d'E analog + Confi Sor Relais + prot contr surch +  
Confi Sor analog

(fréq + puiss + Cos + U + I + vitess) nom du mot + loi (U/f) +  
comp.glissem + Réd bruit

Menu de commande (Réf + forçage...)

Menu rampe (ACC+DEC+Forme+Mode d'arrêt+M Arrière+  
Vitess prés)

Menu Ges defaults (Seuil surch+temp avant redemmarage)

**Annexe D : TIA PORTAL, logiciel de  
programmation des API/IHM  
SIEMENS**

## Le logiciel Somove

### Introduction

TIA Portal (Totally Integrated Automation Portal ), en français Portail d'automatisation totalement intégré, est la nouvelle plateforme de SIEMENS qui englobe deux de ses logiciels : SIMATIC STEP 7 V11 et SIMATIC WinCC V11 pour les solutions d'automatisation et pour faciliter l'environnement de travail avec une nouvelle interface et de nouvelles fonctionnalités. Le logiciel optimise l'ensemble des procédures au niveau planification, machine et processus, son interface utilisateur intuitive, ses fonctions simples et sa transparence totale des données le rendent extrêmement convivial. Des données et projets déjà existants peuvent être intégrés aisément, ce qui garantit la sécurité de l'investissement.

### SIMATIC STEP 7 TIA Portal

#### Présentation générale

SIMATIC STEP 7 est le logiciel de programmation le plus connu au monde et le plus utilisé dans le domaine de l'automatisation industrielle. SIMATIC STEP 7 Version (TIA Portal) ingénierie efficace dans le TIA Portal pour contrôleurs éprouvés et nouveaux contrôleurs SIMATIC. Parmi ses diverses applications on trouve [12] :

- Programmation API : Configuration et programmation des contrôleurs SIMATIC S7-1200, S7-300, S7-400 et du nouvel S7-1500.
- Configuration des appareils et du réseau pour toute la chaîne d'automatisation.
- Diagnostic en ligne pour l'ensemble du projet.
- Motion et Technologie pour les fonctionnalités Motion intégrées.
- Simulation et visualisation du projet.

## Création d'un projet dans TIA Portal

Une fois le logiciel est installé, et après l'ouverture de ce dernier, la première page qui s'affiche montre tous les projets déjà créés, et si on veut importer un projet qu'on a réalisé ailleurs, on clique sur "Browse" et puis on le sélectionne, le logiciel nous donne la possibilité de créer un autre en cliquant sur "Create new project".

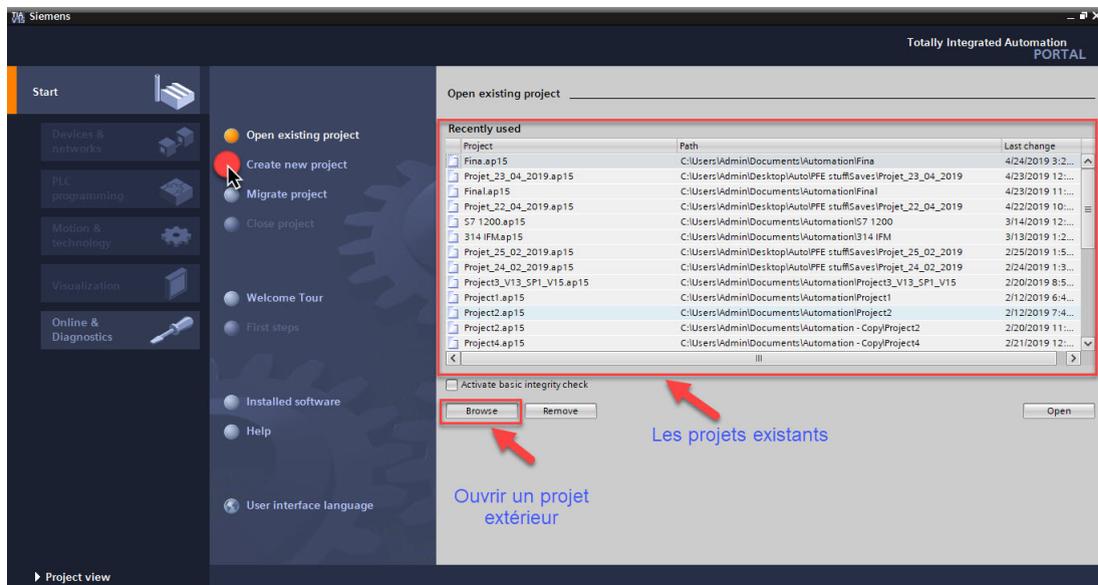


FIGURE IV.37: Ouverture d'un projet dans TIA Portal

La nouvelle vue qui s'affiche nous montre le nom du nouveau projet et le chemin de l'enregistrement, et on peut changer ces deux derniers en modifiant chacun dans le champ correspondant.

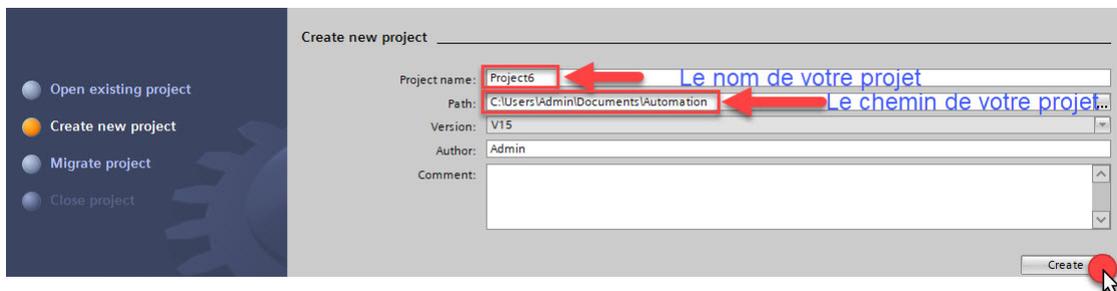


FIGURE IV.38: Création d'un projet dans TIA Portal

## Configuration matériel du projet

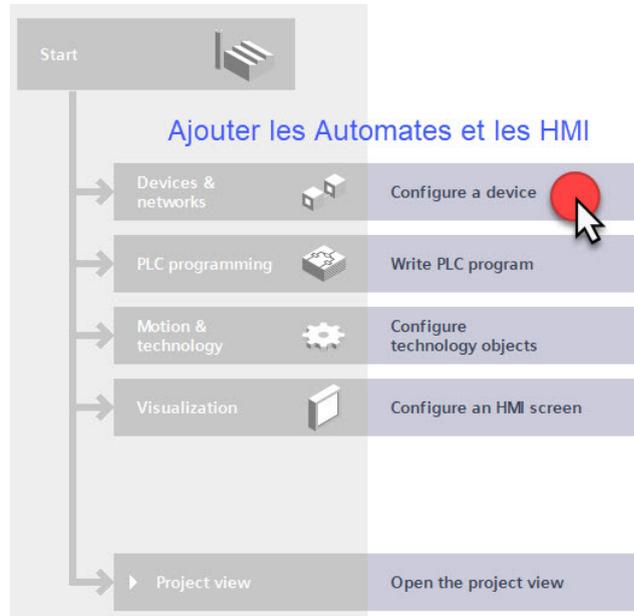


FIGURE IV.39: Les étapes d'un projet d'automatisation dans TIA Portal

Après la sélection de nom du projet avec son chemin, on passe maintenant à la configuration matériel en cliquant sur “Configure a device”, la nouvelle vue nous montre tous les appareils qu'on peut ajouter à notre projet, que ce soit un Automate programmable (PLC) ou une HMI, donc premièrement, on clique sur 1 “Add new device” :

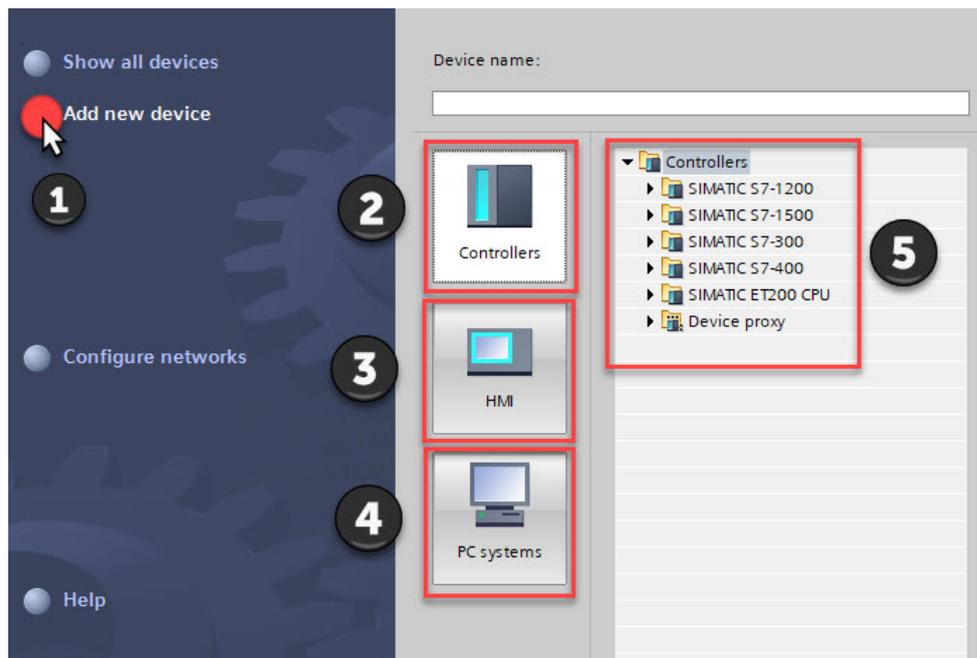


FIGURE IV.40: Configuration et ajout du matériel dans un projet TIA Portal

Le numéro	La signification
2	<p>“Controllers” signifie tous les automates programmables disponible dans TIA Portal.</p> <p>Sur 5, on voit ces PLCs, et leur familles.</p> <p>Les automates disponibles dans TIA Portal sont : S7-1200 et S7-1500 et S7-300 et S7-400.</p> <p>Il ya des modules qu’on peut pas trouver dans TIA Portal et qui se trouve dans SIMATIC Step 7.</p>
3	<p>Les Interfaces Homme-Machine HMI :</p> <p>on peut choisir notre HMI en cliquant sur 2 et puis on cherche la référence du produit souhaité.</p>
4	<p>Signifie le micro-ordinateur qu’on peut utiliser comme HMI et non pas les HMI ordinaires, et en général on les trouve dans les salles de control.</p>
5	<p>Les API disponibles dans TIA Portal</p>

TABLE IV.9: explication de la configuration du matériel

**Le choix d’un PLC** On prend un exemple pour le choix d’un automate programmable dans TIA Portal, l’automate cherché est le suivant : SIMATIC s7-300 avec la référence suivante :6ES 314-1AG13-0AB0.

Et pour le faire, on clique sur “controllers” et on choisit la famille S7-300 et on ouvre les CPU-314 et on trouve notre PLC :

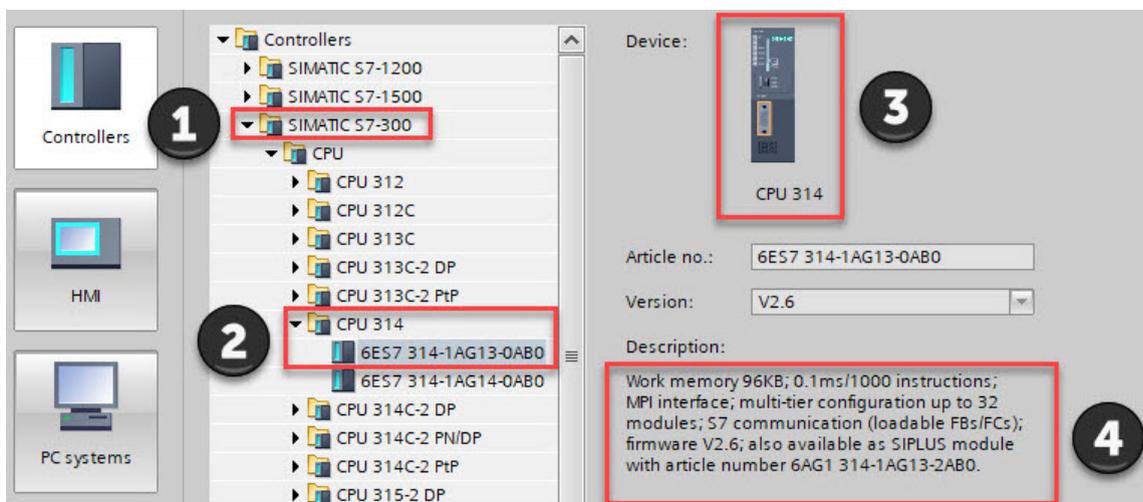


FIGURE IV.41: Choix des API et des CPU

Le numéro 4 nous montre toutes les spécifications de notre PLC, avec une vue sur l'automate "3"

## Vue principale de l'environnement de travail

La vue principale de l'environnement de travail de TIA Portal aide à bien organiser le projet et de trouver les outils nécessaires facilement.

