

الجمهورية الجزائرية الديمقراطية الشعبية  
République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur  
et de la Recherche Scientifique  
Université Akli Mohand Oulhadj - Bouira -  
Tasdawit Akli Muḥend Ulḥağ - Tubirett -



وزارة التعليم العالي والبحث العلمي  
جامعة أكلي محمد أولحاج  
- البويرة -

Faculté des Sciences et des Sciences Appliquées

كلية العلوم والعلوم التطبيقية

Référence : ...../MM/2021

المرجع : ...../م/م / 2021

Mémoire de Master

Présenté au

Département : Génie Électrique

Domaine : Sciences et Technologies

Filière : Electronique

Spécialité : Electronique des systèmes embarqués

Réalisé par :

**BOUDRA Fatiha**

**Thème**

**Automatisme du poste de remplissage et bouchonnage**

Soutenu le: 20/10/2021

Devant la commission composée de :

Mr :	SAIDI Mohammed	M.A.A	Univ. Bouira	Président
	ASRADJ Zahir	M.A.A	Univ. Bouira	Rapporteur
	AGGOUN Gania	M.A.A	Univ. Bouira	Examineur



**Dédicaces**

**Je dédie ce modeste travail**

À ma chère mère

À mon cher père

À mes frères

À toute ma famille

À tous mes amis

À mes encadreurs



**FATIHA**

## Remerciements

Ce travail a été effectué au sein du Département des Sciences et sciences appliquées de l'Université de Bouira.

Avant tout, nous tenons à remercier le Dieu tout puissant qui nous a donné la force et la volonté pour étudier ce travail.

Nous tenons à remercier, en premier lieu, ASRADJ Zahir, Directeur de ce mémoire. notre encadreur pour sa disponibilité, son aide, ses conseils et ses orientations précieuses, sans oublier sa participation à la réalisation de ce mémoire.

nous tenons également tous les membres du jury pour l'intérêt qu'ils ont porté à mon travail :

SAIDI Mohammed

AGGOUN Gania

Nous remercions spécialement, Monsieur KHALED KHODJA Rachid, responsable du chef service de la société SARL BISTA PRODA, qui est aussi notre Co-encadreur pour nous avoir intégrés rapidement au sein de l'entreprise et nous avoir accordé toute sa confiance, pour le temps qu'il nous a consacré tout au long de cette période, sachant répondre à toute notre interrogation, sans oublier sa participation à la réalisation de ce travail.

Nous présentons enfin, notre profonde gratitude à notre famille, ainsi qu'à tous ceux qui nous ont aidés de près ou de loin pour l'élaboration de ce travail.

Enfin, j'associe à ces remerciements tous ceux qui ont contribué à réaliser ce travail.

## **Résumé**

Le travail présenté dans ce mémoire consiste à automatiser le fonctionnement d'une chaîne de production du post de remplissage et de bouchonnage qui contient trois postes. Cette commande à base d'un automate (API), en particulier le Simens modulaire S7-300.

Les données du processus sont obtenues et traitées par le SIMATIC S7-300 intégré au logiciel Steps7. Nous créons encore une interface homme-machine (HMI) et une vue d'alarme avec un logiciel de supervision WinCC flexible 2008, pour faciliter la tâche de contrôle, de surveillance et la productivité de la chaîne de production par le travailleur.

### **Mots clés**

Système automatique, Logiciel de supervision, API, automate Simens modulaire S7-300, logiciel STEPS7, WinCC flexible.

# Table des Matières

Remerciements.....	I
Résumé.....	II
Table des Matières.....	III
Liste des Figure.....	VIII
Liste des Tableaux.....	XI
Listes des Acronymes.....	XII
Introduction Générale.....	1

## Chapitre 1 : Présentation de l'entreprise et généralités sur automatismes.

1.1 Introduction.....	2
1.2 Partie 1 Présentation de l'entreprise SARL BISTA PRODA.....	2
1.2.1 Description de l'entreprise SARL BISTA PRODA.....	2
1.2.2 Situation géographique.....	3
1.2.3 Produit d'usine BISTA PRODA.....	3
1.2.4 Composition des produits BISTA PRODA.....	4
1.3 Partie 2 Généralités sur l'automatisme.....	4
1.3.1 Système automatique.....	4
1.3.2. Objectif de l'automatisation.....	5
1.3.3 Les avantages et les inconvénients de l'automatisation.....	5
1.3.3.1. Les avantages.....	5
1.3.3.2 Les inconvénients.....	5
1.3.4 Structure générale d'un système automatique.....	6
1.3.4.1 Partie contrôle ou supervision.....	6
1.3.4.2 Partie commande ou API.....	6
1.3.4.3 Partie opérative.....	7
1.4 Les automates programmables industriels (API).....	8
1.4.1 Définition d'un API.....	8
1.4.2 Quelques fabricants d'un API dans le monde.....	9
1.4.3 Type d'un API.....	9

1.4.3.1	Automate compact.....	9
1.4.3.2	Automate modulaire.....	10
1.4.4	Matériel d'un API .....	10
1.4.4.1	Processeur .....	11
1.4.4.2	L'unité d'alimentation.....	11
1.4.4.3	Le périphérique de programmation.....	12
1.4.4.4	La mémoire.....	12
1.4.4.5	Les interfaces d'entrées-sorties.....	12
1.4.4.6	L'interface de communication.....	13
1.4.5	Fonctionnements d'un API .....	13
1.4.6	Programmes d'un API.....	14
1.4.6.1	Langage textuels.....	14
1.4.6.2	Langages graphiques.....	15
1.4.7	Critère de choix d'un API .....	17
1.5	L'automate programmable industriel SIEMENS.....	18
1.5.1	Présentation quelque gamme SIEMENS .....	18
1.5.2	L'interface de communication SIMENS.....	18
1.5.3	Adresses des entrées-sorties d'un API SIMATIC.....	18
1.6	Présentation de Win CC Runtime Advanced.....	19
1.6.1	Définition .....	19
1.6.2	Les tache d'un SIMATIC HMI .....	19
1.6.3	Utilisation de SIMATIC WinCC flexible .....	20
1.6.4	Eléments de WinCC flexible.....	20
1.7	Conclusion.....	20

## Chapitre 2 : Fonctionnement d'un système à l'aide d'un automate S7-300 Siemens

2.1	Introduction.....	21
2.2	But et importance de l'étude.....	21
2.3	Partie 1 Composition d'un système.....	22

2.3.1	Partie électrique.....	22
2.3.1.1	Capteurs.....	22
2.3.1.2	Caractéristiques métrologiques d'un capteur.....	22
2.3.1.3	Familles des capteurs.....	23
2.3.1.4	Domaines d'utilisation des capteurs.....	23
2.3.1.5	Critères de choix d'un capteur.....	24
2.3.1.6	Capteurs de proximité inductive.....	25
2.3.1.6	Capteur à chute de pression.....	25
2.3.1.7	Capteurs de niveau.....	26
2.3.1.9	Capteur de débit.....	26
2.3.2	Partie pneumatique.....	27
2.3.2.1	Vérins pneumatiques.....	27
2.3.2.2	Distributeur pneumatique.....	30
2.3.2.3	Moteur pneumatique.....	31
2.3.2.4	Moteurs pas à pas.....	31
2.3.2.5	Une pompe d'eau.....	32
2.3.2.6	Relie électromagnétique.....	33
2.3.2.7	Le convoyeur.....	33
2.3.2.8	Electrovanne.....	33
2.4	Partie 2 Généralité d'automate S7-300 d'un SIMENS.....	34
2.4.1	Présentation de l'automate utilisée S7-300.....	34
2.4.1.1	Module d'alimentation (PS).....	35
2.4.1.2	Module de CPU.....	35
2.4.1.3	Le coupleurs (IM).....	36
2.4.1.4	Modules de signaux (SM).....	36
2.4.1.5	Modules de fonctions (FM).....	37
2.4.1.6	Modules de communication (PC).....	38

2.4.2 Caractéristique d'un automate S7-300 .....	38
2.4.2 Interface (MPI).....	38
2.5 Programmation via STEP 7.....	39
2.5.1 Démarrage du logiciel STEP7.....	39
2.5.2 Création d'un projet STEP7 .....	40
2.5.3 Configuration matérielle .....	41
2.5.4 Création du programme.....	42
2.5.5 Structure du programme STEP7 .....	43
2.5.6 Table des mnémoniques .....	46
2.5.7 Présentation du PLCSIM.....	47
2.6 Conclusion.....	49

### **Chapitre 3 : Simulation et la supervision de notre projet**

3.1 Introduction.....	50
3.2 Cahier de charge.....	50
3.2.1 Principe de fonctionnement.....	50
3.2.2 Choix de technologie.....	50
3.3 Organigramme de programme du notre étude.....	51
3.3.1 Organigramme de programme du post de chargement (post 1).....	51
3.3.2 Organigramme de programme pour le post 2 .....	52
3.3.3 Organigramme de programme de post 3 .....	52
3.3.4 Organigramme de programme du tapie relou .....	53
3.4 Présentation de GRAFCET d'un système.....	55
3.4.1 GRAFCET niveau 01 .....	55
3.4.1.1 Liste d'actionneurs, capteurs d'un système.....	56
3.4.2 GRAFCET niveau 02.....	59
3.5 Simulation du programme avec S7-PLCSIM.....	60
3.5.1 Chargement du programme.....	60

