

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université A. MIRA – BEJAIA  
Faculté de Technologie  
Département de Génie Electrique



En vue de l'obtention du diplôme de

**Master**

En Electrotechnique

Option : Commandes Electriques

*Thème*

---

***Automatisation du circuit d'injection d'air  
stérile dans les bacs de stockage au sein de  
CEVITAL***

---

Réalisé par :

Mr. HANAFI AHCENE

Encadré par :

Mr. BOUZEMBOUA MALEK

Mr. MELAHI AHMED

Mr. BOUZEMBOUA TAKFARINAS

**Promotion 2021/2022**

## **REMERCIEMENTS**

*Ce travail a été réalisé au sein de stockage et chargement du sucre liquide du complexe CEVITAL*

*Tout d'abord, nous tenons à remercier DIEU le tout puissant de nous avoir accordé santé et courage pour accomplir ce travail.*

*Nous tenons à remercier vivement messieurs les membres du jury d'avoir consacré de leurs temps à la lecture de ce manuscrit, et d'accepter de juger et d'évaluer ce travail.*

*Nous remercions chaleureusement notre promoteur, Melahi Ahmed, pour sa démarche et son exigence scientifique qui nous ont permis de prendre de bonnes leçons pour la suite de notre parcours. Il a été le garant des orientations de ce travail.*

*Nous tenons également à remercier vivement notre encadreur d'entreprise Cevital ; Monsieur Bouzemboua Takfarinas de nous avoir permis d'user de son précieux temps et prodigué ses directives précieuses et ses conseils qui nous ont beaucoup aidés afin de réaliser notre travail. Ainsi que tous le personnel de la direction technique de cevital (unité de stockage et de chargement).*



## DEDICACE

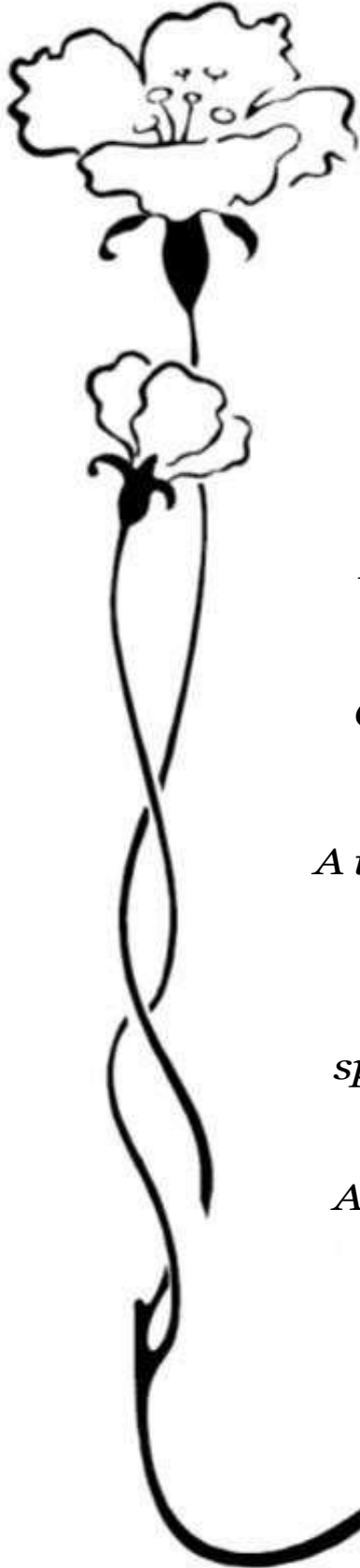
*Je dédie ce modeste travail  
À mes très chers parents qui m'ont  
tant aidé tout au long de mes études.*

*À mes frères et sœurs ainsi que  
tous les membres de ma famille ;*

*À tous mes amis et camarades sans  
exception ;*

*À tous mes proches ;*

*À tous ceux qui aiment partager le  
savoir sans rien en retour.*



## DEDICACE

*Je dédie ce travail*

*A mes parents :*

*Que Dieu les gardes et les  
protège.*

*À mes chère sœur Houria,*

*Et Melaaz, à mes*

*Chers frères, Salem, Fateh,*

*Tacfarinas.*

*A tous ceux qui me sont chères.*

*A tous ceux qui m'aiment.*

*A tous ce que j'aime,  
spécialement à ma très chère  
Katia.*

*A tous ceux qui ont contribué  
de près ou de loin à la  
Réalisation de ce modeste  
travail*

# *Sommaire*

## Sommaire

Remerciement

Dédicace

Sommaire

Liste des figures

Liste des tableaux

Liste d'abréviation

Introduction générale.....	1
Présentation du complexe CEVITAL.....	2
Chapitre I Description et fonctionnement du processus .....	5
I.1    Introduction .....	5
I.2    Présentation du l'unité de sucre liquide et ses différentes sections.....	5
I.2.1    Description général .....	5
I.2.2    Conditions de stockage et durée de vie .....	6
I.3    Fonctionnement des différentes sections de fabrications du sucre liquide .....	6
I.3.1    Section SL100 (Déméralisation).....	6
I.3.2    Section SL200 (Hydrolyse).....	6
I.3.3    Section SL300 (Charbon actif / filtration) .....	6
I.3.4    Section SL400 (Pasteurisation et filtration stérile) .....	6
I.3.5    Section SL500 (Concentration par évaporation).....	7
I.3.6    Section SL600 (Stockage et chargement camions).....	7
I.4    Description du procédé de la section stockage expédition (SL600).....	7
I.4.1    Bac de stockage (T601, T602, T603, T604, T605) .....	7
I.4.2    Quais de désinfection et chargement camion .....	7
I.4.3    Zone de prélavage .....	7
I.4.4    Description du système de filtration .....	8
I.4.4.1    Principe de fonctionnement de la soufflante.....	8
I.4.4.2    Description de la procédure de désinfection des soufflantes .....	9
I.5    Instruments de l'installation .....	9
I.5.1    Vannes.....	9
I.5.2    Capteurs- Transmetteurs .....	12
I.5.2.1    Capteur –transmetteurs de niveau .....	13
I.5.2.2    Capteur de pression .....	14
I.5.2.3    Capteur –transmetteurs de température.....	14
I.5.3    Les filtre .....	15

I.5.4	Pompe de la soufflante 610.....	16
I.6	Procédures de nettoyage d'un bac .....	16
I.7	Problématique de ce fonctionnement .....	17
I.8	Conclusion.....	17
chapitre II L'API et logiciels associés.....		18
II.1	Introduction .....	18
II.2	Structure d'un système automatisé de production (SAP).....	18
II.2.1	La Partie Relation (PR).....	18
II.2.2	La Partie commande .....	18
II.2.3	La Partie Opérative (PO) .....	19
II.3	Présentation de l'automate S7-300 .....	19
II.3.1	Présentation de la CPU S7-300.....	20
II.3.1.1	LED de visualisation d'état et de défaut.....	20
II.3.1.2	Commutateur de mode de fonctionnement .....	21
II.3.1.3	Pile de sauvegarde ou accumulateur .....	21
II.3.1.4	Carte mémoire .....	22
II.3.1.5	Interface MPI (interface multipoint).....	22
II.3.2	Caractéristiques techniques de la CPU S7-300 .....	22
II.3.3	Module d'alimentation.....	23
II.4	Cycle d'un API.....	23
II.5	Programmation de l'automate S7-300.....	24
II.5.1	Mémentos.....	25
II.5.2	Les différents blocs du programme utilisateur .....	25
II.6	Traitement du programme par la CPU.....	26
II.6.1	Traitement linéaire du programme .....	26
II.6.2	Traitement structuré du programme .....	26
II.7	Création d'un projet sous STEP7.....	27
II.7.1	Configuration matérielle .....	27
II.8	Lancement et configuration de S7-PLCSIM .....	29
II.8.1	Mise en route.....	30
II.9	Test du programme STEP7 avec S7-PLCSIM .....	30
II.10	Structure du programme .....	30
II.11	Élaboration de la supervision.....	32
II.11.1	Représentation du processus.....	32

II.11.2	Commande du processus .....	32
II.11.3	Archivage de valeurs processus et d'alarmes .....	32
II.11.4	Documentation de valeurs processus et d'alarmes .....	33
II.11.5	Gestion des paramètres de processus et de machine .....	33
II.12	Outils de la supervision.....	33
II.13	Conclusion.....	33
Chapitre III Elaboration de l'analyse.....		41
III.1	Introduction .....	41
III.2	Modélisation de fonctionnement du bac de stockage et désinfection soufflant de la section sucre liquide.....	41
III.2.1	Description du langage GRAFCET .....	41
III.2.2	Niveau d'un GRAFCET.....	42
III.2.2.1	Spécifications fonctionnelles (Niveau 1).....	42
III.2.2.2	Spécifications technologiques (Niveau 2) .....	42
III.2.3	Règles d'évolution d'un Grafcet .....	43
III.2.4	Les structures de base.....	43
III.2.4.1	Les séquences multiples .....	43
III.2.4.2	Saut d'étapes et reprise de séquence.....	43
III.3	Élaboration du GRAFCET du nouveau système .....	44
III.3.1	Grafcet niveaux 1 pour le bac de stockage .....	46
III.3.2	Grafcet niveaux 1 pour la désinfection de la soufflante 610.....	50
III.4	Conclusion.....	51
Chapitre IV Programmation et validation.....		52
IV.1	Introduction .....	52
IV.2	Procédure suivie pour la programmation de l'automate S7-300.....	52
IV.2.1	Attribution des adresses.....	52
IV.2.1.1	Editeur de mnémoniques.....	52
IV.2.1.2	L'adressage des modules E/S .....	53
IV.2.2	Création de la table des mnémoniques .....	53
IV.3	Elaboration du programme .....	53
IV.3.1	Programmation des blocs .....	54
Fig.IV.32 Programmation et paramétrage du régulateur PID .....		73
IV.4	SUPERVISION .....	75
IV.5	Étape de mise en œuvre .....	75
IV.5.1	Établir une liaison directe.....	75



IV.5.2	Création de la table des variables.....	76
IV.5.3	Création des vues .....	77
IV.5.3.1	Vues du processus .....	77
IV.5.3.1.1	Vue d'ensemble.....	77
IV.5.3.1.2	Vue des Bacs .....	78
IV.5.3.1.3	Compilation et simulation .....	79
IV.6	Conclusion.....	80
	Conclusion générale .....	81

Références bibliographiques

Annexe

# *Liste des figures*

## Présentation du complexe CEVITAL

Figure 1 : Emplacement géographique de l'entreprise..... 3

Figure 2 : Organigramme du complexe CEVITAL ..... 4

## Chapitre I : Description et fonctionnement du processus

Figure I-3 : Schéma explicatif de la fabrication du sucre..... 5

Figure I-4 : La soufflante ..... 8

Figure I-5 : Diagramme fonctionnelle de soufflante..... 8

Figure I-6 : Vanne TOR..... 10

Figure I-7 : La vanne régulatrice (TCV 600) ..... 11

Figure I-8 : La vanne régulatrice (PCV 640) ..... 11

Figure I-9 : Vanne à trois voies ..... 11

Figure I-10 : Vanne de purge ..... 12

Figure I-11 : Capteur de niveau..... 13

Figure I-12 : Capteur de pression ..... 14

Figure I-13 : Capteur utilisé dans notre section ..... 15

Figure I-14: P-SRF N..... 15

Figure I-15 : Pompe..... 16

## Chapitre II : L'API et logiciels associés

Figure II-1 : Structure d'un automatisme.....	18
Figure II-2 : Constituant de l'automate S7300.....	19
Figure II-3 : Vue générale de la CPU S7-300 .....	20
Figure II-4 : Alimentation de l'automate.....	23
Figure II-5 : Cycle d'un API.....	24
Figure II-6 : Organigramme de création d'un projet.....	27
Figure II-7 : configuration matériels .....	29
Figure II-8 : Connexion des bits d'entrées/sorties.....	30
Figure II-9 : Blocs de programme .....	31

## **Chapitre III : Élaboration de l'analyse fonctionnelle et du GRAFCET de**

Figure III-1 : Différent composants d'un grafcet .....	42
Figure III-2 : Saut d'étape.....	44
Figure III-3 : Reprise de séquence.....	44

## **Chapitre IV : Programmation et validation**

Figure IV.1 : Blocs du projet.....	55
Figure IV-2 : Les réseaux de bloc OB1.....	56
Figure IV-3 : Les réseaux de L'OB35.....	56
Figure IV-4 : Le programme de la temporisation .....	57
Figure IV-5 : Le passage entre les étapes du grafcet désinfection soufflante 610 .....	58
Figure IV-6 : Sélection du bac pour le nettoyage .....	58
Figure IV-7 : Nettoyage bac T601 en marche .....	59
Figure IV-8 : Sectionnement bac T601 pour le remplissage .....	59
Figure IV-9 : Remplissage du bac T601 en marche.....	59
Figure IV-10 : Programme compteur cumul du temps de fonctionnement de la soufflante 610 .....	60
Figure IV-11 : Demande désinfection soufflante 610.....	60
Figure IV-12 : Remise à zéros du compteur S0610 .....	60
Figure IV-13 : Action ou sortie du grafcet bac T601.....	61
Figure IV-14 : Programme capteur du niveau du bac LT601.....	62

Figure IV-15 : Programmation capteur de température bac TT601.....	63
Figure IV-16 : Programme capteur de pression soufflante PT610 .....	64
Figure IV-17 : Programme capteur de pression soufflante PT620 .....	65
Figure IV-18 : Programme capteur de pression bac PT601 .....	66
Figure IV-19 : Programme capteur de pression PT640.....	66
Figure IV-20 : Programmation vannes.....	67
Figure IV-21 : Actions ou sorties du grafcet désinfection soufflante .....	67
Figure IV-22 : Programme de la temporisation.....	68
Figure IV-23 : Activation de la régulation .....	69
Figure IV-24 : Programmation de la consigne .....	69
Figure IV-25 : Programmation des paramètres PID (format TI, TD, GAIN).....	70
Figure IV-26 : Mesure du capteur de pression PT640 .....	70
Figure IV-27 : Programmation et paramétrage du régulateur PID.....	71
Figure IV-28 : Activation de la régulation TCV600.....	72
Figure IV-29 : Programmation de la consigne .....	72
Figure IV-30 : Programmation des paramétré PID (format, TI, TD, GAIN) .....	73
Figure IV-31 : Mesure du capteur de température TT601 .....	73
Figure IV-32 : Programmation et paramétrage du régulateur PID .....	74
Figure IV-33 : Bloc de données paramètres bac T601.....	75
Figure IV-34 : Paramètres PCV640 .....	75
Figure IV-35 : Paramètres de régulation TCV600 .....	76
Figure IV-36 : Liaison entre la station et le pupitre .....	77
Figure IV-37 : Table de quelques variables .....	77
Figure IV-38 : Vues pupitre de l'installation .....	78
Figure IV-39 : Vue d'ensemble.....	78
Figure IV-40 : Configuration de l'animation des vannes.....	79
Figure IV-41 : Vue des Bacs .....	79
Figure IV-42 : Paramètres de bac.....	80
Figure IV-43 : Paramètres de soufflante .....	80

# *Liste des tableaux*

## **Chapitre II : L'API et logiciels associés**

Tableau II-1 : LED de visualisation d'état et de défaut .....	20
Tableau II-2 : Position de commutateur du mode de fonctionnement .....	21
Tableau II-3 : Caractéristiques techniques de la CPU S7-300 .....	23

## **Chapitre IV : Programmation et validation**

Tableau IV-1 : Les différents types de variable .....	54
---	----



# *Liste des abréviations*

*Introduction*  
*générale*

## **Introduction générale**

Aujourd'hui, les entreprises utilisent de plus en plus des systèmes de production automatisée car l'automatisation des opérations renforce la productivité, la fiabilité, la disponibilité et les performances.

Actuellement, il serait difficile de concevoir un système de production sans utilisation de systèmes automatisés et de technologies qui nous permettent d'augmenter la capacité de production dans tous les secteurs d'industrie et au même temps un produit de qualité.

En Algérie le groupe Cevital et parmi les grandes industries qui utilise ces technologies d'automatisation. Leur unité de production de sucre liquide contient des équipements automatisés qui assurent la production. D'où l'intervention manuelle se réduisant à la surveillance des différents paramètres de la machine, assurant ainsi le bon fonctionnement de la chaîne de stockage.

Ce projet est réalisé au sein de l'unité T600, c'est l'unité de stockage et d'expédition du sucre liquide, où nous nous sommes intéressés au système de stockage qui actuellement présente des difficultés pour garder la pression idéale dans les bacs de stockage lors du soutirage et chargement camion.

L'objectif de notre travail est l'Automatisation du circuit d'injection d'air stérile dans les bacs de stockage. Pour se faire, nous allons utiliser l'automate S7 300 CPU 315- 2DP, de la firme SIEMENS. Car c'est l'automate qui gère le système actuel et il s'agit d'un automate extensible.

Pour une meilleure présentation de notre travail, on a organisé notre mémoire en 4 chapitres

Dans le premier chapitre nous allons donner une brève présentation du fonctionnement des différentes sections de fabrication du sucre liquide. Par la suite nous allons présenter l'unité de stockage et expédition qui constitue l'objectif de notre travail, et on va conclure ce chapitre par la présentation du problème.

Le deuxième chapitre quant à lui, il sera consacré à la présentation de l'automate programmable et les logiciels de programmation et de supervision qu'on va utiliser dans notre projet

Ensuite le chapitre trois sera dédié à l'élaboration de la solution sous forme d'un Grafset.

Enfin, le dernier chapitre, sera consacré à la programmation de notre solution avec le logiciel step7 et la création des vues de supervision sur WINCC flexible afin d'assurer une bonne supervision du système.

Ce travail sera finalisé par une conclusion générale qui va récapituler le contenu de ce manuscrit.

*Présentation du  
complexe CEVITAL*

**Présentation du complexe CEVITAL**

Cevital (SPA) est parmi les entreprises algériennes qui ont vu le jour dès l'entrée de notre pays en économie du marché. Elle a été créée par des fonds privés en 1998. Son complexe de production se situe dans le port de Bejaia et s'étale sur une superficie de 45000 m<sup>2</sup>.

Le complexe contribue largement au développement de l'industrie agroalimentaire nationale. Son but est de satisfaire le marché national et exporter le surplus, en offrant une largegamme de produit de qualité.

Les nouvelles données économiques nationales dans le marché agroalimentaire font que les meilleurs sont ceux qui maîtrisent d'une façon efficace et optimale les coûts, les charges et ceux qui offrent le meilleur rapport qualité/prix. Ceci est nécessaire pour s'imposer sur les marchés que CEVITAL négocie avec les grandes sociétés commerciales internationales. Ses produits se vendent dans différentes villes africaines (Lagos, Niamey, Bamako, Tunis, Tripoli...).

**Activité de CEVITAL**

Lancé en mai 1998, le complexe CEVITAL a débuté son activité par le conditionnement d'huile en décembre 1998.

En février 1999, les travaux de génie civil de la raffinerie ont débuté, elle est devenue fonctionnelle en août 1999. L'ensemble des activités de CEVITAL est concentré sur la production et la commercialisation des huiles végétales, de margarine et de sucre, ainsi que la production de l'énergie électrique, elles se présentent comme suit :

Raffinage des huiles (1800 tonnes/jour).

- Conditionnement d'huile (1400 tonnes/jour).
- Production de margarine (600 tonnes/jour).
- Fabrication d'emballage (PET) : Poly-Ethylène-téréphtalate (9600 unités/heurs).
- Raffinage du sucre (2000 tonnes/jour et 3000 tonnes/jour).
- Stockage des céréales (120 000 tonnes).

**Situation géographique**

CEVITAL est implanté au nouveau quai du port de Bejaia, à 3 km du sud-ouest de cette ville, à proximité de la route nationale 26. Cette situation géographique lui a beaucoup profité étant donné qu'elle lui confère l'avantage de proximité économique. En effet elle se trouve proche du port et de l'aéroport.



Fig.1 Emplacement géographique de l'entreprise

### Différents organes constituant le complexe CEVITAL

L'organigramme suivant donne une vue générale sur les différents organes constituant le complexe CEVITAL.

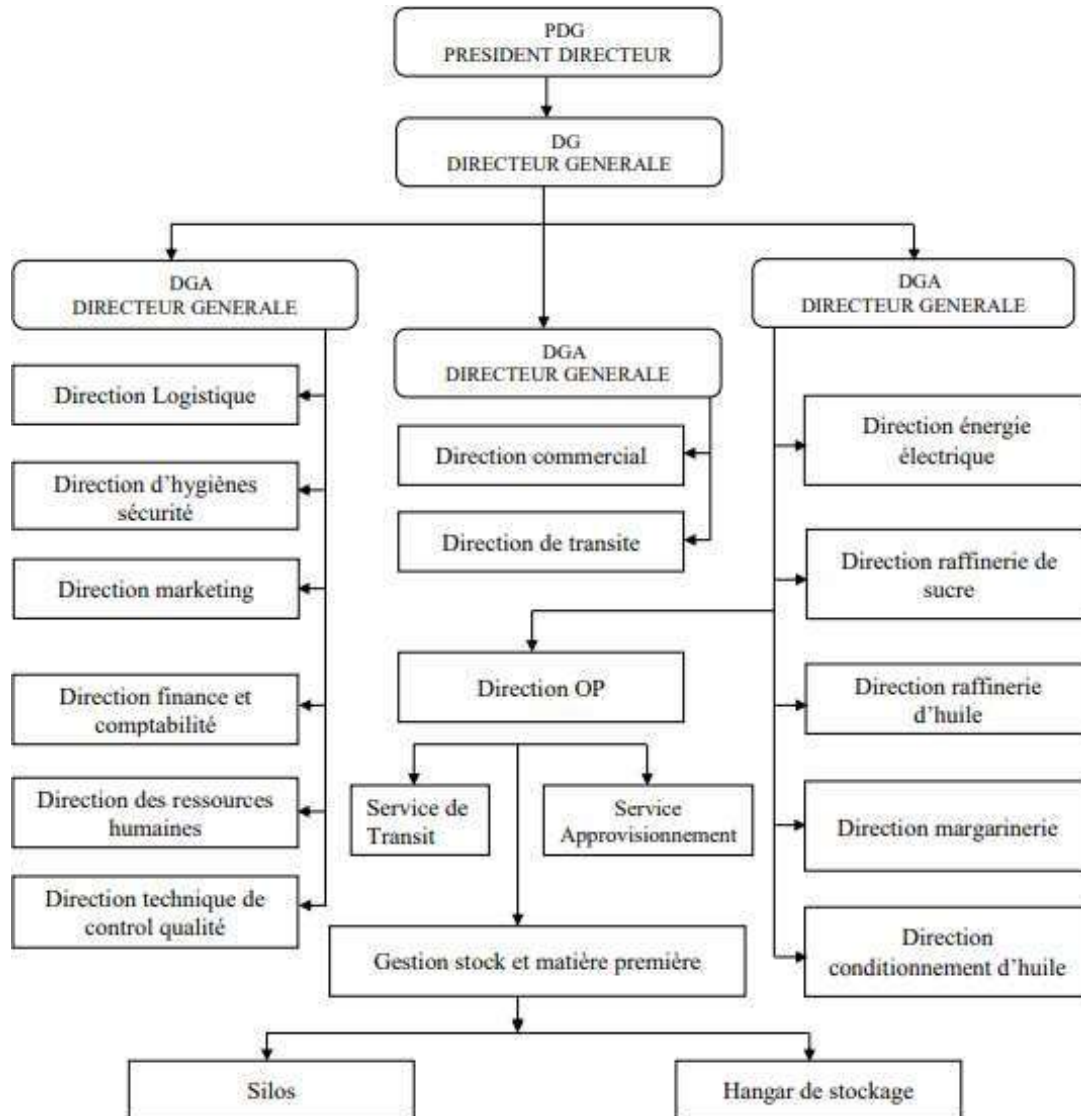


Fig.2 Organigramme du complexe CEVITAL

# *Chapitre I*



## I.1 Introduction

Dans ce présent chapitre nous allons d'abord présenter les différentes étapes et sections de l'unité sucre liquide, ensuite nous donnerons une description détaillée sur l'unité de stockage et chargement (unité 600) et les différents éléments constituant cette unité.

## I.2 Présentation de l'unité de sucre liquide et ses différentes sections

### I.2.1 Description général

Pour la fabrication de sucre liquide et/ou sucre inversé, le sirop décoloré passe par plusieurs sections : d'abord la déminéralisation pour enlever les minéraux, ensuite l'hydrolyse qui se fait par enzyme pour la fabrication du sucre inversé, après cette étape c'est l'étape de filtration au charbon pour enlever les particules de charbon injecté lors de la production du sirop, ce sirop décoloré passe ensuite par une section de pasteurisation puis par la section de concentration pour augmenter le brix du produit par évaporation de l'eau. Enfin le produit sera prêt à être stocké dans des bacs de stockage qui se trouvent à l'unité 600. Le schéma suivant illustre tout le processus de fabrication du sucre liquide

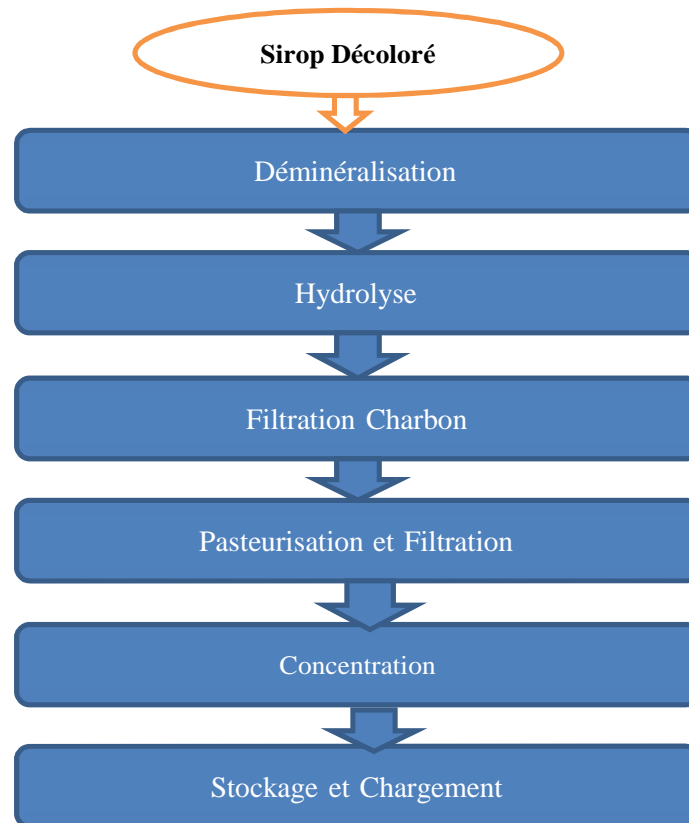


Fig.I.3 Schéma explicatif de la fabrication du sucre liquide [1].

### **I.2.2 Conditions de stockage et durée de vie**

Le sucre liquide saccharose est stocké dans des cuves en inox stériles, typiquement verticales, fond conique, calorifugées avec une capacité de 300m<sup>3</sup>. Température de stockage : Min 15°C - Max 25°C. Durée de stockage 01 mois [1].

### **I.3 Fonctionnement des différentes sections de fabrications du sucre liquide**

Afin d'avoir un produit fini prêt à l'emploi, le sirop décoloré issu de la raffinerie de sucre passe par plusieurs étapes qui peuvent être envisagées comme suit :

#### **I.3.1 Section SL100 (Déminéralisation)**

Cette unité constitue la première phase du cycle de production de sucre liquide, son rôle est d'éliminer les minéraux contenus dans le sucre liquide ainsi corriger sa couleur et son goût. Les minéraux sont des ions, et il en existe deux types : ceux chargés positivement (dits cations) et d'autres chargés négativement (dits anions) si la ligne produit du sucre inverti il sera envoyé vers l'unité 200 (Hydrolyse), sinon, il sera orienté vers unité 300 (Charbon-Filtration) pour la production du sucre liquide [1].

#### **I.3.2 Section SL200 (Hydrolyse)**

Le rôle de cette unité est la production du sucre inverti à partir du sirop déminéralisé en provenance de la section T100.

#### **I.3.3 Section SL300 (Charbon actif / filtration)**

Cette unité permet d'éliminer les dernières traces de couleur, d'odeur et de goût parasites avec du Charbon actif. Puis la filtration du sirop qui est indispensable pour lui enlever le charbon.

#### **I.3.4 Section SL400 (Pasteurisation et filtration stérile)**

La fonction de cette unité est de stériliser le produit venant de l'unité 300 pour détruire les micro-organismes, elle est réalisée à travers deux échangeurs de chaleur et une boucle de rétention pour que le sirop atteigne une température de 105 °C pendant 20 secondes, puis il sera refroidi à 70 °C. après la pasteurisation c'est la filtration stérile le sirop pasteurisé donc passe dans un filtre à plaques stériles comprimées à 200 bars par un vérin hydraulique, ces plaques possèdent des trous de moins de 0.6 mm de diamètre capable de retenir la quasi-totalité des particules de charbon restantes, ce qui assure une très bonne stérilisation du produit, après chaque cycle de filtration, le changement des plaques se fait manuellement, et l'automatisme s'occupe de les stériliser avant de les utiliser.