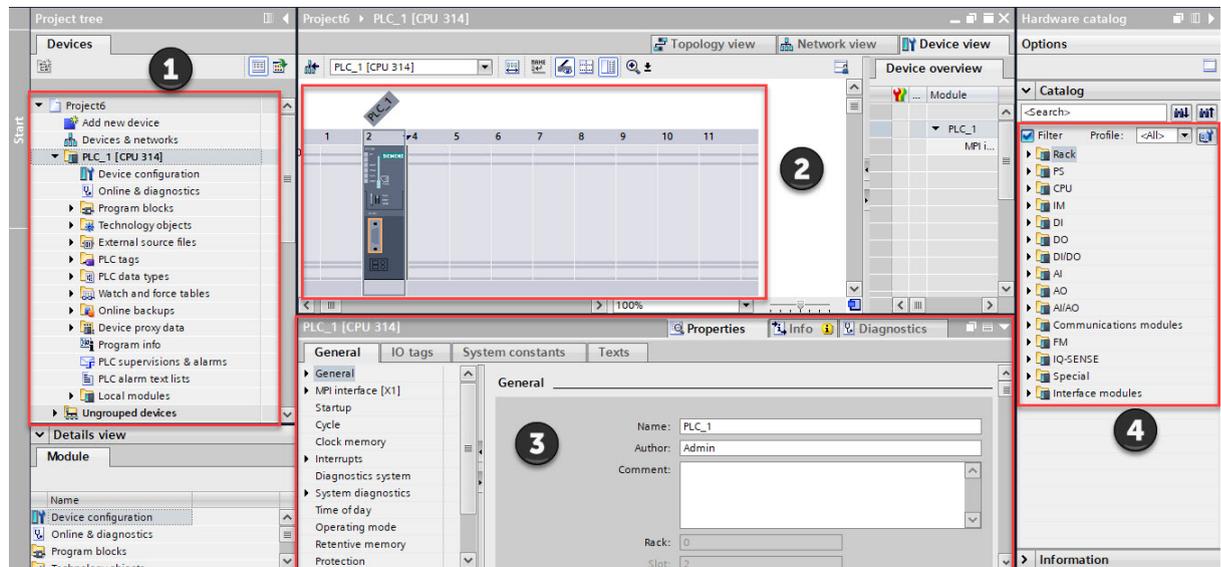


FIGURE IV.42: Environnement de travail dans TIA Portal

Comme illustré dans la figure, il y a différentes rubriques dans la vue :

Le numéro	La signification
1	<b>1</b> contient tout les fichiers du projet : Le main OB1, les fonctions créées, les mnémoniques, les fichiers externes, ainsi que les fichiers du HMI qu'on va bien détailler dans le cinquième chapitre.
2	<b>2</b> montre ceux qui sont déjà met dans le rack (PS, PLC, IOs)
3	<b>3</b> nous montre les propriétés des éléments sélectionnés
4	<b>4</b> les éléments disponibles pour les ajouter dans le rack

TABLE IV.10: expliquer de l'environnement de travail dans TIA Portal

## Ajout des éléments au rack

Pour ce petit projet, on a besoin des éléments de base pour assurer le bon fonctionnement du projet pour cela, on choisit :

- Une boîte d'alimentation de 24V, 5A.
- Un module d'E/S TOR de 8DI et 8DO.
- Un module d'E/S analogique de 4AI et 2AO.

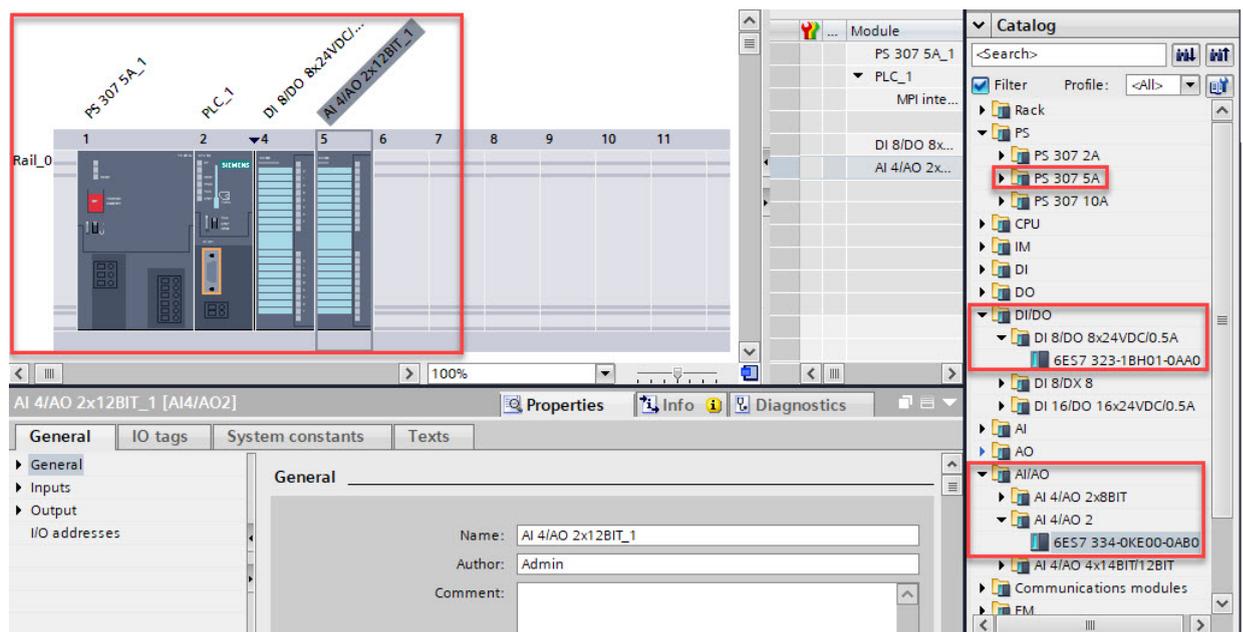


FIGURE IV.43: Ajout des modules et des éléments sur le rack

## Configuration des modules d'E/S

Une fois les modules sont choisis, on doit les configurer selon nos entrées et sorties physiques, et pour le faire, on sélectionne le module qu'on veut modifier, et on appuis sur "IO tags" :

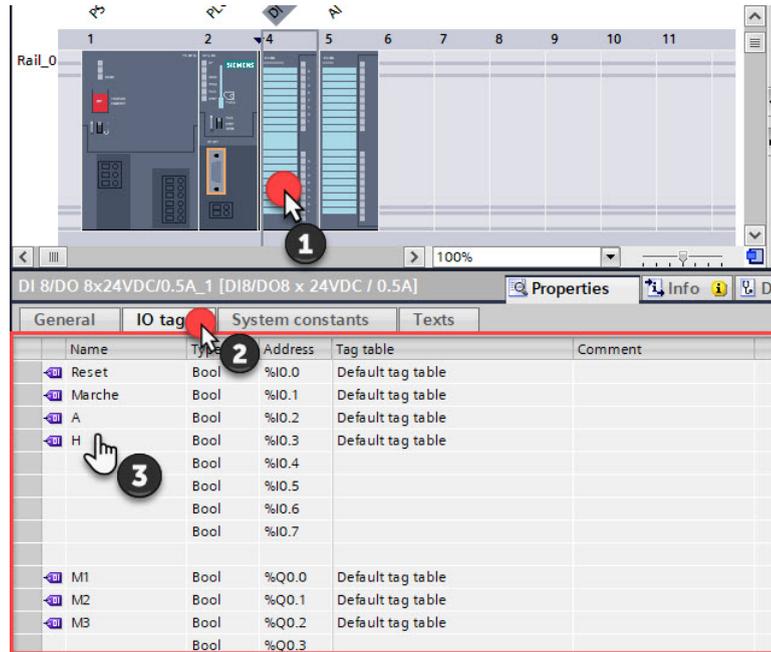


FIGURE IV.44: Configuration des modules E/S

Et on change ce qu'on veut, et pour cet exemple, on a comme entrées : "Reset", "Marche", "A" et "H". et comme sortie : 'M1', 'M2' et 'M3'.

## Table des mnémoniques

La configuration de la mémoire interne de l'automate se fait ailleurs avec les mnémoniques, et pour y accéder on suit le chemin suivant :

$$PLC \Rightarrow PLC_{tags} \Rightarrow Show\ all\ tags \quad (IV.11)$$

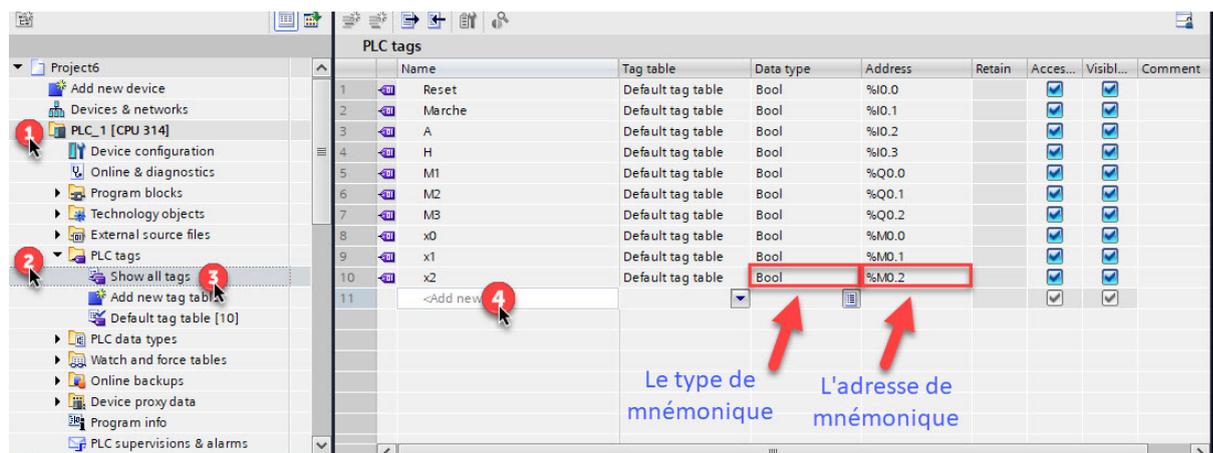


FIGURE IV.45: Création des mnémoniques et édition d'adressage

On choisit le nom de la variable, son type et son address sur la mémoire. Par exemple pour l'état 3 : X3 a comme type : booléen et comme adresse : M1.2 (le bit 2 dans le byte 1).

On peut aussi changer les entrées qu'on a déjà configurées auparavant.

## Programme principal Main[OB1]

Le programme main pour un automate SIEMENS est dans OB1, où on doit mettre notre programme principale :

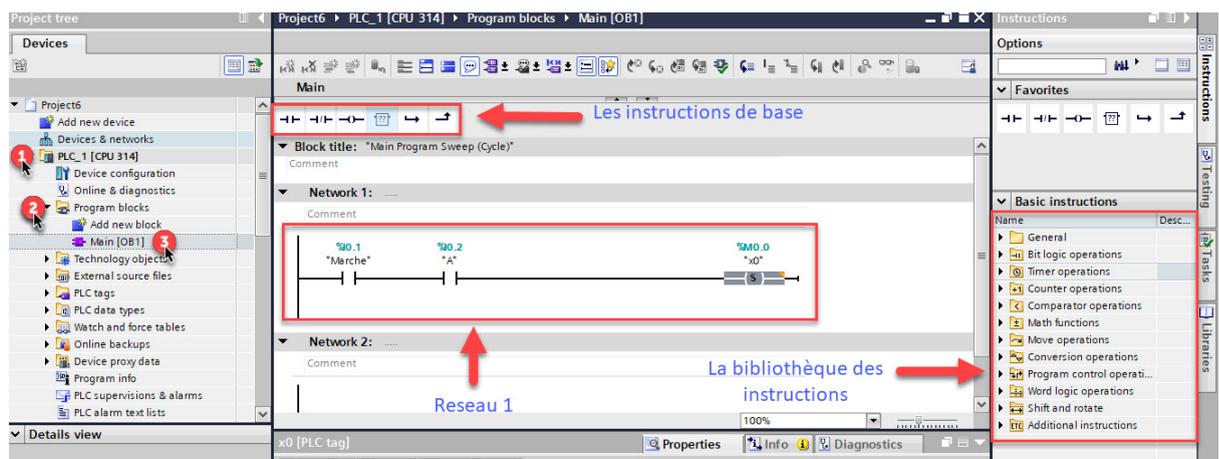


FIGURE IV.46: Environnement de travail de TIA Portal

Pour y accéder, on sélectionne Main[OB1] dans :

$$\boxed{PLC \implies Program\ Blocks} \quad (IV.12)$$

On trouve les réseaux de programmation où on traduit nos équations établies à la base du cahier des charges en Ladder par exemple.