3.5.2 Exécution et visualisation du programme .....	60
3.5.1 Simulation et visualisation du programme de post de chargement (Post 1) .....	62
3.5.2 Simulation et visualisation de programme de post de remplissage (Post 2) .....	63
3.5.3 Simulation et visualisation de poste du bouchonnage des bouteille (Poste 3) .....	64
3.6 La supervision de projet avec le WINCC flexible .....	65
3.6.1 définitions de la supervision .....	65
3.6.2 Création du projet avec WinCC flexible .....	65
3.6.3 Choix le panel du pupitre opérateur .....	66
3.6.4 Démarrage la station SIMATC HMI .....	66
3.6.5 Présentation du Vue dans le WinCC flexible .....	68
3.7 Création de vues du projet de post remplissage et bouchonnage .....	69
3.7.1 Création la vue d’alarme .....	70
3.8 Simulation avec le WinCC flexible Advanced .....	71
3.8.1 Visualisations d’alarme de la vanne .....	72
3.6 Conclusion .....	73
<b>Conclusion Générale</b>	<b>74</b>
<b>Références</b>	<b>75</b>
<b>Annexe</b>	<b>76</b>

## Liste des Figures

Figure 1.1 : Situation géographique de l'entreprise SARL BISTA PRODA.....	3
Figure 1.2 : Produits de l'entreprise SARL BISTA PRODA. ....	4
Figure 1.3 : Présentation d'un système automatisé.....	5
Figure 1.4: Structure générale d'un système automatisé. ....	6
Figure 1.5: Principe d'un capteur.....	7
Figure 1.6: Principe de transmettre d'information.....	8
Figure 1.7: Un automate programmable industriel. [3] .....	8
Figure 1.8: Type d'automate programmable industriel compacte. ....	9
Figure 1.9: Type d'automate programmable industriel modulaire. ....	10
Figure 1.10: Structure d'un automate programmable industriel. ....	11
Figure 1.11: Cycle de fonctionnement d'un API. ....	13
Figure 1.12 : Commande des machines avec l'API. ....	14
Figure 1.13: Commande des machines avec l'API. [3] .....	15
Figure 1.14: Présentation d'un élément de GRAFCET. ....	17
Figure 1.15: Présentation des gammes de SIMATIC. ....	18
Figure 1.16: Format d'une adresse dans l'automate SIMATIC S7 . [3].....	18
Figure 2.1: Exemple d'un chaîne de production .....	21
Figure 2.2: Schéma fonctionnel d'un capteur. [5] .....	22
Figure 2.3: Capteur tout ou rien TOR. ....	23
Figure 2.4: Exemple d'un capteur analogique. ....	24
Figure 2.5: Critères de choix d'un capteur. [5].....	25
Figure 2.6: Capteur à chute de pression.....	25
Figure 2.7: Capteur de niveau.....	26
Figure 2.8: Le principe d'un vérin pneumatique.....	27
Figure 2.9: Schéma d'un vérin pneumatique. ....	27
Figure 2.10: Vérin simple effet. ....	28
Figure 2.11: Vérin double effet.....	30
Figure 2.12: Le distributeur. ....	30
Figure 2.13: Distributeur 5/2.....	31
Figure 2.14: Pilotage d'un vérin double effet par un distributeur 5/2.....	31
Figure 2.15: Moteur pas à pas.....	32
Figure 2.16: Principe de base d'un moteur pas à pas.....	32

Figure 2.17: La pompe d'eau.....	32
Figure 2.18: Relie électromagnétique.....	33
Figure 2.19: Convoyeur.....	33
Figure 2.20: Débitmètre.....	26
Figure 2.21: Schéma d'une électrovanne.....	33
Figure 2.22: Constitution d'un automate programmable industriel S7-300.[2].....	34
Figure 2.23: Schéma de raccordement PS 307-5. [9].....	35
Figure 2.24: Le CPU 314-2 PN/DP de S7-300.....	36
Figure 2.25: Modules Entrées/sorties d'un automate S7-300. [9].....	37
Figure 2.26: L'interface entre l'automate (API) et l'ordinateur (PC).....	38
Figure 2.27: Présentation de l'interface (MPI). ....	39
Figure 2.28: Assistant nouveau projet.....	40
Figure 2.29: Choix de la CPU 412-2PN/DP et du bloc d'organisation OB1.....	40
Figure 2.30: Nom et création du projet.....	41
Figure 2.31: Station SIMATIC S7-300.....	41
Figure 2.32: Configuration matériels.....	42
Figure 2.33: Présentation la fenêtre de programmation.....	42
Figure 2.34: Exemple une partie de programme.....	43
Figure 2.35: Présentation les Blocs de programmation Steps7. [2].....	43
Figure 2.36: L'instruction mise à l'échelle FC105.....	44
Figure 2.37: Programme de niveau d'eau dans la vanne.....	45
Figure 2.38: Présentation de l'instruction MOVE.....	45
Figure 2.39: Programme déplacement des bouteilles.....	<b>Erreur ! Signet non défini.</b>
Figure 2.40: La partie de table des mnémoniques du projet.....	46
Figure 2.41: Paramétrage de l'interface PG/PC.....	47
Figure 2.42: Présentation de simulateur S7-PLCSIM.....	48
Figure 2.43: Présentation les éléments de S7 PLCSIM.....	48
Figure 2.44: Présentation le type de la liaison S7 PLCISIM.....	49
Figure 3.1: Présentation de post remplissage et bouchage.....	51
Figure 3.2: Organigramme de programme pour le post 1.....	51
Figure 3.3: Organigramme de programme post de remplissage.....	52
Figure 3.4: Organigramme de programme pour le post 3.....	53
Figure 3.5: Organigramme de programme pour déplacement des bouteilles.....	54

Figure 3.6 : Organigramme de programme pour le tapis par un moteur pas à pas. ....	54
Figure 3.7: Présentation le GRAFCET niveau 01.....	56
Figure 3.8: Présentation de GRAFCTE de notre projet.....	59
Figure 3.9: Chargement de programme dans l'API de simulation. ....	60
Figure 3.10: Simulation et visualisation de programme sur S7 PLCISIM. ....	61
Figure 3.11: Présentation de Fenêtre de la table de variable. ....	61
Figure 3.12: Simulation de programme par le TABLE DE VARIABLE.....	62
Figure 3.13: Simulation et visualisation de post de chargement des bouteilles.....	62
Figure 3.14: Simulation et visualisation le niveau de la vanne.....	63
Figure 3.15: Simulation et visualisation de programme post 2.....	63
Figure 3.16: Simulation et visualisation de programme du débitmètre. ....	64
Figure 3.17: Présentation la visualisation de post du bouchage des bouteille. ....	64
Figure 3.18:Schéma explicatif l'introduction de la supervision. ....	65
Figure 3.19: WinCC flexible 2008.....	65
Figure 3.20: Création de station SIMATIC HMI de WinCC flexible. ....	66
Figure 3.21: Présentation la fenêtre de choix du pupitre. ....	66
Figure 3.22: Liaison entre station SIMATC HMI et l'automate S7-300.....	67
Figure 3.23: Présentation le connecteur entre le panel et l'automate. ....	67
Figure 3.24: Présentation le tableau de variable dans le WinCC flexible.....	68
Figure 3.25: Présentation du Vue dans le WinCC flexible.....	68
Figure 3.26: Les éléments d'une vue continue dans le WinCC flexible.....	69
Figure 3.27:Présentation La vue modèle du chine de production piloter. ....	70
Figure 3.28: Vue d'alarme pour contrôler la vanne de remplissage. ....	70
Figure 3.29: Présentation la vue PROJET d'un chainepiloter sur WinCC fllexible.....	71
Figure 3.30 : Vue principal.....	71
Figure 3.31 : Simulation de la chine de production piloter.....	72
Figure 3.32 : Présentions de la simulation de vue d'alarme .....	73

## Liste des Tableaux

Tableau 1.1: Les principaux éléments d'un réseau LD.....	16
Tableau 2.1: Différents domaines d'application des capteurs.....	24
Tableau 2.2: Mode fonctionnement d'un vérin pneumatique.....	28
Tableau 2.3: Blocs d'organisations utilisées dans notre travail.....	43

## Listes des Acronymes

TOR	Toute ou rien.
CPU	Centrale Processing Unit.
PC	Partie commande.
PO	Partie opérative.
PS	Partie supervision.
API	Automate programmable industriel.
PLC	Programmable Logic Controller.
CAN	Conversion analogique numérique.
PC	Modules de communication.
FM	Modules de fonctions.
SM	Modules de signaux.
MPI	Multi Point Interface.
IM	Le coupleur.
RUN-P	Marche.
RUN	Marche en lecture seule.
MRES	Module Reset.
SF	Signalisation groupée de défauts.
OB	Bloc d'organisation.
FB	Bloc fonctionnel.
SFB	Bloc fonctionnel spécialisé.
FC	Fonction.
SFC	Fonction spécialisé.
WinCC	Windows control centre.
IEC	Commission Électrotechnique Internationale

## Introduction générale

L'émergence des nouvelles technologies capable de résoudre la majorité des difficultés à l'évolution rapide du monde industriel, ainsi ce dernier apporte des solutions et optimise au mieux des paramètres essentiels au bon déroulement tel que la production, la sécurité du personnel et du produit, le coût de production, la productivité et les conditions de travail.