### **I.3.5 Section SL500 (Concentration par évaporation)**

La fonction de cette unité est de concentrer le produit venant de l'unité de pasteurisation et filtration stérile (unité 400) avant d'être stocker.

Pour éviter un risque de contamination microbiologique pendant le stockage et le transport, il est indispensable de concentrer le sirop à un brix de 73 grâce à un évaporateur à plaques double effet. Ensuite, à la sortie de l'évaporateur, une unité de refroidissement est prévue pour refroidir le sirop à 25°C. Le brix (Concentration) minimum que doit avoir le sucre liquide est de 67% tandis que celui du sucre inverti est de 73%, le pH du produit est ajusté dans le bac d'alimentation T500, il ne doit être ni trop acide ni trop basique, le pH doit être le plus proche possible de la valeur neutre [1].

### **I.3.6 Section SL600 (Stockage et chargement camions)**

Notre travail va se spécifier dans cette unité, qui est l'unité de stockage et d'expédition. La fonction de cette unité est le stockage et le chargement du produit final concentré. Cette unité comprend 5 bacs de stockage (T601, T602, T603, T604, et T605), chacun a une capacité de 300 m<sup>3</sup>, 3 quais de désinfection et chargement camion et d'un prélavage camion avec possibilité de désinfection. Le sirop arrive à cette section, prêt à être vendu.

## **I.4 Description du procédé de la section stockage expédition (SL600)**

### **I.4.1 Bac de stockage (T601, T602, T603, T604, T605)**

L'opération de stockage se fait par un seul remplissage à la fois. Le bac à remplir devra être à une température inférieure à 40 c° (inverti à 35 c° et sucre liquide à 25 c°), une sélection du bac à remplir est prévue. Une fois le bac est vide, il est possible d'effectuer une opération de nettoyage (NEP) à l'eau chaude, suivie d'une stérilisation avec la vapeur [1].

### **I.4.2 Quais de désinfection et chargement camion**

Avant chargement, tout camion devra être stérilisé avec la vapeur. Le chargement d'un camion s'effectue en sélectionnant un bac, à condition que ce dernier soit mémorisé « bon pour chargement camion ». Deux quais au maximum peuvent utiliser le même bac. Une pompe par quai assure le transfert du bac sélectionné vers le camion (par flexible). La fin du chargement est donnée par un volume [1]. Il est possible d'effectuer un nettoyage de la ligne de chargement de camion, suivi d'une stérilisation.

### **I.4.3 Zone de prélavage**

Deux postes sont mis à disposition pour les camions devant être nettoyés, les citernes sont nettoyées à l'aide d'eau chaude. Cette opération s'effectue avant la stérilisation.

#### I.4.4 Description du système de filtration

Le système de filtration est une unité compacte intégrant une préfiltration, une soufflante basse pression et une filtration stérilisante.

L'air stérile est transporté avec une légère surpression vers la cuve de stockage. L'échange constant dans le ciel de cuve évite le développement des bactéries, virus ou phages contenus dans l'air ambiant.

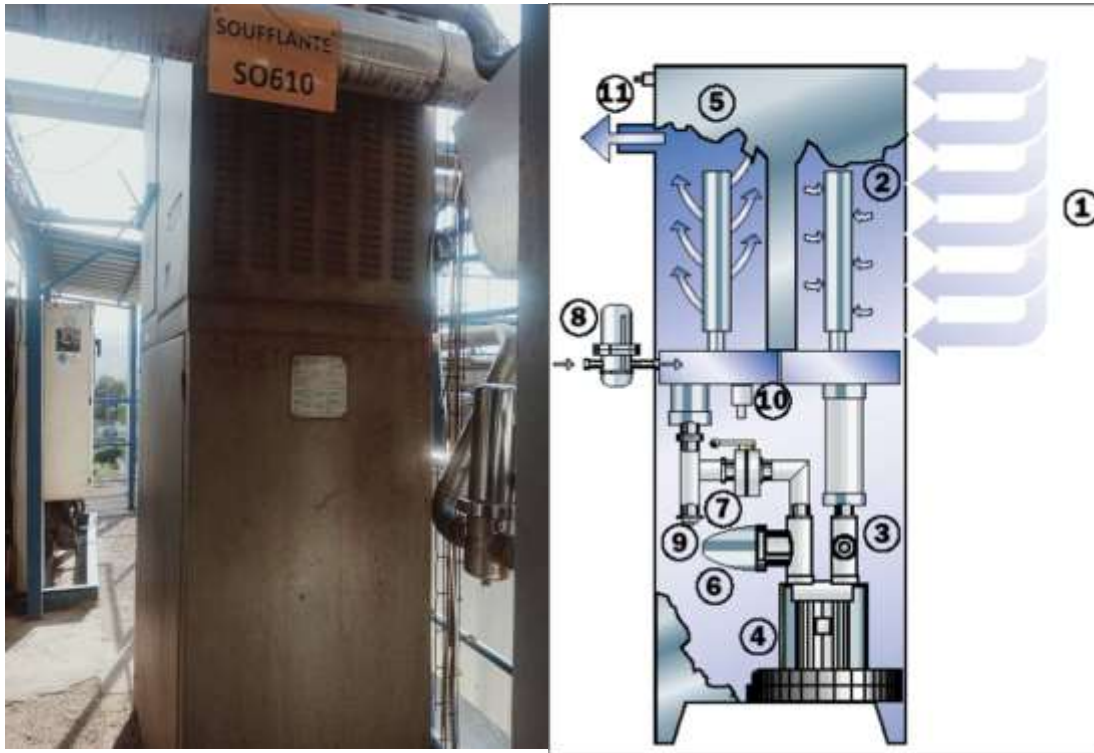


Fig.I.4 La soufflante

Fig.I.5 Diagramme fonctionnelle de soufflant [1]

##### I.4.4.1 Principe de fonctionnement de la soufflante

La soufflante aspire l'air ambiant sec, mais pollué (H.R. < 95%) par la prise d'aspiration (1) vers la chambre (2) (voir la figure (I.5)).

L'air entrant est nettoyé par l'étage de pré filtration en amont de la soufflante. Les particules retenues pourront créer une perte de charge dans le temps. Pour maintenir et protéger la soufflante (4) une soupape de sécurité (3) est installée.

La soufflante (4) compresse l'air à une pression d'environ 0,1 bar. L'air comprimé passe par la chambre stérile (5). Le filtre stérile retient les micro-organismes et toutes les bactéries et autres contaminants.

Pour protéger la soufflante, il y a une soupape de sécurité qui s'ouvre si le filtre stérile est colmaté (6).

L'air stérile sort en repère (11) pour alimenter la cuve de stockage ou toute autre application. Pendant la phase de stérilisation des éléments filtrants stériles, la vanne (7) doit être fermée pour empêcher la vapeur d'entrer dans la soufflante. Pour une bonne qualité de la vapeur pour la stérilisation, le filtre à vapeur (8) est monté en standard sur l'unité P-SLF. Après stérilisation, les condensats peuvent être évacués par les vannes manuelles (9 +10). (Voir également le guide de stérilisation) [1].

Notre unité dispose deux soufflant qui fonctionne on permanence, un basculement de la soufflant en marche ver la stérilisation chaque 200 heures.

#### **I.4.4.2 Description de la procédure de désinfection des soufflantes**

La désinfection de l'une des deux soufflantes (P-SLF SO610,P-SLF SO620) se fait comme suit: arrêt de la soufflant SO6010,ou bien SO620 et l'ouverture des vannes xv610.23/xv610.00 ou bien pour SO620 les vannes xv620.23/xv620.00 puis ouverture des vannes distribution vapeur xv630.01/xv610.21 et pour SO620 xv620.21 et l'injection de la vapeur pendant 100 minute (paramètre réglable pour le temps ) pour une pression 150-250 mb on lance l'opération de refroidissement purge pendant 120 minute le basculement de la soufflant ce fait chaque 200 heure.

### **I.5 Instruments de l'installation**

Nous allons présenter dans ce qui suit, les actionneurs présents dans l'installation.

#### **I.5.1 Vannes**

Une vanne est un dispositif qui sert à arrêter ou modifier le débit d'un fluide liquide ou gazeux, en milieu libre (canal) ou en milieu fermé (canalisation)

##### **➤ Vanne tout ou rien**

Une vanne TOR (Tout Ou Rien) est utilisée pour le contrôle de débit des fluides en tout ou rien, c'est-à-dire, elle exécute une action discontinue qui prend deux positions ou deux états 0 ou 1 (fermée ou bien ouverte à 100%). Les vannes TOR sont utilisées pour la commande des systèmes ayant une grande inertie où la précision de régulation n'est pas importante. Il existe deux types de vannes (TOR) l'une est automatique et l'autre manuelle.

Dans notre installation il existe un autre type de vanne (TOR) (fermée 30% ou bien ouvert à 100%).



Fig.I.6 Vanne TOR

➤ **Vanne de régulation**

La vanne de régulation est un dispositif conçu pour contrôler le débit de toutes sortes de fluides (liquide ou gaz) dans un système de commande de processus. La variation peut aller de la fermeture à l'ouverture totale (de 0% à 100%)

La vanne est commandée par un régulateur qui utilise l'action de l'air comme fluide d'asservissement. Ainsi l'ouverture, la fermeture ou l'action modulée de la vanne est produite par les variations de pression de sortie d'un instrument de mesure et de contrôle.

La vanne est actionnée mécaniquement. Elle est reliée à un actionneur capable de faire varier la position d'un organe de fermeture dans la vanne. L'actionneur peut être mû par une énergie pneumatique, hydraulique ou électrique.

La vanne régulatrice est constituée de trois éléments principaux :

Le servomoteur : c'est l'élément qui assure la conversion du signal de commande en mouvement de la vanne

Le corps de vanne : c'est l'élément qui assure le réglage du débit.



Fig.I.7 La vanne régulatrice (TCV 600)



Fig.I.8 La vanne régulatrice (PCV640)

### ➤ Les vannes à trois voies

Une vanne classique permet d'ouvrir, fermer ou ajuster le débit de liquide circulant dans un circuit.

Elle est généralement une vanne droite possédant un raccord d'entrée et un raccord de sortie. En forme de T, elle est donc munie de 3 raccords afin de pouvoir ajouter une conduite d'entrée à un circuit existant.

Munie d'un volant ou d'une manette, elle permet de doser le débit de liquide, d'ouvrir ou de fermer chacune de ses sorties. La vanne à 3 voies est notamment utilisée en régulation [2].



Fig.I.9 Vanne à trois voies.

➤ **Vanne de purge**

Organe placé sur une canalisation pour permettre le puisage et le réglage du débit du fluide qui y circule.



Fig.I.10 Vanne de purge

### I.5.2 Capteurs- Transmetteurs

➤ **Un capteur**

Un capteur c'est un composant qui permet de capter une grandeur physique, et de restituer sous forme de signal exploitable par la partie commande de ce même système. Ce signal est généralement électrique.

Il existe deux types de capteurs :

1. Les capteurs actifs : les capteurs actifs transforment directement le mesurande en grandeur électrique.

Un capteur actif fonctionne assez souvent en électromoteur.

2. Les Capteurs passifs : Ils nécessitent dans la plupart des cas une source d'énergie extérieure pour fonctionner (exemples. Thermistance, photorésistance, potentiomètre).

L'information détectée par un capteur peut être d'une grande variété, ce qui implique une grande variété de besoins en capteur, on cite parmi les plus connus et fréquents (pression, niveau, température, ...etc.).



➤ **Un Transmetteurs**

D'après la norme NFC 46-303, un transmetteur est un appareil qui, recevant une vraie variable mesurée, produit un signal de sortie normalisé pouvant être transmis et ayant une relation continue et définie avec la valeur de la variable mesurée. Le transmetteur est constitué d'un capteur associé à son conditionneur.

Il permet ainsi la transmission de signaux exploitables pour la plupart des appareils de contrôle des commandes industrielles [3].

**I.5.2.1 Capteur –transmetteurs de niveau**

Un capteur de niveau est un dispositif électronique, ce sont des appareils perfectionnés à deux ou quatre fils, à sécurité intrinsèque, reposant sur un principe de flottabilité simple pour détecter des variations de niveau de liquide et les convertir en un signal de sortie 4-20 mA stable. La conception verticale en ligne du transmetteur permet d'alléger le dispositif et de simplifier son installation.



Fig.I.11 Capteur de niveau.

### I.5.2.2 Capteur de pression

Le capteur de pression utilisé dans l'unité est DS 401 est la combinaison réussie des éléments suivants

- Pressostat intelligent ;
- D'un affichage numérique.

Et a été spécialement conçu pour une utilisation universelle dans les applications industrielles ; avec sa membrane affleurant, le DS 401 est adapté à une utilisation dans des milieux visqueux.



Fig.I.12 Capteur de pression

### I.5.2.3 Capteur –transmetteurs de température

Il est possible de mesurer la température de plusieurs façons différentes qui se distinguent par le coût des équipements et la précision ainsi que le temps de réponse.

Les capteurs que nous allons utiliser dans notre cas sont des capteurs transmetteurs de température type Omnigrad M TR12 Thermomètre RTD PT100 modulaire, qui est une technologie de mesure de température standard et complète pour presque toutes les industries Commander, Sélectionner et dimensionner, Configurer, Comparer.

De marque ©Endress+Hauser référence :TR 12-AAA2SAM30000 et TR12-AAA2SAH70000



Fig.I.13 Capteur utilisé dans notre section.

### I.5.3 Les filtre

- **P-SRF N** : est un filtre en profondeur plissé pour la filtration stérilisante de l'air comprimé, l'air process, les gaz techniques et les applications d'aération. Le P-SRF N associe une meilleure robustesse, une durée de vie plus longue et une grande sécurité. Son utilisation permet de réduire considérablement les coûts d'exploitation [1].



Fig.I.14 P-SRF N.

### I.5.4 Pompe de la soufflante 610

La soufflante utilise une pompe qui comprime l'air à une pression d'environ ( -290/280) mbar.



Fig.I.15 Pompe de la soufflante 610

### I.6 Procédures de nettoyage d'un bac

#### ➤ Lavage et désinfection des bacs :

La désinfection des bacs se fait en deux étapes :

-une fois on sélectionne l'un des bacs pour le nettoyer, l'ouverture des vannes 600-02 et 60x-04 ,60x-05 se fait pour injecter l'eau bouillante > 80 °C qui ruisselle sur le bord par un système de boule centrale. Dans le bac sélectionne par suit on fait évacuer l'eau par la vanne 60x-01 pour purge lavage

-on lance la désinfection par la vapeur basse pression séquencée par le haut et par le bas par l'ouverture des vannes 600-01, 60x-04,600-02 et fixer les paramètres à 240 min, après on ouvre la vanne 60x-02 purge vapeur pendant 20 minute pour une température pas moins de 90 C°.

#### ➤ Refroidissement des bacs

- Afin de refroidir les bacs et d'offrir un environnement idéal pour le remplissage et afin d'éviter le phénomène de caramélisation du produit fini à stocker. L'air généré par la soufflante SO610/SO620 est envoyé en permanence dans toutes les bacs en production dans sa phase stérilisation, chaque bac est isolé indépendamment de l'air processus par une vanne pneumatique TOR (60X.10). A la fin de chaque désinfection, on ouvre la vanne. L'injection à

faible pression de l'air stérile se fait pour faire baisser la température dans le bac à moins de 35 C° pour se préparer au remplissage.

### **I.7 Problématique de ce fonctionnement**

Lors du soutirage et le chargement des camions, en cas de colmatage des évènements stériles les pompes de soutirage créent une dépression dans les bacs de stockage concernés ce qui provoque l'implosion de ces bacs de stockage ce qui conduit par la suite à des pertes du produit.

Afin de remédier à ce problème, on a donc pensé à faire un nouveau système qui nous a permis d'effectuer les opérations de lavage, désinfection et refroidissement bacs dans un seul point de chargement (un seul grafcet), où nous allons automatiser le circuit d'injection d'air stérile sur pressé venant de la soufflante P-SLF lors de la phase de refroidissement, de remplissage et de soutirage pour but de :

- ✓ Chasser la phase vapeur lors de la phase de refroidissement jusqu'à obtenir une température de 30 C (pour éviter le flash thermique lors du remplissage avec le produit).
- ✓ L'air injecté permettra de remplacer le produit soutiré et évitera ainsi les dépressions qui provoquent l'implosion des bacs.

### **I.8 Conclusion**

Dans ce chapitre, nous avons décrit en premier lieu les différentes unités de production du sucre liquide de Cevital. Ensuite, nous avons détaillé le principe de fonctionnement de l'unité 600 où se trouve l'unité de stockage et expédition qui constitue l'objectif de notre travail. Ainsi qu'une description des différents instruments utilisés dans ces unités. Dans le chapitre qui va suivre nous allons parler sur l'automate programmable utilisé dans notre projet.

# *Chapitre II*

## II.1 Introduction

L'automate programmable industriel API (ou Programmable Logic Controller PLC) est aujourd'hui le constituant le plus répandu des automatismes. On le trouve pratiquement dans tous les domaines industriels vue sa grande flexibilité et son aptitude à s'adapter.

Dans ce chapitre, nous allons présenter la structure d'un système automatisé de production. Ensuite nous allons présenter spécifiquement l'automate programmable utilisé dans notre projet à savoir l'automate S7-300, ses principaux éléments qui le constitue. Nous allons présenter par la suite le logiciel de programmation **STEP7**, les différentes étapes pour la réalisation du programme, et nous concluons ce chapitre par une présentation du logiciel de simulation PLCSIM, ainsi l'interface graphique (HMI) sous WinCC flexible.

## II.2 Structure d'un système automatisé de production (SAP)

Il est constitué de trois parties :

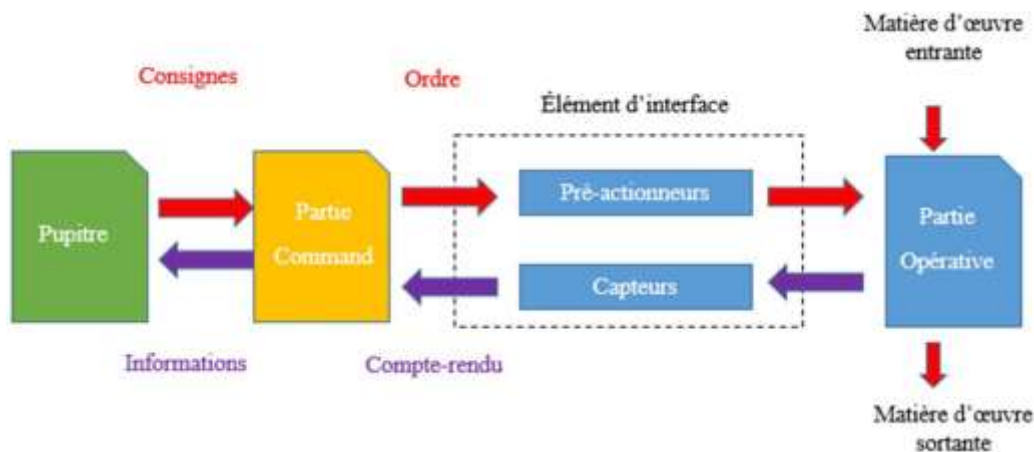


Fig.II.1 Structure d'un automatisme.

### II.2.1 La Partie Relation (PR)

Le pupitre permet le dialogue entre l'opérateur et la partie commande. L'opérateur envoie des consignes opérateurs et reçoit des informations visuelles.

### II.2.2 La Partie commande

C'est l'ensemble des moyens de traitement des informations émises par les organes de commande de la PR et capteurs de la PO. Les ordres résultants sont transmis aux pré actionneurs de la PO et aux composants de signalisation de la PR afin d'indiquer à l'opérateur l'état et la situation du système. A89

### II.2.3 La Partie Opérative (PO)

C'est l'ensemble des moyens techniques qui permettent d'apporter une valeur ajoutée à la matière d'œuvre, Elle regroupe :

- Les effecteurs : dispositifs terminaux qui agissent directement sur la matière d'œuvre pour lui donner sa valeur ajoutée (outils de coupe, pompes, têtes de soudure, etc.) ;
- Les actionneurs : éléments chargés de convertir l'énergie afin de l'adapter au besoin de la partie opérative ; cette énergie étant ensuite consommée par les effecteurs (moteur, vérin, électroaimant, résistance de chauffage, etc.) ;
- Les pré actionneurs : éléments chargés :
  - 1-D'adapter le faible niveau énergétique disponible en sortie de la P.C. au besoin de la P.O ;
  - 2-De distribuer ou de moduler l'énergie délivrée aux actionneurs (contacteur, distributeur, variateur de vitesse,).
- Les capteurs qui assument l'ensemble des fonctions de la chaîne d'acquisition de données (fin de course de vérin, détecteur de position, capteur de température, etc.).

### II.3 Présentation de l'automate S7-300

L'automate programmable utilisé dans ce projet est un S7-300, c'est un automate modulaire qui se compose de la CPU (Central Processing Unit), d'un module d'alimentation et des modules d'entrées/sorties (TOR ou Analogique.). La figure ci-dessous présente les constituants externes de l'automate S7-300.

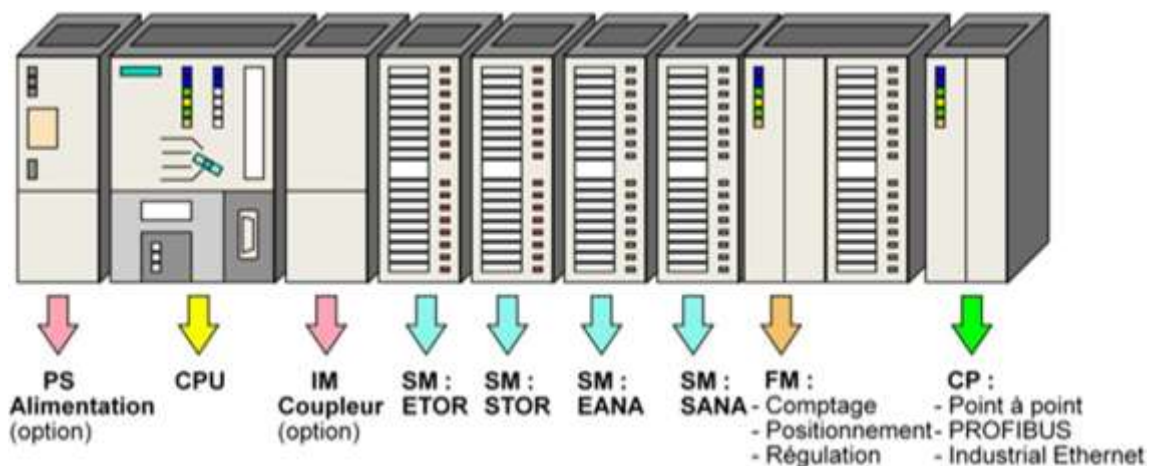


Fig.II.2 Constituants de l'automate S7-300.



**II.3.1 Présentation de la CPU S7-300**

La CPU c'est l'unité centrale qui gère l'automate programmable : elle reçoit, mémorise et traite les données entrantes et détermine l'état des données sortantes en fonction du programme établi. La figure ci-dessous montre une vue générale de la CPU S7-300 [4].

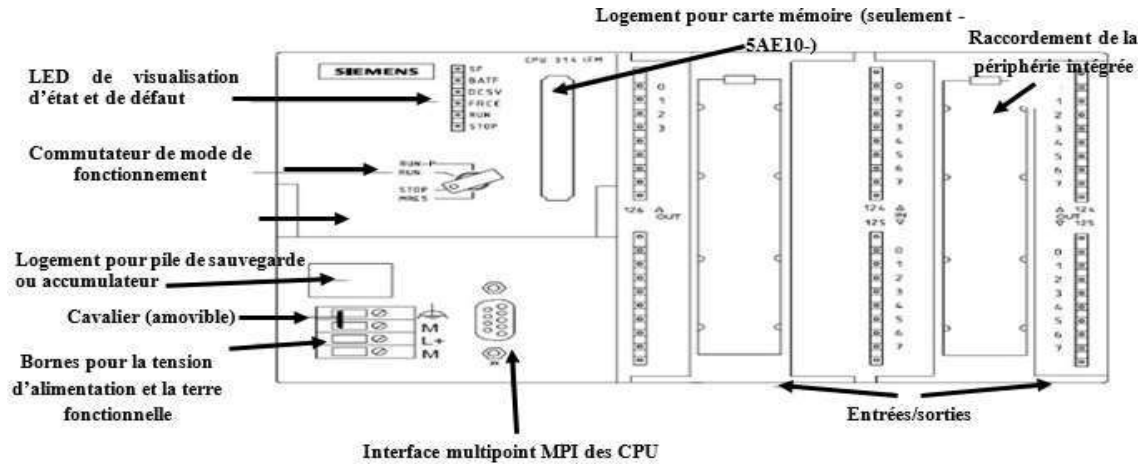


Fig.II.3 Vue générale de la CPU S7-300.

**II.3.1.1 LED de visualisation d'état et de défaut**

La visualisation d'état et de défaut se fait à travers des LED comme le montre le tableau II.1.

(Rouge)SF	Défaut matériel ou logiciel
(Rouge) BATF	Défaillance de la pile.
(Vert) 5V DC	L'alimentation 5V DC est correcte.
(Jaune) FRCE	Le forçage permanent est actif.
(Verte) RUN	CPU en RUN.
(Jaune) STOP	CPU en STOP ou en ATTENTE ou en démarrage.

Tableau II.1 LED de visualisation d'état et de défaut.

### II.3.1.2 Commutateur de mode de fonctionnement

Le changement de mode se fait à l'aide d'une clé comme le présente le tableau suivant.

Position	Signalisation	Explication
RUN-P	Mode de fonctionnement RUN-PROGRAMME	La CPU traite le programme utilisateur. Le programme peut être modifié. Dans cette position la clef ne peut être retirée.
STOP RUN	Mode de Fonctionnement STOP	La CPU ne traite aucun programme utilisateur. La clef peut être retirée. La CPU traite le programme utilisateur. Le programme ne peut être modifié qu'avec légitimation par mot de passe. La clef peut être retirée.
MRES	Effacement général	Position instable du cumulateur, pour effectuer l'effacement général il faut respecter un ordre particulier de commutation.

Tableau II.2 Position de commutateur du mode de fonctionnement.

### II.3.1.3 Pile de sauvegarde ou accumulateur

L'utilisation de l'accumulateur ou de la pile de sauvegarde est nécessaire pour l'horlogetemps réel.

La pile de sauvegarde est aussi utilisée pour :

- La sauvegarde du programme utilisateur s'il n'est pas enregistré dans la mémoire morte,
- Pour étendre la zone rémanente de données.

L'accumulateur est rechargé à chaque mise sous tension de la CPU. Son autonomie est de quelques jours voire quelques semaines au maximum. La pile de sauvegarde n'est pas rechargeable mais son autonomie peut aller jusqu'à une année.

**II.3.1.4 Carte mémoire**

La plupart des CPU possèdent une carte mémoire. Son rôle est de sauvegarder le programme utilisateur, le système d'exploitation et les paramètres qui déterminent le comportement de la CPU et des modules en cas de coupure du courant.

**II.3.1.5 Interface MPI (interface multipoint)**

L'interface MPI est l'interface de la CPU utilisée pour la console de programmation(PG), le pupitre opérateur(OP) ou par la communication au sein d'un réseau MPI.

**II.3.2 Caractéristiques techniques de la CPU S7-300**

Les tableaux suivants résument les principales caractéristiques techniques de la CPU S7-300

CPU	315-2-DP	CPU	315-2-DP
Temps d'exécution en us		Blocs d'organisation	N°OB
• Instruction binaire	0.3 – 06	Cycle libre	1
• Opération sur mot	1.0	Alarmes horaires	10
• Entier (+/-)	2.0	Alarmes temporisé	20
• Réel (+/-)	50.0	Alarmes cycliques	35
		Alarmes de processus	40
		Traitement d'arrière-plan	-
		Démarrage	100
		erreur, asynchrone	80-82, 85,87
		erreur, synchrone	121,122
Mémoire utilisateur		Données locales (octets)	1536
• Mémoire de travail	64 Ko	Longueur de bloc maxi.	15Ko
• Mém.chargem.intégr	96 Ko	Profondeur d'imbrication	8
• Mém.chargem.ext	4 Mo	des blocs/niveau d'exéc.	
Opérande		Communication :	4/8
• Mémentos (bits)	2048	Nombre maximum. de	
• Mémentos de cadence	8	liaisons	
• Temporisation	128	statique/dynamique	

• Compteur	64		
Type de blocs/Nombre		Communication par	
FB	192	donn.globales via MPI :	
FC	192	Cercles GD par CPU	4
DB	255		
Taille de la mémoire image (entrée/sorties)	128 octets chacune	Emission Paquet GD par cercle GD	1
Espace maxi. D'adresses périphéries	1024 octets chacun	Réception Paquets GD par cercle GD	1
Interfaces	MPI	Taille maxi.des données utiles d'un paquet	22 octets

Tableau II.3 Caractéristiques techniques de la CPU S7-300.

**II.3.3 Module d'alimentation**

Divers modules d'alimentation sont mis à disposition pour l'alimentation du S7-300.