A droite, on trouve tous les instructions et les fonctions prédéfinies qui nous aident à compléter rapidement notre programme.

Les instructions de base on les trouve au dessus du programme, et on peut aussi ajouter les instructions qu'on utilise plus fréquemment.

## Simulation des programmes dans TIA Portal

Une fois le programme finalisé, on peut vérifier si ce dernier fonctionne correctement avant de le transférer dans l'automate, et pour cela on lance **la simulation**

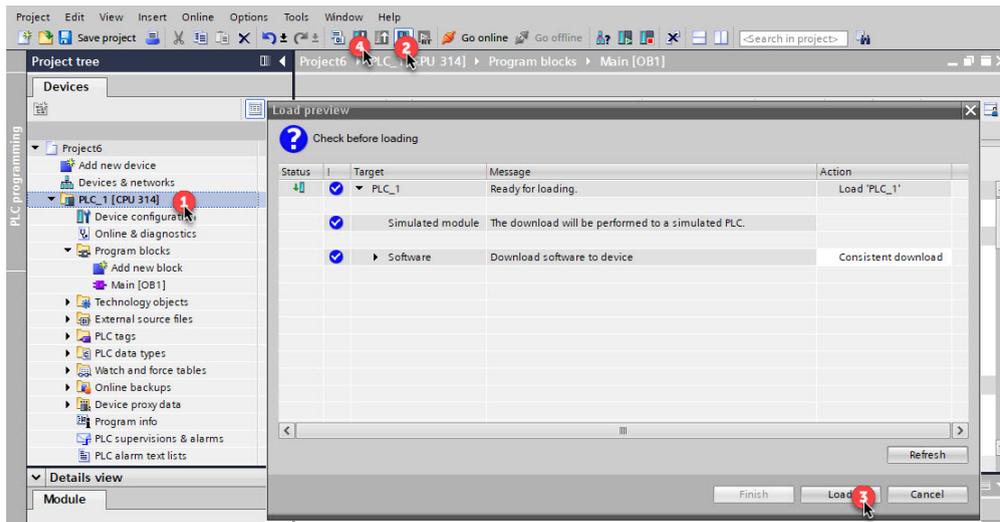


FIGURE IV.47: Simulation du projet dans TIA Portal

Pour simuler le programme pour la première fois, on doit charger tout le matériel qu'on a utilisé, donc on sélectionne l'automate (PLC\_1) et on appuie sur le bouton (voir figure III.11), et après on appuis sur "Load" pour charger le programme. Si après la simulation, on trouve que le programme nécessite un changement, on doit d'abord arrêter la simulation en appuyant sur le bouton "Stop", après on peut effectuer des modifications et puis recharger une nouvelle fois.

## L'interface de simulation

SIEMENS propose une interface simple pour simuler les programmes :

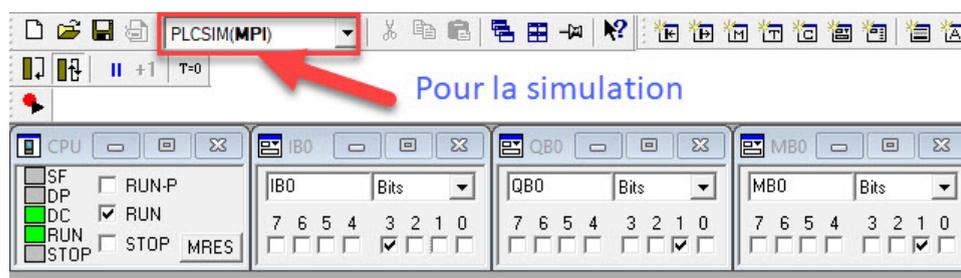


FIGURE IV.48: Interface du simulateur de TIA Portal

D'abord on choisit l'interface de communication entre le simulateur et l'ordinateur (PLCSIM), et après on lance la simulation en appuyant sur "RUN".

On trouve que l'interface est facile à manipuler intuitivement. IB0 représente les 8 entrées TOR du Byte 0, et QB0 représente les 8 sorties TOR du Byte 0, et MB0 représente les 8 bits du Byte 0 de la mémoire de l'automate.

Donc on manipule les entrées et la mémoire si nécessaire, et on observe le changement dans les sorties ou le changement dans la mémoire qu'on a programmé pour qu'elle soit changée.

Afin de voir exactement ce qui se passe dans les réseaux pour faire un bon diagnostic du programme, on peut visualiser le changement des variables en temps réel, en appuyant sur "Go online" et après sur le raccourci (2).

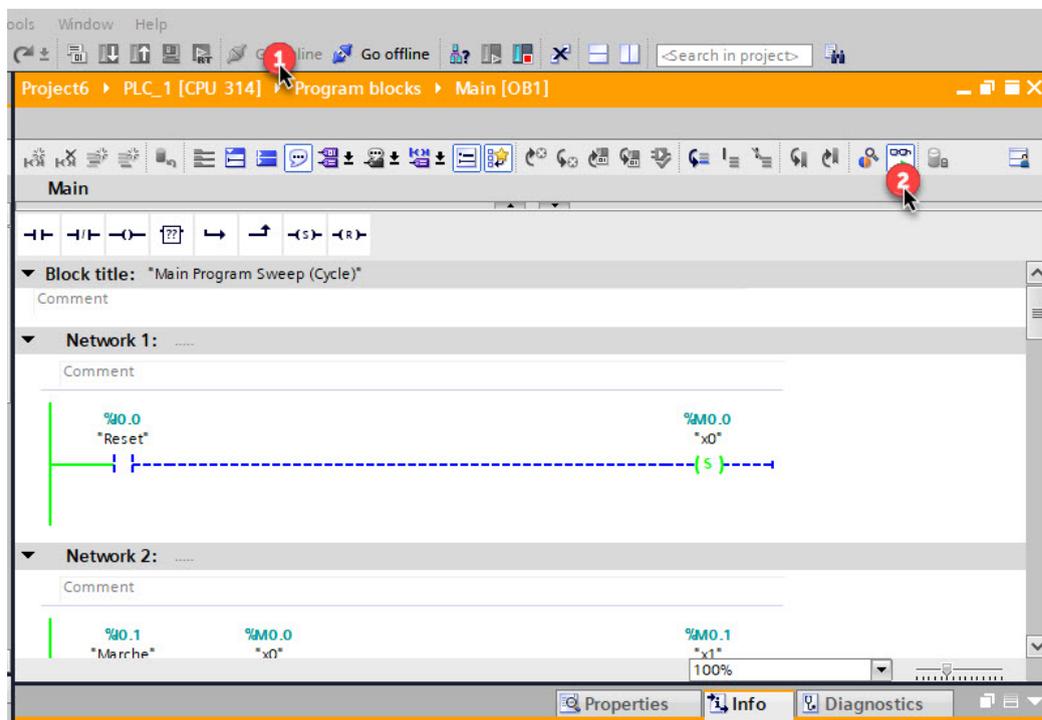


FIGURE IV.49: Visualisation des état lors de l'exécution du projet

## Usage des fonctions dans TIA Portal

Comme pour tous les programmes, il doit y avoir des fonctions qui nous aident à avancer plus rapidement et à éviter la redondance, et pour cela on appuis sur le bouton droit de la souris sur "Program Block"s et on choisit "Add new Block" :

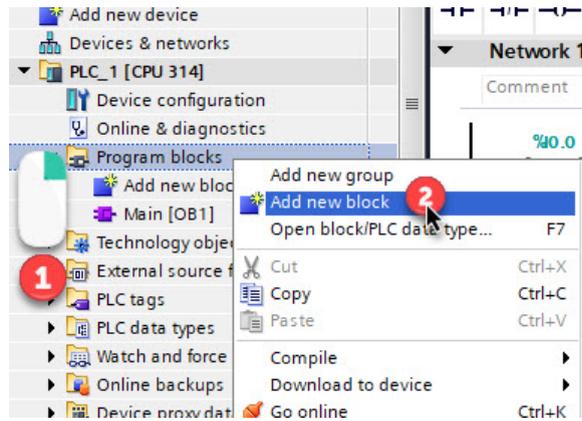


FIGURE IV.50: Ajout des blocs dans TIA Portal

Et on choisit le type de bloc qu'on souhaite :

- **Organisation Block** : Pour les interruptions.
- **Function Block** : Pour un bloc de fonctions.
- **Function** : pour créer une fonction.
- **Data Block** : Pour un bloc de donnée.

Pour notre exemple, on choisit un bloc de fonctions avec SCL comme langage de programmation :

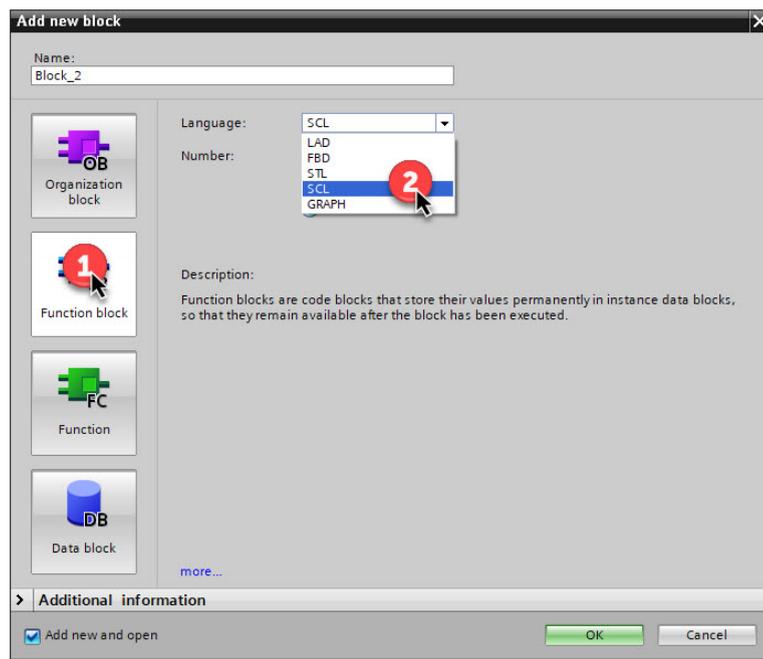


FIGURE IV.51: Configuration des langages de programmation des blocs de fonctions

Une fois le bloc choisi, on procède à la déclaration des variables (entrées, sorties, les entrées sorties) et leurs types (Bool, Real,...).  
 et en dessous, on écrit notre programme :

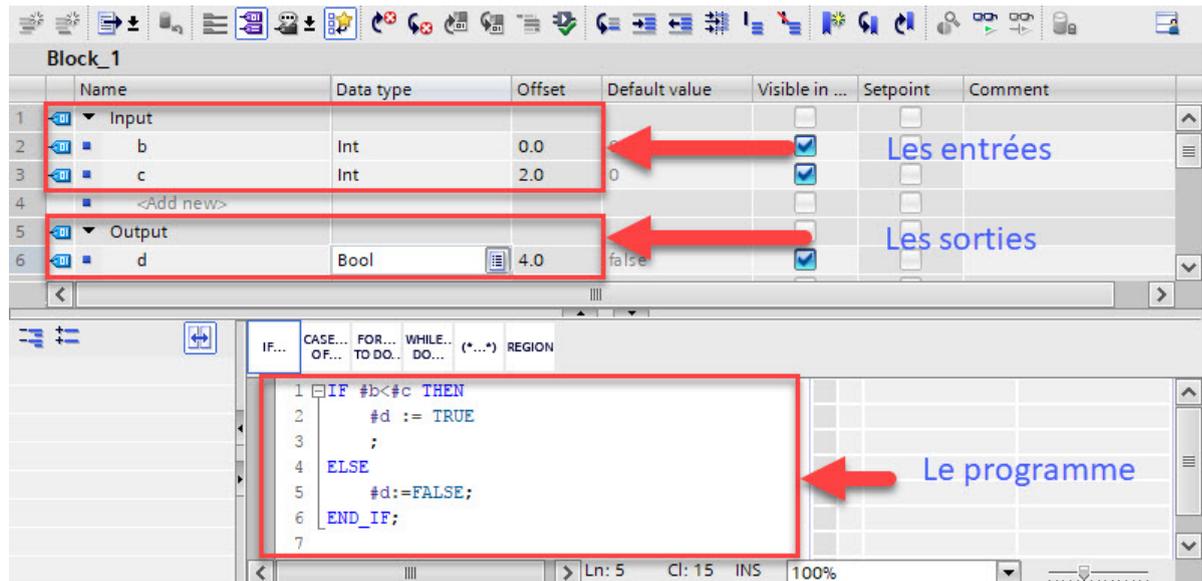


FIGURE IV.52: Programmation d'un bloc de fonction

Cet exemple de programme envoie la réponse de la question : est ce que “b” est inférieur à “c”, et ces derniers sont des nombres naturels.