Les automates programmables industriels (API) apportent la solution pour tous besoins d'adaptation et de flexibilité de nombreuses activités industrielles.

Le fond de notre travail est effectué au sein de l'entreprise SARL BISTA PRODA, l'objectif est d'étudier le fonctionnement d'une chaîne de production de poste remplissage et bouchonnage à base d'un automate S7-300 Siemens par logiciel STEPS 7.

L'élaboration du programme sera présentée sous le logiciel SIMATIC Manager de Siemens. La simulation et le test du programme se fera sous S7-PLCSIM, la supervision quant à elle se fera sous WinCC Flexible.

Pour une meilleure présentation de notre travail. Le mémoire a été organisé en trois chapitres.

- Le premier chapitre est consacré à la présentation de l'entreprise et généralités sur l'automatismes, nous allons présenter les gammes d'API Siemens et le logiciel WinCC flexible.
- Le deuxième chapitre expose le principe et le fonctionnement des différents composants utilisés dans notre projet comme première partie. Nous allons identifier l'automate de Siemens S7-300, et la présentation du logiciel de travail comme deuxième partie.
- Le dernier chapitre est consacré à la présentation de cahier de charge et leurs GRAFCET, et l'étude d'un système de supervision avec le logiciel WinCC flexible pour visualiser l'état de fonctionnement du poste de remplissage et bouchonnage. Les deux simulations (Sous STEPS 7 et WinCC) ont été présentées, ainsi que leurs détails. La simulation sous STEPS 7 est effectuée par deux manières : S7-PLCSIM et Table de variable.

## 1.1 Introduction

Les automates programmables industriels (API) représentent l'outil de base d'automatisation des systèmes de production. Le but d'exécution d'un ensemble d'opération exécuté séquentiellement.

Dans ce chapitre, nous évoquons en premier partie L'entreprise SARL BISTA PRODA, situation géographique, ses différentes activités industrielles. En suite la deuxième partie, nous avons développé une étude détaillée des automatismes.

## 1.2 Partie 1 Présentation de l'entreprise SARL BISTA PRODA

### 1.2.1 Description de l'entreprise SARL BISTA PRODA

SARL BISTA PRODA (Unité de traitement et de valorisation des produits Agro-alimentaire), Avec le nom commercial « OREVIC », elle est fondée en 2013 par Mr RABAH AMARI.

Le début de « OREVIC » a commencé en magasin de stockages des légumes mais avec le temps l'entreprise SARL BISTA PRODA a commencé la production de minoterie d'aliment de Bétails. Cette unité est composée de quatre zones :

- 1) : Zone de réception.
- 2) : Zone de dosage et broyage.
- 3) : Zone de granulation.
- 4) : Zone de conditionnement en sac et vrac la capacité de production et de TJ/H.

**Zone 1** : Elle est composée de

- Trémie de réception de matière première tel que : Mais, soja Corfou et l'orge.
- Une visse qui permet de transporter ces matières vers les silos.
- Un élévateur a godé.
- Visse de distribution sur les silos.

**Zone 2** : Elle est composée de

- Visse de dosage.
- Peseuse programmable.
- Mélangeuse de produits poussé.
- Broyeur à mentaux RM16.
- Aspirateur des matières broyées.

**Zone 3** : Elle est composée de

- Un pressé BLM92.
- Un maturateur.
- Vise d'alimentation de la presse.
- Le refroidisseur horizontal.
- Silos de stockage des produits granulé.
- Une ensacheuse de sac (25 :50 Kg).

**Zone 4** : Elle est composée de

- Trois silos de stockage de matière granulé.
- Tapis de distribution sur les trois silos.
- Un pont bascule.
- Le port de chargement.

### 1.2.2 Situation géographique

L'unité de production Sarl BISTA PRODA est située à BOUIRA dans la zone de parcs sur la route cent-quarante (Voir la figure 1.1).

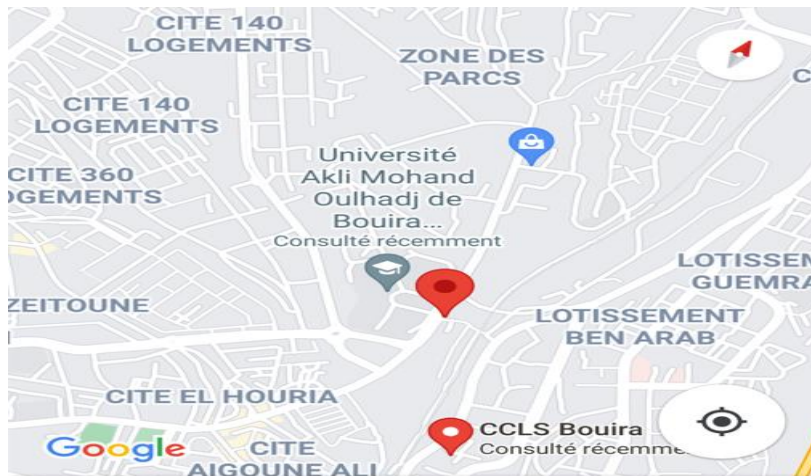


Figure 1.1 : Situation géographique de l'entreprise SARL BISTA PRODA.

### 1.2.3 Produit d'usine BISTA PRODA

Le produit de BISTA PRODA est un produit qui appartient à l'alimentation animale qui décrit le besoin alimentaire des animaux d'élevage et les moyens et méthodes permettant de les satisfaire comme le montre la figure suivante.

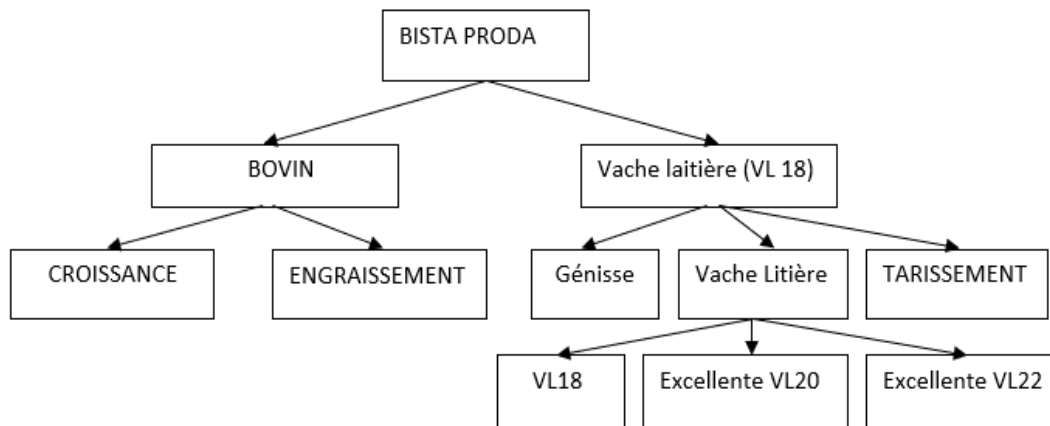


Figure 1.2 : Produits de l'entreprise SARL BISTA PRODA.

L'aliment BISTA PRODA VL 18 apporte l'énergie et les protéines au bon fonctionnement du rumen.

#### 1.2.4 Composition des produits BISTA PRODA

Les produit de l'entreprise SARL BISTA PRODA contient du maïs, son, semoule basse, soja, mélasse, phosphate, sel, levucelle, paille calcaire, carbonate sodium.

### 1.3 Partie 2 Généralités sur l'automatisme

#### 1.3.1 Système automatique

L'automatisme est un sous-ensemble ou un organe de machines à remplacer de façon automatique une action ou, une décision habituelle et prédéfinie sans intervention de l'être humain. Le domaine d'automatisme est très large dans le monde, en générale comme l'industriel, transport, médicale et aviation...etc. (Voir figure 1.3)

#### L'automatisme est caractérisé par :

- Simplifie considérablement le travail de l'homme est qui, libéré vis-à-vis de la machine, peut se consacrer à des activités plus nobles.
- Réduit les tâches complexes, pénibles ou indésirables en les faisant exécuter par la machine.
- Facilite les changements de fabrication en permettant de passer d'une quantité ou d'un type de production à un autre. [1]

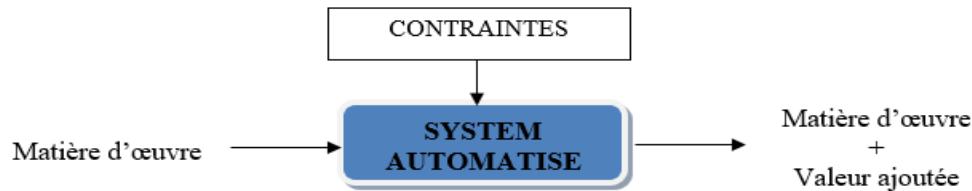


Figure 1.3 : Présentation d'un système automatique.