Le module d'alimentation utilisé dans ce projet est : PS3075 A qui présente les propriétés suivantes :

- Courant de sortie 5 A,
- Tension nominale de sortie 24 Vcc, stabilisée, tenue aux court-circuit et à la marche à vide,
- Raccordement à un réseau alternatif monophasé (tension nominale d'entrée 120/130 Vcc, 50/60Hz),
- Séparation de sécurité des circuits,
- Peut servir de tension d'alimentation des capteurs et actionneurs.

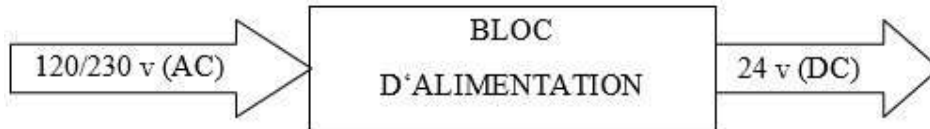


Fig. II.4 Alimentation de l'automate.

**II.4 Cycle d'un API**

Tous les automates fonctionnent selon le même mode opératoire

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

Ces quatre opérations sont effectuées continuellement par l'automate (fonctionnement cyclique) [5]

Ces quatre opérations sont effectuées continuellement par l'automate (fonctionnement cyclique) [5].

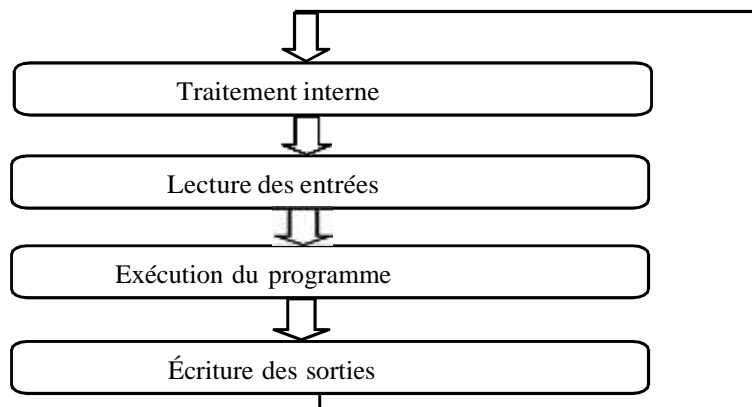


Fig.II.5 Cycle d'un API.

## II.5 Programmation de l'automate S7-300

La programmation des automates de la famille S7 se fait par la console de programmation PG ou PC et sous un environnement Windows, via le langage de programmation STEP7.

Le logiciel STEP7 est un outil de base pour la configuration et la programmation des systèmes d'automatisation SIMATIC.

Le STEP7 offre les fonctions suivantes pour l'automatisation d'une installation [6] :

- Configuration et paramétrage du matériel et de la communication,

- La création et la gestion des projets,
- La gestion des mnémoniques,
- La création des programmes pour système cible S7,
- Le chargement des programmes dans des systèmes cibles,
- Le test de l'installation d'automatisation.

La programmation en STEP7 présente trois modes de représentations possibles qui peuvent être combinés dans une même application :

- Le schéma logique (LOG),
- Le schéma contact (CONT),
- Liste d'instructions (LIST).

Chaque mode du programme a ses avantages mais aussi ses limites. Si les règles de programmations ont été respectées lors de la programmation, la compilation est possible dans les troismodes de représentation. Les programmes en CONT ou en LOG sont en principe toujours traduisibles en LIST.

Dans la mémoire du programme de l'automate, le programme est toujours stocké en LIST (plus exactement en langage machine).

### **II.5.1 Mémentos**

Les mémentos sont utilisés pour les opérations internes à l'automate pour lesquelles l'émission d'un signal n'est pas nécessaire. Les mémentos sont des bistables servant à mémoriser les états logique « 0 » ou « 1 ».

Chaque automate programmable dispose d'un grand nombre de mémentos (S7-300 dispose de 2048 bits de mémentos). On programme ces derniers comme des sorties. En cas de panne de la tension. De service, le contenu sauvegardé dans les mémentos est perdu.

### **II.5.2 Les différents blocs du programme utilisateur**

Le logiciel de base STEP7 dans ces différents langages de programmation possède un nombre important de blocs d'utilisateur, destiné à structurer le programme utilisateur. Dont on peut citer les blocs important suivant :

➤ Bloc d'organisation (OB)

Le programme utilisateur et le système d'exploitation. L'OB contient des instructions d'appels des blocs indiquant à l'unité de commande de l'automate l'ordre dans lequel il doit traiter les blocs [6].

➤ Bloc fonctionnel (FB)

Un bloc fonctionnel contient un programme qui est exécuté, lorsque ce bloc fonctionnel est appelé par un autre bloc de code. Il facilite la programmation de fonctions complexes, la commande de moteur (accélération...etc.).

Un bloc de données d'instance est associé à chaque FB qui en constitue la mémoire, paramètres transmis au FB ainsi que les variables statiques sont sauvegardés dans le bloc de données d'instance, les variables temporaires sont rangées dans la pile des données locales [6].

➤ Bloc de données (DB)

Les DB sont utilisés pour la mise à la disposition de l'espace mémoire pour des variables de type données [6].

## **II.6 Traitement du programme par la CPU**

Selon la manière de programmation choisie pour le problème d'automatisation, la CPU peut traiter le programme comme suit [6] :

### **II.6.1 Traitement linéaire du programme**

La CPU exécute le cycle habituel, en appelant le bloc OB1 dans le programme principal une fois par cycle. Ce type de traitement est requis lors de la commande de procédé simple exigeant un programme de taille et de complexité réduite.

### **II.6.2 Traitement structuré du programme**

Dans le cas d'un procédé complexe possédant plusieurs fonctions (postes), il est recommandé de partager le programme utilisateur en fonctions principales que l'on programme à l'aide des blocs de codes (OB, FB), d'une manière à faciliter la programmation et le test des parties du programme.

Dans ce cas la CPU exécute le programme principal dans OB1 et fait appel quand il faut aux autres blocs (FB). A la fin de l'exécution du bloc appelé, la CPU revient pour poursuivre l'exécution du programme appelant.

## II.7 Création d'un projet sous STEP7

Pour créer un projet sous STEP7 on a deux solutions possibles :

- **Alternative 1** : commencer par la configuration matérielle,
- **Alternative 2** : commencer par la création du programme.

Le schéma suivant illustre les deux solutions possibles lors de la conception d'une solution d'automatisation :

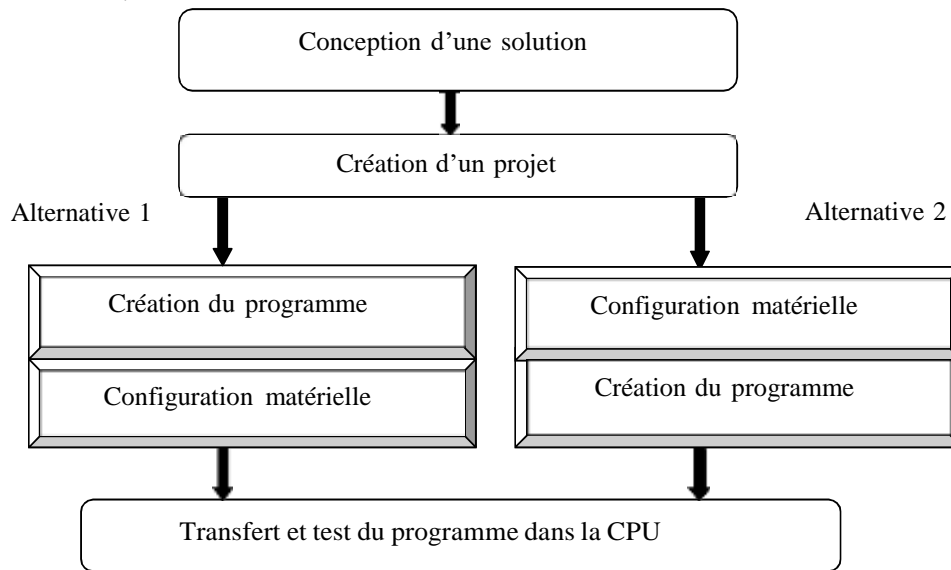


Fig.II-6 Organigramme de création d'un projet [6].


Toutes fois il est recommandé de commencer par la configuration matérielle pour les installations qui contiennent beaucoup d'entrées et de sorties, l'application de la configuration matérielle de STEP7 présente l'avantage de sélection automatique des adresses. Si on commence par la création du programme, il faudra rechercher les adresses en fonction des constituants choisis, dans ce cas on n'aura pas bénéficié de la fonction d'adressage automatique de STEP7. Les procédures qui vont nous permettre la création du projet sous STEP7 sont comme suit :

### II.7.1 Configuration matérielle

Après l'installation de logiciel Step7, pour pouvoir configurer le matériel, on doit au préalable avoir créé un projet. Ces données de configuration sont ensuite chargées dans le système d'automatisation. La procédure à suivre est la suivante :

- On double-clique sur Matériel, la fenêtre « HW config » s'ouvre (→matériel).
- La CPU qui a été sélectionnée à la création du projet est affichée.



➤ on ouvre le catalogue matériel en cliquant sur l'icône  on le trouve réparti en différents répertoires : PROFIBUS-DP, SIMATIC 300, SIMATIC 400 et SIMATIC PC Based Control 300/400, le support, les modules et les interfaces à la disposition pour l'étude de l'assemblage matériel.

➤ On insère un profilé support en double cliquant sur (→SIMATIC 300→RACK 300→Profilésupport).

Ensuite, un tableau de configuration pour l'assemblage du rack 0 est inséré automatiquement. Toutefois, le profilé support a des règles à respecter :

À l'emplacement 1, on place l'alimentation.

À l'emplacement 2, on place la CPU (Unité Centrale). L'emplacement 3 est réservé pour les cartes spéciales d'extension.

Dans les emplacements de 4 à 11, on place : les modules de signaux ou fonctionnels, le processeur de communication. On sélectionne :

- Deux modules d'entrées TOR (DI, Digital Input).
- Un module de sortie TOR (DO, Digital Output).
- De la même manière on sélectionne trois modules d'entrée analogique (AI, Analogique Input).
- On sélectionne ensuite un module de sortir analogique (AO, Analogique Output)

La figure ci-dessous montre les modules d'entrées/sortie sélectionnée.

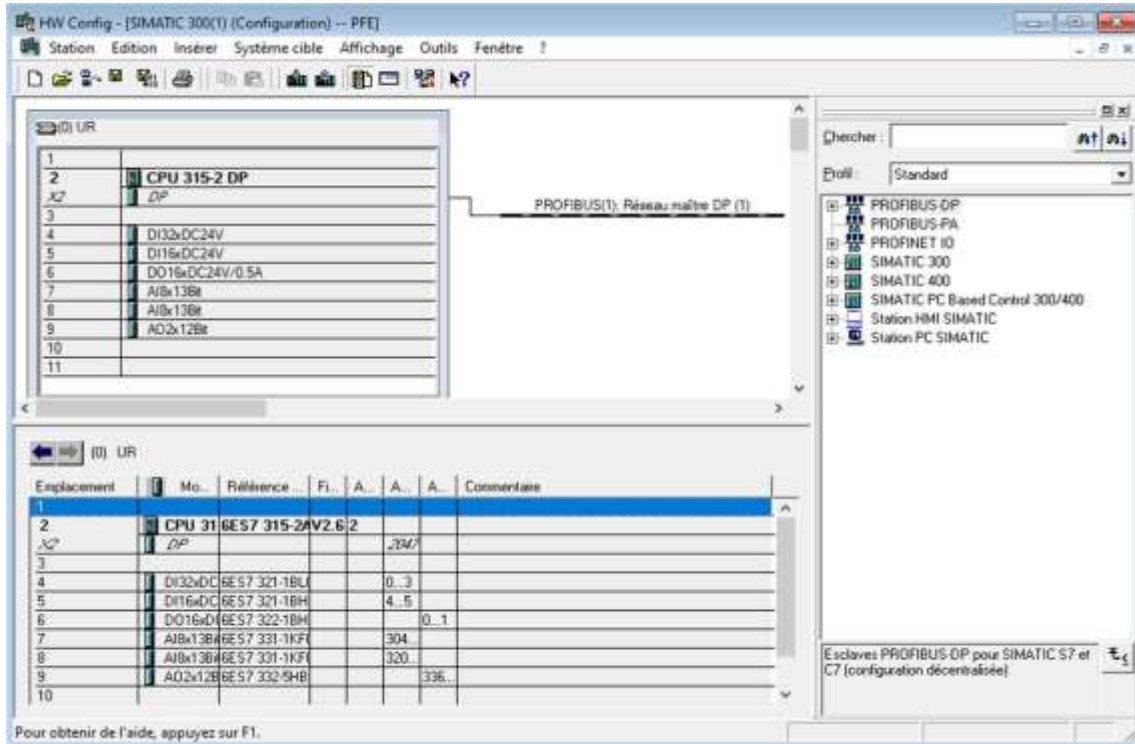




Fig.II.7 Configuration matérielle

On finalise par enregistrer et compiler en cliquant sur l'icône  puis charger en cliquant sur l'icône .

Après avoir fermé « HW Config » une nouvelle icône apparaît dans le dossier Bloc. Il s'agit des données système.

### II.8 Lancement et configuration de S7-PLCSIM

L'application S7-PLCSIM permet de simuler le fonctionnement d'un automate programmable S7.

On peut tester les programmes de commande à partir de S7-PLCSIM sans devoir se connecter à du matériel S7. S7-PLCSIM fournit une interface utilisateur graphique permettant de visualiser et de modifier les variables du programme de commande, d'exécuter la CPU de simulation en mode "Cycle unique" ou "Cycle continu", ainsi que la possibilité de modifier l'état de fonctionnement de l'AP de simulation.

### II.8.1 Mise en route

Pour tester maintenant ce programme, sans pour cela relier le PC avec un Hardware API, il suffit d'activer le simulateur.



Tous les accès lancés autrement sur une interface du Software API, sont alors simulés en interne avec S7-PLCSIM.

- Pour lancer PLCSIM, le Simulateur doit être exécuté. (→Activer/désactiver la simulation)
- Toutes les entrées et sorties qui ont été utilisées dans le programme à tester, doivent être insérées avec le menu « Insertion ». (Insertion→Entrée F2), (Insertion→Sortie F3).
- Les mémentos, temporisation et compteur peuvent également être affichés. (Exemple : Insertion→Mémento F4)
- Les adresses souhaitées par exemple ici, EB0 et AB0 doivent être déclarées, et le mode de présentation est précisé, ici c'est « Bits ».

### II.9 Test du programme STEP7 avec S7-PLCSIM

Le programme STEP7 à tester peut maintenant être chargé dans l'API et le simuler.

Dans notre cas, on prend par exemple OB1.

- on sélectionne le bloc (OB1) et on clique sur charger . (→OB1→charger .
- Les bits d'entrées peuvent être connectés par clic de souris. Les sorties actives sont représentées par le symbole «  ». (→RUN→).

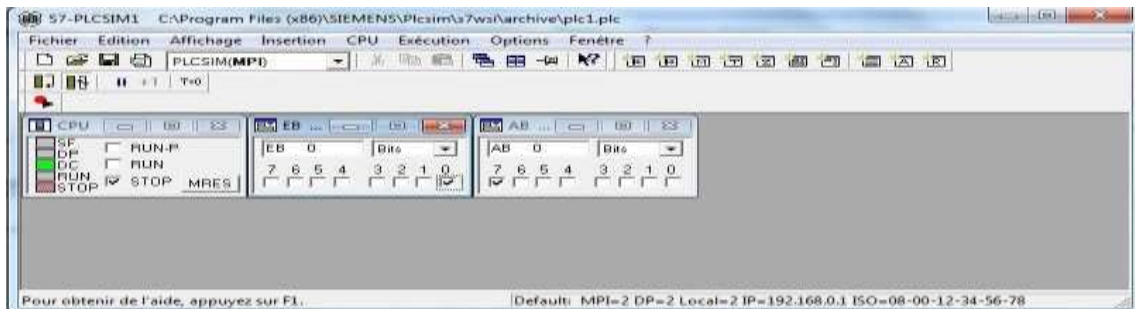


Fig.II.8 Connexion des bits d'entrées/sorties.

### II.10 Structure du programme

La programmation structurée permet la rédaction claire et transparente de programmes et la construction d'un programme complet à l'aide de modules qui peuvent être échangés et/ou modifiés volonté.

Pour permettre une programmation structurée confortable, il faut prévoir plusieurs types de modules : (OB) Bloc d'organisation, (FB) Bloc fonctionnel, (FC) Fonction, (SFB) Bloc fonctionnel système, (SFC) Fonction système, (DB) Bloc de données [4].

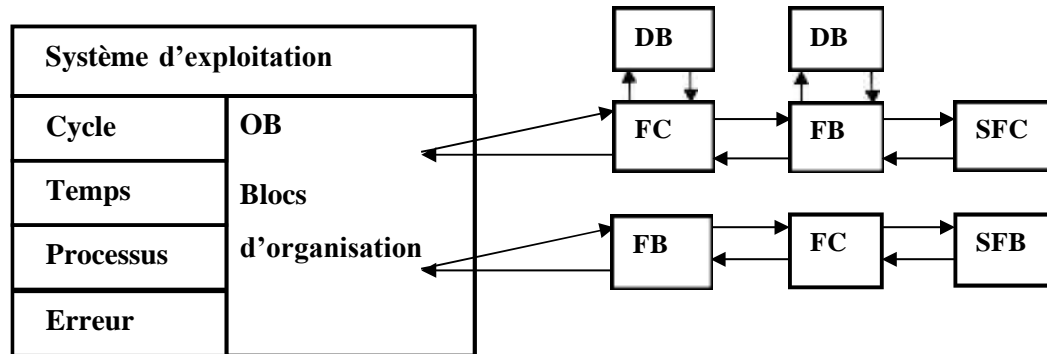


Fig.II.9 Blocs de programme.

**Blocs** : le système d'automatisation utilise différents types de blocs dans lesquels peuvent être mémorisés le programme utilisateur et les données correspondantes. Selon les exigences du processus, le programme peut être structuré en différents blocs [6] :

- **Bloc d'organisation** : les blocs d'organisation (OB) constituent l'interface entre le système OB d'exploitation et le programme utilisateur. L'ensemble de programme peut être concaténé dans un seul bloc OB1 (programme linéaire) appelé de manière cyclique par le système d'exploitation ou être structuré dans plusieurs blocs (programme structuré).
- **Fonction** : Une fonction (FC) assure une fonctionnalité spécifique du programme. **FC, SFC** ; Les fonctions peuvent être paramétrables. Dans ce cas, des paramètres sont transmis à la fonction lorsqu'elle est appelée. Les fonctions conviennent donc pour la programmation de fonctionnalités récurrentes et complexes, par exemple pour effectuer des calculs.
- **Les fonctions système (SFC)** : Elles sont des fonctions paramétrables, intégrées au système d'exploitation de la CPU, dont le numéro et la fonctionnalité sont définis de manière fixe.
- **Bloc fonctionnel** : Du point de vue du programme, les blocs fonctionnels s'apparentent aux fonctions FB, SFB, mais ils disposent en plus de zones mémoire spécifiques, sous forme de blocs de données d'instance. Les blocs fonctionnels conviennent donc pour la programmation de fonctionnalités récurrentes encore plus complexes, par exemple pour assurer des tâches de régulation.

- **Bloc fonctionnels système (SFB) :** Ils sont des blocs fonctionnels paramétrables, intégrés au système d'exploitation de la CPU, dont le numéro et la fonctionnalité sont définis de manière fixe.
- **Blocs de données :** Les blocs de données (DB) sont des zones de données du programme DB utilisateur.

## **II.11 Élaboration de la supervision**

Lorsque la complexité des processus augmente et que les machines et l'installation doivent répondre à des spécifications de fonctionnalité toujours plus sévères, l'opérateur a besoin d'un maximum de transparence. Cette transparence s'obtient au moyen de l'Interface Homme-Machine (IHM).

Un système IHM constitue l'interface entre l'homme (opérateur) et le processus (machine/installation). Le contrôle proprement dit du processus est assuré par le système d'automatisation. Il existe par conséquent une interface entre l'opérateur et WinCC flexible (sur le pupitre opérateur) et une interface entre WinCC flexible et le système d'automatisation. Un système IHM se charge des tâches suivantes :

### **II.11.1 Représentation du processus**

Le processus est représenté sur le pupitre opérateur. Lorsqu'un état du processus évolue par exemple, l'affichage du pupitre opérateur est mis à jour.

### **II.11.2 Commande du processus**

L'opérateur peut commander le processus via l'interface utilisateur graphique. Il peut par exemple, définir une valeur de consigne pour un automate ou démarrer un moteur.

Vue des alarmes

Lorsque des états critiques surviennent dans le processus, une alarme est immédiatement déclenchée, par exemple, lorsqu'une valeur limite est franchie.

### **II.11.3 Archivage de valeurs processus et d'alarmes**

Les alarmes et valeurs processus peuvent être archivées par le système IHM. Vous pouvez ainsi documenter la marche du processus et accéder ultérieurement aux données de la production écoulée.

### **II.11.4 Documentation de valeurs processus et d'alarmes**

Les alarmes et valeurs processus peuvent être éditées par le système IHM sous forme de journal.

Vous pouvez ainsi consulter les données de production à la fin d'une équipe par exemple,

### **II.11.5 Gestion des paramètres de processus et de machine**

Les paramètres du processus et des machines peuvent être enregistrés au sein du système IHM dans des recettes. Ces paramètres sont alors transférables en une seule opération sur l'automate pour démarrer la production d'une variante du produit par exemple SIMATIC HMI.

SIMATIC HMI offre une gamme complète permettant de couvrir toutes les tâches de contrôle-commande. SIMATIC HMI permet de maîtriser le processus à tout instant et de maintenir les machines et l'installation en état de marche.

WinCC flexible est le logiciel IHM pour la réalisation, par des moyens d'ingénierie simples et efficaces, de concepts d'automatisation évolutifs, au niveau machine. WinCC flexible réunit les avantages suivants :

- Simplicité,
  
- Ouverture,
  
- Flexibilité.

### **II.12 Outils de la supervision**

Un système de supervision et de contrôle est constitué d'une partie matérielle et d'une partie logicielle. La partie matérielle permet de relever les paramètres et d'interagir physiquement avec l'installation, alors que le logiciel est le cerveau du système.

### **II.13 Conclusion**

Dans ce chapitre nous avons donné un aperçu général sur l'automate programmable industriel S7- 300 son architecture interne et sa constitution, ainsi que sa structure modulaire. Par la suite, on a mis en avant ses caractéristiques techniques pour une meilleure exploitation pendant sa programmation.

Ensuite nous avons présenté le logiciel de programmation **STEP7**, les différentes étapes pour l'effectuer la configuration matérielle.

Nous avons vu aussi la présentation du logiciel de simulation PLCSIM, ainsi l'interface graphique (HMI) sous WinCC flexible pour assurer la supervision

# *Chapitre III*



### **III.1 Introduction**

L'automatisation de chaque système demande une étude et une analyse du mode de fonctionnement. L'analyse permet d'identifier les problématiques existantes puis des modifications s'imposent sur le système.

Dans ce chapitre nous allons d'abord donner une description globale de fonctionnement du notre système de stockage actuel ainsi le problème que nous avons confronté durant notre projet, ensuite nous présenterons la solution de la problématique par un grafcet qui va gérer le nouveau système.

### **III.2 Modélisation de fonctionnement du bac de stockage et désinfection soufflant de la section sucre liquide**

Dans cette partie, on va faire la modélisation de notre système par Grafcet.

#### **III.2.1 Description du langage GRAFCET**

Le grafcet (Grphe Fonctionnel de Commande Étape Transition) est un outil graphique de description du comportement attendu de la Partie Commande. Il décrit les relations à travers la frontière d'isolement de la Partie Commande et de la Partie Opérative d'un système automatisé [7].

L'établissement d'un grafcet suppose la définition préalable :

- Du système,
- De la frontière PO-PC, spécifiant la Partie Commande,
- Des Entrées et des Sorties de la Partie Commande.

La description du fonctionnement d'un automatisme logique peut alors être représenté graphiquement par un ensemble :

- D'ETAPES auxquelles sont associées des ACTIONS,
- De TRANSITIONS auxquelles sont associées des RECEPTIVITES,
- De LIAISONS (ou ARCS) ORIENTEES.

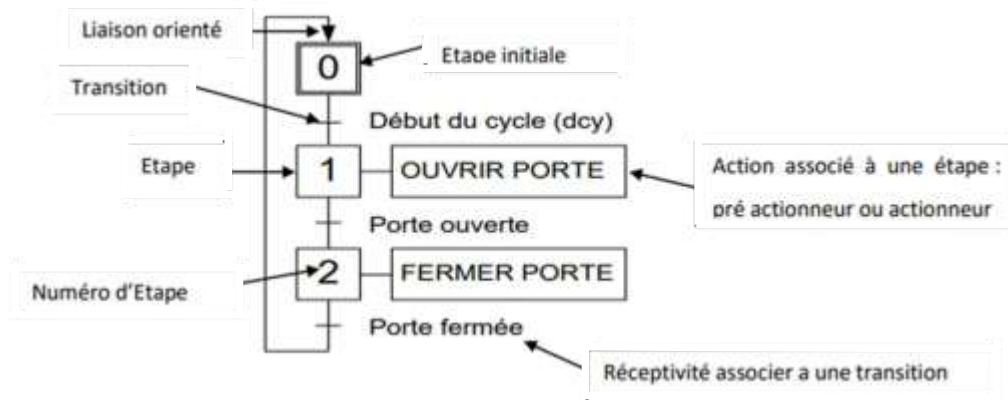


Fig.III.1 Différents composants d'un GRAFCET.

### III.2.2 Niveau d'un GRAFCET

#### III.2.2.1 Spécifications fonctionnelles (Niveau 1)

Le grafcet niveau 1 donner une interprétation de la solution retenue pour un problème posé.

Elle permet une compréhension globale du système.

Les spécifications fonctionnelles précisent [8] :

- Le dispositif à automatiser sous forme d'un dessin simplifié, ou d'un schéma fonctionnel,
- Les fonctions assurées par chaque partie du dispositif,
- Les informations permettant de contrôler ces fonctions.

#### III.2.2.2 Spécifications technologiques (Niveau 2)

La représentation technologique donne une interprétation en tenant compte des choix technologiques relatifs à la partie de commande et opérative de l'automatisme.

C'est à ce niveau que sont prises les options relatives [8] :

- La nature des actionneurs : moteurs, vérins, pompes ...etc.
- Les types de capteurs : détecteurs, contacts, avec les contraintes technologiques qui à ces choix, en particulier les interfaces utilisées soient :

A l'entrée de l'automatisme (amplificateur, mise en forme du signal, adaptation)

A la sortie (utilisation de relais, contacteurs, distributeurs, etc.).

### **III.2.3 Règles d'évolution d'un Grafcet**

- Première règle condition initiale : A l'instant initial, seules les étapes initiales sont actives.
- Deuxième règle franchissement d'une transition Pour qu'une transition soit validée, il faut que toutes ses étapes amont (immédiatement précédentes reliées à cette transition) soient actives. Le franchissement d'une transition : se produit lorsque la transition est validée, ET seulement si la réceptivité associée est vraie.
- Troisième règle évolution des étapes actives : Le franchissement d'une transition entraîne obligatoirement l'activation de toutes les étapes immédiatement suivantes et la désactivation de toutes les étapes immédiatement précédentes.
- Quatrième règle franchissement simultané : Toutes les transitions simultanément franchissables à un instant donné sont simultanément franchies
- Cinquième règle conflit d'activation : Si une étape doit être simultanément désactivée par le franchissement d'une transition aval, et activée par le franchissement d'une transition amont, alors elle reste active. On évite ainsi des commandes transitoires (néfastes à la partie opérative).

### **III.2.4 Les structures de base**

#### **III.2.4.1 Les séquences multiples**

Un cycle d'automatisme peut être linéaire, ou multiples. Les cycles simples que nous avons vus plus tôt sont des cycles avec une seule séquence, c'est-à-dire est une suite d'étapes à exécuter l'une après l'autre. Les séquences multiples comporte plusieurs séquences. La sélection de ces séquences se fera par aiguillage appelés divergences en OU, en ET [7].

#### **III.2.4.2 Saut d'étapes et reprise de séquence**

Le saut d'étapes permet de sauter une ou plusieurs étapes lorsque les actions associées sont inutiles à réaliser, La reprise de séquence (ou boucle) permet de reprendre, une ou plusieurs fois, une séquence tant qu'une condition n'est pas obtenue [7]

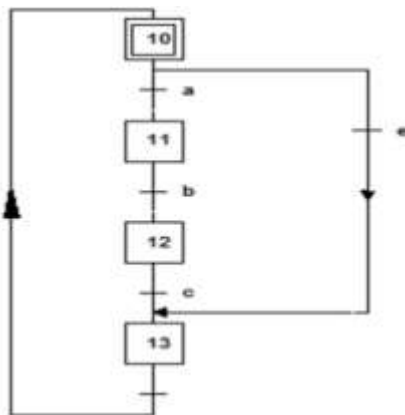


Fig. III.2 Saut d'étape.