## Utilisation des fonctions dans le programme

Une fois les fonctions sont créées et compilées, on peut les appeler dans le main en cas de besoin :

le bloc de fonctions nécessite un bloc de données, et on définit les entrées sorties de ce bloc de fonction :

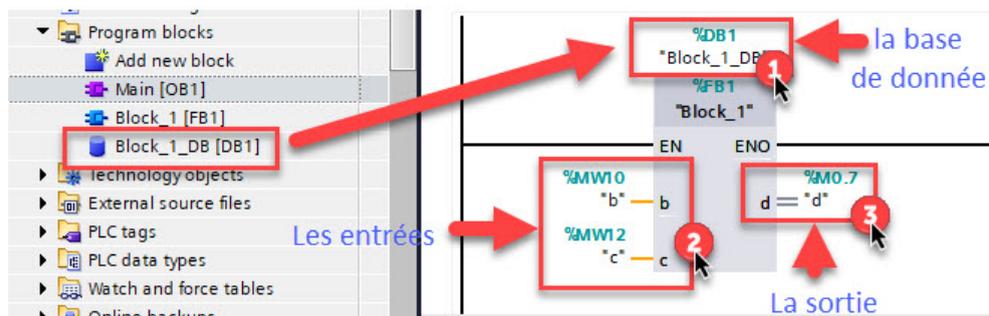


FIGURE IV.53: Insertion d'une instance de bloc de fonction dans le programme principale

## SIMATIC WinCC TIA Portal

### présentation générale

Le SIMATIC WinCC dans le Totally Integrated Automation Portal (TIA Portal) fait partie d'un nouveau concept d'ingénierie intégré qui offre un environnement d'ingénierie homogène pour la programmation et la configuration de solutions de commande, de visualisation et d'entraînement.

WinCC dans le TIA Portal est le logiciel pour toutes les applications IHM allant de solutions de commande simples avec des Basic Panels aux applications SCADA pour systèmes multipostes basés sur PC. La gamme de solutions offerte par le prédécesseur de SIMATIC WinCC flexible s'en trouve considérablement élargie.

### Création d'une IHM dans un projet

Pour pouvoir visualiser le process d'un projet dans le but d'avoir une meilleur compréhension ou même d'un point de vue de contrôle, on ajoute à notre projet une IHM de la manière suivante :

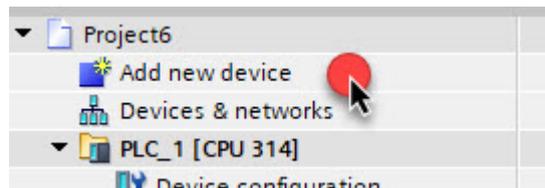


FIGURE IV.54: Ajout d'une IHM dans TIA Portal

On appuie sur "Add new device" et ensuite, on selectionne "IHM" dans le tableau qui apparaît, et on cherche notre IHM dans la liste donnée :

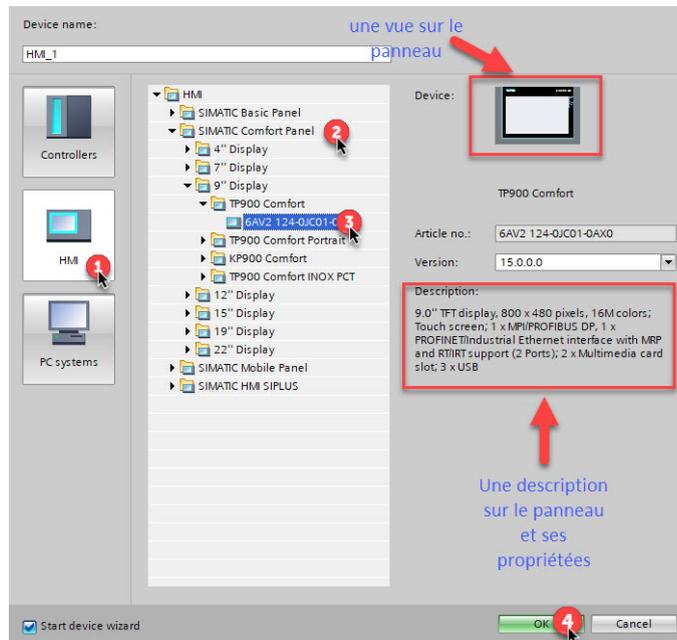


FIGURE IV.55: Choix d'une IHM dans un projet TIA Portal

On sélectionne TP900 Comfort, la même utilisée dans notre projet de fin d'études. On a la possibilité d'introduire un système PC, dans le cas où l'IHM n'est pas disponible, donc un micro ordinateur peut remplacer et effectuer les fonctionnalités d'un IHM..

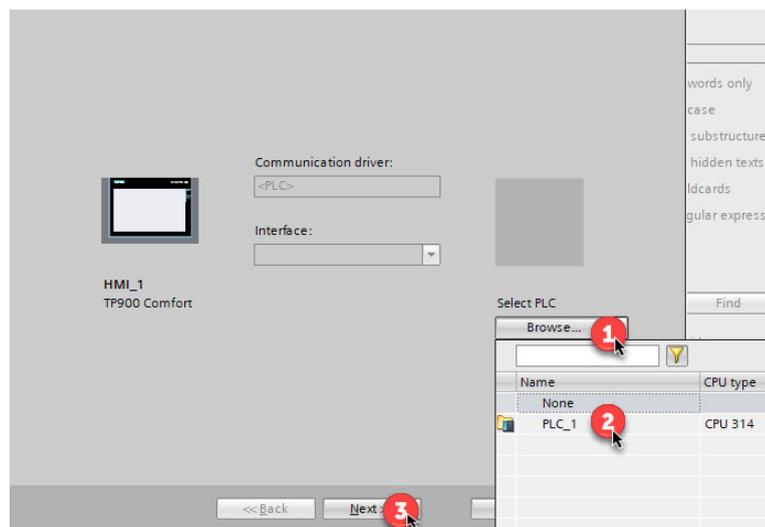


FIGURE IV.56: Configuration d'une IHM dans TIA Portal

Une fois le choix est fait, on passe à la configuration de cette IHM, on crée un réseau de communication entre l'IHM et l'API, dans ce cas, on a un seul automate disponible pour créer un réseau (PLC\_1), ensuite on appuis sur "Next".

Maintenant, il nous reste seulement de compléter la configuration de base de notre IHM.

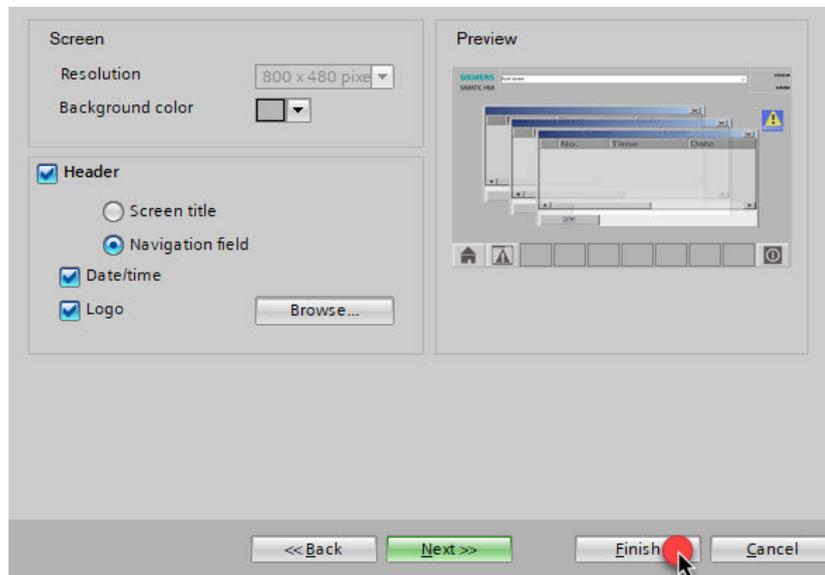


FIGURE IV.57: Configuration d'une IHM dans TIA Portal\_2

## Vue principale du projet et table des mnémoniques

La vue principale du projet ressemble à celle de l'automate (Step7) et elle est composée d'un illustrateur de projet contenant tous les fichiers de ce dernier, et de la vue qu'on est en train de travailler sur.

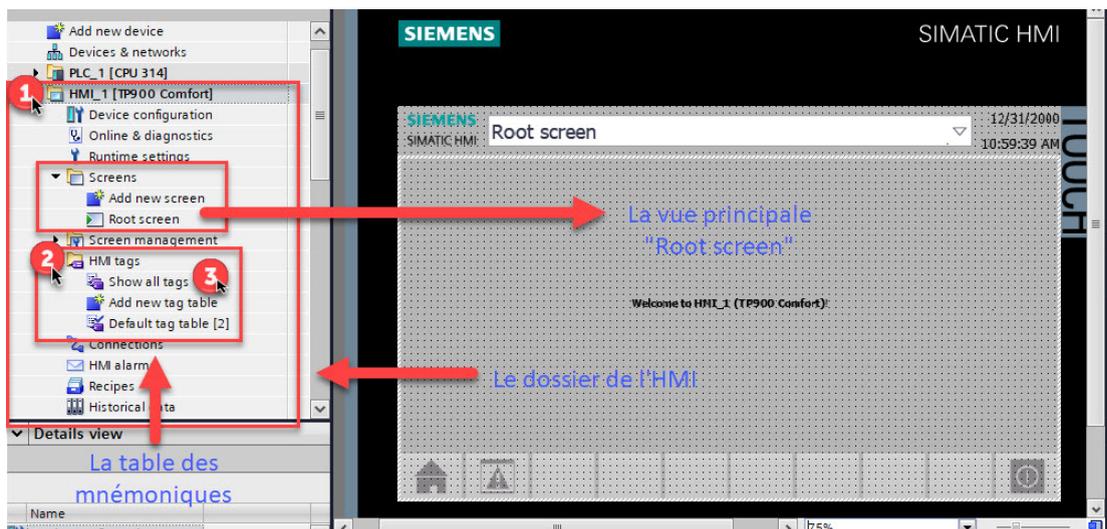


FIGURE IV.58: vue principale sur l'environnement de travail

Pour accéder à la table des mnémoniques, on ouvre le dossier “HMI tags” et on appuie sur “Show all tags”.

On importe toutes les variables nécessaire pour notre projet depuis l’automate, et pour le faire, on donne un nom à notre variable, pas forcément le même nom déclaré dans l’automate, et on le sélectionne pour que l’IHM puisse reconnaître leurs emplacements dans la mémoire de l’automate.

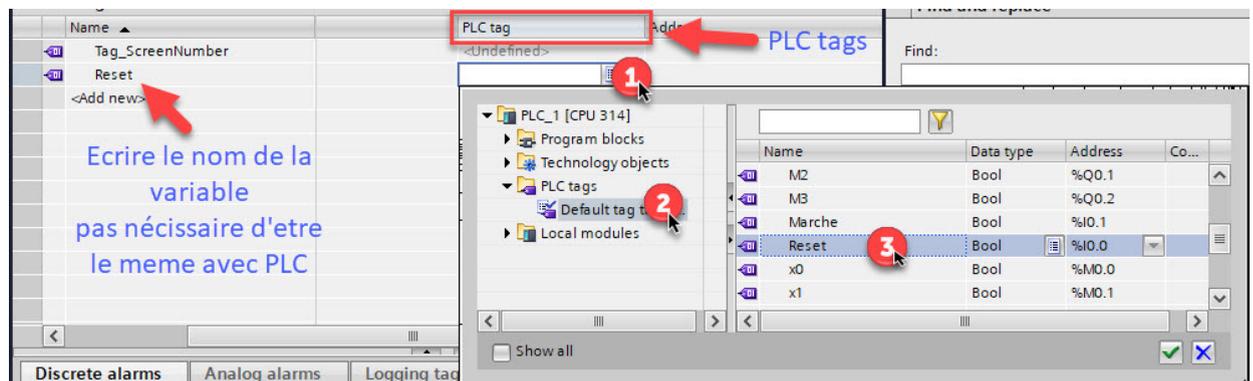


FIGURE IV.59: Table des variables

On doit d’abord établir la connexion entre l’automate et l’IHM si on la pas fait dans sa création.

## Les objets dans Simatic WinCC et la création d’un bouton

Le principe de travail sur WinCC est d’établir des fonctions aux objets prédéfinis dans le logiciel, et on peut distinguer deux types d’objet :

- **Basic Objets** : Destinés construire des fonctions de base comme le dessin des formes géométriques comme, un cercle par exemple ou pour créer un texte dans une vue..
- **Éléments de control** : Ce logiciel nous propose une gamme d’éléments pour faciliter le travail, chaque élément possède une fonction prédéfinie par le constructeur, et prêt à être utilisé dans notre projet.
- **Control** : Cette rubrique contient des éléments qui aident à tracer des graphes ou à afficher des alarme, ou même ouvrir un fichier pdf ou exposer une séquence de vidéo.

- **Bibliothèque :** La bibliothèque globale contient davantage d'éléments avec plus de détail (plusieurs boutons) pour qu'on puisse choisir les plus convenables selon nos besoins.