### 1.3.2 Objectif de l'automatisation

L'objectif de l'automatisation des systèmes est de produire, en ayant recours le moins possible à l'homme, des produits de qualité et ce pour un coût le plus faible possible. Elle permet de :

- ✓ Effectuer les opérations répétitives par les machines.
- ✓ Rendre le système autonome, productif.
- ✓ Minimiser l'intervention de l'être humain.
- ✓ Accroître la productivité de l'entreprise en réduisant les coûts de production.
- ✓ Augmenter la sécurité de personnel.
- ✓ Contrôler et protéger les installations et les machines. [1]

### 1.3.3 Les avantages et les inconvénients de l'automatisation

#### 1.3.3.1. Les avantages

- ✓ La capacité de production accélérée.
- ✓ L'aptitude à convenir à tous les milieux de production.
- ✓ La souplesse d'utilisation.
- ✓ La création de postes d'automaticiens.
- ✓ La simplification des tâches de production et la vie courante.
- ✓ Évité la perte de temps et tous les problèmes des machines industrielles.

#### 1.3.3.2 Les inconvénients

- ✓ Le coût élevé du matériel, principalement avec les systèmes hydrauliques.
- ✓ La maintenance doit être structurée.
- ✓ La complexité de la maintenance.
- ✓ Basés sur les capteurs.
- ✓ La suppression d'emplois (chômage).

### 1.3.4 Structure générale d'un système automatique

Un système automatisé est toujours composé d'une partie supervision (PS), partie commande (PC) et une partie opérative (PO), pour faire fonctionner les systèmes de grandes productivités, l'opérateur va donner des consignes à la (PC), celle-ci va traduire ces consignes en ordre et transmettre à la partie (PO) à partir d'une partie (PC), et la partie (PO) va le signaler à la (PC), par un retour d'information par des capteur (Voire la figure 1.4). [2]

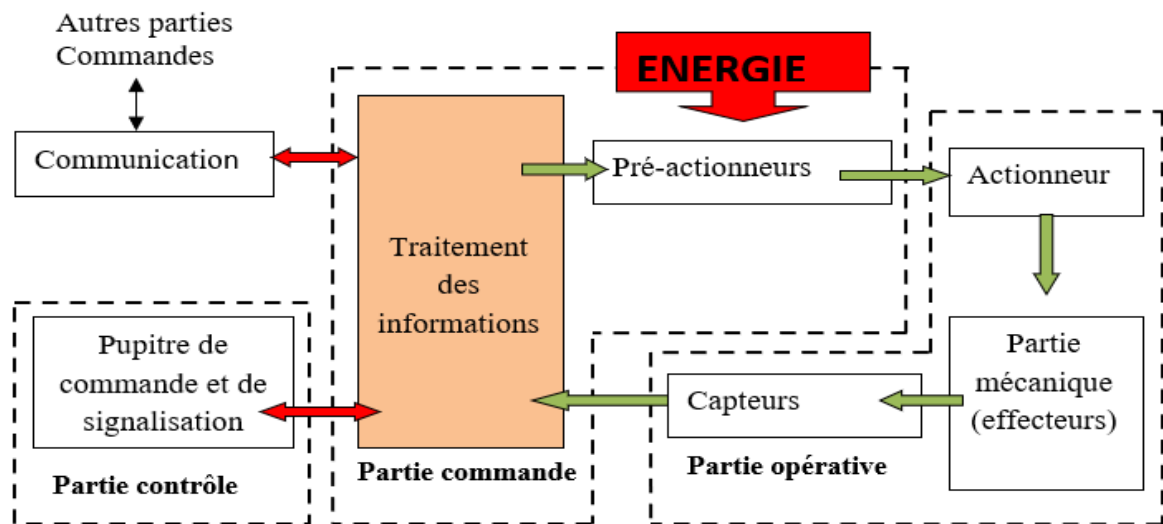


Figure 1.4: Structure générale d'un système automatisé.

Dès sa conception, un système à automatiser doit être décomposé en trois parties :

#### 1.3.4.1 Partie contrôle ou supervision

Représente l'échange des informations entre la partie commande et l'opérateur (pilote, usage, surveillant...) dont elle reçoit des consignes et à qui elle fournit des comptes rendus visuels ou sonores [1]. Elle est composée des pupitres de commande et de signalisation. Il permet à l'opérateur de commander le système (marche, arrêt, départ cycle...) et visualiser les différents états du système à l'aide de voyants, de terminal de dialogue ou d'interface homme-machine (IHM).

#### 1.3.4.2 Partie commande ou API

Elle transmet l'ordre de fonctionnement au pré-actionneur ou actionneurs (partie opérative) à partir : [3]

- ✓ Du programme qu'elle contient.
- ✓ Des informations reçues par les capteurs.
- ✓ Des consignes données par l'utilisation ou l'opérateur.

### 1.3.4.3 Partie opérative

Elle consomme de l'énergie électrique, pneumatique (air) ou hydraulique (eau ou huile). Elle envoie l'information à la partie commande à partir de capteur elle contient :

**-Actionneurs** : qui transforment l'énergie reçue en mouvement de rotation ou translation comme vérin, moteur, électrovanne...etc. [3] Ces actionneurs appartiennent à trois technologies :

- a) Actionneurs pneumatiques (vérins, moteurs).
- b) Actionneur hydraulique (vérins).
- c) Actionneurs électriques (moteurs électriques).

**Pré-actionneur** : est l'élément compris entre les deux parties opératives et commandes, utilisé pour distribuer l'énergie nécessaire qui permet de commander les actionneurs. Le pré-actionneur peut être : [3]

- a) Contacteurs pour moteurs électriques.
- b) Variateurs de vitesse pour moteurs électriques.
- c) Distributeurs pour vérins pneumatiques ou hydrauliques.

**-Capteurs** : qui transforment les variations de grandeurs physiques (température, luminosité, présence, position,) en signaux électrique produisant une information qui permet au système automatisé de déclencher des actions.[3]



Figure 1.5: Principe de fonctionnement d'un capteur.

Afin de mesurer ou de commander le capteur obtient une information qui transmettra la partie commande par les périphériques d'entrée /sortie. Cette information doit être logique (TOR), numérique et analogique. [3]

✓ **Information Tout Ou Rien TOR** : l'information ne peut prendre que deux états (vrai /faux, 0 ou 1...). C'est le type d'information délivrée par un détecteur, un bouton poussoir... [3]

✓ **Information Analogique** : l'information est continue et peut prendre une valeur comprise dans une plage bien déterminée. C'est le type d'information délivrée par un capteur (pression, température...).[3]

- ✓ **Information Numérique** : l'information est continue dans des mots codés sous forme binaire ou bien hexadécimale. C'est le type d'information délivrée par un ordinateur ou un système intelligent.

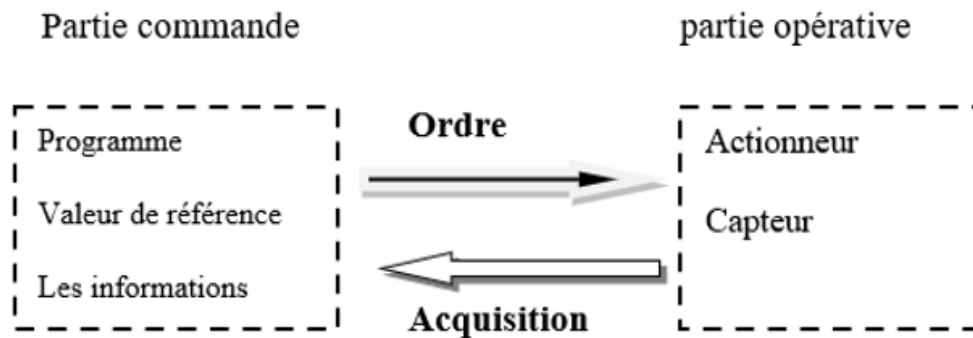


Figure 1.6: Principe de transmission d'information.