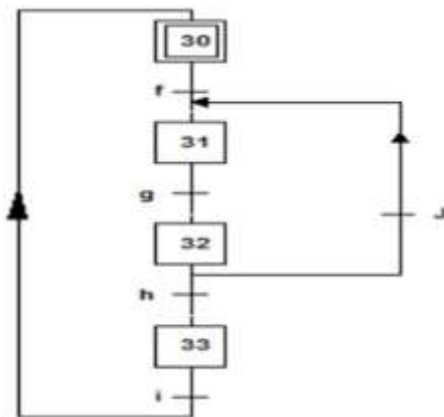


Fig. III. 3 Reprise de séquence.

### III.3 **Élaboration du GRAFCET du nouveau système**

Lors du soutirage et le chargement des camions, en cas de colmatage des évènements stérile les pompes de soutirage créent une dépression dans les bacs de stockage concernés ce qui provoque l'implosion de ces bacs de stockage ce qui conduit par la suite à des pertes du produit.

Le système de stockage actuel contient 5 bacs de stockage (T601, T602, T603, T604, T605), et deux soufflantes (PSF-610 ET PSF-620) qui sert à injecter l'air stérile dans les bacs afin de les refroidir à la fin de la phase de désinfection. Chaque bac de stockage est géré par deux grafjets ; un grafjet géré par l'automate S7-300 qui gère à la fois le lavage, la désinfection et le remplissage, et un autre grafjet qui est géré par l'automate S7-400 qui s'occupe à la fois du refroidissement des bacs ainsi de la procédure de la désinfection des soufflantes (PSF-610 et PSF-620) et d'autres tâches. Sachant que les deux automates sont interconnectés afin d'assurer

le bon déroulement du système.

Afin de remédier ce problème, on a donc pensé à automatiser le circuit d'injection d'aire dans les bacs de stockage qui va continuer à assurer d'une part le refroidissement et d'autre part la protection des bacs contre les implosions et cela en injectant l'air stérile même dans les phases de remplissage et soutirage.

L'injection d'air stérile sur pressé venant de la soufflante P-SLF permet de chasser la phase vapeur lors de la phase de refroidissement jusqu'à obtenir une température de 30 C (pour éviter le flash thermique lors du remplissage avec le produit). Lors du remplissage le volume d'aire sera évacué par l'évent stérile, en cas de colmatage de l'évent, la soupape mécanique déclenche.

Tandis que l'air qui sera injecter lors du soutirage permettra de remplacer le produit soutiré et évitera ainsi les dépressions des bacs qui provoquent leur implosion.

Pour la mise en œuvre de cette solution nous avons deux proposition :

- ✓ La première, en laissant le système telle qu'il est, c'est à dire de travailler avec les deux automates (S7-300 et S7-400) et modifier le programme du sort que ces deux automates vont gérer le circuit d'injection d'aire
- ✓ La deuxième proposition c'est d'utiliser uniquement l'automate S7-300 qui va nous permettre de gérer le nouveau système.

Pour des raisons de cout et pour éviter l'encombrement et surtout pour faire un travail bien organisé nous avons opté pour la deuxième solution, c'est-à-dire de travailler uniquement avec l'automate S7-300. Cette automate va s'occuper donc de la gestion :

- ✓ Des opérations de lavage, désinfection, refroidissement et remplissage des bacs ;
- ✓ De la procédure de désinfection des soufflantes ;
- ✓ De circuit d'injection d'air stérile dans les bacs de stockage.

Dans notre projet nous avons choisi de travailler sur un seul bac (T601) et sur une seul soufflante (PSF-610) pour des raisons de simplifier le travail. Et la solution s'applique pour tous les autres bacs.

Dans ce qui suit nous allons présenter le grafcet élaboré pour le nouveau système.

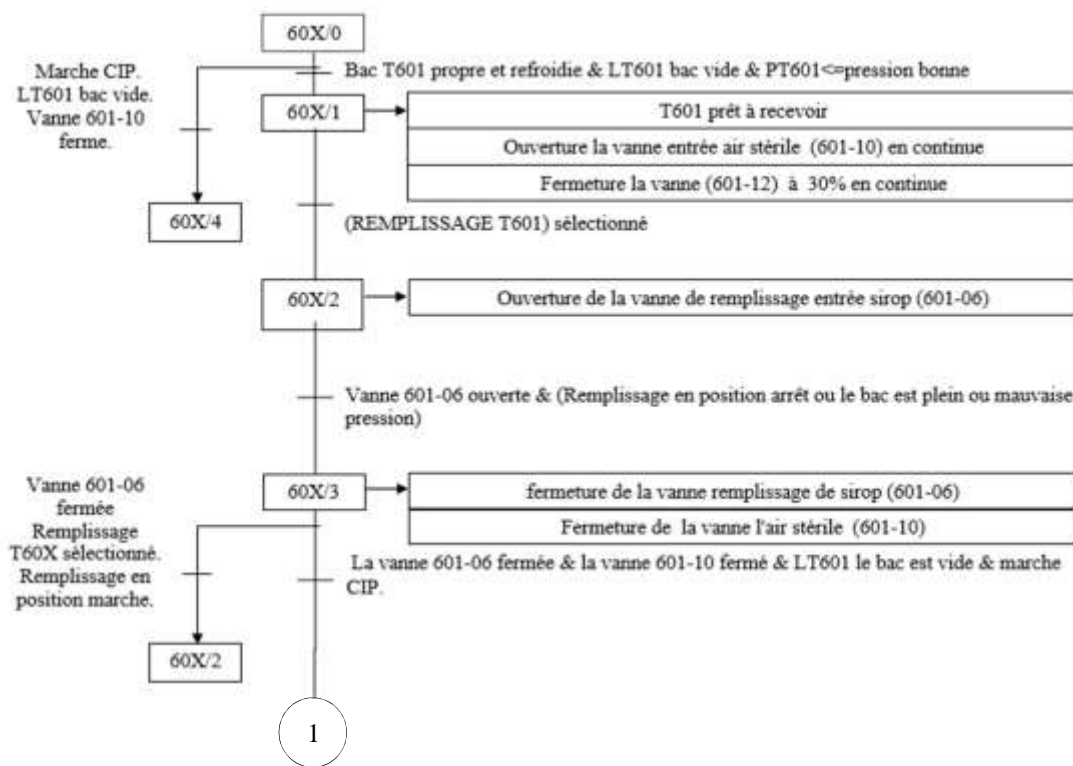
### III.3.1 Grafcet niveaux 1 pour le bac de stockage

Le grafcet qui suit présente les étapes à suivre pour effectuer les opérations de CHARGEMENT, NETTOYAGE, DESINFECTION, REFROIDISSEMENT

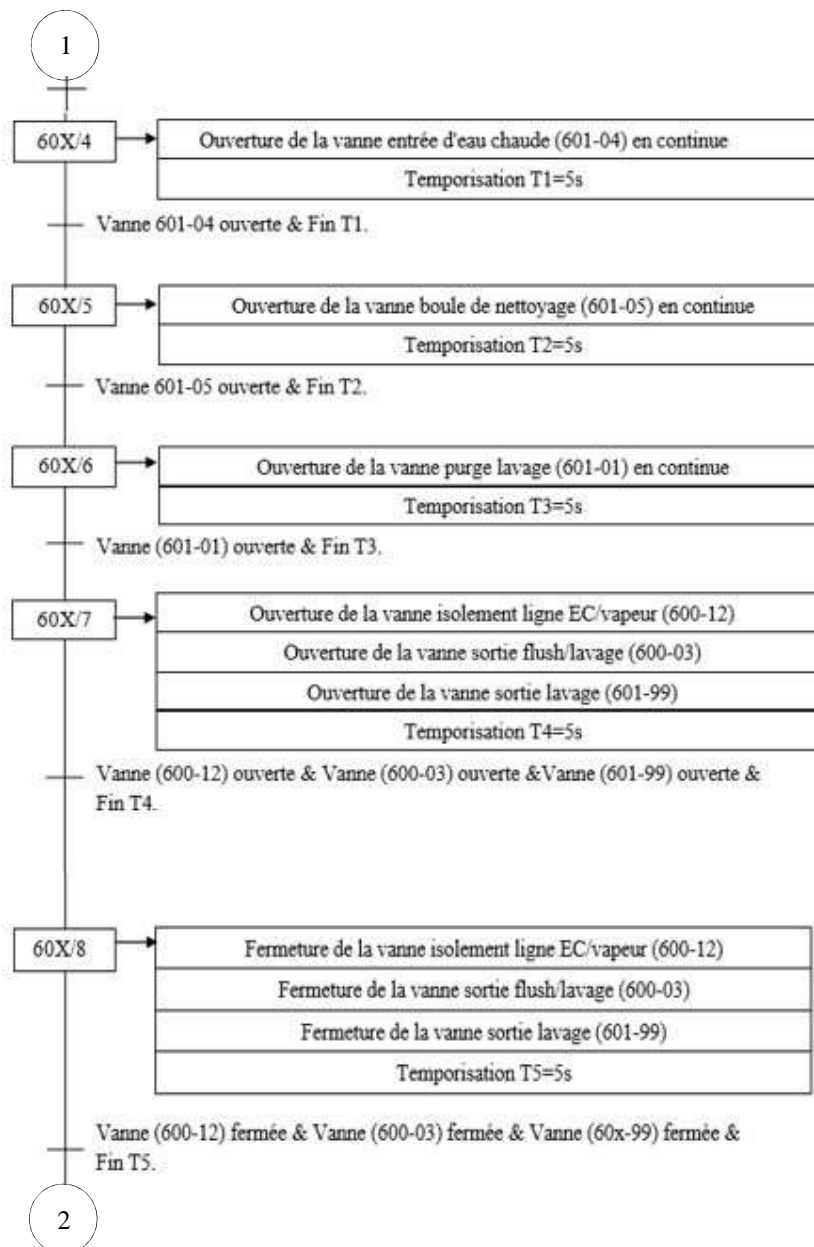
60x=601 pour T601

De l'étape 0 jusqu'à l'étape 3 présente la phase de remplissage, à l'étape 1 le bac affiche un message « prêt à recevoir » sur l'afficheur de l'opérateur, la vanne d'entrée d'air stérile 601-10 est maintenu ouvert et la vanne 601-12 qui en série avec la vanne précédente elle se ferme à 30% pour permettre l'injection d'air stérile en continue dans le bac.

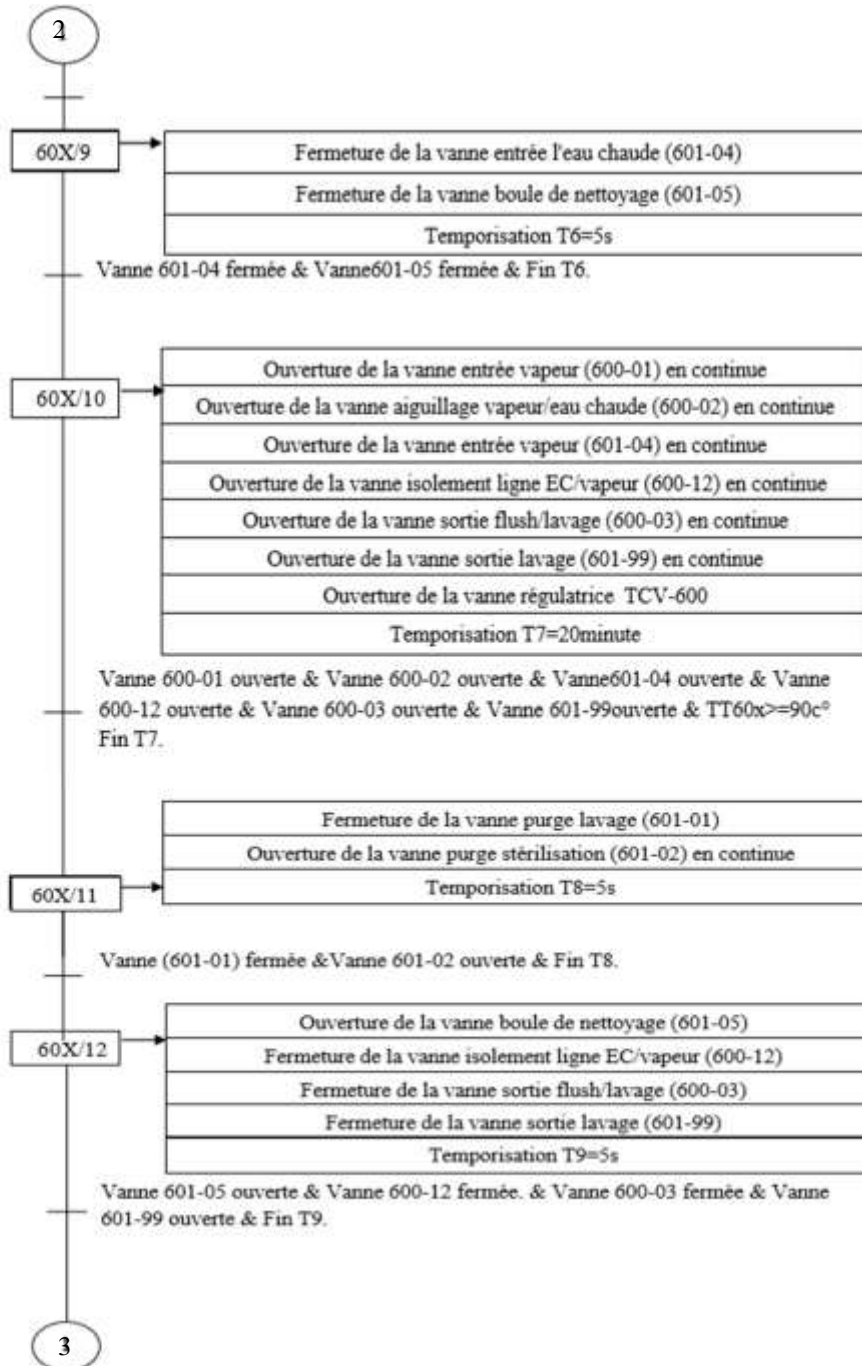
Quand l'opérateur sélectionne le bac concerné à remplir, l'ouverture de la vanne 601-06 et le remplissage s'active. Quand le niveau du produit atteint le seuil limite haut, ou quand l'opérateur presse sur le bouton « arrêt remplissage », le remplissage s'arrête par la fermeture de la vanne 601-06. Et la vanne 601-10 se ferme ainsi pour empêcher l'entrée d'air stérile dans ce bac.



De l'étape 4 jusqu'à l'étape 8 c'est la phase de lavage du bac. A l'étape 4 entrée d'eau chaud par l'ouverture de la vanne 601-04 pour une certaine temporisation paramétrable, dans l'étape 5 l'ouverture de la vanne 601-05 qui permet le ruissellement d'eau vers la boule de nettoyage qui se trouve au centre du bac, a l'étape 6 ouverture de la vanne purge lavage 601-01, après une certain temporisation l'étape 7 s'active ; ouverture des vanne 600-12,600-03 et601-99 pour évacuer l'eau du bac. Après une certaine temporisations ces vannes se ferment a l'étape 8 et le bac sera prêt pour la désinfection.

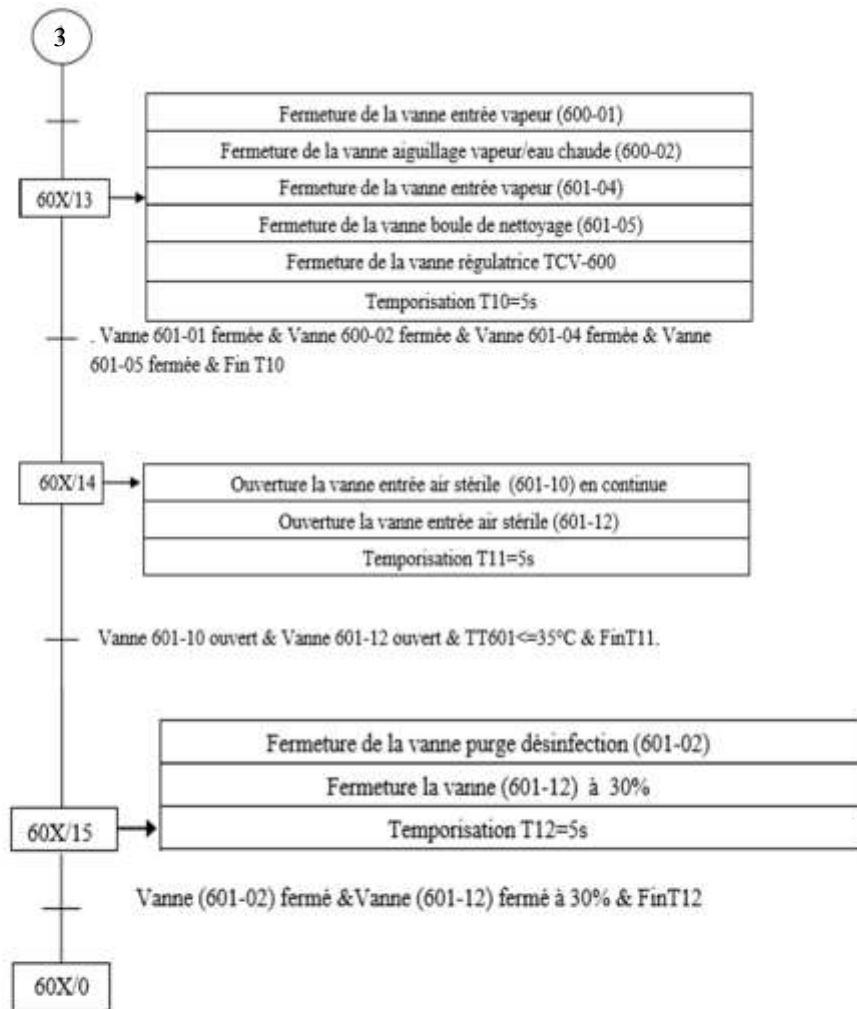


Dans cette phase de désinfection les vanne d'entre d'eau chaud 601-04 et la vanne boule de nettoyage 601-05 se ferment. Après une certaine temporisation, l'étape 10 s'active ce qui conduit à l'ouverture des vannes qui sont concernés par la désinfection (voir l'étape 10 du grafcet ci-dessous). A l'étape 11 fermeture de la vanne purge lavage et ouverture de la vanne purge stérilisation, après une certaine temporisation l'étape 12 s'active ce qui permet l'ouverture de la vanne de boule de nettoyage 601-05 et fermeture des vannes mentionnées dans l'étape 12 (voir le grafcet ci-dessous)



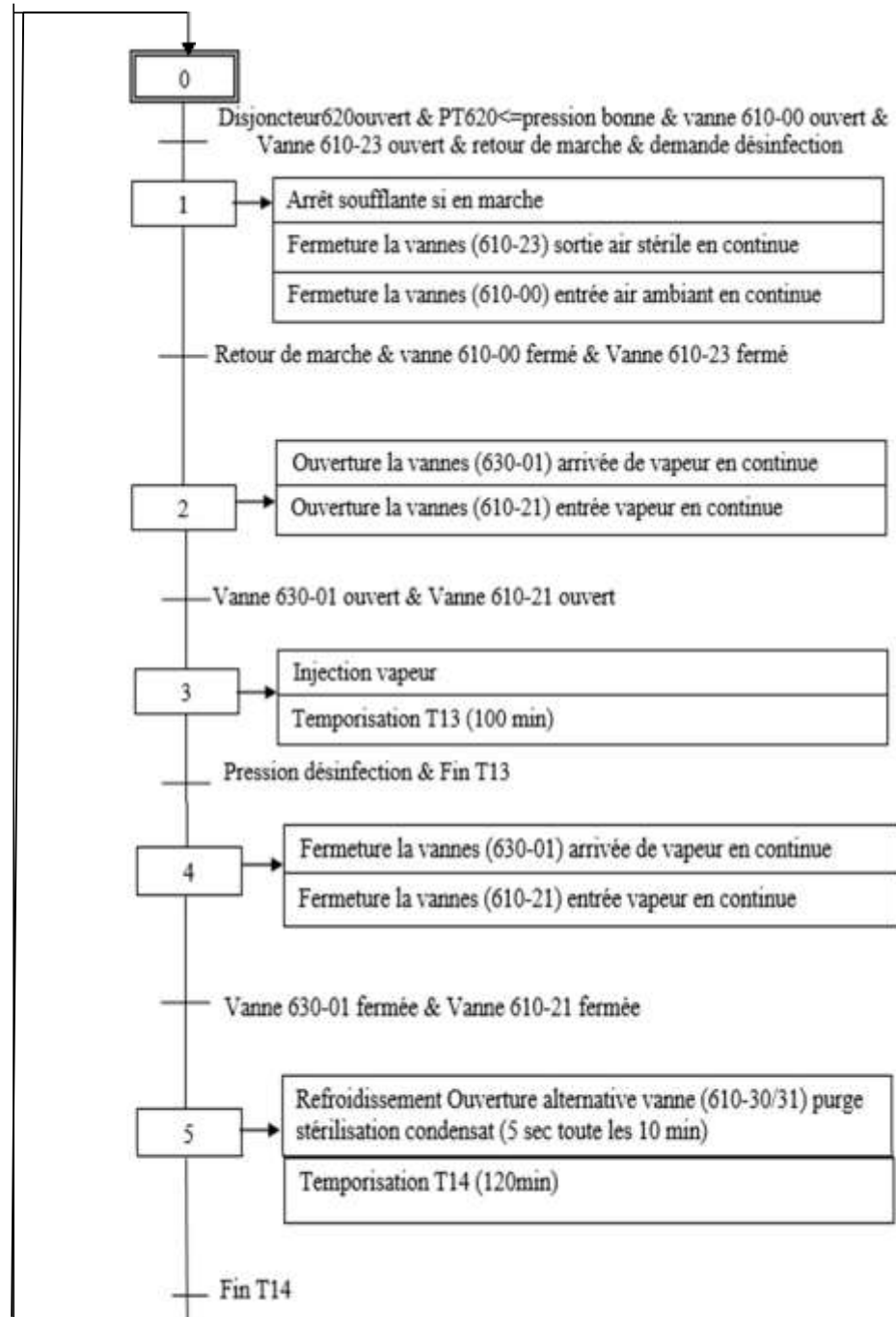


De l'étape 13 à l'étape 14 c'est la phase de refroidissement. A l'étape 13 les vannes de circuit de désinfection se ferment. Et à l'étape 14 les vanne concerné pour l'entrées d'air stérile s'ouvrent, quand la température du bac diminue à 35 degrés l'étape 15 s'active. A l'étape 15 la vanne purge désinfection 601-02 se ferme et la vanne 601-12 se ferme à 30% (voir le grafcet ci-dessous).



### III.3.2 Grafcet niveaux 1 pour la désinfection de la soufflante 610

Le grafcet qui suit présente les étapes à suivre pour la désinfection de la soufflante 610



#### **III.4 Conclusion**

Dans ce chapitre, nous avons donné les règles de bases pour l'élaboration d'un grafcet, après ensuite, nous avons présenté la solution élaborée, cette solution est présentée par un grafcet. Le premier grafcet permet le stockage du sirop dans les cinq bacs (T60X) et le deuxième grafcet décrit la procédure de désinfection de la soufflant PSF-610.

Le Grafcet nous a facilité considérablement le passage de la description à la modélisation et cela nous permettra dans le chapitre suivant de réaliser un programme en langage contact sur STEP 7 v5.

# *Chapitre IV*

## **IV.1 Introduction**

Dans le chapitre précédent, nous avons présenté le grafset élaborée pour notre nouveau système, ce grafset a facilité considérablement le passage de la description à la modélisation, ce qui nous facilitera ça programmation.

Pour piloter l'installation, nous avons besoin de réaliser un programme que nous devons intégrer dans l'automate. C'est pour cela ce présent chapitre est dédié a la programmation de ce nouveau système.

Pour la réalisation de ce programme nous allons utiliser le logiciel STEP7 que nous avons décrit dans le chapitre 2 et nous avons choisi d'utiliser le langage de schémas à contacts.

En premier lieux nous allons donner une brève présentation de la procédure suivie pour la programmation de l'automate S7-300. Ensuite, nous allons présenter le programme élaboré qui gère une fois le chargement, le lavage, la désinfection et le refroidissement du bac de stockage T601 ainsi le programme de désinfection de la soufflante 610. Enfin, nous allons conclure ce chapitre par la présentation de la supervision du système étudié

## **IV.2 Procédure suivie pour la programmation de l'automate S7-300**

### **IV.2.1 Attribution des adresses**

Pour que l'automate entre en contact avec la partie opérative et coordonne la continuité et le déroulement des étapes, des informations doivent être échangées en permanence avec la partie opérative, qui correspond à des adresses physiques sur les modules d'entrées/sorties à travers différents capteurs qui doivent être relié à l'automate dans des emplacements spécifiques.

#### **IV.2.1.1 Editeur de mnémoniques**

Il permet la gestion de toutes les variables globales. En effet il définit les désignations symboliques et les commentaires pour les signaux du processus (entrées/sorties), les mémentos, les blocs de données, les temporisations et les compteurs.

La table des mnémoniques qui en résulte est mise à disposition de toutes les applications.

Les paramètres peuvent être de type simple tel que : BOOL, WORD, INT, REAL...etc., ou de type complexe tel que : TIMER, COUNTER...etc.

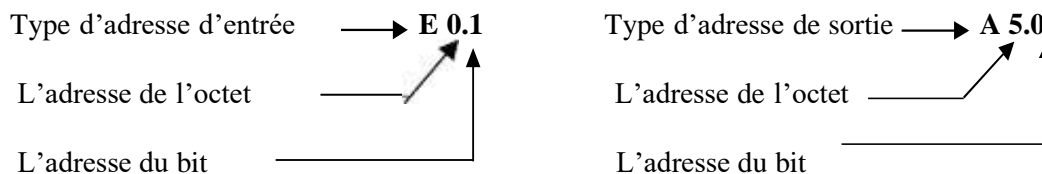
Les différents types de variables utilisés sont présentés dans le tableau (Table IV.1).

Opérations	Désignation	Type de données	Plage d'adresse	
<b>E</b>	Bit d'entrée	<b>BOOL</b>	0.0	127. 7
<b>A</b>	Bit de sortie	<b>BOOL</b>	0.0	127.7
<b>M</b>	Bit de memento	<b>BOOL</b>	0.0	255.7
<b>MB</b>	Octet de memento	<b>BYTE, CHAR (8 bit)</b>	0	255
<b>MW</b>	Mot de memento	<b>WORD, INTEGER (16 bit)</b>	0	254
<b>MD</b>	Double mot de memento	<b>WORD, REAL (32 bit)</b>	0	254
<b>T</b>	Temporisation	<b>TIMER</b>	0	127
<b>VAT</b>	Table de variable		0	127

Table IV.2 Les différents types de variables.

#### IV.2.1.2 L'adressage des modules E/S

Une entrée ou une sortie est désignée dans le programme à l'aide d'une adresse qui indique clairement quel est son emplacement sur l'automate. Pour adresser une entrée ou une sortie il faut entrer une adresse comme suit :



#### IV.2.2 Création de la table des mnémoniques

Pour améliorer la disponibilité et la clarté de notre programme, nous avons utilisé des mnémoniques à la place des adresses absolues. Pour cela nous avons créé une table de mnémoniques dans laquelle nous avons défini pour chaque opérande utilisé un nom d'adresse absolue, le type de données ainsi qu'un commentaire. Les mnémoniques ainsi définies pourront être utilisées dans l'ensemble du programme. (Voir l'annexe 1).

#### IV.3 Elaboration du programme

Après avoir créé le projet et configuré le matériel, pour commencer la programmation, nous devons créer des blocs de fonction ou insérer un langage à contacts. Chaque bloc fonction exécute une tâche bien définie dans des conditions spécifiques. Ces blocs fonctions seront assemblés dans le bloc OB1 pour former le programme principal qui sera

exécuté d'une façon cyclique.