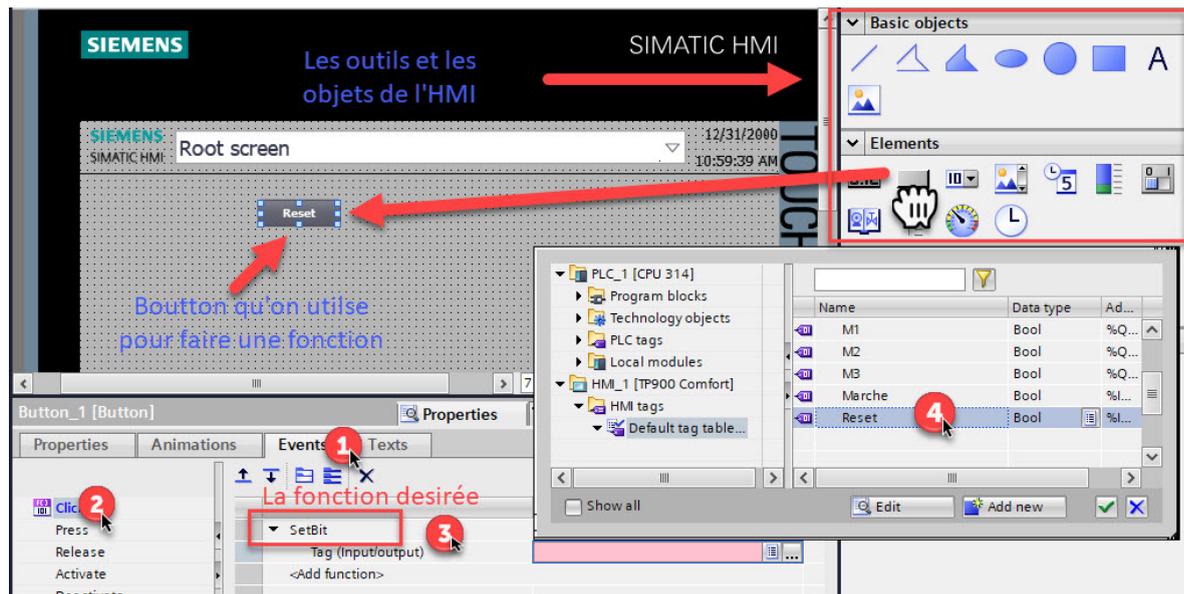


FIGURE IV.60: Création des objets

Maintenant pour créer un bouton qui possède une fonction, on glisse le bouton qui se trouve dans la rubrique des éléments, puis on appuis sur “Événements” dans propriétés, et selon notre besoin on choisit le type et le moment de exécution de l'événement.

Tia portal nous propose plusieurs fonctions préprogrammées qu'on peut utiliser, et de même Tia portal nous donne la possibilité de créer nos propre fonctions par des scripts et les exécuter en cas de besoin.

Dans ce cas là, on a choisi la mise à 1 comme fonction qui s'exécute une fois on appuis sur le bouton, et ensuite on choisit les variables attribuées à cette fonction.

## Création d'un voyant

Le travail est similaire pour la création d'un voyant, mais cette fois-ci, on a pas directement un voyant préprogrammé, donc on doit dessiner un cercle et on introduit des fonctions pour l'utiliser comme un voyant.

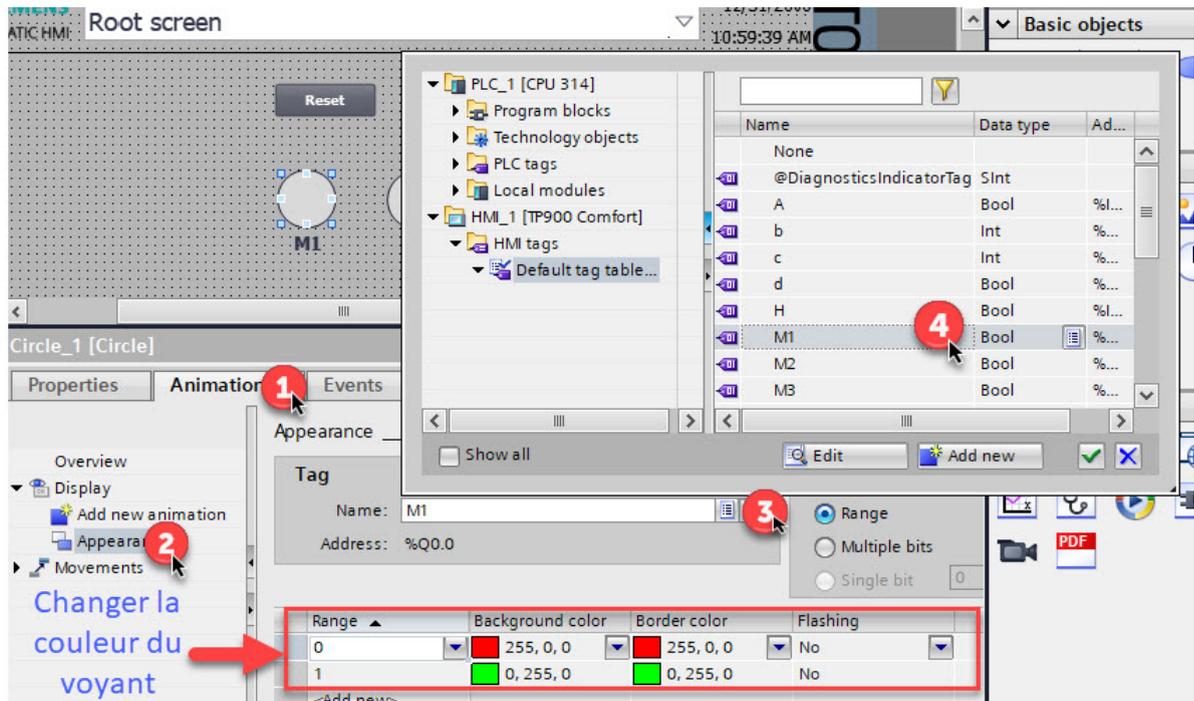


FIGURE IV.61: Affectation des variables

Après avoir dessiné le cercle, on choisit animation dans ses propriétés, ensuite on appuis sur “Apparence”, dans notre cas, on veut avoir un voyant qui s’allume en rouge en cas de l’arrêt du moteur 1 et en vert quand il est en marche.

Donc on choisit la variable attachée au moteur 1 “M1” et on crée deux pages 0 et 1 avec les couleurs correspondantes.

## Simulation de l'IHM

On peut tester notre projet IHM dans TIA Portal avant de l'implémenter sur la vraie IHM, en procédant de la même façon qu'on a fait pour la simulation de l'automate, mais cette fois-ci on sélectionne l'IHM et l'automate pour la simulation.

Après le lancement de la simulation, on teste notre IHM, si le programme fonctionne de la façon désirée on procède à l'implémentation, sinon on a toujours la possibilité d'effectuer des modifications sur l'IHM afin de l'améliorer.

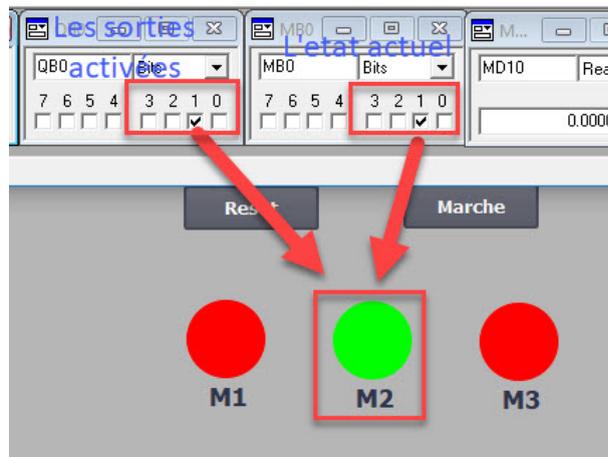


FIGURE IV.62: Simulation du projet

## Conclusion

Ce chapitre regroupe les fonctionnalités essentielles du logiciel de programmation de SIEMENS le TIA Portal, utilisé pour la configuration matériel, la programmation en différents langages, la simulation des programmes ainsi que la création et paramétrage des interfaces HMI pour la visualisation des process.

# Annexe E : Logiciel de configuration des variateurs "Somove"

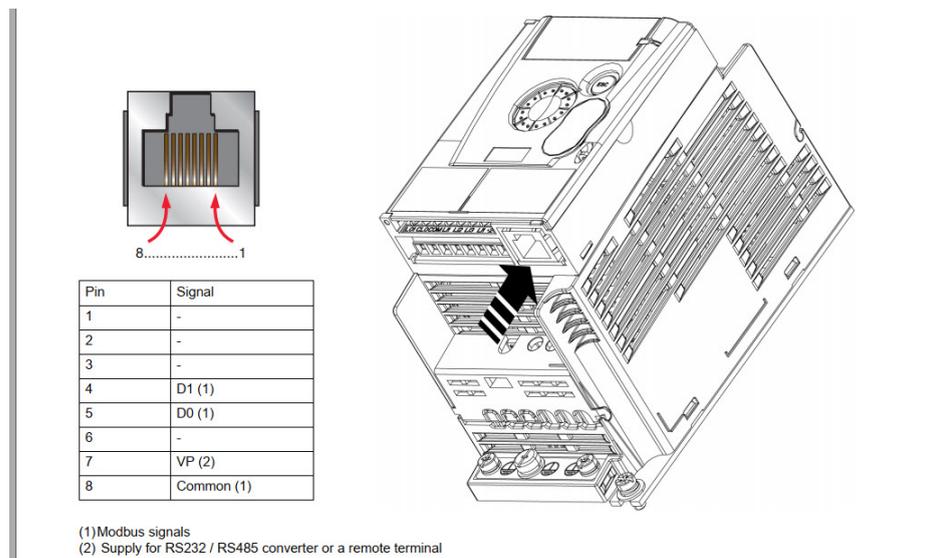
## Le logiciel Somove

### Description du logiciel

Ce logiciel appartient à *SchneiderElectric*, il sert à la configurer les variateurs de vitesse industriels, la tâche de la configuration est simplifiée par le constructeur car l'interface du logiciel est bien détaillée, contrairement à la configuration manuelle où toutes les informations sont codées, et il faut se référer à chaque fois au manuel fourni par Schneider Electric pour comprendre la signification de ces codes.

### Le câble de communication

Le câble utilisé pour communiquer ATV12 avec d'autres appareils est *RJ45* connu par (cable réseau) qui apparaît dans la figure ci-dessous :



Le Protocol de communication utilisé par Schneider est le *Modbus* où seulement les pines : 4,5 et 8 (voir la figure ci-dessus).

## L'environnement de travail



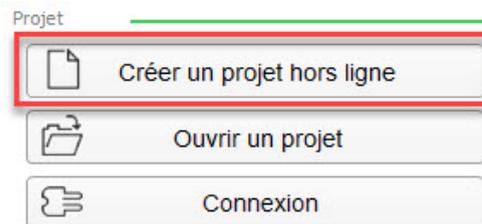
La figure ci-dessus représente l'environnement de travail du logiciel *Somove*, il contient le menu principal et elle donne la main pour charger la configuration déjà existante dans le logiciel.

**Remarque** La licence du logiciel est gratuite, il suffit juste de s'inscrire sur le site de *Schneider* et de remplir les formulaires.

## La création d'un projet

Pour commencer, on doit créer un projet :

### La création d'un projet hors ligne

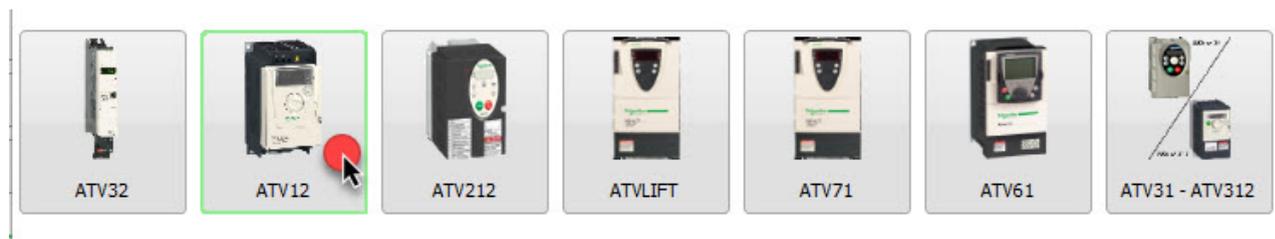


On peut créer un projet hors ligne, et une fois la configuration voulue est prête, on peut la transférer vers le variateur via le câble *Modbus*.

Pour créer le projet, on appuis sur (Créer un projet hors ligne).