## 1.4 Les automates programmables industriels (API)

### 1.4.1 Définition d'un API

Un automate programmable industriel (API) est une forme particulière d'automate à base de microprocesseur qui se base sur une mémoire programmable pour enregistrer et mettre en œuvre des fonctions, qu'elles soient logiques, de séquençement, de temporisation, de comptage ou arithmétiques, pour contrôler des machines et des processus. Il est conçu pour être manipulé par des ingénieurs ayant, potentiellement, une connaissance limitée en informatique et en langages de programmation. [3]

✚ **Un automate programmable industriel (API) est caractérisé par :**

- Connexion directe aux différents capteurs et actionneurs grâce à ces entrées/sortie.
- Fonctionnement dans des conditions industrielles sévères (température, vibrations, humidité, microcoupure de l'alimentation en énergie électrique...).
- Son aspect pratique grâce à la possibilité de sa programmation en utilisant un langage spécialement développé pour le traitement de fonctions d'automate (Step7). [1]

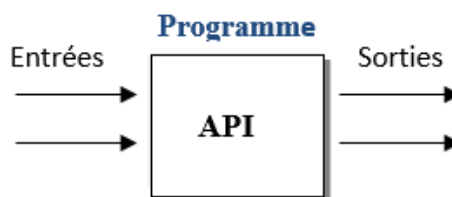


Figure 1.7: Automate programmable industriel. [3]

### 1.4.2 Quelques fabricants d'API dans le monde

Les entreprises les plus influentes dans le domaine des automatismes sont :

- Siemens (S7-200, S7-300, S7-400, S7-400, S7-1200...).
- Schneider électrique (TSX 17/37/57, TSX micro, premium...).
- ABB (AC500, AC800C, S500...).
- Omron (ZEN, CPM 1/2A/2C, CS1, CJ1...) ...etc. [3]

### 1.4.3 Type d'un API

Les systèmes (API) sont principalement disponibles sous deux formes : en boîtier unique qui s'appelle COMPACT et en version modulaire rack qui s'appelle MODULAIRE.

#### 1.4.3.1 Automate compacte

Les automates de type compact ou micro automates, est un modèle en boîtier unique, ou coffret, est seront utilisés pour les petits automates programmables et correspond à un système compact complet équipé d'unités d'alimentation, de processeur (CPU), de mémoire et d'entrée-sortie limité. [3]

✚ L'automate compacte est caractérisé par :

- Disposer de six, huit, douze ou vingt-quatre entrées, de quatre, huit ou seize sorties.
- Une mémoire permettant d'enregistrer entre 300 et 1000 instructions.



Figure 1.8: Automate programmable industriel de type compacte.

Exemples : LOGO de Siemens, ZELIO de Schneider, S7-200 de Siemens...

Ces automates sont de fonctionnement simple et sont généralement destinés à la commande de petits automatismes.

### 1.4.3.2 Automate modulaire

Un automate modulaire est constitué de modules séparés pour l'alimentation, le processeur (CPU)... etc. souvent montés sur des rails dans une armoire métallique, et ces modules sont ensuite branchés dans le rack. [3]

**✚ L'automate modulaire est caractérisé par :**

- ✓ Son employabilité pour toutes les tailles d'automates programmables.
- ✓ Le choix des modules nécessaires à la réalisation d'un projet précis est décidé par l'utilisateur.
- ✓ La facile d'étendre le nombre d'entrées-sorties ou la quantité de mémoire.



Figure 1.9: Automate programmable industriel de type modulaire.

Exemple : automate Siemens S7-300, S7-400, tta potail S7-1200...etc.

### 1.4.4 Architecture d'un API

Un API est structuré autour de plusieurs éléments de base que sont l'unité de traitement, la mémoire, l'unité d'alimentation, les interfaces d'entrées-sorties, l'interface de communication et le périphérique. (Voir Figure 1.10). [3]

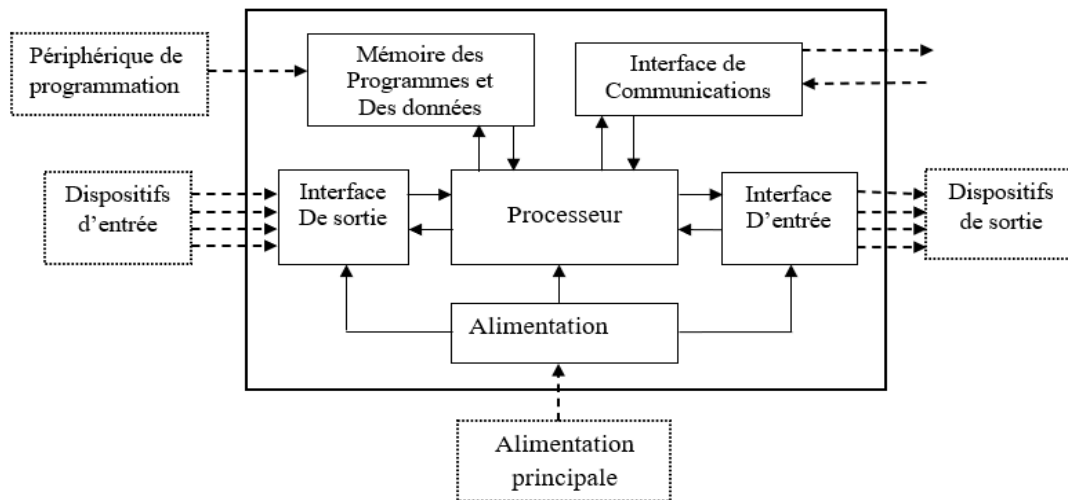


Figure 1.10: Structure d'un automate programmable industriel.

#### 1.4.4.1 Processeur

Unité centrale de traitement (CPU, Centrale Processing Unit) contient le microprocesseur. Le CPU interprète les signaux d'entrée et effectue les actions de commande conformément au programme stocké en mémoire, en communiquant aux sorties les décisions sous forme de signaux d'action[3]. En général, le CPU lie l'information d'entrée, puis exécute le programme mis en ROM et puis commandé les sorties.

✚ **Le processeur est caractérisé par :**

- 1-capacité de la mémoire.
- 2-nombre d'entrée-sortie maximale.
- 3-vitesse d'exécution d'une instruction.
- 4-fonction spéciale...etc. [2]

#### 1.4.4.2 L'unité d'alimentation

L'alimentation est indispensable puisqu'elle convertit une tension d'alimentation alternative de en une basse tension continue (5 V) nécessaire au processeur et aux modules d'entrées-sorties.

L'automate est alimenté généralement par le réseau monophasé 230V ; 50 Hz mais d'autres alimentations sont possibles (110V etc. ...).

#### 1.4.4.3 Le périphérique de programmation

Utilisé pour entrer le programme dans la mémoire du processeur. Ce programme est développé sur le périphérique, puis transféré dans la mémoire de l'API. [3]

#### 1.4.4.4 La mémoire

La mémoire d'un système (API) contient le programme qui définit les actions de commande effectuées par le microprocesseur. Elle contient également les données qui proviennent des entrées en vue de leur traitement, ainsi que celles des sorties.[3]

La capacité de stockage d'une unité de mémoire est déterminée par le nombre de mots qu'elle peut enregistrer. Les types des mémoires utilisés sont :

- ✓ **RAM** (Random Access Memory) utilisé pour stocker les données et résultats temporaires.
- ✓ **ROM** (Read Only Memory) ou **PROM** (Rom Programmable) utilisées pour stocker les instructions de programme d'un système qui doit fonctionner.

#### 1.4.4.5 Les interfaces d'entrées-sorties

Permettent au processeur de recevoir et d'envoyer des informations aux dispositifs extérieurs. Les entrées peuvent être des interrupteurs, comme dans le cas de la perceuse automatique, ou d'autres capteurs, des sondes de température, des débitmètres, etc. Les sorties peuvent être des bobines de moteur, des électrovannes, etc. Les dispositifs d'entrées-sorties peuvent être classés en trois catégories, selon qu'ils produisent des signaux discrets, numériques ou analogiques. [3]

##### A) L'interface d'entrée

Elle permet d'acquérir les informations des organes de commande et des capteurs (saisie l'information). Contient des formes suivantes :

- ✓ Entée tout ou rien (TOR)
- ✓ Entrée analogique
- ✓ Entrée numérique

##### B) L'interface de sortie

Elle permet d'effectuer la commination entre le CPU et les actionneurs, et transmettre l'ordre à la partie opérative. Elle contient des formes suivantes :

- ✓ Sortie tout ou rien (TOR)
- ✓ Sortie analogique
- ✓ Sortie numérique

#### 1.4.4.6 L'interface de communication

Est utilisée pour recevoir et transmettre des données sur des réseaux de communication qui relie l'API à d'autres (API) distants.

### 1.4.5 Fonctionnements d'un API

L'automate programmable industriel est un appareil qui traite les informations selon un programme préétabli, son fonctionnement est basé sur l'emploi, d'un micro-processeur et de mémoire. Le fonctionnement de l'API peut se résumer de la manière suivante (Voir Figure 1.11) :

1. Scruter toutes les entrées et les copier en RAM.
2. Lire, décoder et exécuter toutes les instructions du programme en séquence, en copiant le résultat des instructions en RAM.
3. Affecter toutes les sorties.
4. Répéter la procédure.

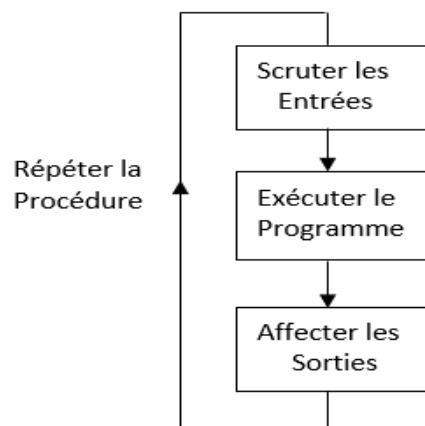


Figure 1.11: Cycle de fonctionnement d'un API.