Dans notre projet, nous avons utilisé la programmation structurée qui contient les blocs illustrés par la figure suivante

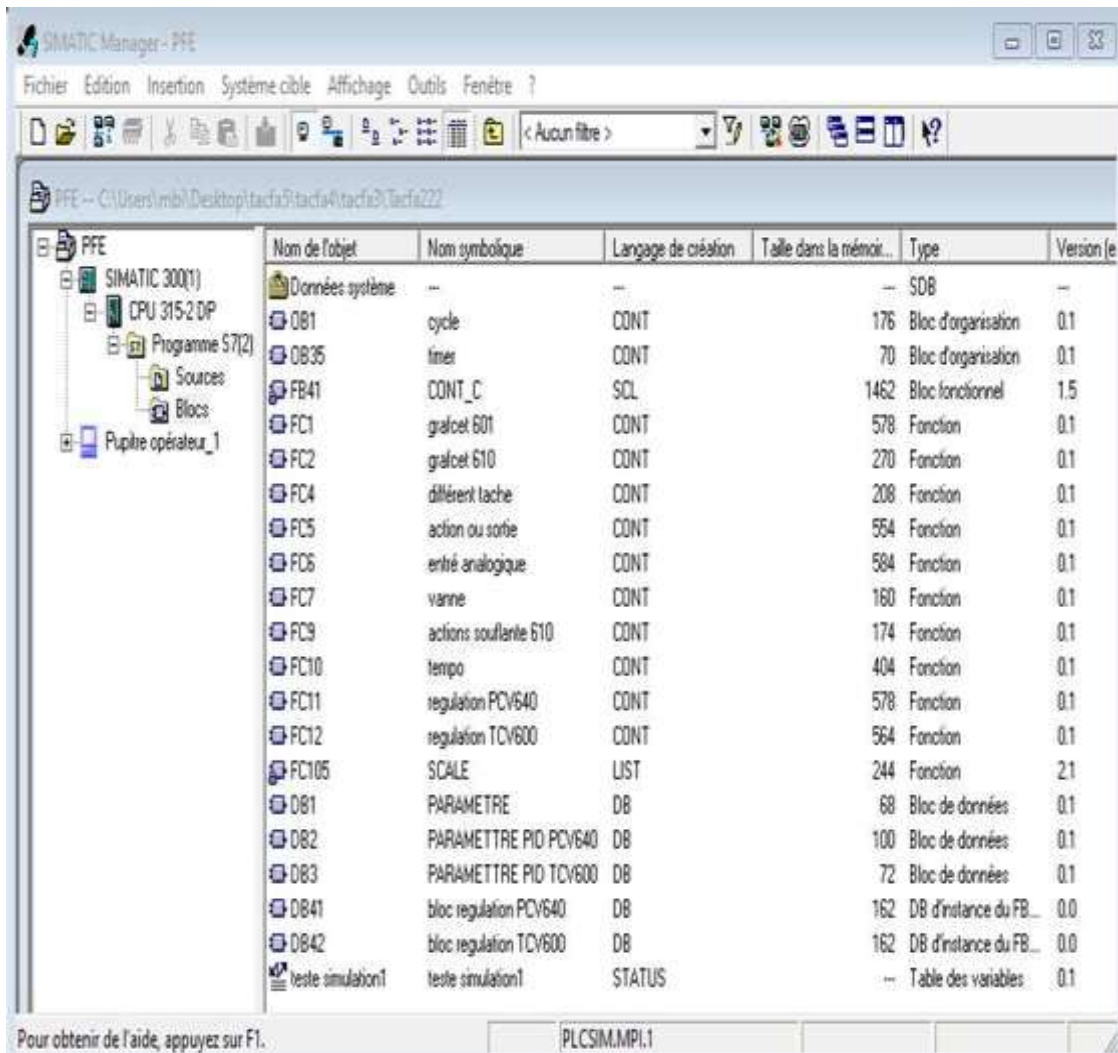


Fig. IV. 1 Blocs du projet.

### IV.3.1 Programmation des blocs

#### ➤ OB1

Le bloc d'organisation OB1 sert à l'exécution cyclique du programme utilisateur. Dans ce bloc on fait appel aux blocs fonctionnels FB ou aux fonctions FC ou à d'autres types de structures que nous avons créées. L'OB1 ne peut être appelé que par le programme système dès que l'exécution du programme. La figure IV.2 montre les fonctions appelées par ce bloc.

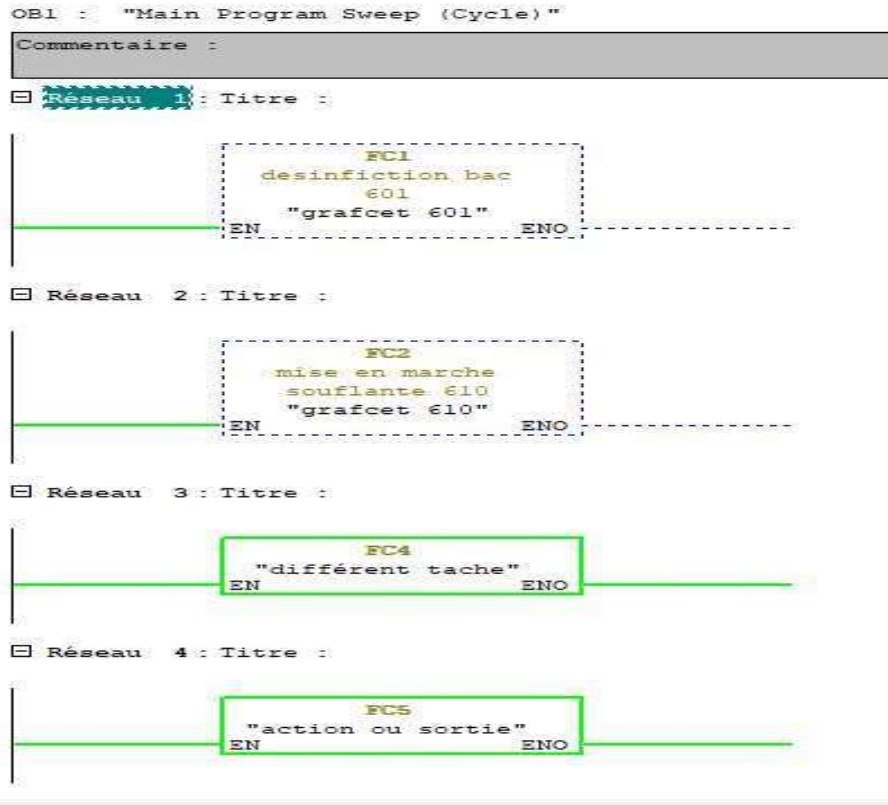


Fig. IV. 2 Les réseaux du bloc OB1.

➤ OB35

L'OB35 est parmi les OB d'alarme cyclique, ils sont mis à la disposition de l'utilisateur par les CPU S7 dont le but est d'interrompre le traitement de programme cyclique à des intervalles de temps précis. Le moment de déclenchement de la période est le passage de l'état de fonctionnement "Arrêt" (STOP) à l'état "Marche" (RUN). Dans notre programme l'OB35 fait l'appelle aux fonctions de régulation FC11 et FC 12. L'exécution de cette OB se fait chaque 100ms. La figure IV.3 illustre les réseaux de l'OB35.

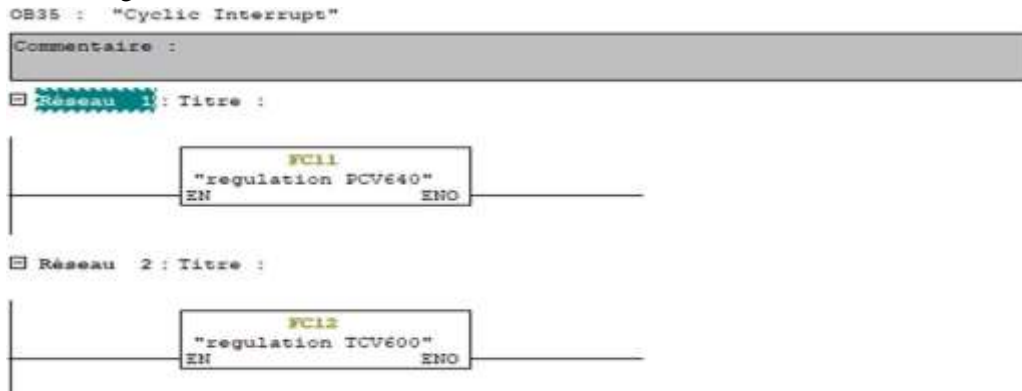


Fig. IV. 3 Les réseaux de L'OB35.



FB41

Le bloc SFB41/FB41 "CONT\_C" (Continuous Controller) sert à régler des processus industriels à grandeurs d'entrée et de sortie continues sur les automates programmables SIMATIC S7. Le paramétrage nous permet d'activer ou de désactiver des fonctions partielles du régulateur PID et donc d'adapter ce dernier au système réglé.

➤ FC1

Dans cette fonction nous avons programmé les étapes du grafcet chargement, lavage, désinfection refroidissement bac T601, elle contient 16 étapes de l'étape 0 jusqu' à l'étape 15. la figure ci-dessous présente le passage entre les étapes du grafcet du bac T601.

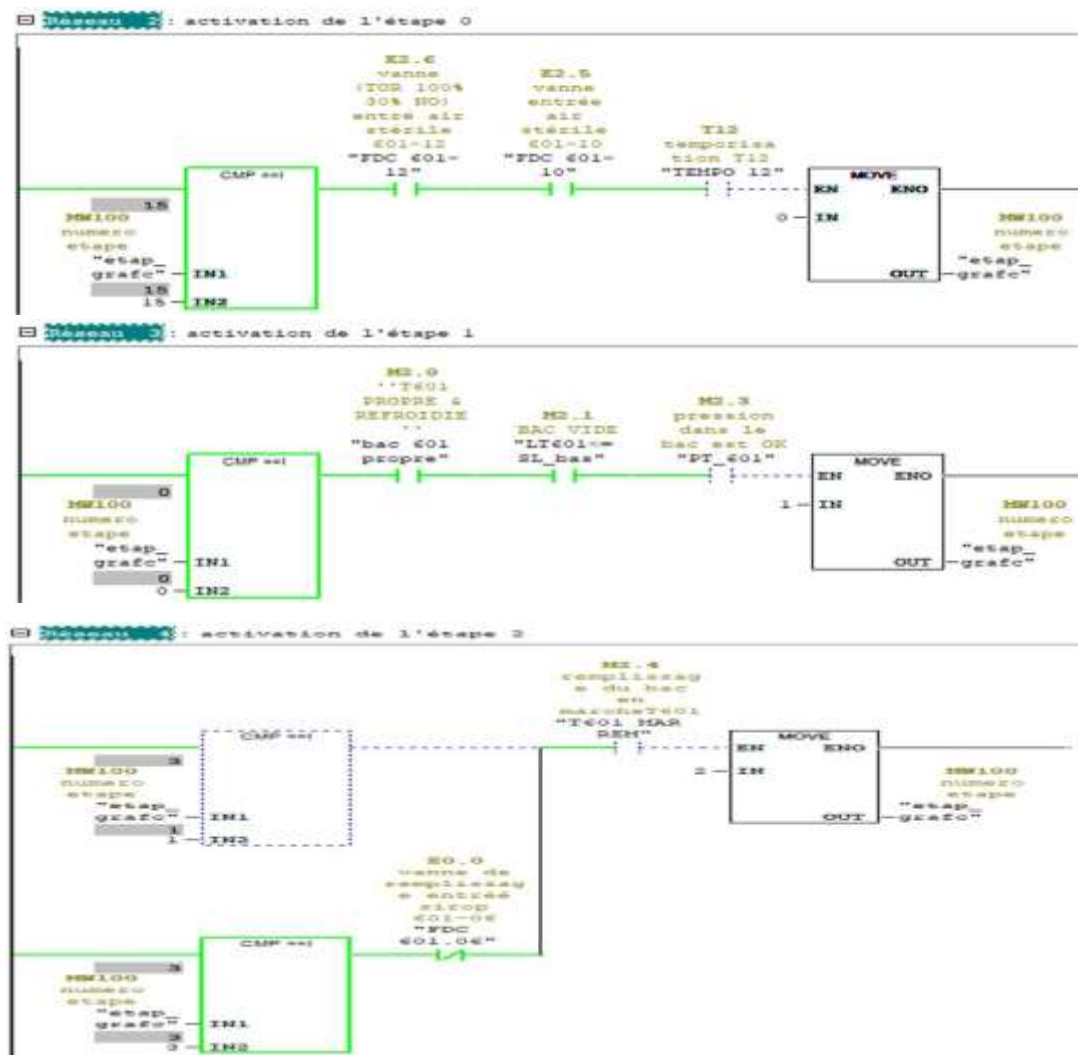


Fig.IV.4 Passage entre les étapes du grafcet BAC T601.

➤ FC2

Cette fonction contient le programme des étapes du grafcet désinfection soufflante 610, elle contient 6 étapes de l'étape 0 jusqu'à l'étape 05. La figure ci-dessous montre quelque réseau de cette fonction qui illustre l'activation des étapes du grafcet concerné.

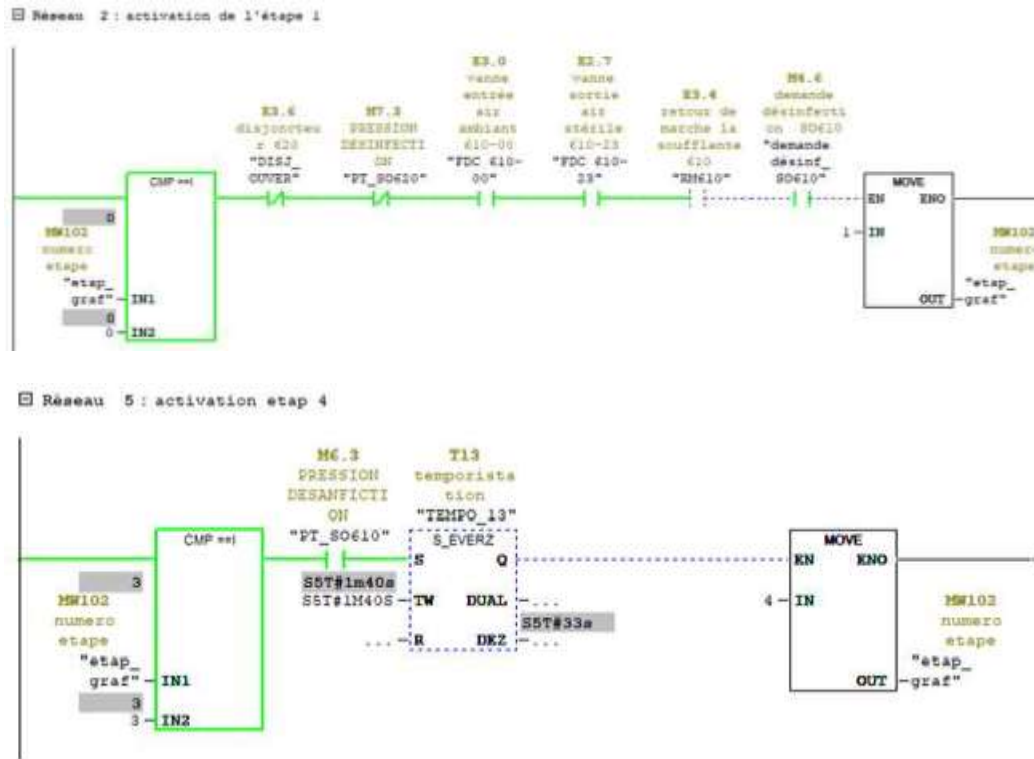


Fig.IV.5 Le passage entre les étapes du grafcet désinfection soufflante 610.

➤ FC4

Dans cette fonction nous avons programmé les différentes taches, c'est-à-dire la sélection des bacs, la mise en marche, le comptage ...etc. Les figures IV.6, IV.7 IV.8, IV.9, IV.10, IV.11, et IV.12 montrent tous les réseaux de cette fonction.

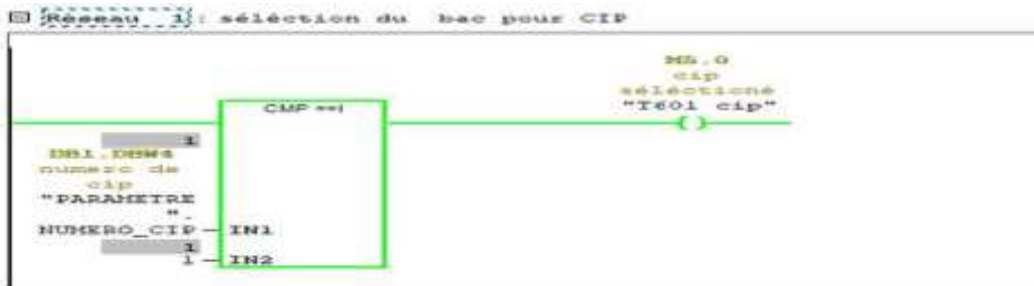


Fig.IV.6 Sélection du bac pour le nettoyage.

Réseau 2 : mise en marche cip

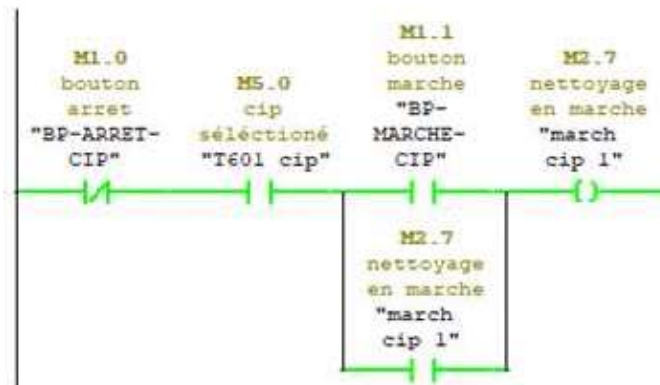


Fig.IV.7 Nettoyage bac T601 en marche.

Réseau 3 : sélectionnement bac 601 pour le remplissage

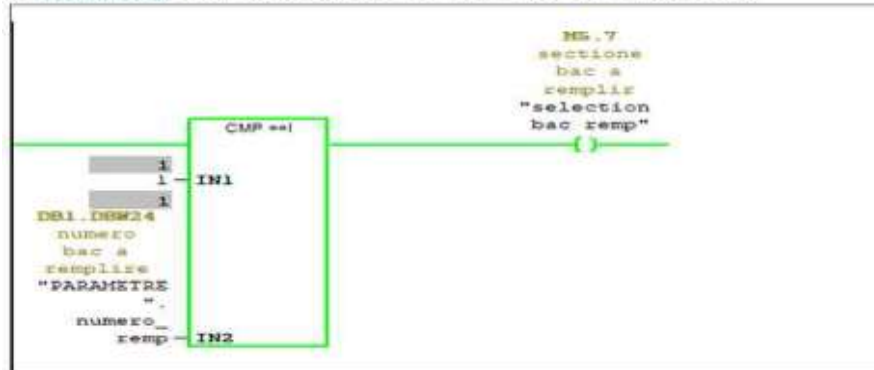


Fig.IV.8 Sectionnement bac T601 pour le remplissage.

Réseau 4 : remplissage du bac en marcheT601

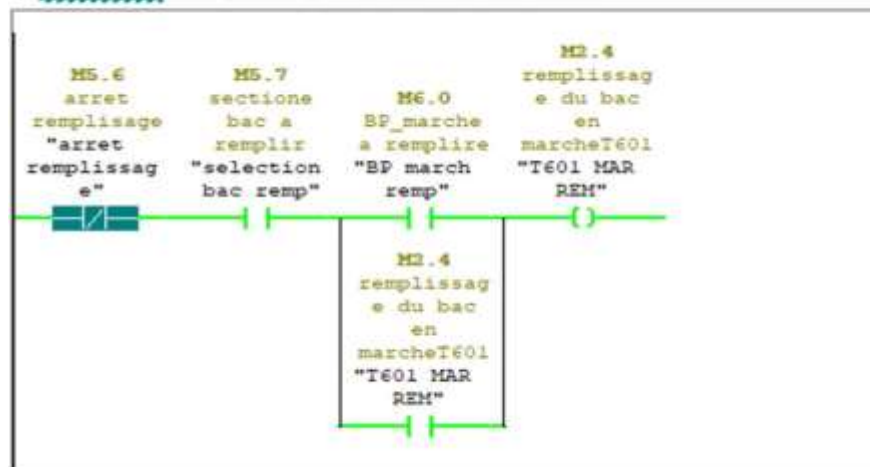


Fig.IV.9 Remplissage du bac T601 en marche.

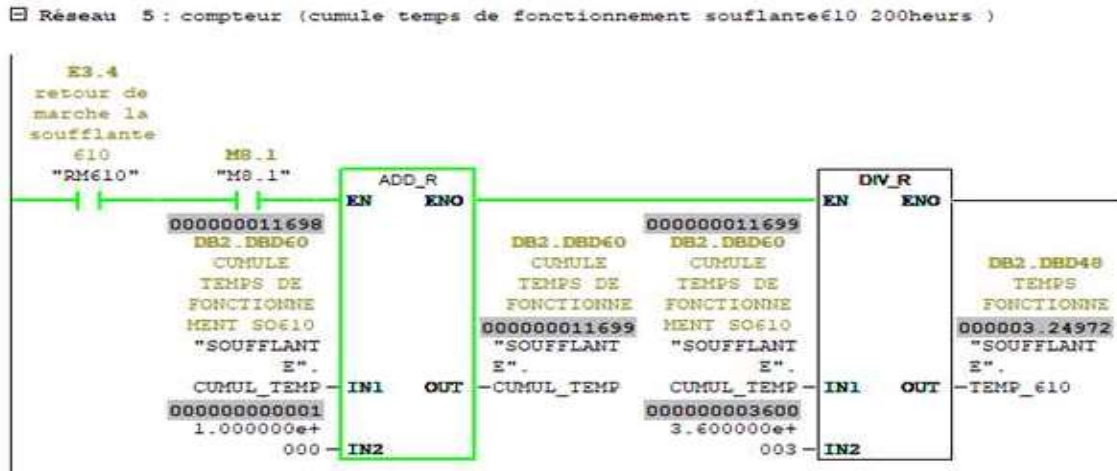


Fig.IV.10 Programme compteur cumul du temps de fonctionnement de la soufflante 610.

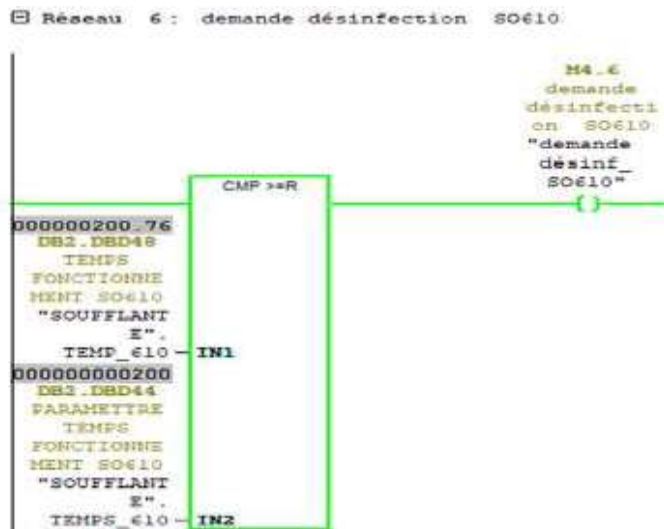


Fig.IV.11 Demande désinfection soufflante 610.

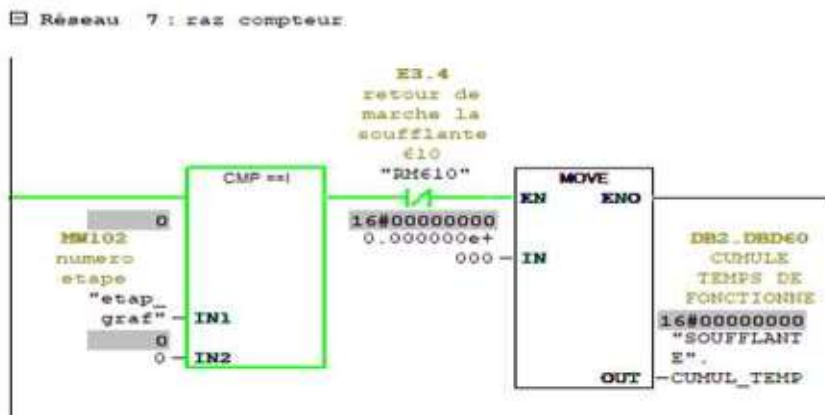


Fig.IV.12 Remise à zéros du compteur S0610.

➤ FC5

Cette fonction illustre le programme des actions correspondantes au grafcet du bac T601. La figure IV.13 montre deux réseaux de cette fonction, le premier réseau montre l'action de l'étape quatre et quinze, et le réseau sept montre les actions des étapes septe, dix et onze.

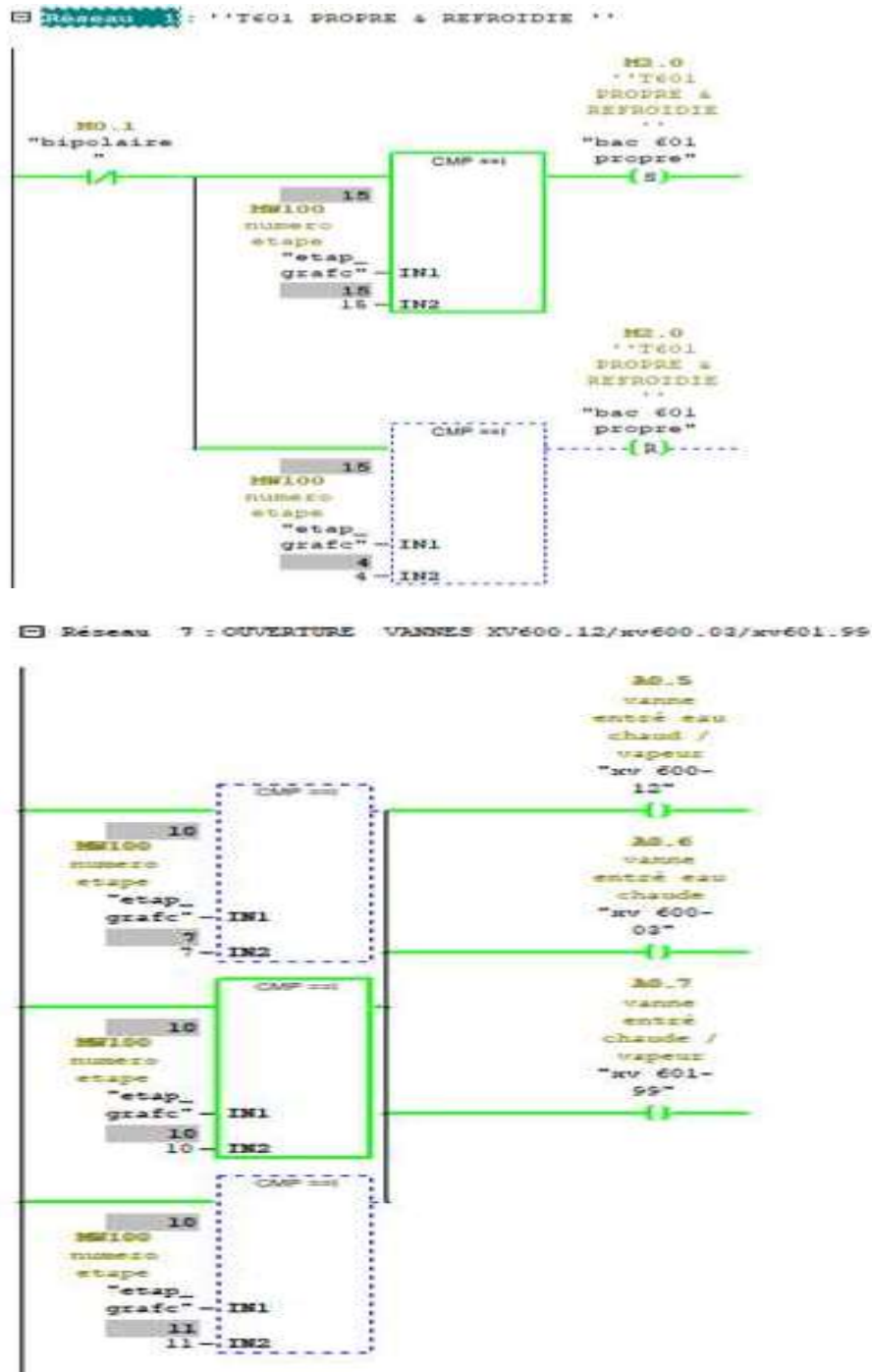


Fig.IV.13 Action ou sortie du grafcet bac T601.

➤ FC6

Dans cette fonction nous avons programmé les capteurs utilisés dans notre projet à savoir leur mise à l'échelle et les seuils paramétrables. Cependant cette fonction fait appel au bloc de données DB1 et DB2.

Les figures IV.14, IV.15, IV.16, IV.17, IV.18, IV.19 illustre le programme de chaque capteur utilisé.