### Le choix d'une série

Après qu'on a créé un projet hors ligne, on doit choisir l'une des séries qui existe dans la gamme des variateurs :

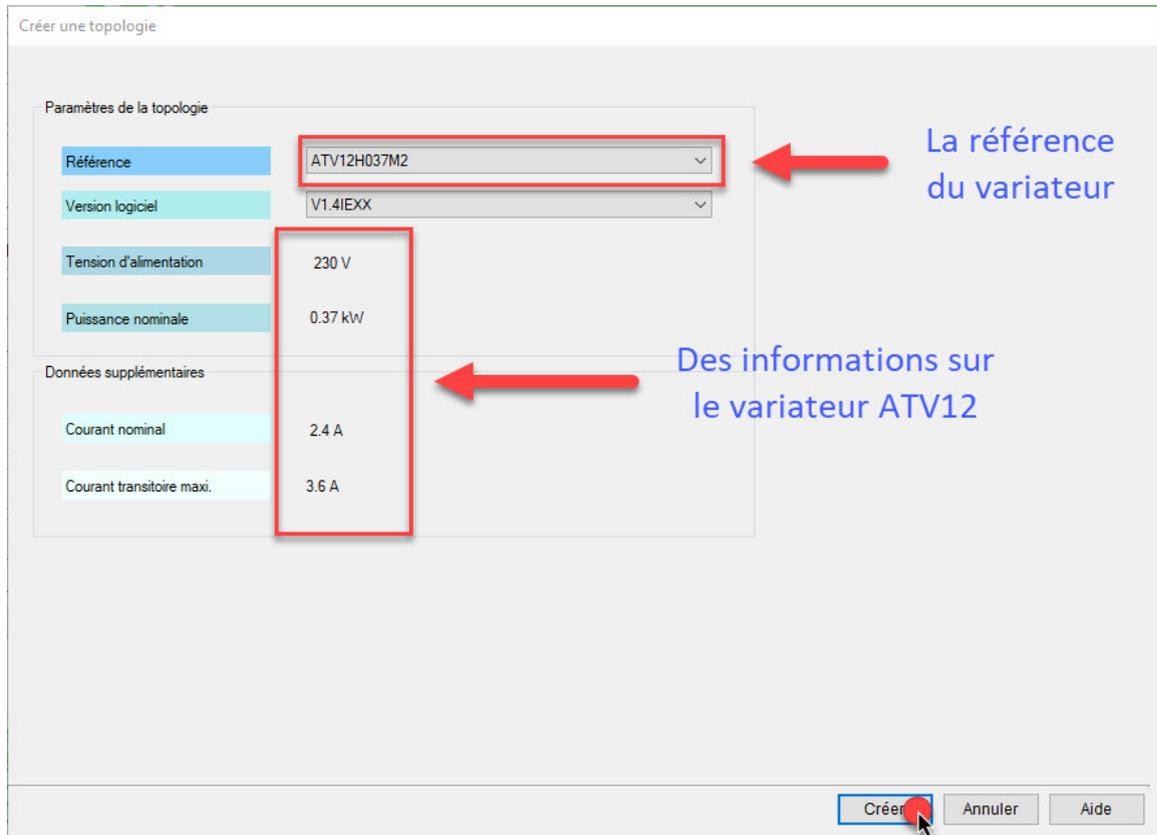


On a choisi ATV12 par ce qu'on travaillé avec dans notre projet.

### Le choix d'un model

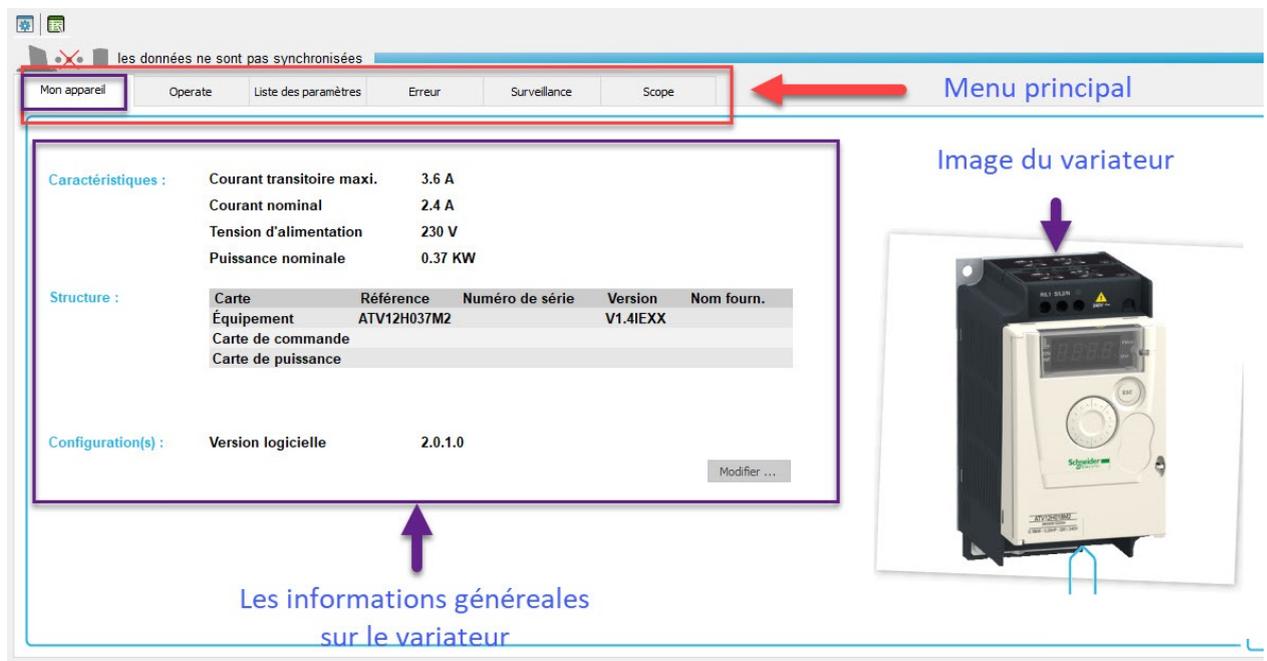
La série ATV12 possède différents models qui ont différentes caractéristiques. Notre model est *ATV12H037M2* qui possède les caractéristiques mentionnées dans la figure ci-dessus.

Puis, on appuis sur Créer :



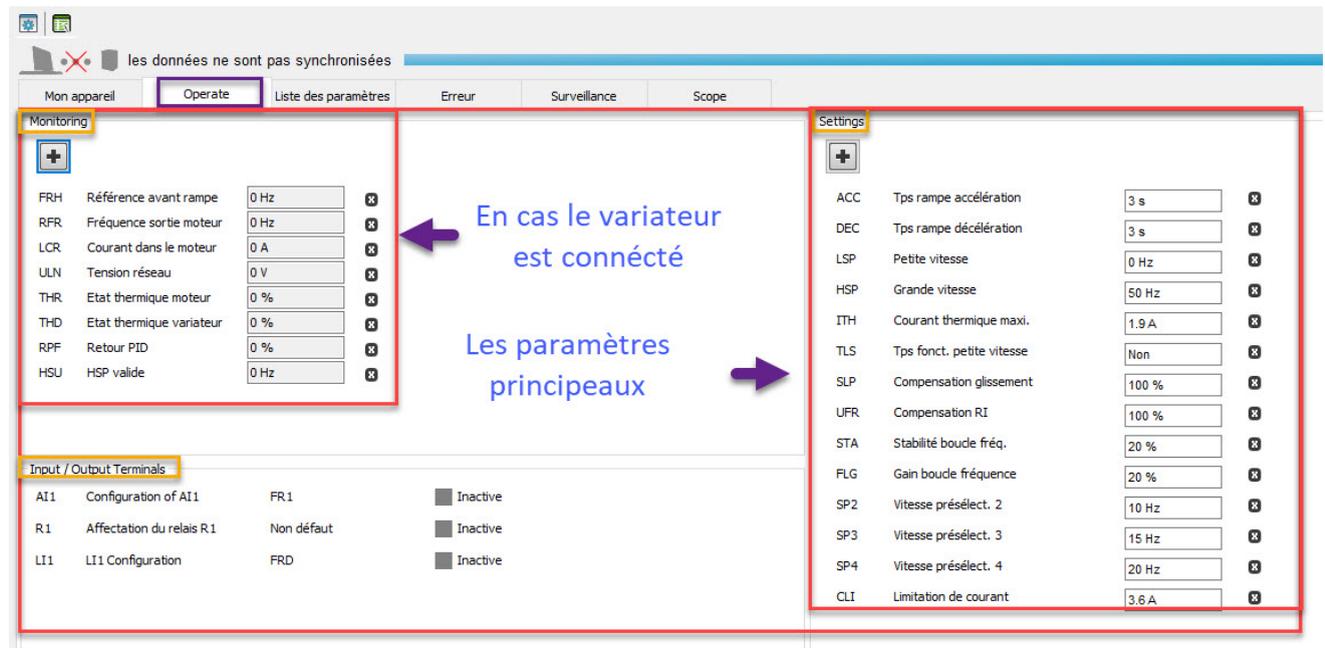
## Menu principal du projet

Il apparaît sur le menu principal, une rubrique 'Mon appareil' qui nous donne une image du variateur avec les caractéristiques générales de ce variateur.



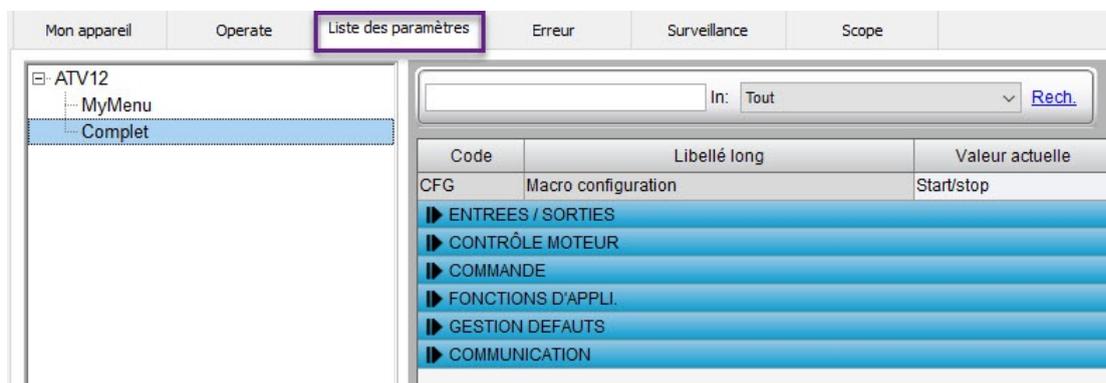
## Rubrique 'Operate'

Dans cette rubrique, on peut configurer les paramètres principaux du variateur (Setting), et aussi voir ce qui est affiché sur l'IHM du variateur en temps réel en cas où le variateur est connecté.



## Rubrique 'Liste des paramètres'

Dans cette rubrique, on trouve tous les paramètres qu'on peut configurer ou régler dans le variateur, les paramètres sont classés en catégories (Entrées/Sorties ; Contrôle moteur ;...) :



### Catégorie 'Entrées/Sorties'

Dans cette catégorie, on trouve tous ce qui est en relation avec les entrées sorties du variateur, et contient aussi une sous-catégorie :

Code	Libellé long	Valeur actuelle	Valeur par défaut	Valeur min	Valeur max
CFG	Macro configuration	Start/stop	Start/stop		
<b>ENTREES / SORTIES</b> <i>rubrique principale</i>					
TCC	Mode commande 2 / 3 fils	Cde 2 fils	Cde 2 fils		
TCT	Type de commande 2 fils	Transition	Transition		
NPL	LI actif à	POS	POS		
R1	Affectation du relais R1	Non défaut	Non défaut		
TOL	Tps détection surcharge	0 s	0 s	0 s	100 s
LOC	Seuil détection surcharge	90 %	90 %	70 %	150 %
FTO	Temps surcharge av. réd.	0 min	0 min	0 min	6 min
ULT	Tempo.Dét. Sous charge	0 s	0 s	0 s	100 s
LUL	Seuil couple freq.nulle	60 %	60 %	20 %	100 %
FTU	Temps s/charge av. réd.	0 min	0 min	0 min	6 min
FTD	Seuil fréquence moteur	50 Hz	50 Hz	0 Hz	400 Hz
CTD	Seuil de courant moteur	2.4 A	2.4 A	0 A	3.6 A
TTD	Dét therm. moteur	100 %	100 %	0 %	118 %
<b>CONFIGURATION AI1</b> <i>rubriques secondaires</i>					
▶ CONFIGURATION LO1					
▶ CONFIGURATION AO1					

**Exemple sur la rubrique principale :** Dans cette exemple, on essaye de changer la commande du variateur de 2 à 3 fils, et pour le faire, on appuis sur 'Cde 2 fils', qui se trouve dans la catégorie 'Entrées/Sorties' et on choisit '3 fils' :

Code	Libellé long	Valeur actuelle	Valeur par défaut	Valeur min	Valeur max
CFG	Macro configuration	Start/stop	Start/stop		
<b>ENTREES / SORTIES</b>					
TCC	Mode commande 2 / 3 fils	Cde 2 fils	Cde 2 fils		
TCT	Type de commande 2 fils	Transition	Transition		
NPL	LI actif à	POS	POS		
R1	Affectation du relais R1	Non défaut	Non défaut		
TOL	Tps détection surcharge	0 s	0 s	0 s	100 s
LOC	Seuil détection surcharge	90 %	90 %	70 %	150 %

**Exemple sur la sous-rubrique :** On veut dans cette exemple de changer le type d'entrée du variateur, on doit choisir, soit courant ou tension.