✚ **Le temps de cycle** ou de scrutation d'un API, c'est-à-dire son temps de réponse, est déterminé par les facteurs suivants :

1. Le CPU employé.
2. La taille du programme à analyser.
3. Le nombre d'entrées-sorties à gérer.
4. Les fonctions systèmes utilisées ; plus leur nombre est grand, plus le temps de scrutation est long. [3]

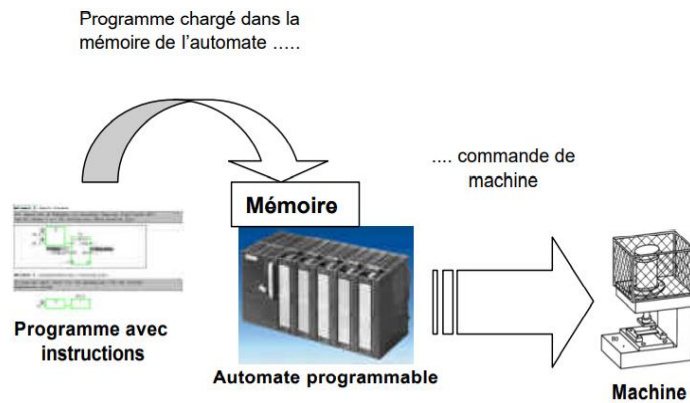


Figure 1.12 : Commande des machines avec l'API.

### 1.4.6 Programmes d'un API

Les programmes utilisés avec les API peuvent être écrits dans différents formats. Pour que les ingénieurs ayant peu de connaissances en programmation puissent élaborer des programmes pour les API, le langage à contacts a été conçu. La plupart des fabricants d'automates ont adopté cette méthode d'écriture des programmes. [3]

Pour toutes les méthodes de programmation employées avec les API, on trouve une norme désignée sous la référence CEI 61131-3, publiée en 1993 par la commission électrotechnique internationale – qui contient cinq langages et sont le langage à contacts (LD, Ladder Diagram), les listes d'instructions (IL, Instruction List), les graphes de fonction séquentielle (SFC, Sequential Function Charts), le texte structuré (ST, Structured Text) et les diagrammes de schémas fonctionnels (FBD, Function Block Diagram). La norme comprend une bibliothèque de fonctions préprogrammées et des blocs fonctionnels. Parmi ces langages, deux sont textuels, et les autres sont graphiques. [3]

#### 1.4.6.1 Langage textuels

Le langage textuel contient le langage listes d'instruction, et le texte structuré, seront saisis sur l'appareil de programmation à partir d'un clavier, une ligne à la fois. [3]

**a. Langage liste d'instruction** de même nature que l'assembleur (programmation des microcontrôleurs). Très peu utilisé par les automaticiens. [3]

**b. Langage texte structuré** de même nature que le pascal, (langage informatique). Peu utilisé par les automaticiens. [3]

#### 1.4.6.2 Langages graphiques

Le langage graphique est un programme peut donc être développé à partir d'éléments graphiques sur l'écran de l'appareil de programmation. Et ce langage qui contient les langages à contacts, les graphes de fonction séquentielle et les diagrammes de schémas fonctionnels. [3]

**a. Langage à contacts** est un langage développé pour les électriciens, se trouver dans les schémas de câblage électrique pour les systèmes de commande à relais. Ce langage est essentiellement dédié à la programmation d'équation booléenne (true/false). (Voire la figure I.13) [3]

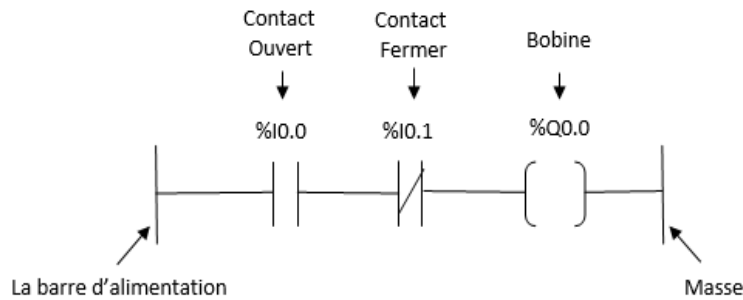


Figure 1.13: Commande des machines avec l'API. [3]

C'est un langage volontairement simple et graphique pour être compréhensible. Il a permis dans les années 1990 d'être utilisé sans lourde formation par un électricien. Il est aujourd'hui un peu dépassé. La puissance de calcul des CPU actuelles permettent de travailler directement en langage objets avec des notions de classe et d'héritage. [3]

Il existe 3 types d'élément de langage Ladder (Ladder Diagram) :

- Les entrées (ou contact), qui permettent de lire la valeur d'une variable booléenne,
- Les sorties (ou bobines) qui permettent d'écrire la valeur d'une variable booléenne,
- Les blocs fonctionnels qui permettent de réaliser des fonctions avancées.

Le tableau 1.1 donne les principaux éléments (contacts et bobines) d'un réseau LD.

Objet graphique	Nom
-  -	Contact normalement ouvert
- / -	Contact normalement fermé
- P -	Contact fermé au front montant
- N -	Contact fermé au front descendant
-( )-	Bobine normalement ouverte
-( / )-	Bobine normalement fermée

-(S)- ou-(L)-	Bobine Latch (maintenu à 1 une fois actionné)
-(R)-ou -(U)-	Bobine Reset (remise à 0 de la bobine latch)
-(P)-	Bobine active au front montant de son entrée
-(N)-	Bobine active au front descendant de son entrée

Tableau 1.1: Principaux éléments d'un réseau LD. [3]

### b. Graphe de fonction séquentielle ou GRAFCET

Le mot GRAFCET signifie graphe fonction de commande étape-transition, est un langage graphique pour décrire, étudier, réaliser, exploiter et présenté les automatismes. Cette représentation graphique permet une meilleure compréhension de l'état de système d'automatisme par tous les intervenants. [3] (Voir la figure 1.14)

Il existe deux éléments de base d'un langage GRAFCET :

✚ **Les éléments graphiques** : contient trois éléments de base ; les étapes, les transitions, les liaisons. [3]

**-Les étapes** : Une étape est une situation dans laquelle le comportement d'une partie ou de totalité du système automatisé est invariant par rapport à ses entrées et ses sorties.[3]

**-Les transitions** : la transition associée à l'intermédiaire de deux étapes. Chaque transition doit avoir une condition. Si la transition doit toujours se produire, sa condition doit être True. [3]

**-Les liaisons** : Une liaison est le lien qui lie une étape à une transition ou l'inverse. [3]

✚ **Les éléments de l'interprétation** : contient deux éléments de base ; les actions, les réceptivités. [3]

**-Les actions** : L'action associée à l'étape peut être de 3 types : continue, conditionnelle ou mémorisée. Et l'action activée lorsque l'étape associée à cette action est active. [3]

**-Les réceptivités** : Une réceptivité est associée à chaque transition et peut être 4 types :

- une étape
- un capteur
- une temporisation
- un compteur

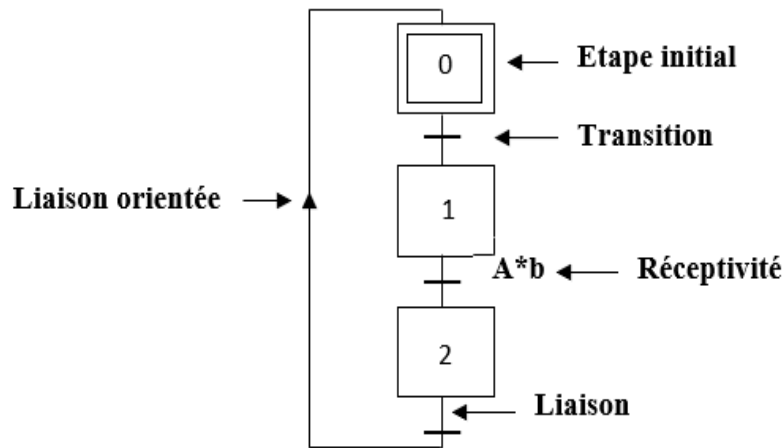


Figure 1.14: Présentation d'un élément de GRAFCET.

c. **Les diagrammes de schémas fonctionnels** est une instruction du programme qui, lorsqu'elle est exécutée, produit une ou plusieurs valeurs de sortie qui montre la représentation d'un bloc utilisé par l'automaticien.

#### 1.4.7 Critère de choix d'un API

Après avoir étudié notre système dans les chapitres précédent, le choix des API revient considérer certains critères important tels que :

- Le fabricant d'un automate.
- La possession d'un logiciel de programmation.
- Nombre d'entrées / sorties.
- Type de processeur.
- Fonctions ou modules spéciaux.
- Fonctions de communication.

Pour notre travail, nous avons choisi l'automate SIEMENS S7-300 :

- 19 entrées et 18 sorties ;
- Unité centrale CPU 412C-2 PN/DP ;
- Configuration multi rangée jusqu'à 32 modules.