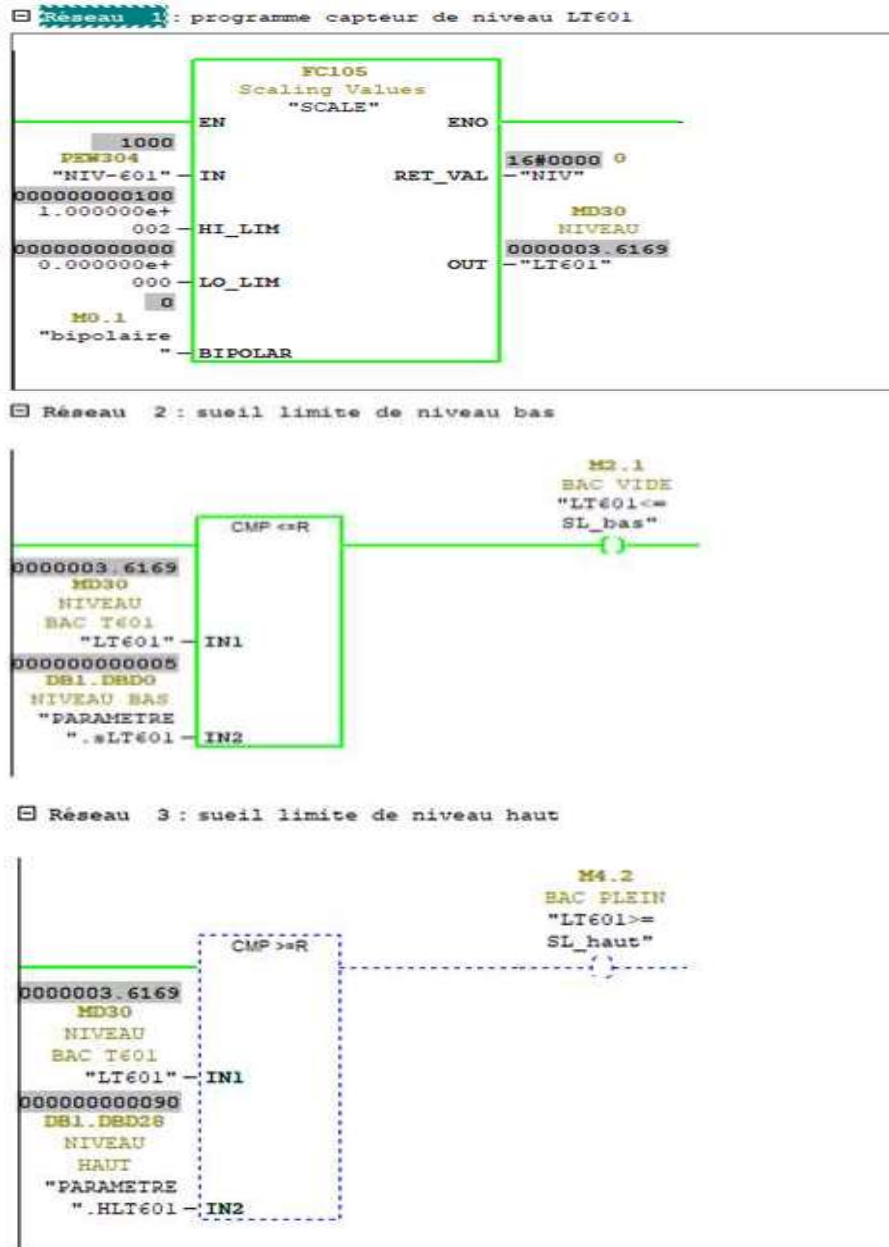
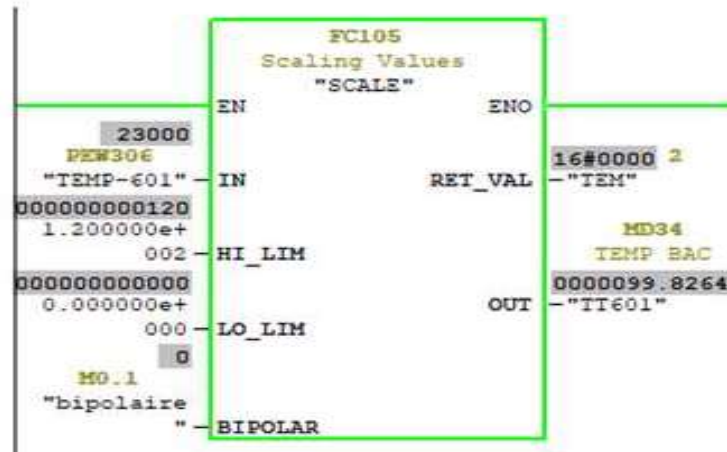
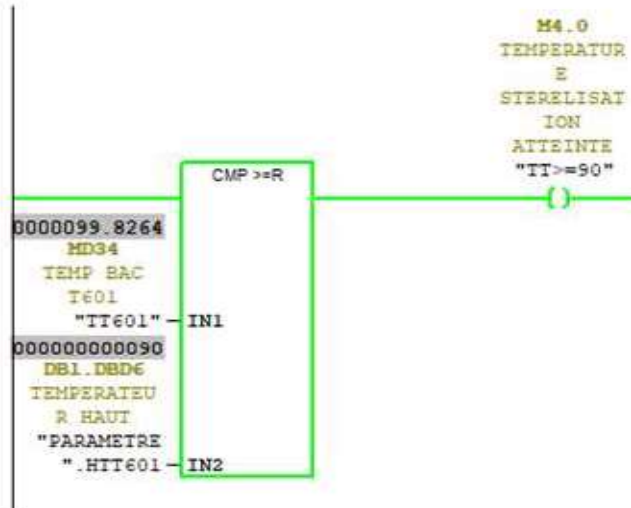


Fig.IV.14 Programme capteur du niveau du bac LT601.

▣ Réseau 4 : programme capteur de température TT601



▣ Réseau 5 : seuil de température haute



▣ Réseau 6 : seuil de température basse

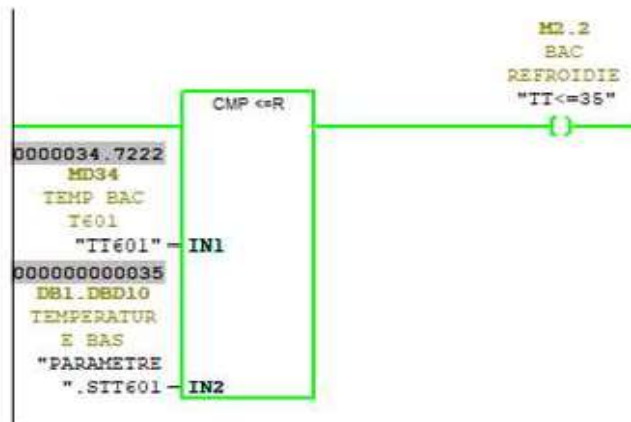
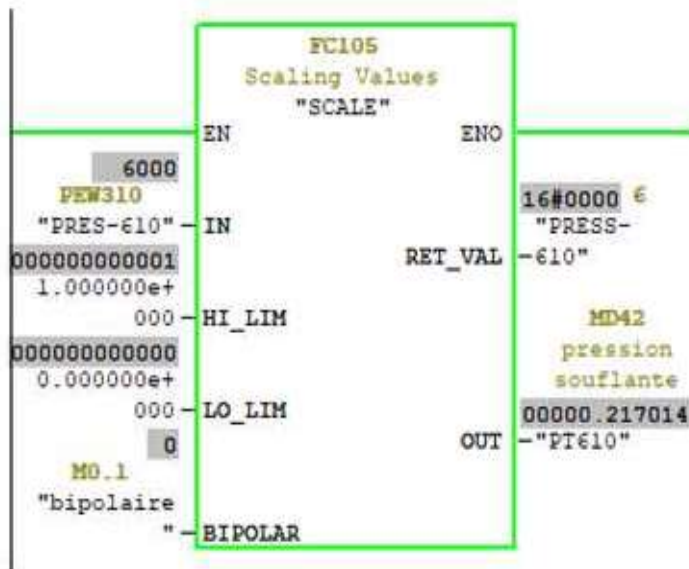


Fig.IV.15 Programme capteur de température BAC TT601.

▣ Réseau 7 : programme capteur de pression de la soufflante SO610 / PT610



▣ Réseau 8 : pression desinfection soufflante 610

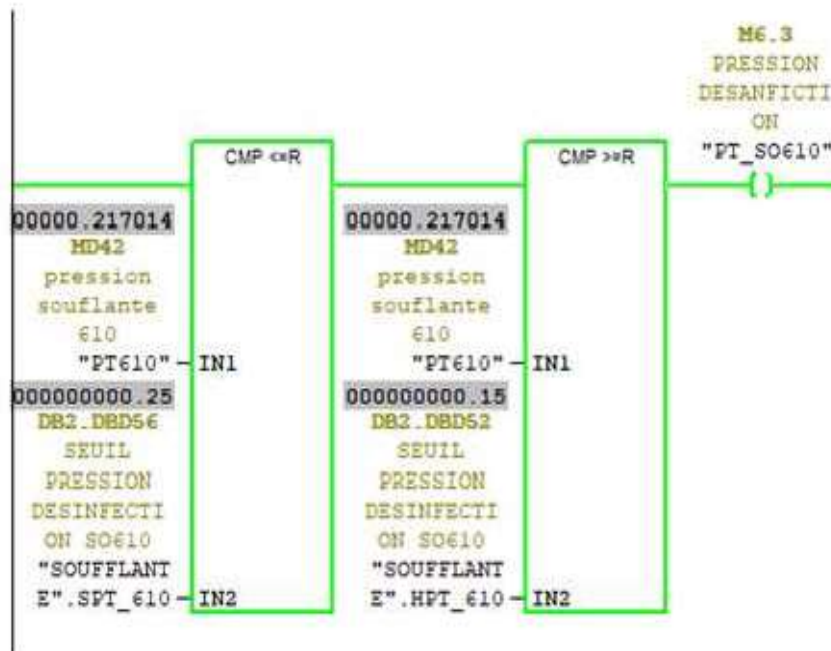
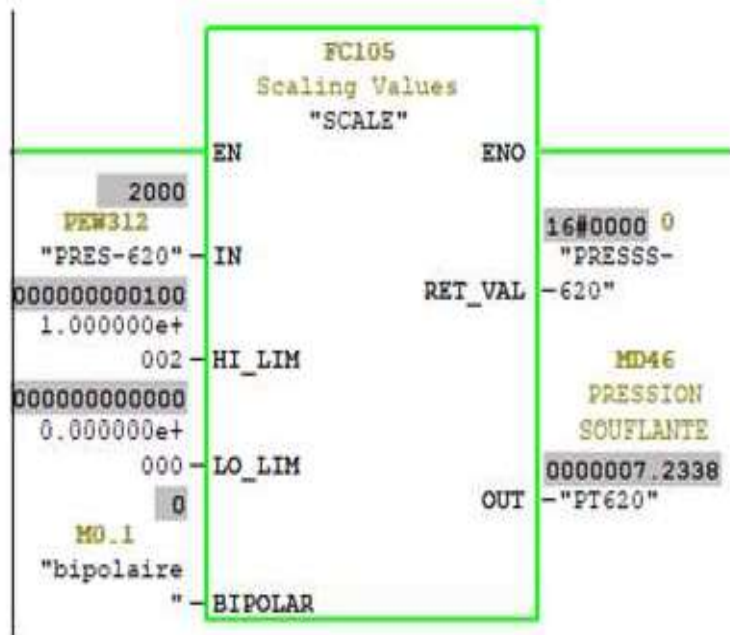


Fig.IV.16 Programme capteur de pression soufflante PT610.



▣ Réseau 9 : programme capteur de pression de la soufflante SO620 / PT620



▣ Réseau 10 : pression désinfection soufflante 620

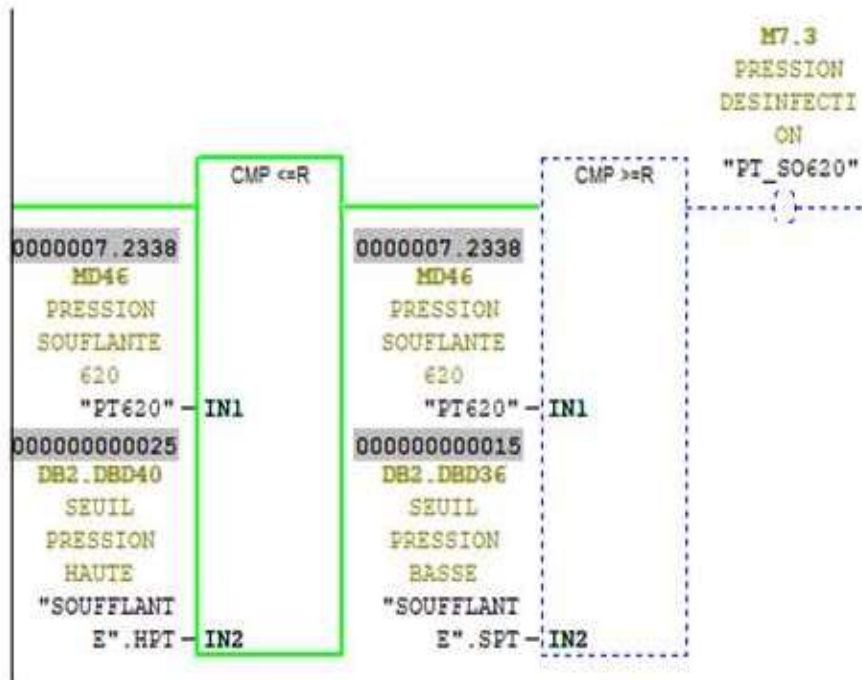
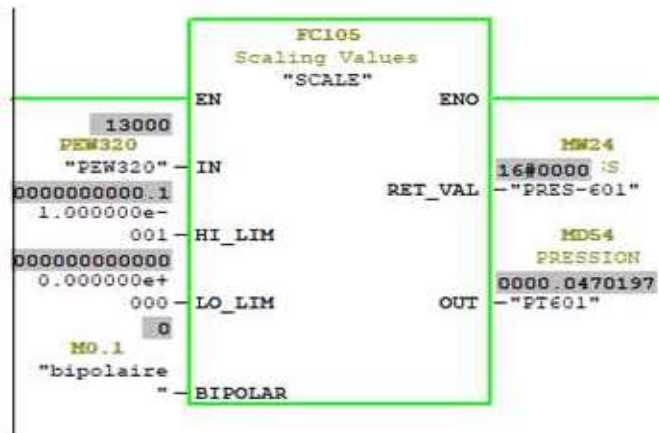


Fig.IV.17 Programme capteur de pression soufflante PT620.

▣ Réseau 12 : CAPTEUR DE PRESSION PT601



▣ Réseau 13 : SUEIL DE PRESSION PT601

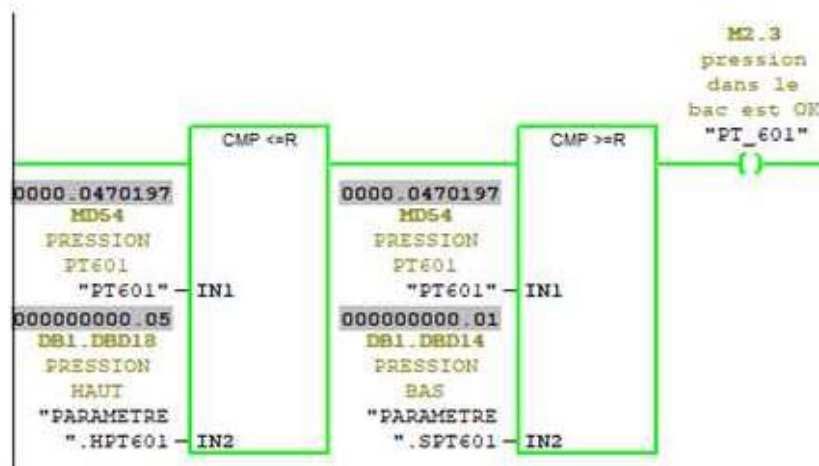


Fig.IV.18 Programme capteur de pression bac PT601.

▣ Réseau 11 : programme capteur de pression PT640

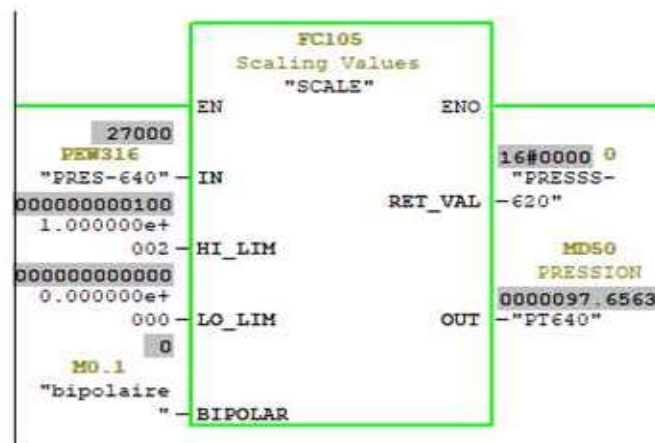


Fig.IV.19 Programme capteur de pression PT640.

➤ FC7

Dans cette fonction nous avons le programme de toutes les vannes qu'on a utilisé dans notre projet. Le réseau "un" et le réseau "quatre" de cette fonction sont illustrés dans la figure IV.20 qui est présenté ci-dessous.

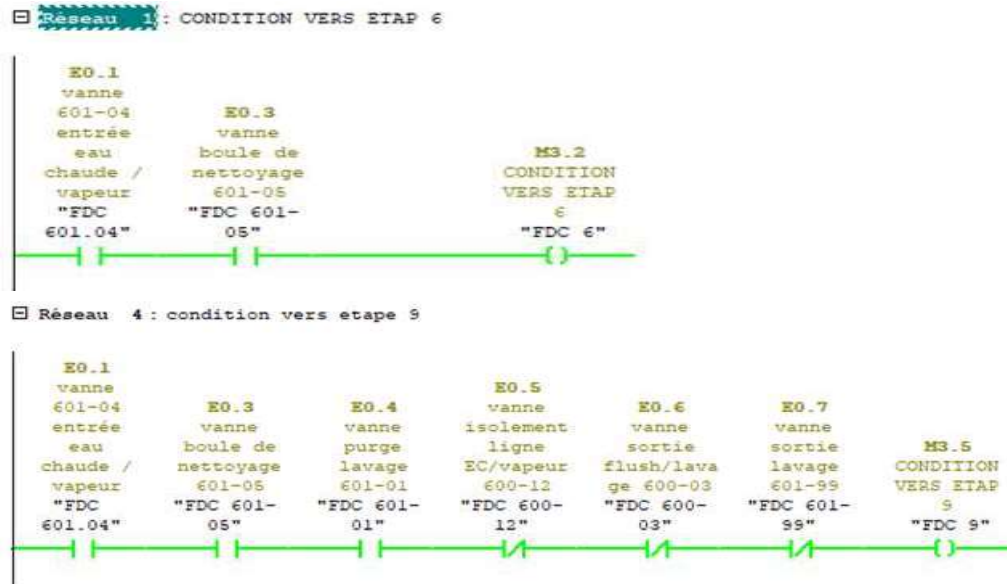


Fig.IV.20 Programmation vannes.

FC9

Quant à cette fonction, nous avons les sorti ou les actions du grafcet désinfection soufflante 610. La figure IV.21 présente les actions de l'étape "zéro" qui 'sont t activées.

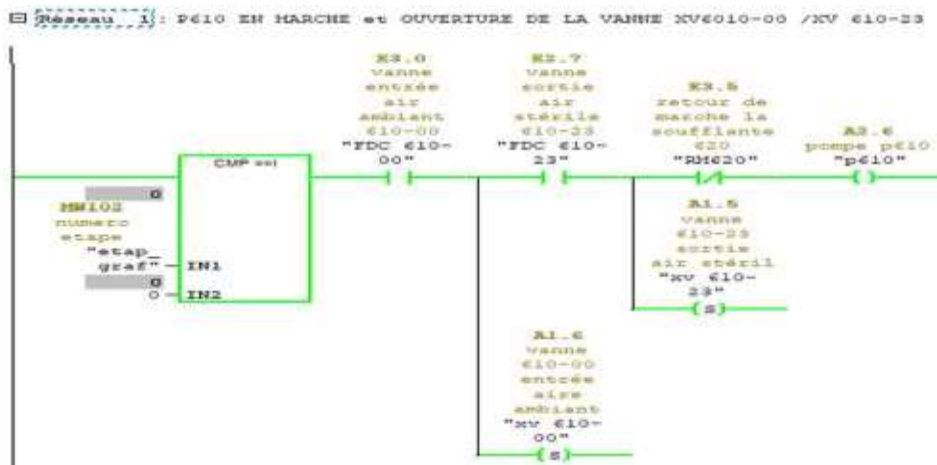


Fig.IV.21 Actions ou sorties du grafcet désinfection soufflante.

## ➤ FC10

Cette fonction contient le programme de la temporisation, nous avons illustré le réseau "un" et le réseau "7" de cette fonction dans la figure IV.22.

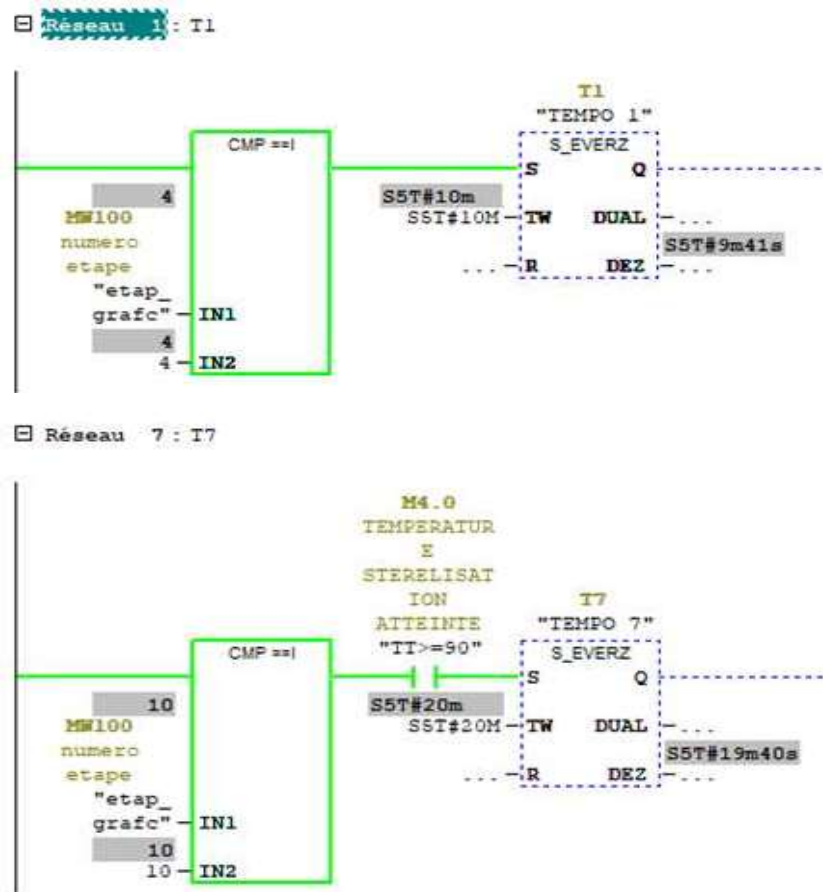


Fig.IV.22 Programme de la temporisation.

## ➤ FC11

Cette fonction contient le programme de régulation de pression PCV640. Cependant cette fonction est appelée par le bloc d'organisation OB35.

Dans le réseau 1 de cette fonction nous avons programmé les conditions d'activation de la régulation (voir figure IV.23), quant au réseau 2 nous avons programmé la consigne que nous avons donnée à la vanne régulatrice 640 (voir figure IV.24), et dans le réseau 3 se trouve les paramètres PID (Proportionnel, Intégrateur, Dérivé) qu'on va injecter dans le régulateur (voir figure IV.25). Dans le réseau 4 nous avons la mesure captée par le capteur de pression PT640 (voir figure IV.26) et en fin dans le réseau 5 nous avons le régulateur lui-même dont nous avons

Programmé ces entrées et ces sorties en injectant les paramètres de régulation (le gain, l'intégrateur, le dérivateur) que nous avons vue dans le réseau 3 (voir figure IV.27).

Ce régulateur en fonction de la consigne qu'on lui donne, et la mesure captée par le capteur PT640, il va nous générer un signal de sortie "PAW336" qui va commander l'ouverture et la fermeture de la vanne régulatrice PCV640.

Le tableau de (l'annexe 2) nous donne des informations utiles pour le bon paramétrage des entrées sortis de ce régulateur.

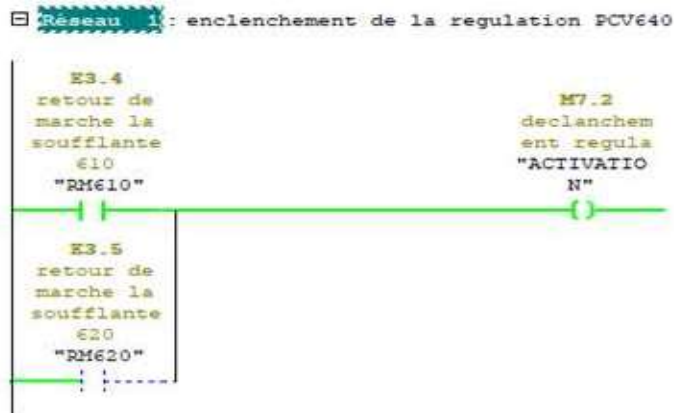


Fig. IV. 23 Activation de la régulation

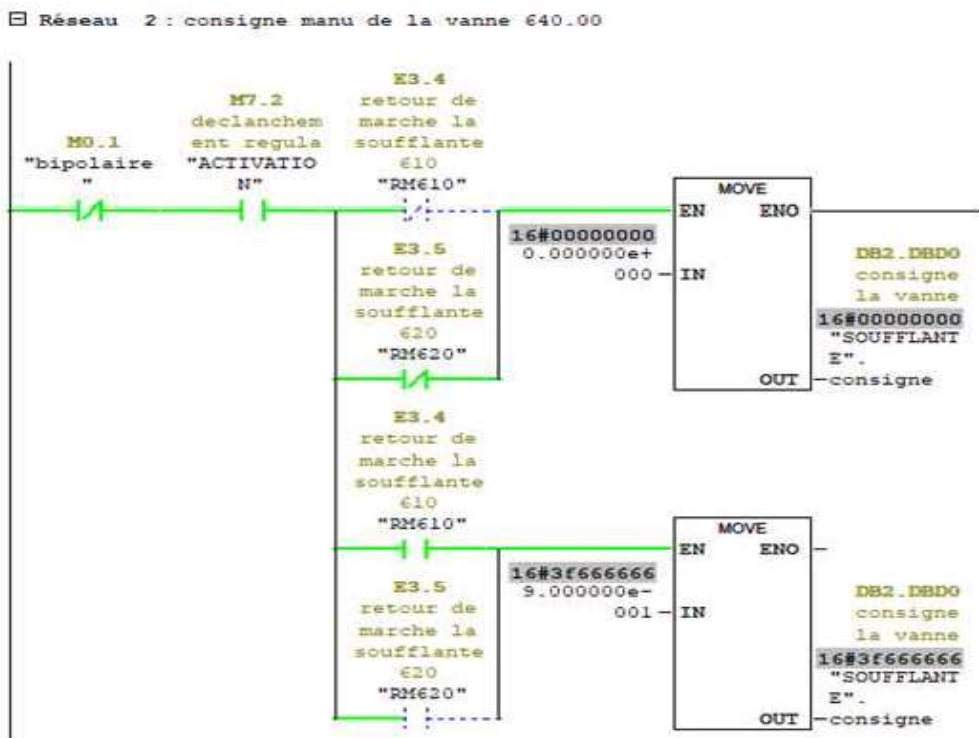


Fig.IV.24 Programmation de la consigne.

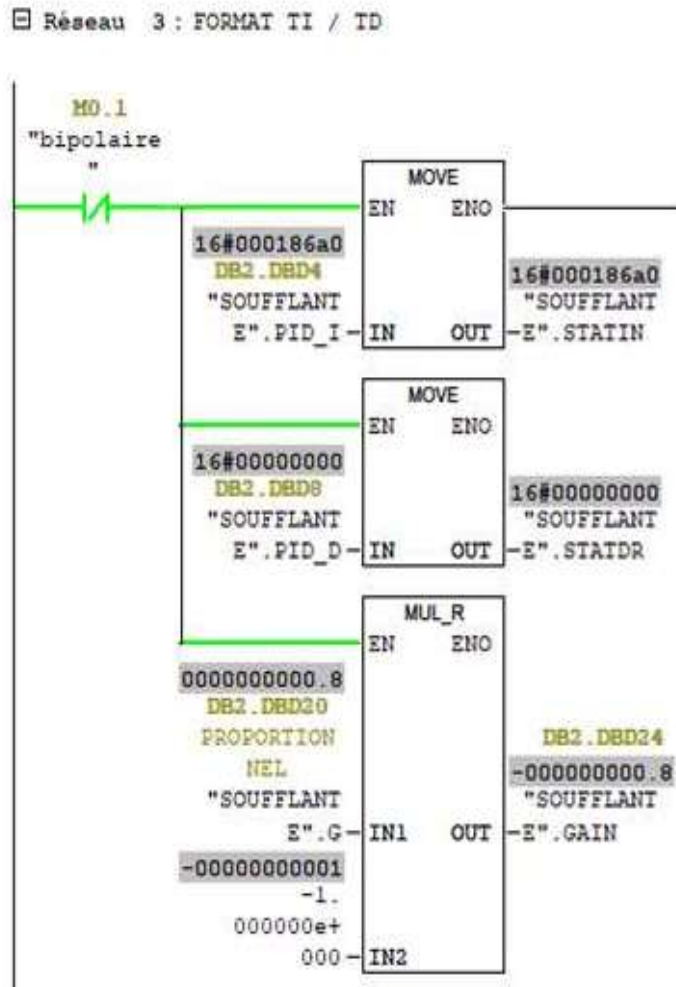


Fig.IV.25 Programmation des paramètres PID (format TI, TD, GAIN).

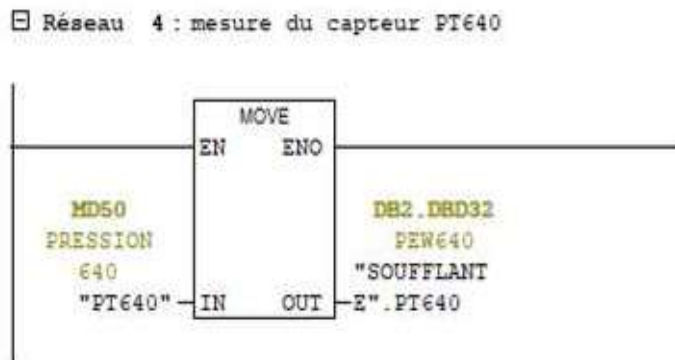


Fig.IV.26 Mesure du capteur de pression PT640.