<b>CONFIGURATION AI1</b> <i>rubrique secondaire</i>					
AI1T	Type AI1	5V	5V	0V	4432
CRL1	Valeur minimum AI1	4 mA	4 mA	0 mA	20 mA
CRH1	Valeur maximum AI1	20 mA	20 mA	0 mA	4442

← Type d'entrée analogique  
← courant minimum/maximum

En cas où le choix est courant, on doit spécifier la plage de variation du courant (0 ⇒ 20mA ou 4 ⇒ 20mA) :

▼ CONFIGURATION AI1					
AI1T	Type AI1	5V	5V		
CRL1	Valeur minimum AI1	5V	4 mA	0 mA	20 mA
CRH1	Valeur maximum AI1	Tension Courant	20 mA	0 mA	20 mA

### Catégorie 'Contrôle moteur'

Dans cette catégorie, on définit tous ce qui est en relation avec les caractéristiques internes du moteur (fréquence standard, puissance nominale, compensation glissement, type de commande (U/f); ...):

La figure ci-dessous montre les paramètres à régler dans cette rubrique :

▼ CONTROLE MOTEUR					
BFR	Fréquence stand. moteur	50Hz IEC	50Hz IEC		
NPR	Puissance nom. mot	0.37	0.37	0.1	0.75
COS	Cosinus phi moteur 1	0.75	0.75	0.5	1
UNS	Tension nominale moteur	230 V	230 V	100 V	480 V
NCR	Courant nominal moteur	1.9 A	1.9 A	0.6 A	3.6 A
FRS	Fréq. nominale moteur	50 Hz	50 Hz	10 Hz	400 Hz
NSP	Vitesse nominale moteur	1400 rpm	1400 rpm	0 rpm	24000 rpm
TFR	Fréquence max. de sortie	60 Hz	60 Hz	10 Hz	400 Hz
CTT	Type commande moteur	Standard	Standard		
UFR	Compensation RI	100 %	100 %	25 %	200 %
SLP	Compensation glissement	100 %	100 %	0 %	150 %
STA	Stabilité boucle fréq.	20 %	20 %	0 %	100 %
FLG	Gain boucle fréquence	20 %	20 %	0 %	100 %
SFC	K filtre boucle vit.	0	0	0	100
PFL	Profil U/F	20 %	20 %	0 %	100 %
SFR	Fréquence de découpage	4 kHz	4 kHz	2 kHz	16 kHz
SFT	Type découpage	SFR type 1	SFR type 1		
NRD	Fr. découpage aléatoire	Non	Non		
TUN	Autoréglage	Non	Non		
MPC	Choix parametre moteur	P mot	P mot		

### Catégorie 'Commande'

Dans cette catégorie, on définit Les commandes reçues par le variateur et comment doit-il les gérer (source, priorité) :

▼ COMMANDE			
FR1	Configuration consigne 1	AI1	AI1
AIV1	Image entrée AIV1	0 %	0 %
RIN	Inhibition sens RV	Non	Non
PST	Priorité touche stop	Oui	Oui
CHCF	Configuration des canaux	Non séparé	Non séparé
CD1	Configuration canal cmd1	Borniers	Borniers
FLO	Affectation forçage local	Non	Non
FLOC	Canal de forçage local	Non	Non

## Catégorie 'Fonctions d'appli'

Dans cette catégorie, on s'intéresse plus à la régulation de la vitesse, et le temps de réponse, le temps d'accélération, le temps de décélération et les vitesses présélectionnées :

▼ FONCTIONS D'APPLI			
RRS	Affectat. marche arrière	Non	Non
JOG	Affectation pas à pas	Non	Non
JPF	Fréquence occultée	0 Hz	0 Hz
▶ RAMPE			
▶ CONFIGURATION ARRET			
▶ INJECTION DC AUTO			
▶ VITESSES PRESELECT.			
▶ REGULATEUR PID			
▶ SECONDE LIM. COURANT			
▶ BOUCLE DE VITESSE			

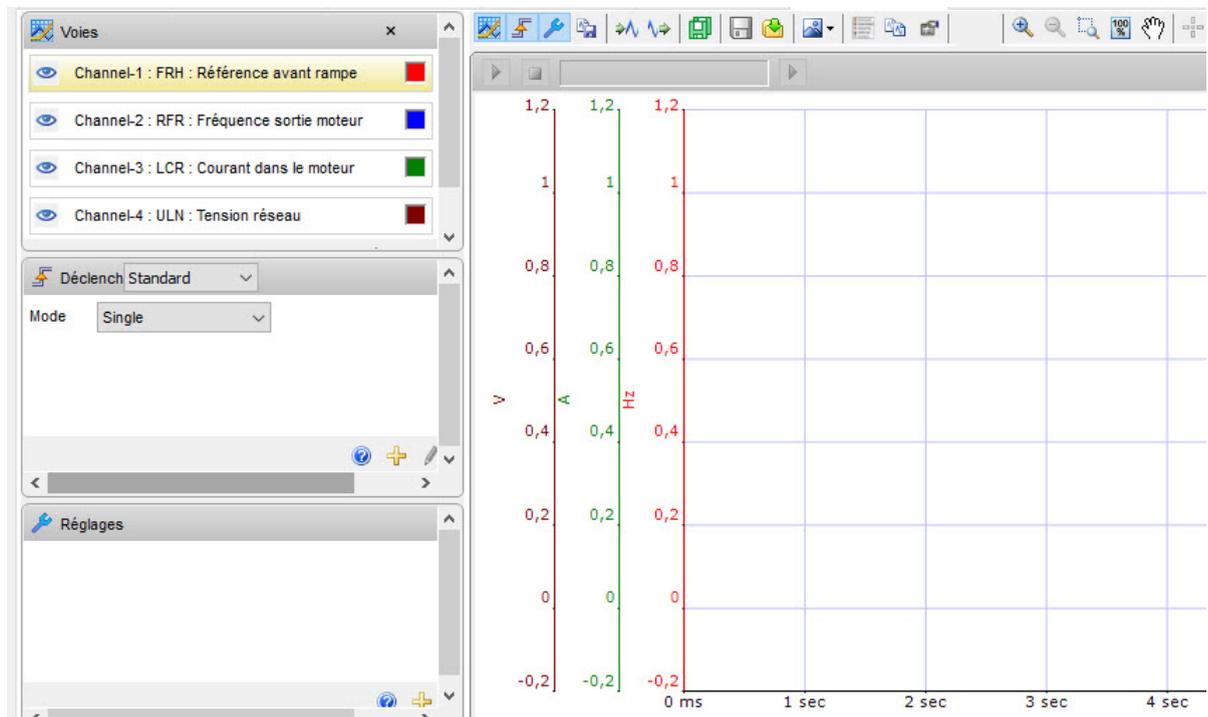
**Exemple des sous-rubriques de la catégorie :** Dans ces sous-rubriques, on trouve en plus les détails des paramètres :

▼ RAMPE			
ACC	Tps rampe accélération	3 s	3 s
DEC	Tps rampe décélération	3 s	3 s
RPT	Type de rampe	Linéaire	Linéaire
RPS	Affect. commut. rampe	Non	Non
AC2	Tps rampe accélération 2	5 s	5 s
DE2	Tps rampe décélération 2	5 s	5 s
BRA	Adapt. rampe décélér.	Oui	Oui
▶ CONFIGURATION ARRET			
▶ INJECTION DC AUTO			
▼ VITESSES PRESELECT.			
PS2	Affect. 2 vitesses présél.	Non	Non
PS4	Affect. 4 vitesses présél.	Non	Non
PS8	Affect. 8 vitesses présél.	Non	Non
SP2	Vitesse présélect. 2	10 Hz	10 Hz
SP3	Vitesse présélect. 3	15 Hz	15 Hz
SP4	Vitesse présélect. 4	20 Hz	20 Hz
SP5	Vitesse présélect. 5	25 Hz	25 Hz
SP6	Vitesse présélect. 6	30 Hz	30 Hz
SP7	Vitesse présélect. 7	35 Hz	35 Hz
SP8	Vitesse présélect. 8	40 Hz	40 Hz

## Rubrique 'Scope'

Dans cette rubrique, on peut visualiser, la commande et le temps de réponse avec le courant consommé en temps réel, avec la possibilité de changer les paramètres en

temps réel aussi :



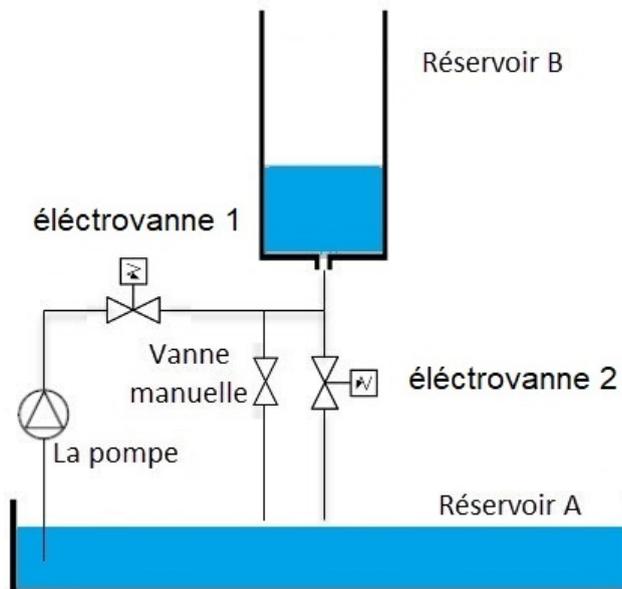
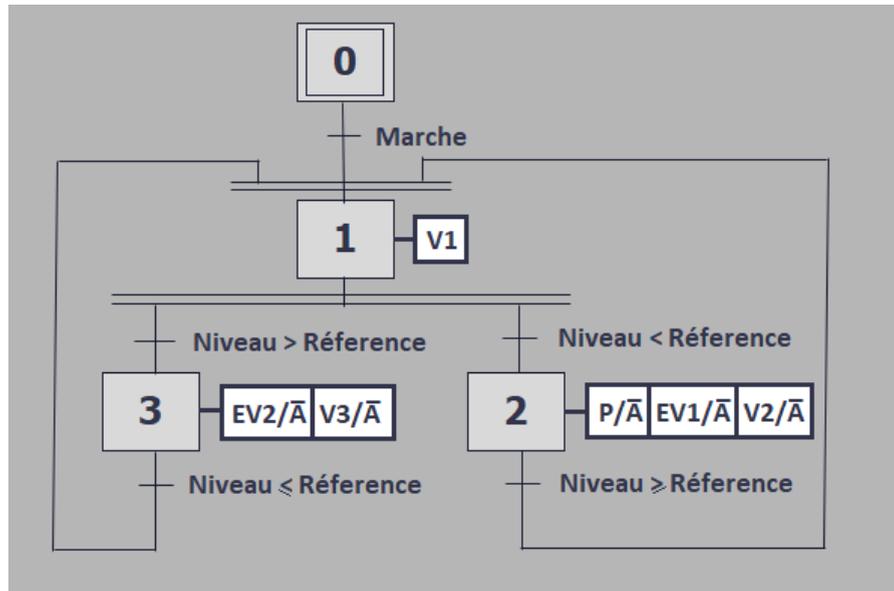
Annexe F : Travaux pratiques  
proposés pour exploiter la maquette

## TP1 : Manip TOR

Le but de ce TP est de contrôler le niveau d'eau dans le réservoir B, la référence de niveau est introduite par l'utilisateur via l'IHM.

Dans ce TP, l'étudiant doit réaliser un programme sur Step7 répondant au cahier des charges proposé, après la vérification du bon fonctionnement du programme sur le simulateur Step7, il passe à la maquette pour l'essayer en suivant les instructions.

Le cahier des charges proposé pour ce TP est traduit par le grafcet suivant :



## **Explication du cahier des charges :**

Le grafcet proposé fonctionne de la manière suivante :

Au démarrage du système, le grafcet est à l'état 0, une fois on appuis sur marche, on passe à l'état 1 (voyant rouge allumé), on vérifie si le niveau dans le réservoir est inférieur ou supérieur par rapport à la référence introduite.

Si le niveau est inférieur à la référence, on passe à l'état 2 : on allume la pompe avec l'électrovanne (anti-retour) et le voyant qui indique que la pompe est allumée. tout ces actions sont conditionnées par le bouton d'arrêt d'urgence.