### 1.5 L'automate programmable industriel SIEMENS

#### 1.5.1 Présentation quelque gamme SIEMENS

Le nom SIMATIC est portemanteau de « Siemens » et « Automatic », et est une série de Programmable Logic Controller et systèmes d'automatisation, développés par Siemens. Lancée en 1958, la série a traversé quatre générations majeures, SIMATIC S5 programmé par logicielle steps 5

et la dernière étant la SIMATIC S7 programmé par logiciel steps 7 et contient les différents automates S7-300, S7-400, S7-1200, S7-1500...etc. (Voir la figure 1.15)

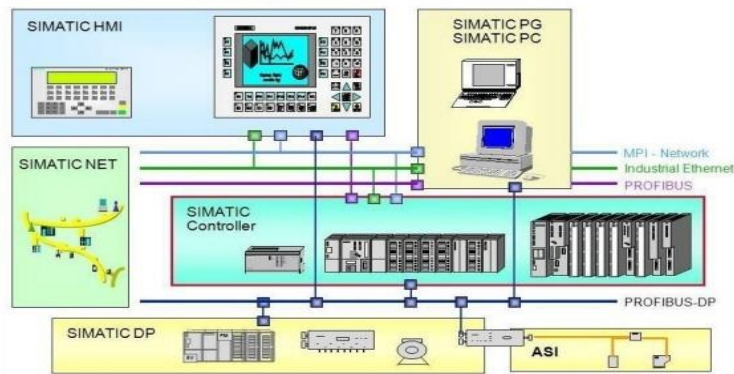


Figure 1.15: Présentation des gammes de SIMATIC.

### 1.5.2 L'interface de communication SIMENS

Interfaces de communication que vous pouvez trouver sur un automate programmable SIMENS est :

- La communication série RS 232, RS 422, et EIA-485.
- La communication MPI.
- La communication via profibus DP.
- La communication via profinet IO.

### 1.5.3 Adresses des entrées-sorties d'un API SIMATIC

La déclaration d'une entrée ou sortie pour le modèle SIMATIC S7 de siemens donné à l'intérieur d'un programme s'appelle l'adressage. [3]

Les entrées-sorties des automates sont regroupées par huit, chaque groupe est un OCTET et chaque entrée ou sortie d'un groupe est un BIT. Ce constitue un numéro de module suivie d'un numéro de port, séparés par un point (.) (Voire la figure 1.16).

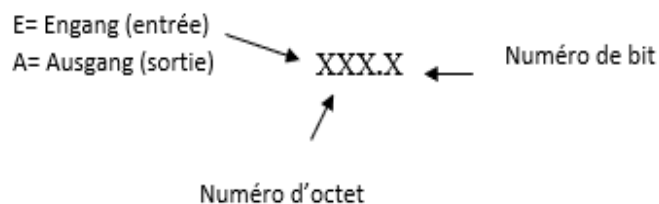


Figure 1.16: Format d'une adresse dans l'automate SIMATIC S7 . [3]

## **1.6 Présentation de Win CC Runtime Advanced**

### **1.6.1 Définition**

SIMATIC WinCC est un système SCADA qui permet la visualisation et l'automatisation des processus.

Lorsque la complexité des processus augmente et que les machines et installations 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). [2]

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. [2]

### **1.6.2 Les tache d'un SIMATIC HMI**

SIMATIC HMI offre une gamme complète permettant de couvrir toutes les tâches de contrôle-commande. Est permet de :

- Représentation du process.
- Commande du processus.
- Vue des alarmes.
- Archivage de valeurs processus et d'alarmes.
- Documentation de valeurs processus et d'alarmes.
- Gestion des paramètres de processus et de machine. [2]

### **1.6.3 Utilisation de SIMATIC WinCC flexible**

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 : [2]

- Simplicité
- Ouverture
- Flexibilité

#### **1.6.4. Eléments de WinCC flexible**

##### **A. WinCC flexible Engineering System**

WinCC flexible Engineering System est le logiciel avec lequel vous réalisez toutes les tâches de configuration requises. L'édition WinCC flexible détermine les pupitres opérateurs de la gamme SIMATIC (HMI) pouvant être configurés. [2]

##### **B. WinCC flexible Runtime**

WinCC flexible Runtime est le logiciel de visualisation de process. Dans Runtime, vous exécutez le projet en mode process. [2]

##### **C. Options WinCC flexible**

Les options WinCC flexible permettent d'étendre les fonctionnalités de base de WinCC flexible. Chaque option nécessite une licence particulière. [2]

#### **1.7. Conclusion**

Ce chapitre donne un aperçu sur les différentes zones de l'usine de production SARL BISTA PRODA et leur activité comme première partie, ensuite on a donné d'une façon détaillée sur l'automate programmable industriel que nous allons identifier les familles et matériel d'un API.

Le choix de l'automate dépend des éléments suivants : coût, CPU, le nombre des entrées et sorties dont le système que doit étudier.

Enfin, on a présenté l'automate programmable industriel SIEMENS et logicielle d'un commande-contrôle d'une machine le WINCC flexible Advanced.

## 2.1 Introduction

Dans ce chapitre nous allons identifier les deux parties matérielle et logicielle de notre projet qui consiste en l'automatisation du poste de remplissage et bouchonnage des bouteilles.

Nous allons présenter dans la première partie les différents matériels d'une machine et leur fonctionnement. Et dans la deuxième partie nous allons donner une description détaillée de l'automate S7-300 programmé par logiciel steps 7.

## 2.2 But et importance de l'étude

Le système de remplissage et bouchonnage est mis en place dans le but de remplir et boucher des bouteilles d'une façon automatique et fiable. Donc sans perte de temps et sans l'intervention de l'être humain.

Ce système est commandé par un automate programmable industrielle S7-300 supportant son environnement agressif, qui contrôle, guide et assure la bonne démarche du travail.

En plus, ce système contient une partie supervision de logiciel WINCC flexible runtime (RT) pour contrôler l'état du système et pour éviter tous les problèmes de la partie opérative.

Ce projet de poste de remplissage et bouchage des bouteilles est composé de trois postes :

- ✚ Poste de transfert : consiste au chargement des bouteilles à base de tapis roulant à l'intermédiaire du vérin A.
- ✚ Poste de remplissage : consiste au remplissage des bouteilles avec une pompe mue commandée par une électrovanne EV.
- ✚ Poste de bouchonnage : consiste au transfert de bouchons par le vérin G, d'un moteur pneumatique F pour tourner et visser le bouchon.



Figure 2.1: Exemple d'un chaine de production .

## 2.3 Partie 1 Composition du système

Ce système est constitué par deux parties principales dont la partie opérative est :

### 2.3.1 Partie électrique

Dans cette partie on définit les différents capteurs utilisés dans ce système pour transmettre les informations à la partie commande.

#### 2.3.1.1 Capteurs

Un capteur est un composant technique qui détecte un événement physique se rapportant au fonctionnement du système (présence d'une pièce, détecteur niveau etc.) et traduit cet événement en un signal électrique sous forme d'un signal basse tension. [4]

On distingue essentiellement les capteurs de chute de pression pour la position d'un vérin pneumatique et de capteur de proximité inductif. La figure (2.2) donne le principe d'un capteur.



Figure 2.2: Schéma fonctionnel d'un capteur. [5]

#### 2.3.1.2 Caractéristiques métrologiques d'un capteur

Les liens entre un capteur et la grandeur qu'il mesure sont définis par ses caractéristiques d'emploi.

##### a. Etendue de mesure

L'étendue de mesure est la valeur extrême pouvant être mesurée par le capteur. [5]

##### b. Sensibilité

C'est le rapport de la variation du signal de sortie à la variation correspondante de la grandeur à mesurer. [5]

##### c. précision

C'est l'aptitude du capteur à donner des indications proches de la valeur vraie de la grandeur mesurée. [5]

#### **d. Fidélité**

Un capteur est dit fidèle si le signal qu'il délivre ne varie pas pour une série de mesures concernant la même valeur de la grandeur d'entrée. [5]

#### **e. Rapidité**

C'est l'aptitude du capteur à suivre dans le temps les variations de la grandeur à mesurer. Il faut donc tenir compte du temps de réponse, de la bande passante et la fréquence de coupure du capteur. [5]

### **2.3.1.3 Familles des capteurs**

#### **2.3.1.3.1 Capteurs TOR**

Ce type de capteur permet de détecter un événement ou un objet lié au fonctionnement du système. On les appelle aussi les capteurs « tout ou rien » (TOR) car le signal électrique fourni est de type logique (0 ou 1). [1]

Exemple du capteur de niveau : Ce capteur, de type logique (TOR), est conçu pour la détection de niveau de liquides de toutes natures (niveau liquide entre la bouteille dans notre système). Son rôle est de détecter le niveau de la vanne.



Figure 2.3: Capteur tout ou rien TOR.

#### **2.3.1.3.2 Capteurs analogique**

Dans la pratique industrielle, on donne à ce type de matériel le nom de capteur, si on lui inclut une électrique de puissance dans son boîtier, on parle de capteurs transmetteurs,[1] l'information ce forme un signal analogique souvent une sortie courante 4MA à 20MA, ou tension 0V à 10V.

Exemple d'un capteur analogique : PT-100 sonde température, détecteur de niveau (ultrason).