Fig.IV.27 Programmation et paramétrage du régulateur PID.

➤ FC12

Comme la fonction précédente, cette fonction contient le programme de régulation de température TCV600. Cependant cette fonction est appelée par le bloc d'organisation OB35. Dans le réseau 1 de cette fonction nous avons programmé les conditions d'activation de la régulation (voir figure IV.28), quant au réseau 2 nous avons programmé la consigne que nous avons donnée à la vanne régulatrice TCV600 (voir figure IV.29), et dans le réseau 3 se trouve les paramètres PID (Proportionnel, Intégrateur, Dérivé) qu'on va injecter dans le régulateur (voir figure IV.30). Dans le réseau 4 nous avons la mesure captée par le capteur de température TT601 (voir figure IV.31) et en fin dans le réseau 5 nous avons le régulateur lui-même dont nous avons programmé ces entrées et ces sorties en injectant les paramètres de régulation (le gain, l'intégrateur, le dérivateur) que nous avons vue dans le réseau 3 (voir figure IV.32). Ce régulateur en fonction de la consigne qu'on lui donne, et la mesure captée par le capteur TT601, il va nous générer un signal de sortie "PAW338" qui va commander l'ouverture et la fermeture de la vanne régulatrice TCV600.



Fig.IV.28 Activation de la régulation TCV600.

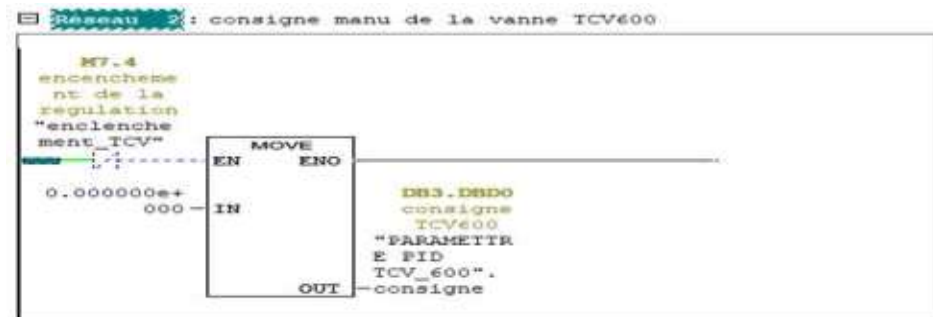


Fig.IV.29 Programmation de la consigne.



☐ Réseau 3 : parametres PID (GAIN, TI, TD)

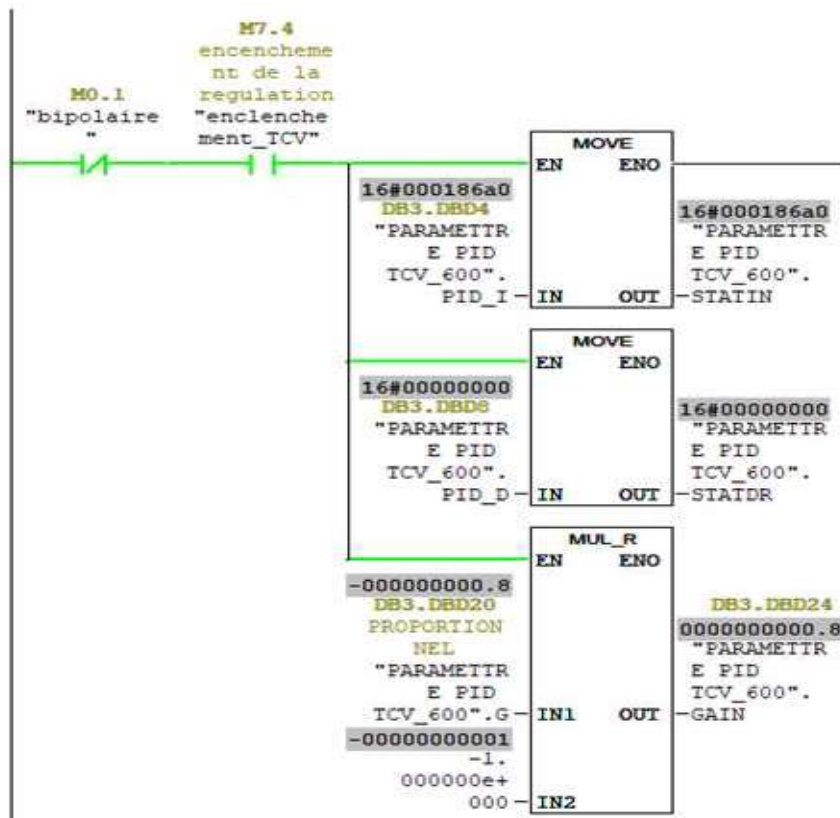


Fig.IV.30 Programmation des paramètres PID (format TI, TD, GAIN).

☐ Réseau 4 : RETOUR CAPTEUR ( la valeur mesurée)

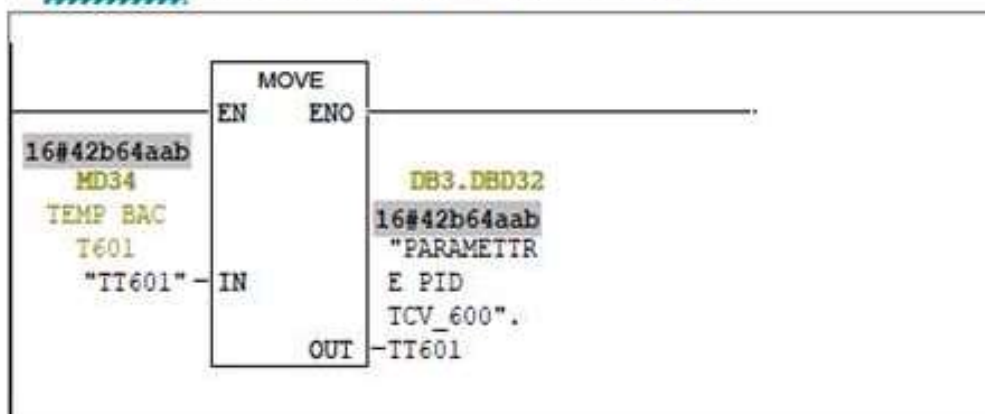


Fig.IV.31 Mesure du capteur de température TT601.



Fig.IV.32 Programmation et paramétrage du régulateur PID.

➤ FC105

La fonction Mise à l'échelle (SCALE) prend une valeur entière (IN) et la convertit en une valeur réelle exprimée en unités physiques, comprises entre une limite inférieure (LO\_LIM) et une limite supérieure (HI\_LIM).

➤ DB1

Ce bloc de données contient les paramètres du programme du BAC T601, elles sont présentées dans la figure suivante.

Adresse	Nom	Type	Valeur initiale	Commentaire
0.0		STRUCT		
+0.0	sLT601	REAL	0.000000e+000	NIVEAU BAS
+4.0	NUMERO_CIP	INT	0	numero de cip
+6.0	HTI601	REAL	0.000000e+000	TEMPERATEUR HAUT
+10.0	STI601	REAL	0.000000e+000	TEMPERATURE BAS
+14.0	SPT601	REAL	0.000000e+000	PRESSION BAS
+18.0	HPT601	REAL	0.000000e+000	PRESSION HAUT
+22.0	DB_VAR	INT	0	Variable temporaire de réservation
+24.0	numero_remp	INT	0	numero bac a remplir
+26.0	DM	INT	0	Demande desinfection
+28.0	HLT601	REAL	0.000000e+000	NIVEAU HAUT
+32.0		END_STRUCT		

Fig. IV. 33 Bloc de données paramètres bac T601

➤ DB2, DB3

Le bloc de donnée DB2 contient les paramètres de régulation PCV640, quant au bloque DB3 il contient les paramètres de régulation TCV 600.les figures ci-dessous montre l'ensemble de ces paramètres.

Adresse	Nom	Type	Valeur initiale	Commentaire
0.0		STRUCT		
+0.0	consigne	REAL	0.000000e+000	consigne la vanne regulation
+4.0	PID_I	DINT	L#0	
+8.0	PID_D	DINT	L#0	
+12.0	STATIN	TIME	T#0MS	
+16.0	STATDR	TIME	T#0MS	
+20.0	G	REAL	0.000000e+000	PROPORTIONNEL
+24.0	GAIN	REAL	0.000000e+000	GAIN
+28.0	sortie_vanne	REAL	0.000000e+000	ouverture la vanne regulatrice
+32.0	PI640	REAL	0.000000e+000	PI640
+36.0	SPT	REAL	0.000000e+000	SEUIL PRESSION BASSE
+40.0	HPT	REAL	0.000000e+000	SEUIL PRESSION HAUTE
+44.0	TEMPS_610	REAL	0.000000e+000	PARAMETRE TEMPS FONCTIONNEMENT SO610
+48.0	TEMD_610	REAL	0.000000e+000	TEMPS FONCTIONNEMENT SO610
+52.0	HPT_610	REAL	0.000000e+000	SEUIL PRESSION DESINFECTION SO610
+56.0	SPT_610	REAL	0.000000e+000	SEUIL PRESSION DESINFECTION SO610
+60.0	CURSUL_TKMP	REAL	0.000000e+000	CURSULE TEMPS DE FONCTIONNEMENT SO610
+64.0		END_STRUCT		

Fig. IV. 34 Paramètres PCV640

Adresse	Nom	Type	Valeur initiale	Commentaire
0.0		STRUCT		
+0.0	consigne	REAL	0.000000e+000	consigne TCV600
+4.0	PID_I	DINT	L#0	
+8.0	PID_D	DINT	L#0	
+12.0	STATIN	TIME	T#0MS	
+16.0	STATDR	TIME	T#0MS	
+20.0	G	REAL	0.000000e+000	PROPORTIONNEL
+24.0	GAIN	REAL	0.000000e+000	GAIN
+28.0	sortie_vanne	REAL	0.000000e+000	ouverture la vanne regulatrice
+32.0	IT601	REAL	0.000000e+000	FEW601
=36.0		END_STRUCT		

Fig. IV. 35 Paramètres de régulation TCV600.

#### IV.4 SUPERVISION

La supervision est une technique industrielle, elle sert à représenter et à surveiller l'état de fonctionnement d'un procédé et le pilotage informatique. Elle consiste à l'acquisition de données et des paramètres de commande des processus, fournir aux opérateurs les informations qui permettent de prendre, au bon moment, les bonnes décisions

#### IV.5 Étape de mise en œuvre

Pour créer une interface Homme/Machine, il faut prendre connaissance des éléments de l'installation ainsi que le logiciel de programmation de l'automate utilisé. L'interface de supervision est réalisée sous STEP 7 comme suite [9] :

Dans la fenêtre principale du projet, en cliquant sur "Insertion" puis sur "Station SIMATIC IHM", une fenêtre s'ouvre et permet le choix de l'écran à utiliser dans la supervision et nous avons choisis le panel ("MP 370 15" Touch).

##### IV.5.1 Établir une liaison directe

La première chose à effectuer est de créer une liaison directe entre WinCC et notre automate. Ceci dans le but que WinCC puisse aller lire les données qui se trouvent dans la mémoire de l'automate. Après avoir créé notre projet sous WinCC, nous cliquons sur l'onglet liaison afin de créer une nouvelle liaison que nous nommerons « liaison », nous indiquons ensuite ses différents paramètres :

- Interface : MPI/DP : Notre automate est relié par un MPI-DP,
- Adresse : Permet de spécifier l'adresse de la station, dans ce cas l'adresse MPI.

La figure IV.36 ci-dessous représente une vue de la liaison entre la station et le panel MP 370 15" Touch (pupitre).

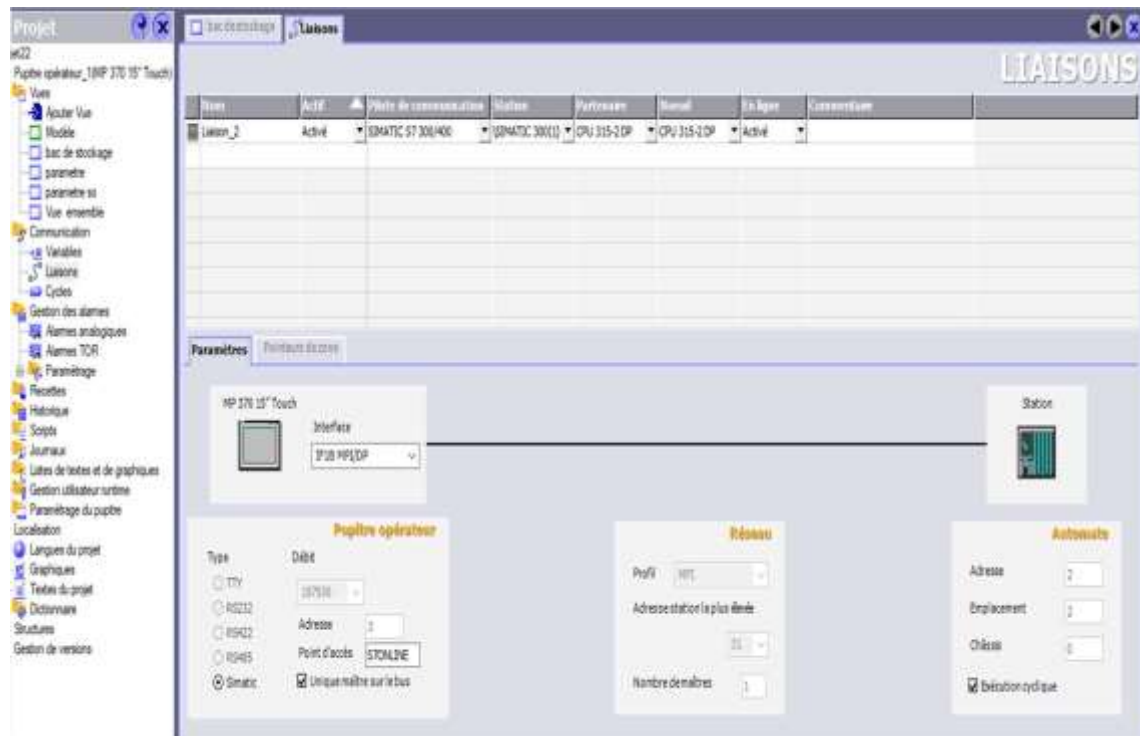


Fig. IV.36 Liaison entre la station et le pupitre

**IV.5.2 Création de la table des variables :**

La liaison entre notre projet WinCC flexible et l'automate est établie, il est donc possible d'accéder à toutes les zones mémoires de l'automate, Une table de correspondance des variables IHM est créé à travers l'onglet variables, chaque ligne correspond à une variable de l'IHM qui est spécifiée par : nom, type de données, mnémonique, adresse, et le commentaire [9].

La figure IV.37 représente les variables de l'IHM

Nom	Connexion	Type de données	Mnémonique	Adresse	Intervalle du tableau	Cycle d'écoulement	Commentaire
VDC 600-01	Liaison_2	Bool	VDC 600-01	I 1.6	1	1 s	vanne entrée vapeur 600-01
L7001	Liaison_2	DWord	L7001	MD 30	1	100 ms	NEVEAU BAC 7001
p630	Liaison_2	Bool	p630	Q 2.6	1	1 s	pompe p630
PT_601	Liaison_2	Bool	PT_601	M 2.3	1	1 s	pression dans le bac est OK
T7001	Liaison_2	DWord	T7001	MD 34	1	1 s	TEMP BAC 7001
xv 600-01	Liaison_2	Bool	xv 600-01	Q 5.0	1	1 s	vanne entrée vapeur
xv 600-02	Liaison_2	Bool	xv 600-02	Q 5.1	1	1 s	vanne a trois voies , entrée vapeur
xv 601-04	Liaison_2	Bool	xv 601-04	Q 0.2	1	1 s	vanne entrée eau chaude / vapeur
xv 601-05	Liaison_2	Bool	xv 601-05	Q 0.3	1	1 s	vanne boule de nettoyage
xv 601-06	Liaison_2	Bool	xv 601-06	Q 0.8	1	1 s	vanne de remplissage 5L
xv 601-10	Liaison_2	Bool	xv 601-10	Q 0.1	1	1 s	vanne injection aire stérile
xv 601-12	Liaison_2	Bool	xv 601-12	Q 1.3	1	1 s	vanne injection aire stérile
xv 630-00	Liaison_2	Bool	xv 630-00	Q 1.6	1	1 s	vanne 630-00 entrée aire arborant
xv 630-21	Liaison_2	Bool	xv 630-21	Q 1.7	1	1 s	vanne 630-21 entrée vapeur p630
xv 630-23	Liaison_2	Bool	xv 630-23	Q 1.5	1	1 s	vanne 630-23 sortie air stéril
xv 630-30/31	Liaison_2	Bool	xv 630-30/31	Q 2.3	1	1 s	ouverture alternative vanne 630-30/31 conde
xv 630-01	Liaison_2	Bool	xv 630-01	Q 2.4	1	1 s	vanne 630-01 arrivée vapeur

Fig. IV.37 Table de quelques variables

### IV.5.3 Création des vues

Sous WinCC flexible, on peut créer des vues pour le contrôle et la commande des machines et de l'installation. Pour cela, on dispose d'objets prédéfinis permettant de représenter l'installation et de définir les valeurs du processus.

Une vue peut être composée d'éléments statiques et d'éléments dynamiques.

- Les éléments statiques : le texte et le graphique, ne changent pas au runtime.
- Les éléments dynamiques varient en fonction de la procédure. Ils visualisent les valeurs de processus actuelles à partir de la mémoire de l'automate ou du pupitre (pompes, vannes, champ E/S...etc.).

#### IV.5.3.1 Vues du processus

Les processus partiels peuvent être représentés dans des vues séparées, puis regroupés en une vue principale (vue d'ensemble).

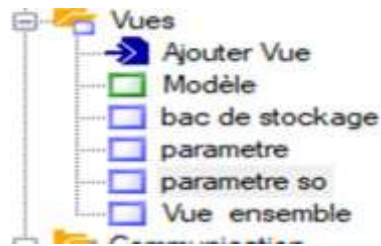


Fig. IV.38 Vues pupitre de l'installation

##### IV.5.3.1.1 Vue d'ensemble

La vue d'ensemble contient le logo de groupe cevital et celui de l'université de Bejaïa elle a aussi trois boutons pour permettre à l'opérateur de choisir l'accès vers les différentes vues de l'installation.



Fig. IV.39 Vue d'ensemble

### IV.5.3.1.2 Vue des Bacs

Cette vue permet à l'opérateur de visualiser :

- Visualiser l'ensemble des équipements de la station,
- Visualiser le déroulement du processus dans le bac en temps réel, ainsi que les mesures des différents capteurs configurés avec un champ d'E/S.
- Commander la mise en marche et l'arrêt de l'installation grâce aux boutons « ON » et « OFF »,
- Visualiser l'état des vannes, ouverte (couleur verte), fermée (couleur blanche).
- Ouverture et fermeture des différentes vannes concernées,
- Le niveau et la pression des bacs.

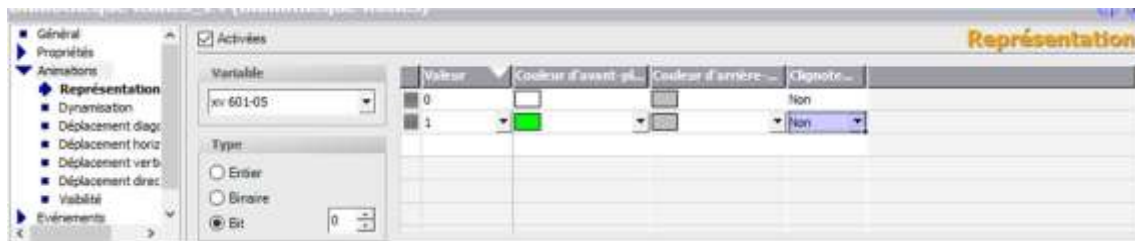


Fig. IV.40. Configuration de l'animation des vannes.

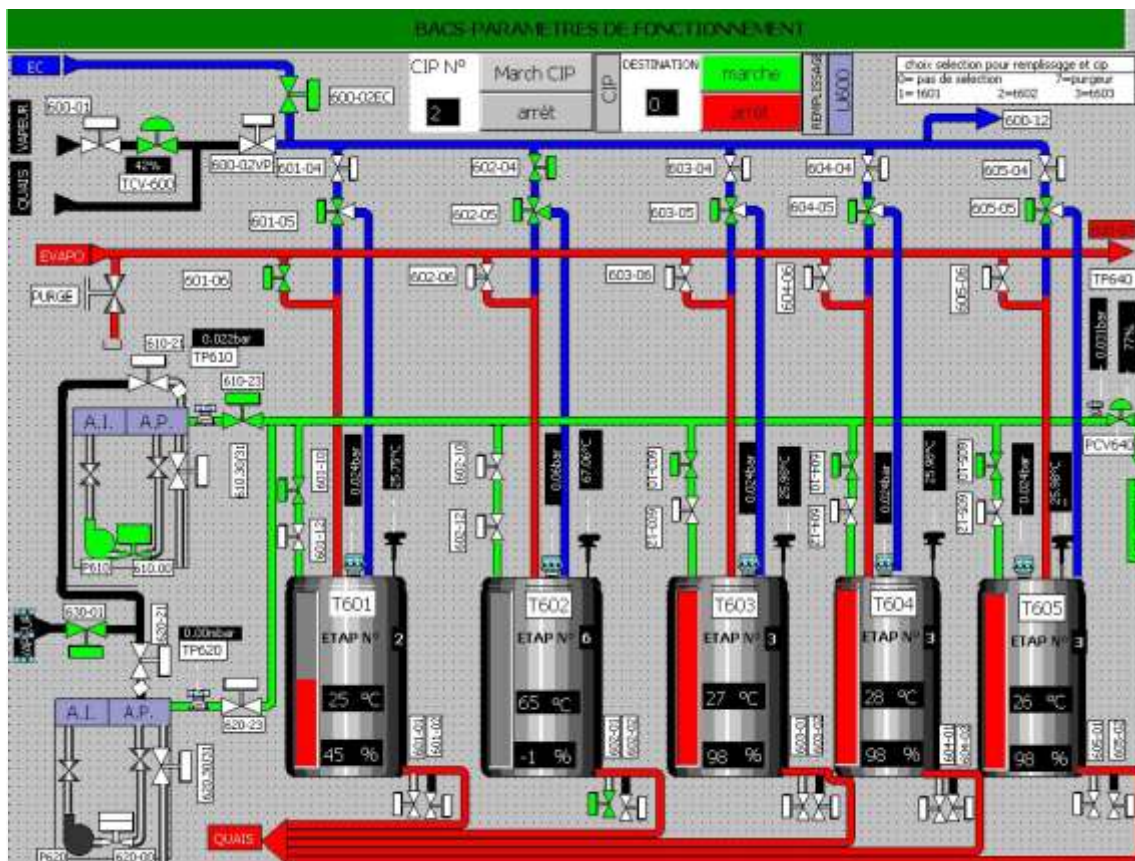


Fig. IV.41 Vue des bacs.

Les figures suivantes représentent une interface des paramètres utilisée dans la supervision

ETAPE STOCKAGE BACS		FINS D'ETAPE	
		TEMPS	
		mn	ss
0	ETAPE INITIALE		
1	T60x: PRET A RECEVOIR		
2	REMPLISSAGE		
3	ARRET REMPLISSAGE		
4	T60x: CIP 1	10	0
5	T60x: CIP 1	5	0
6	T60x: CIP 3	5	0
7	T60x: CIP 4	5	0
8	T60x: CIP 5	5	0
9	T60x: CIP 6	5	0
10	T60x: DESINFECTION 1	20	0
11	T60x: DESINFECTION 2	5	0
12	T60x: DESINFECTION 3	5	0
13	T60x: DESINFECTION 4	5	0
14	T60x: SOUFFLANTE	20	0
15	set "T60x PROPRE EST REFROIDI"	5	0

seuil pression haute	00000
seuil pression basse	00000
seuil de niveau haute	00000
seuil de niveau basse	00000
seuil de temperature haute	00000
seuil de temperature basse	00000

Fig. IV.42 Paramètres de bac.

ETAPE DESINFECTION SO 620	FINS D'ETAPE	
	TEMPS	
	mn	ss
ETAPE INITIALE		
arret soufflante		
ouverture distribution vapeur		
desinfection	100	0
fermeture distribution vapeur		
refroidissement purge	120	0

seuil pression haute TP 620	00000
seuil pression basse TP 620	00000

ETAPE DESINFECTION SO 610	FINS D'ETAPE	
	TEMPS	
	mn	ss
ETAPE INITIALE		
arret soufflante		
ouverture distribution vapeur		
desinfection	100	0
fermeture distribution vapeur		
refroidissement purge	120	0

seuil pression haute TP 610	00000
seuil pression basse TP 610	00000

Fig. IV.43 Paramètres de soufflante.

### IV.5.3.1.3 Compilation et simulation

Après avoir créé le projet et terminé la configuration, il est indispensable de vérifier la cohérence du projet, et de détecter les erreurs, à l'aide de la commande sur la barre du menu "contrôle de la cohérence", après le contrôle de cohérence, le système crée un fichier de projet compilé.

La simulation permet de détecter des erreurs logiques de configuration, par exemple, des valeurs limites incorrectes, et cela à l'aide du simulateur RUNTIME par la commande « démarrer le système RUNTIME du simulateur » [9].



**IV.6 Conclusion**

Dans ce chapitre, nous avons présenté le logiciel utilisé et les différentes étapes de la création de notre programme sous STEP 7. Nous avons donné aussi un aperçu sur les blocs utilisés lors de la programmation, ce dernier sera exécuté par l'automate S7-300 utilisé. Nous avons aussi présenté la procédure à suivre pour la création d'une Interface Homme Machine (IHM) pour le contrôle et la commande de notre système. On a testé le programme et la supervision par simulation, en utilisant le logiciel STEP 7.

La création de l'IHM exige une bonne connaissance du fonctionnement de notre système, et du langage avec lequel est programmé l'automate, afin de communiquer et de prélever l'adresse des variables qui nous intéressent.

*Conclusion*

*général*

## Conclusion générale

Dans ce présent mémoire nous avons présenté la démarche à suivre pour l'automatisation d'injection d'air stérile dans les bacs de stockage qui présente elle-même l'amélioration ajoutée au système de stockage afin d'éviter les déformations des bacs en cas des sous pression lors du soutirage et chargement camion.

Pour atteindre l'objectif de notre projet, nous avons d'abord commencé par prendre connaissance de l'installation, et nous avons identifié les éléments qui constitue notre système. Par la suite nous avons pensé à faire un nouveau système qui nous permet d'effectuer les opérations de lavage, désinfection, refroidissement et remplissage des bacs dans un seul grafcet.

Il était nécessaire de concevoir un système de commande basé sur un automate programmable industriel. Dans notre projet l'automate S7-300 de la firme SIEMENS est utilisé pour commander et gérer notre système ; c'est à dire gérer à la fois le prélavage des bacs, la désinfection, le refroidissement, le remplissage des bacs, la désinfections des soufflantes ainsi que l'injection d'air stérile dans les bacs de stockage d'une façon continue lors du remplissage et du soutirage afin d'éviter le phénomène de sous pression, qui provoque les déformations de ces bacs. C'est- à-dire d'éviter l'implosion des bacs voire les fissurations qui conduisent à la perte du produit stocké.

Pour la programmation de l'automate s7-300 nous avons utilisé le logiciel Step7 que nous avons présenté dans le chapitre "deux". Et pour le langage de programmation, nous avons programmé notre solution en langage ladder (ou schémas a contacts). Nous avons donné un aperçu global de chaque bloc de notre programme dans le chapitre "quatre". Après la programmation de notre solution, la récupération de l'état des variables est nécessaire pour la réalisation d'une interface homme-machine (IHM). Cette interface permet à l'opérateur un diagnostic rapide, un meilleur suivi, et un bon contrôle en temps réel du processus.

CEVITAL possède, maintenant, une solution répondant aux attentes du projet, à savoir d'une part la protection des bacs contre les sous pressions qui conduisent aux fissurations et les pertes du produit, et d'autre part bénéficier d'une flexibilité de travail plus importante.

Au-delà d'enrichir nos connaissances dans les différents domaines techniques que requière cette expérience professionnelle, ce stage nous a beaucoup apporté au niveau des connaissances personnelles. La gestion d'un tel projet favorisant le travail en équipe, tout en exigeant une grande autonomie, nous a permis d'apprécier le déroulement de ce projet.