Si le niveau est supérieur à la référence, on passe à l'état 3 : on met sous tension la deuxième électrovanne pour faire le vidange, avec l'allumage du voyant qui indique l'activation de la phase du vidange.

Toutes ces actions s'effectuent sous la condition que le bouton d'arrêt d'urgence ne soit pas appuyé.

### **Les entrées :**

- capteur de niveau analogique
- Bouton Marche
- Bouton reset

### **Les sorties :**

- Deux électrovannes, une pour le vidange, et l'autre fonctionne comme clapet anti-retour.
- Une pompe centrifuge qui pompe l'eau dans le réservoir B.
- 3 Voyants qui indiquent les états actifs.

## Partie 1 :

- En se basant sur le grafcet donné en haut, écrire les équations du système en suivant la logique RS, ainsi que les équations des sorties.
- Ouvrez le logiciel de programmation des API SIEMENS Step7, et configurez le matériel suivant : CPU s7-1214 6ES7214-1AG40-0XB0.
- Remplissez la table des mnémonique.
- Passez à la programmation de l'API en Ladder selon les équations faites précédemment.
- Une fois le programme terminé, lancez le simulateur et appelez le chargé de TP pour vérifier votre travail.

## Partie 2 :

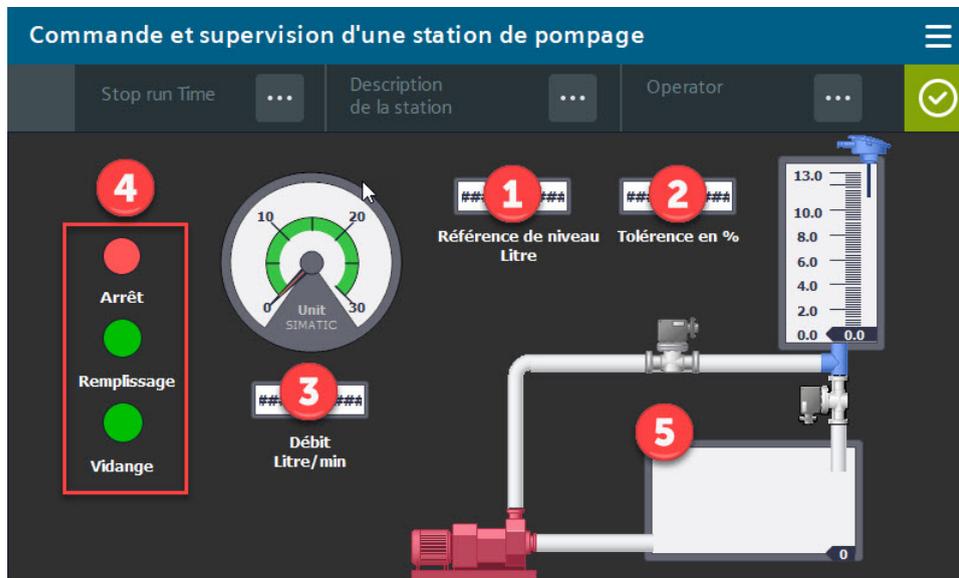
Une fois le programme est simulé, on peut passer à la manip.

Les instructions à suivre :

- Assurez que l'armoire est alimentée, et que le disjoncteur est fermé.
- Attendez que l'interface IHM affiche la figure ci-dessous :



- appuyez sur “suivant” jusqu’à ce que vous soyez sur la vue de la station :

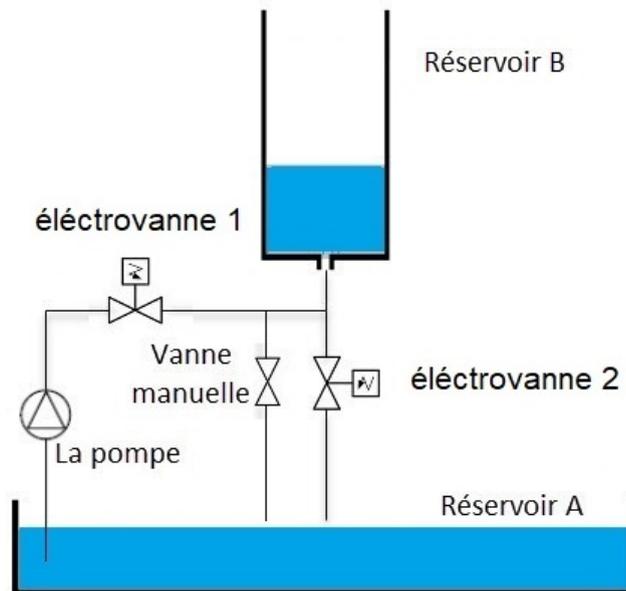


- appuyez sur le switch TOR (doit être ON).
- Choisissez une référence de niveau (par exemple : 7), et puis allez vers le menu des vues et choisissez la vue du Grafcet.
- Assurez vous que la vanne manuelle soit fermée.
- Appuyez sur “Reset”, l’état 0 doit s’afficher en vert.
- Maintenant vous pouvez commencer en appuyant sur “Marche”, et vous constatez que l’état 2 devient vert (phase de remplissage).
- Le moteur commence a pomper l’eau jusqu’à ce qu’il arrive à la référence, et il s’arrête automatiquement, s’il y avait un dépassement, l’électrovanne de vidange s’allume, et après un moment on revient vers l’état 1 (l’arrêt).

## TP2 : Identification du système

**Le but du TP :** Dans ce TP, l'étudiant sera capable d'identifier un système à l'aide d'un fichier E/S et de le modéliser par une fonction de transfert.

**Explication de la manip :** Le TP donne à l'étudiant la possibilité de récupérer en format .csv (fichier excel) les mesures E/S du système suivant :



Le système hydraulique est régi par l'équation différentielle suivante :

$$\frac{dh_1}{dt} = (1 - \gamma) \frac{a}{A} v - \gamma \frac{a}{A} \sqrt{2\rho g h_1}$$

Où  $h_1$  représente la hauteur du réservoir B, et  $\gamma$  représente l'ouverture de la vanne manuelle (1 ouverte et 0 fermée).

### Partie 1 :

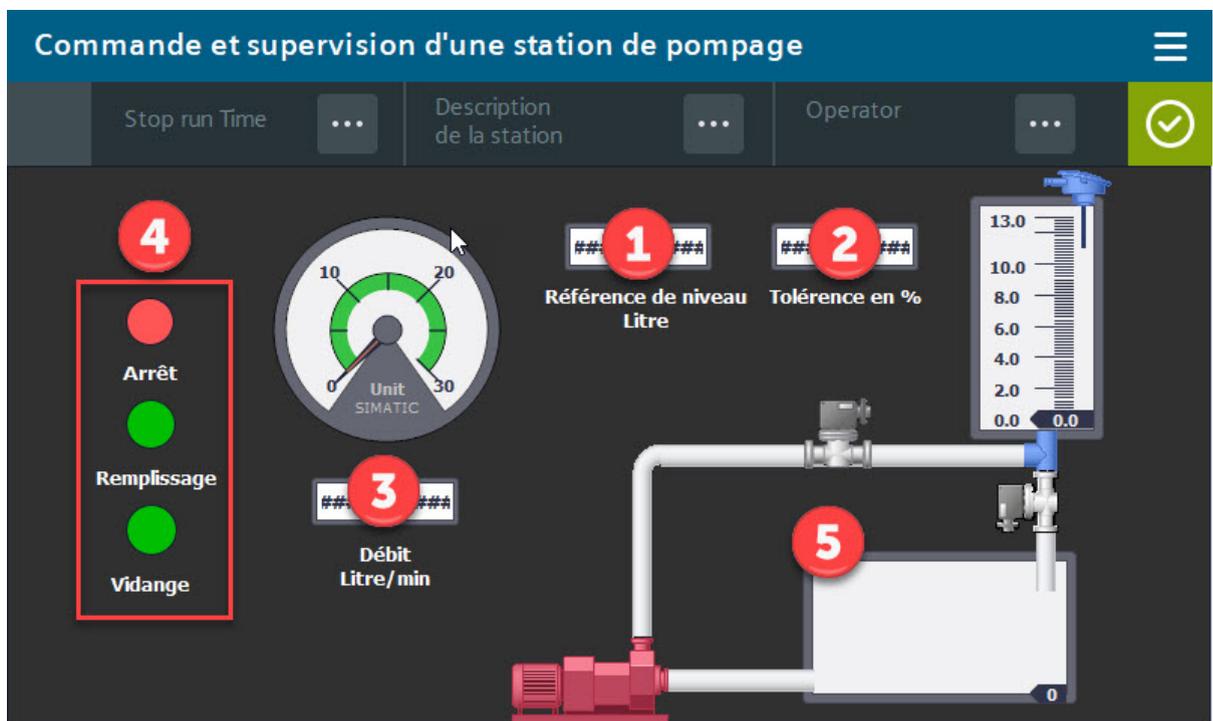
Les instructions à suivre :

- Assurez vous que l'armoire est mise sous tension, et que le disjoncteur qui se trouve à l'intérieur soit bien fermé.

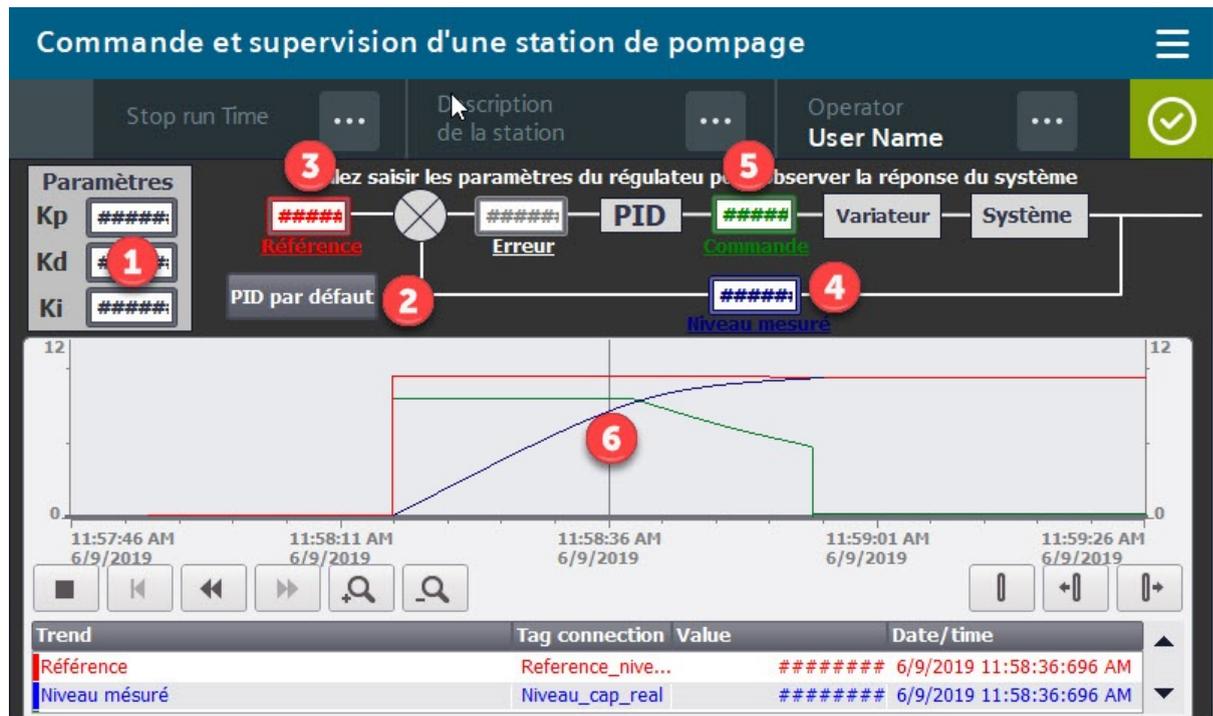
- Attendez que l'interface IHM affiche la figure ci-dessous :



- appuyez sur “suivant” jusqu'à ce que vous soyez sur la vue de la station :



- Allez y vers la “vue PID” depuis le menu de l'HMI, ou vous pouvez visualiser la réponse de votre système :



- Choisissez une référence (point de fonctionnement) par exemple 9L.
- Assurez vous que votre clé USB est insérée dans le port de l'IHM.
- Ouvrez la vanne manuelle à moitié  $\gamma = 0.5$  .
- Appuyez sur Reset, puis sur Marche pour actionner la pompe.
- Une fois le système arrive à la référence, vous pouvez retirer votre clé USB.
- Appuyez sur Reset et attendez que le vidange soit terminé.

## Partie 2 :

Une fois les données du système sont récupérées, vous pouvez passer vers l'étape de l'identification avec le professeur chargé du TP.

Le fichier obtenu contient toutes les données du systèmes prêtes pour le projet de l'identification. Une fois vous avez identifié le système et synthétisé un régulateur adéquat, vous pouvez l'implémenter sur la vue PID et observer la courbe de la réponse.