***REFERENCES  
BIBLIOGRAPHIQUE***

***S***

## **REFERENCES BIBLIOGRAPHIQUES**

- [1] Document de CEVITAL, « Manuel opératoire sucre liquide »,2021.
- [2] Sami RAKIK, « Circuits pneumatique, conception et maintenance ».
- [3] JUMO Pressure instrument, « [www.JUMO.com](http://www.JUMO.com)».
- [4] AUDER Simon, « Automate programmable industriel », édition l'ELAN, liège 1991.
- [5] Information et formation, automatisation et entrainements, Programmation niveau 1, édition Siemens AG, 2003.
- [6] MANUEL SIEMENS,'PROGRAMMATION AVEC STEP7',2000.
- [7] Mr. Abdelkrim Laifaoui, 'AUTOMATISMES INDUSTRIELS', Cours 2016
- [8] J.C. BOSSY, P. BRAND, P. FAUGERE, C. MERLAUD « Le grafcet sa pratique et ses applications », édition CASTEILLA. France 5.3.
- [9] WinCC flexible 2008 Compact / Standard / Advanced Manuel utilisation, 07/2008, 6AV6691-1AB01-3AC0.

# *Annexe*

## *1*

**Propriétés de la table des mnémoniques**

Nom :	Mnémoniques
Auteur :	
Commentaire :	
Date de création :	20/06/2022 22:38:59
Dernière modification :	25/06/2022 00:29:15
Dernier filtre sélectionné :	Tous les mnémoniques
Nombre de mnémoniques :	151/151
Dernier tri :	Mnémonique ordre croissant

Etat	Mnémonique	Opérande	Type de données	Commentaire
	action ou sortie	FC 5	FC 5	
	actions soufflante 610	FC 9	FC 9	
	ACTIVATION	M 7.2	BOOL	declanchement regula
	arret remplissage	M 5.6	BOOL	arret remplissage
	ARRET SOUF610	M 6.6	BOOL	arret soufflante 610
	arret soufflante	M 7.0	BOOL	arret soufflante so610 si en fonctionnement
	B.P demande desinfection	M 6.7	BOOL	
	bac 601 propre	M 2.0	BOOL	"T601 PROPRE & REFROIDIE "
	bipolaire	M 0.1	BOOL	
	bit_cadence	M 200.5	BOOL	
	bloc regulation PCV640	DB 41	FB 41	
	bloc regulation TCV600	DB 42	FB 41	
	BP	M 1.2	BOOL	BP cip
	BP-ARRET-CIP	M 1.0	BOOL	bouton arret
	BP-MARCHE-CIP	M 1.1	BOOL	bouton marche
	BP arret so 610	A 1.4	BOOL	arret soufflante 610
	BP march remp	M 6.0	BOOL	BP_marche a remplir
	bp marche soufflante	M 7.1	BOOL	
	CONT_C	FB 41	FB 41	Continuous Control
	CONT_S	FB 42	FB 42	Step Control
	cycle	OB 1	OB 1	
	demande désinf_SO610	M 4.6	BOOL	demande désinfection SO610
	différent tache	FC 4	FC 4	
	DISJ_OUVER	E 3.6	BOOL	disjoncteur 620
	enclenchement_TCV	M 7.4	BOOL	encenchement de la regulation
	entrée analogique	FC 6	FC 6	
	etap_graf	MW 102	WORD	numero etape
	etap_grafc	MW 100	WORD	numero etape
	FD10	M 3.6	BOOL	CONDITION VERS ETAP 10
	FDC 11	M 3.7	BOOL	CONDITION VERS ETAPE 11
	FDC 12	M 4.1	BOOL	CONDITON VERS ETAP 12
	FDC 13	M 4.3	BOOL	CONDITION VERS ETAPE 13
	FDC 14	M 4.4	BOOL	CONDITION VERS ETAP 14
	FDC 15	M 4.5	BOOL	CONDITION VERS ETAP 15
	FDC 6	M 3.2	BOOL	CONDITION VERS ETAP 6
	FDC 600-01	E 1.6	BOOL	vanne entrée vapeur 600-01
	FDC 600-02	E 1.5	BOOL	vanne aiguillage vapeur/eau 600-02
	FDC 600-03	E 0.6	BOOL	vanne sortie flush/lavage 600-03
	FDC 600-12	E 0.5	BOOL	vanne isolement ligne EC/vapeur 600-12
	FDC 601-01	E 0.4	BOOL	vanne purge lavage 601-01
	FDC 601-02	E 1.7	BOOL	vanne purge désinfection 601-02
	FDC 601-05	E 0.3	BOOL	vanne boule de nettoyage 601-05
	FDC 601-10	E 2.5	BOOL	vanne entrée air stérile 601-10
	FDC 601-12	E 2.6	BOOL	vanne (TOR 100% 30% NO) entré air stérile 601-12
	FDC 601-99	E 0.7	BOOL	vanne sortie lavage 601-99
	FDC 601.04	E 0.1	BOOL	vanne 601-04 entrée eau chaude / vapeur
	FDC 601.06	E 0.0	BOOL	vanne de remplissage entrée sirop 601-06
	FDC 610-00	E 3.0	BOOL	vanne entrée air ambiant 610-00
	FDC 610-21	E 3.1	BOOL	vanne entrée vapeur vers S610/620
	FDC 610-23	E 2.7	BOOL	vanne sortie air stérile 610-23
	FDC 630-01	E 3.3	BOOL	vanne arrivée de vapeur 0.9 bar 630-01
	FDC 7	M 3.3	BOOL	CONDITION VERS ETAP 7
	FDC 8	M 3.4	BOOL	CONDITION VERS ETAP 8

Etat	Mnémorique	Opérande	Type de données	Commentaire
	FDC 9	M 3.5	BOOL	CONDITION VERS ETAP 9
	FDC04	M 3.1	BOOL	CONDITION VERS ETAP 5
	grafcet 601	FC 1	FC 1	desinfection bac 601
	grafcet 610	FC 2	FC 2	mise en marche soufflante 610
	grafcet 620	FC 3	FC 3	mise en marche soufflante 620
	IMPIL_S	M 8.0	BOOL	
	LT601	MD 30	DWORD	NIVEAU BAC T601
	LT601<=SL_bas	M 2.1	BOOL	BAC VIDE
	LT601>=SL_haut	M 4.2	BOOL	BAC PLEIN
	M5.5	M 5.5	BOOL	T601 PRES A RECEVOIR
	M8.1	M 8.1	BOOL	
	march cip 1	M 2.7	BOOL	nettoyage en marche
	MD38	MD 38	DWORD	
	NIV	MW 10	WORD	
	NIV-601	PEW 304	WORD	
	ok_bac	M 3.0	BOOL	selection numero bac active
	p610	A 2.6	BOOL	pompe p610
	PARAMETRE	DB 1	DB 1	
	PARAMETTRE PID PCV640	DB 2	DB 2	
	PARAMETTRE PID TCV600	DB 3	DB 3	
	PEW314	PEW 314	WORD	
	PEW318	PEW 318	WORD	
	PEW320	PEW 320	WORD	
	PRES-601	MW 24	WORD	PRESS
	PRES-610	PEW 310	WORD	
	PRES-620	PEW 312	WORD	
	PRES-640	PEW 316	WORD	
	PRESS	MW 14	WORD	
	PRESS-601	PEW 308	WORD	
	PRESS-610	MW 16	WORD	
	PRESS-620	MW 18	WORD	
	PRESSS-620	MW 20	WORD	
	PT_601	M 2.3	BOOL	pression dans le bac est OK
	PT_SO610	M 6.3	BOOL	PRESSION DESANFICTION
	PT_SO620	M 7.3	BOOL	PRESSION DESINFECTIION
	PT601	MD 54	DWORD	PRESSION PT601
	PT610	MD 42	DWORD	pression soufflante 610
	PT620	MD 46	DWORD	PRESSION SOUFLANTE 620
	PT640	MD 50	DWORD	PRESSION 640
	regulation PCV640	FC 11	FC 11	
	regulation TCV600	FC 12	FC 12	
	RM610	E 3.4	BOOL	retour de marche la soufflante 610
	RM620	E 3.5	BOOL	retour de marche la soufflante 620
	SCALE	FC 105	FC 105	Scaling Values
	selection bac rempl	M 5.7	BOOL	sectione bac a remplir
	SH TT 601	M 2.5	BOOL	temperature est pas bonne
	SORTIE	PAW 336	WORD	PCV640
	SORTIE_VANNE	PAW 338	WORD	TCV600
	soufflante en marche	M 4.7	BOOL	
	T 12	A 2.0	BOOL	lancement temporisation T12 désinfection (parametrable)
	T 13	A 2.2	BOOL	lancement T 13 refroidissement (parametrable)
	T36	T 36	TIMER	
	T47	T 47	TIMER	
	T601 arret FC	M 5.3	BOOL	arretet bac en fin de cycle
	T601 arret FP	M 5.2	BOOL	arretet bac en fin de production
	T601 arret opp	M 5.1	BOOL	arret bac par opérateur
	T601 cip	M 5.0	BOOL	cip sélectioné
	T601 MAR REM	M 2.4	BOOL	remplissage du bac en marcheT601
	TEM	MW 12	WORD	
	TEMP-601	PEW 306	WORD	
	tempo	FC 10	FC 10	
	TEMPO 1	T 1	TIMER	



Etat	Mnémorique	Opérande	Type de données	Commentaire
	TEMPO 10	T 10	TIMER	
	TEMPO 11	T 11	TIMER	
	TEMPO 12	T 12	TIMER	temporisation T12
	TEMPO 14	T 14	TIMER	fin temporisation refroidissement
	TEMPO 5	T 5	TIMER	
	TEMPO 6	T 6	TIMER	
	TEMPO 7	T 7	TIMER	
	TEMPO 8	T 8	TIMER	
	TEMPO 9	T 9	TIMER	
	TEMPO T2	T 2	TIMER	
	TEMPO T3	T 3	TIMER	
	TEMPO T4	T 4	TIMER	
	TEMPO_13	T 13	TIMER	temporisation
	teste simulation1	VAT 1		
	timer	OB 35	OB 35	
	TT<=35	M 2.2	BOOL	BAC REFROIDIE
	TT>=90	M 4.0	BOOL	TEMPERATURE STERELISATION ATTEINTE
	TT601	MD 34	DWORD	TEMP BAC T601
	vanne	FC 7	FC 7	
	xv 600-01	A 1.0	BOOL	vanne entré vapeur
	xv 600-02	A 1.1	BOOL	vanne a trois vois , entré vapeur
	xv 600-03	A 0.6	BOOL	vanne entré eau chaude
	xv 600-12	A 0.5	BOOL	vanne entré eau chaud / vapeur
	xv 601-01	A 0.4	BOOL	vanne purge lavage
	xv 601-02	A 1.2	BOOL	vanne purge stérilisation
	xv 601-04	A 0.2	BOOL	vanne entré eau chaude / vapeur
	xv 601-05	A 0.3	BOOL	vanne boule de nettoyage
	xv 601-06	A 0.0	BOOL	vanne de remplissage SL
	xv 601-10	A 0.1	BOOL	vanne injection aire stérile
	xv 601-12	A 1.3	BOOL	vanne injection aire stérile
	xv 601-99	A 0.7	BOOL	vanne entré chaude / vapeur
	xv 610-00	A 1.6	BOOL	vanne 610-00 entrée aire ambient
	xv 610-21	A 1.7	BOOL	vanne 610-21 entrée vapeur s610
	xv 610-23	A 1.5	BOOL	vanne 610-23 sortie air stéril
	xv 610-30/31	A 2.3	BOOL	ouverture alternative vanne 610-30/31 condensat (5 sec toute les 10 min)
	xv 630-01	A 2.4	BOOL	vanne 630-01 arrivée vapeur

# *Annexe*

## *2*

## Régulation continue avec SFB41/FB41 "CONT\_C"

### Introduction

Le bloc SFB41/FB41 "CONT\_C" (continuous controller) sert à régler des processus industriels à grandeurs d'entrée et de sortie continues sur les automates programmables SIMATIC S7. Le paramétrage vous permet d'activer ou de désactiver des fonctions partielles du régulateur PID et donc d'adapter ce dernier au système réglé. Vous pouvez aisément réaliser ceci à l'aide de l'outil de paramétrage (appel, p. ex. : Démarrer > Simatic > STEP 7 > Paramétrage de la régulation PID). Le manuel électronique se trouve, p. ex., sous **Démarrer > Simatic > Manuels S7 > Régulateurs PID**.

### Utilisation

Vous pouvez utiliser le régulateur comme régulateur PID de maintien individuel, mais aussi comme régulateur en cascade, proportionnel ou de rapport dans des régulations à plusieurs boucles. Sa méthode de travail se base sur l'algorithme PID du régulateur d'échantillonnage à sortie analogique, complété le cas échéant par un niveau de formateur d'impulsions assurant la formation de sorties à impulsions modulées en durée pour régulations à deux ou trois échelons avec actionneurs proportionnels.

---

### Nota

Le calcul des valeurs dans les blocs de régulation n'est effectué correctement que si le bloc est appelé à intervalles réguliers. C'est pourquoi il convient d'appeler les blocs de régulation dans un OB d'alarme cyclique (OB30 à OB38). Précisez la période dans le paramètre CYCLE.

---

### Description

En plus des fonctions traitant la consigne et la mesure, le SFB/FB réalise un régulateur PID prêt à l'emploi avec sortie continue de grandeur réglante et possibilité d'influencer la valeur de réglage à la main.

Il propose les fonctions partielles suivantes.

#### Branche de consigne

La consigne est entrée en format de virgule flottante à l'entrée **SP\_INT**.

#### Branche de mesure

La mesure peut être lue en format de périphérie ou de virgule flottante. La fonction CRP\_IN convertit la valeur de périphérie PV\_PER en un nombre à virgule flottante compris entre -100 et +100 % selon la règle suivante :

$$\text{Sortie de CPR\_IN} = \text{PV\_PER} \cdot \frac{100}{27648}$$

La fonction PV\_NORM normalise la sortie de CRP\_IN selon la règle suivante :

$$\text{Sortie de PV\_NORM} = (\text{Sortie de CPR\_IN}) \cdot \text{PV\_FAC} + \text{PV\_OFF}$$

La valeur par défaut de PV\_FAC est 1 et celle de PV\_OFF est 0.

#### Formation du signal d'erreur

La différence entre consigne et mesure donne le signal d'erreur. Il est conduit par une zone morte (DEADBAND) pour atténuer une petite oscillation entretenue causée par la quantification de grandeur réglante (par exemple en cas de modulation de durée d'impulsion avec PULSEGEN). Quand DEADB\_W égale 0, la zone morte est désactivée.

#### Algorithme PID

L'algorithme PID travaille dans l'algorithme de position. Les parties proportionnelle, intégrale (INT) et dérivée (DIF) sont en parallèle et peuvent être activées ou désactivées séparément. Ceci permet de paramétrer des régulateurs P, PI, PD et PID, mais aussi des régulateurs I et D purs.

### Traitement de valeur manuelle

Vous pouvez passer du mode automatique au mode manuel et inversement. En mode manuel, la grandeur réglante est adaptée à une valeur manuelle. L'intégrateur (INT) est forcé de façon interne à LMN - LMN\_P - DISV et le dérivateur (DIF) est forcé à 0 et égalisé de façon interne. Ainsi, le passage au mode automatique est exempt de chocs.

### Traitement de valeur de réglage

La fonction LMNLIMIT permet de limiter la valeur de réglage à des valeurs que vous indiquez. Si la grandeur d'entrée dépasse ces limites, des bits le signalent. La fonction LMN\_NORM normalise la sortie de LMNLIMIT selon la règle suivante :

$$LMN = (\text{Sortie de LMNLIMIT}) * LMN\_FAC + LMN\_OFF$$

La valeur par défaut de LMN\_FAC est 1 et celle de LMN\_OFF est 0.

La valeur de réglage est disponible aussi en format de périphérie. La fonction CRP\_OUT convertit la valeur à virgule flottante LMN en une valeur de périphérie selon la règle :

$$LMN\_PER = LMN * \frac{27648}{100}$$

### Compensation de perturbation

Il est possible d'appliquer une grandeur perturbatrice à l'entrée **DISV**.

### Initialisation

Le bloc SFB41/FB41 "CONT\_C" dispose d'un sous-programme d'initialisation qui est exécuté quand le paramètre d'entrée COM\_RST est à 1.

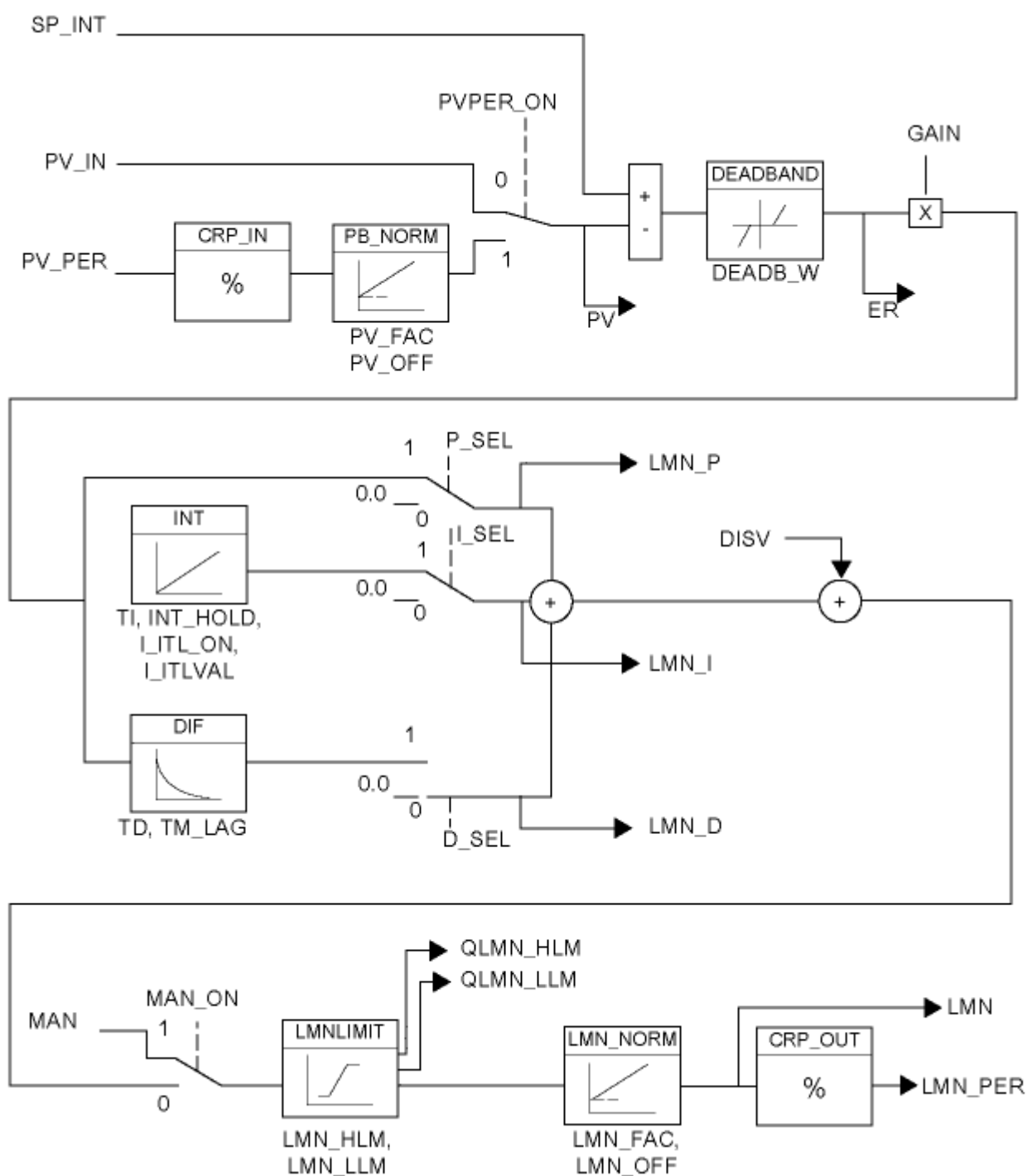
Lors de l'initialisation, l'intégrateur est forcé de façon interne à la valeur d'initialisation I\_ITVAL. En cas d'appel dans un niveau d'alarme d'horloge, il continue à travailler à partir de cette valeur.

Toutes les autres sorties sont forcées à leur valeur par défaut.

### Informations d'erreur

Le mot d'indication d'erreur RET\_VAL n'est pas employé.

### Schéma fonctionnel



### Paramètres d'entrée

Le tableau suivant présente les paramètres d'entrée du bloc SFB41/FB41 "CONT\_C".

Paramètre	Type de données	Valeurs admises	Par défaut	Description
COM_RST	<a href="#">BOOL</a>		FALSE	COMPLETE RESTART Le bloc a un sous-programme d'initialisation qui est exécuté quand cette entrée est à 1.
MAN_ON	BOOL		TRUE	MANUAL VALUE ON / Activation du mode manuel Quand cette entrée est à 1, la boucle de régulation est interrompue. C'est une valeur manuelle qui est introduite comme valeur de réglage.
PVPER_ON	BOOL		FALSE	PROCESS VARIABLE PERIPHERY ON / Activation de la

				mesure de périphérie Pour que la mesure soit lue dans la périphérie, il faut relier l'entrée PV_PER à la périphérie et mettre à 1 l'entrée PVPER_ON.
P_SEL	BOOL		TRUE	PROPORTIONAL ACTION ON / Activation de l'action proportionnelle Dans l'algorithme PID, il est possible d'activer et de désactiver séparément chacune des actions. L'action P est active quand cette entrée est à 1.
I_SEL	BOOL		TRUE	INTEGRAL ACTION ON / Activation de l'action par intégration Dans l'algorithme PID, il est possible d'activer et de désactiver séparément chacune des actions. L'action I est active quand cette entrée est à 1.
INT_HOLD	BOOL		FALSE	INTEGRAL ACTION HOLD / Gel de l'action par intégration La sortie de l'intégrateur peut être gelée. Pour cela, il faut mettre à 1 cette entrée.
I_ITL_ON	BOOL		FALSE	INITIALIZATION OF THE INTEGRAL ACTION / Initialisation de l'action par intégration La sortie de l'intégrateur peut être forcée à la valeur initiale I_ITL_VAL. Pour cela, il faut mettre à 1 cette entrée.
D_SEL	BOOL		FALSE	DERIVATIVE ACTION ON / Activation de l'action par dérivation Dans l'algorithme PID, il est possible d'activer et de désactiver séparément chacune des actions. L'action D est active quand cette entrée est à 1.
CYCLE	<u>TIME</u>	>= 1 ms	T#1s	SAMPLE TIME / Période d'échantillonnage Le temps s'écoulant entre les appels de bloc doit être constant. Il est indiqué par cette entrée.
SP_INT	REAL	-100,0 à 100,0 (%) ou grandeur physique <sup>1</sup>	0,0	INTERNAL SETPOINT / Consigne interne Cette entrée sert à introduire une consigne.
PV_IN	REAL	-100,0 à 100,0 (%) ou grandeur physique <sup>1</sup>	0,0	PROCESS VARIABLE IN / Mesure interne Cette entrée permet de paramétrer une valeur de mise en service ou de relier une mesure externe en virgule flottante.
PV_PER	<u>WORD</u>		W#16#0000	PROCESS VARIABLE PERIPHERIE / Mesure de périphérie La mesure en format de périphérie est reliée au régulateur à cette entrée.
MAN	REAL	-100,0 à 100,0 (%) ou grandeur physique <sup>2</sup>	0,0	MANUAL VALUE / Valeur manuelle Cette entrée sert à introduire une valeur manuelle grâce à des fonctions de contrôle-commande.
GAIN	REAL		2,0	PROPORTIONAL GAIN / Coefficient d'action proportionnelle Cette entrée indique le gain du régulateur.
TI	TIME	>= CYCLE	T#20 s	RESET TIME / Temps d'intégration Cette entrée détermine le comportement dans le temps de l'intégrateur.
TD	TIME	>= CYCLE	T#10 s	DERIVATIVE TIME / Temps de dérivation Cette entrée détermine le comportement dans le temps du dérivateur.
TM_LAG	TIME	>= CYCLE/2	T#2 s	TIME LAG OF THE DERIVATE ACTION / Retard de l'action par dérivation L'algorithme de l'action D contient un retard qui peut être paramétré à cette entrée.
DEADB_W	REAL	>= 0,0 (%) ou grandeur	0,0	DEAD BAND WIDTH / Largeur de zone morte

		physique <sup>1</sup>		Le signal d'erreur est conduit par une zone morte. Cette entrée détermine la taille de la zone morte.
LMN_HLM	REAL	LMN_LLM à 100,0 (%) ou grandeur physique <sup>2</sup>	100,0	MANIPULATED VALUE HIGH LIMIT / Limite supérieure de la valeur de réglage La valeur de réglage est toujours limitée vers le haut et vers le bas. Cette entrée indique sa limite supérieure.
LMN_LLM	REAL	-100,0 à LMN_HLM (%) ou grandeur physique <sup>2</sup>	0,0	MANIPULATED VALUE LOW LIMIT / Limite inférieure de la valeur de réglage La valeur de réglage est toujours limitée vers le haut et vers le bas. Cette entrée indique sa limite inférieure.
PV_FAC	REAL		1,0	PROCESS VARIABLE FACTOR / Facteur de normalisation Cette entrée est multipliée par la mesure. Elle sert à adapter la plage de la mesure.
PV_OFF	REAL		0,0	PROCESS VARIABLE OFFSET / Décalage de normalisation Cette entrée est ajoutée à la mesure. Elle sert à adapter la plage de la mesure.
LMN_FAC	REAL		1,0	MANIPULATED VALUE FACTOR / Facteur de valeur de réglage Cette entrée est multipliée par la valeur de réglage. Elle sert à adapter la plage de la valeur de réglage.
LMN_OFF	REAL		0,0	MANIPULATED VALUE OFFSET / Décalage de valeur de réglage Cette entrée est ajoutée à la valeur de réglage. Elle sert à adapter la plage de la valeur de réglage.
I_ITLVAL	REAL	-100,0 à 100,0 (%) ou grandeur physique <sup>2</sup>	0,0	INITIALIZATION VALUE OF THE INTEGRAL ACTION / Valeur d'initialisation pour l'action par intégration La sortie de l'intégrateur peut être forcée par l'entrée I_ITL_ON. La valeur d'initialisation est à cette entrée.
DISV	REAL	-100,0 à 100,0 (%) ou grandeur physique <sup>2</sup>	0,0	DISTURBANCE VARIABLE / Grandeur perturbatrice La grandeur perturbatrice de compensation est reliée à cette entrée.

1. Paramètres dans les branches de consigne et de mesure avec même unité.

2. Paramètres dans la branche de valeur de réglage avec même unité.

### Paramètres de sortie

Le tableau suivant présente les paramètres de sortie du bloc SFB41/FB41 "CONT\_C".

Paramètre	Type de données	Valeurs admises	Par défaut	Description
LMN	REAL		0,0	MANIPULATED VALUE / Valeur de réglage Cette sortie donne en virgule flottante la valeur de réglage agissant réellement.
LMN_PER	<a href="#">WORD</a>		W#16#0000	MANIPULATED VALUE PERIPHERY / Valeur de réglage de périphérie La valeur de réglage en format de périphérie est reliée avec le régulateur à cette sortie.
QLMN_HLM	<a href="#">BOOL</a>		FALSE	HIGH LIMIT OF MANIPULATED VALUE REACHED / Limite supérieure de la valeur de réglage atteinte La valeur de réglage est toujours limitée vers le haut et vers le bas. Cette sortie signale le dépassement de sa limite supérieure.
QLMN_LLM	BOOL		FALSE	LOW LIMIT OF MANIPULATED VALUE REACHED / Limite inférieure de la valeur de réglage atteinte La valeur de réglage est toujours limitée vers le haut et vers le bas. Cette sortie signale le dépassement de sa limite inférieure.

LMN_P	REAL	0,0	PROPORTIONALITY COMPONENT / Composant P Cette sortie contient le composant proportionnel de la grandeur réglante.
LMN_I	REAL	0,0	INTEGRAL COMPONENT / Composant I Cette sortie contient le composant intégral de la grandeur réglante.
LMN_D	REAL	0,0	DERIVATIVE COMPONENT / Composant D Cette sortie contient le composant dérivé de la grandeur réglante.
PV	REAL	0,0	PROCESS VARIABLE / Mesure Cette sortie donne la mesure agissant réellement.
ER	REAL	0,0	ERROR SIGNAL / Signal d'erreur Cette sortie donne le signal d'erreur agissant réellement.



## Résumé

Travail à CEVITAL consiste à apporter une amélioration à la section de stockage et chargement de l'unité sucre liquide. Cette amélioration consiste à automatiser le circuit d'injection d'air stérile dans les bacs de stockage. Le but de cette démarche est la protection des bacs contre les dépressions qui provoquent les implosions des bacs. Dans ce travail, il nous a été demandé d'automatiser, à l'aide d'un automate programmable industriel, le système d'injection d'air stérile de la section SL600. D'autre part, la connaissance des différents étages utilisés, permettra au service maintenance d'élaborer des documents techniques qui serviront pour une meilleure connaissance du système par les nouvelles recrues de l'entreprise. Pour la conception de ce système, nous avons introduit des nouvelles contraintes au cahier des charges qui montre et explique les différentes étapes de fonctionnement. Enfin, on élabore le GRAFCET qui permettra d'introduire ces nouvelles contraintes facilement programmé avec le langage schémas à contacts (Ladder) dans le logiciel de base STEP7 et ensuite on le valide par une simulation avec le simulateur S7 -PLCSIM qui donne de bons résultats et on termine par la visualisation de système par le logiciel WinCC.