Figure 2.4: Capteur analogique.

### 2.3.1.3.3 Capteur numérique

Selon le cas portent les noms en pratique de codeurs ou de compteurs, l'information fournie par ce capteur permet à la partie commande d'en déduire un nombre binaire sur n bits ou hexadécimale, on parle alors d'un capteur numérique. [1]

### 2.3.1.4 Domaines d'utilisation des capteurs

Le tableau suivant montre les différents domaines d'utilisation des capteurs. (Voir le tableau 2.1)

Mécanique	Electrique	Climatique	Divers
Indication de présence	Tension	Température	Rayonnement
Déplacement linéaire	Courant	Humidité	Luminosité
Déplacement angulaire	Puissance	Vent	Niveau acoustique
Vitesse	Fréquence	Pluviométrie	Célérité acoustique
Accélération	Champ électrique	Ensoleillement	Champ magnétique
Force	Charge	Durée d'insolation	Débit
Pression	/	Durée de pluie	Couleur
Couple	/	/	Salinité

Tableau 2.2: Différents domaines d'application des capteurs. [5]

### 2.3.1.5 Critères de choix d'un capteur

D'après la figure (2.5), nous allons montrer comment bien choisir les défèrent capteurs qui repondent aux besoins d'étude ou de de réalisation d'une chaine de production.

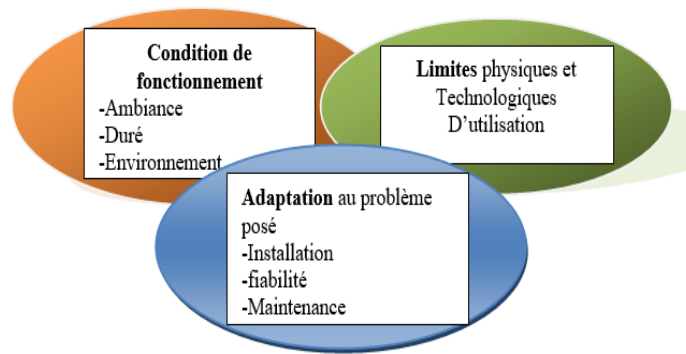


Figure 2.5: Critères de choix d'un capteur. [5]

### 2.3.1.6 Capteurs de proximité inductive

Le capteurs de proximité inductive permet de détecter les objets magnétiques, il est constitué essentiellement de deux éléments principaux, à savoir : une tête détectrice et un oscillateur sous forme de bobinage, le tout forme une forme sensible créant ainsi un champ magnétique.[6]

Dans ce travail nous allons utiliser les capteurs de proximité magnétique pour détecter la position du piston des vérins pneumatiques. Ils sont fixés directement sur le vérin. Nous avons utilisé 12 capteur pour les six vérins.

### 2.3.1.7 Capteur à chute de pression

Le capteur à chute de pression est un capteur qui est principalement utilisé pour capter la fin de courses d'un vérin. Il est généralement placé sur l'entrée et la sortie d'air d'un vérin. Quand le vérin est en mouvement, la chambre qui n'est pas sous pression subit une réduction de son volume. [7]

L'information obtenue par ce type de capteurs et de type tout ou rien TOR, utilisé dans ce système pour contrôler la position de moteur pneumatique F pour visser la bouteille.



Figure 2.6: Capteur à chute de pression.

### 2.3.1.8 Capteurs de niveau

Le mot « niveau » a de nombreux sens dans la langue française. En physique, le niveau correspond à la distance entre la surface d'un liquide et le fond du réservoir le contenant. [8]

Le contrôle de niveau est utilisé dans de nombreux et divers domaines, afin de connaître l'état de remplissage de bouteille. Ce type d'un capteur est analogique utilisé dans ce système pour contrôler le niveau de l'eau dans la vanne.



Figure 2.7: Capteur de niveau.

L'association d'une détection de niveau d'eau haut et d'une détection de niveau bas permet d'automatiser le remplissage d'une bouteille. Ils servent en général à fournir un signal électrique d'entrée au système d'alarme ou d'arrêt, aux automates programmables industrielles. [8]

### 2.3.1.9 Capteur de débit

Un débitmètre est un instrument utilisé pour mesurer le débit linéaire, non linéaire, de masse ou volumétrique d'un fluide. Utilisé dans ce système pour mesurer la quantité de liquide dans la bouteille.



Figure 2.20: Le débitmètre

### 2.3.2 Partie pneumatique

Le but de cette partie est de décrire les pré-actionneurs et les actionneurs pneumatiques, que l'on peut utiliser pour fonctionner notre système.

- ✓ Parmi les actionneurs utilisés dans ce système les vérins pneumatique, moteur etc.
- ✓ Parmi les pré-actionneurs utilisés dans ce système les distributeurs pour actionner les vérins, l'électrovanne etc.

#### 2.3.2.1 Vérins pneumatiques

Un vérin pneumatique est un dispositif électronique utilisé comme actionneur, dans la partie opérative d'un (API). Permettant de convertir la pression de l'aire en une force et un mouvement linéaire. Ils produisent l'énergie mécanique sous forme d'un mouvement permettant de provoquer un déplacement ou de créés une force. [4]

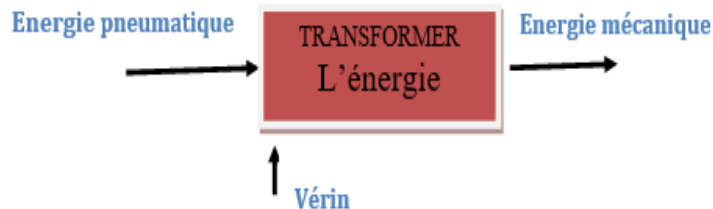


Figure 2.8: Le principe d'un vérin pneumatique.

Il existe deux types de vérins pneumatiques, vérin simple effet et vérin double effet, contiennent principalement les mêmes éléments, le piston est solidaire à la tige qui peut se déplacer à l'intérieur du corps.

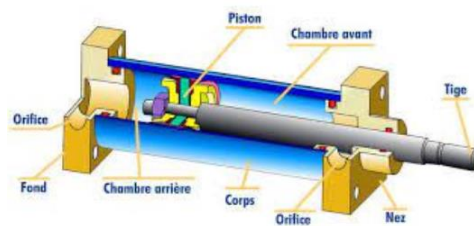


Figure 2.9: Schéma d'un vérin pneumatique.

#### ✚ Principe de fonctionnement du vérin pneumatique

Le vérin pneumatique est un composant qui a besoin d'un pré-actionneur principalement un distributeur pour travailler. (Voir le tableau 2.2)

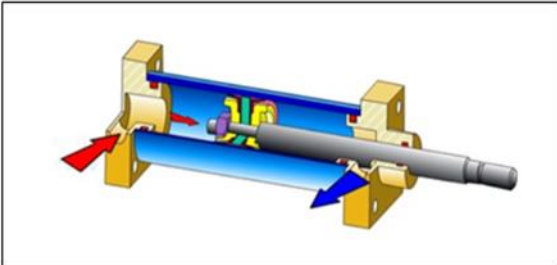
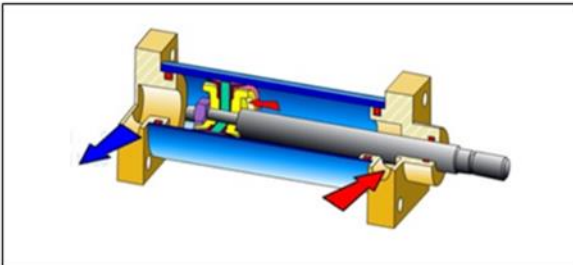
Principe de fonctionnement la sortie de tige de vérin	Principe de fonctionnement la rentrée de tige de vérin
C'est l'air comprimé qui, en pénétrant dans l'une des chambres, pousse le piston. La tige se déplace. L'air présent dans l'autre chambre est donc chassé et évacué du corps du vérin	Le mouvement contraire est obtenu en inversant le sens de déplacement de l'air comprimé
	

Tableau 2.2: Mode de fonctionnement d'un vérin pneumatique.

### 2.3.2.1.1 Le vérin simple effet

L'ensemble tige-piston se déplace dans un seul sens sous l'action du fluide sous pression. Le retour est effectué par un autre moyen que l'air comprimé : le ressort charge. Pendant le retour l'orifice d'admission de l'air comprimé est mis à l'échappement. (Voir la figure 2.10) [4]

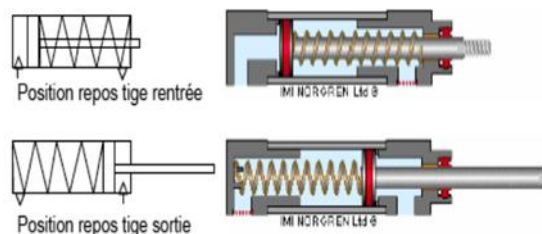


Figure 2.10: Vérin simple effet.

### 2.3.2.1.2 Vérin double effet

Un vérin double effet a deux directions de travail. Il comporte deux orifices d'alimentation et la pression est appliquée alternativement de chaque côté du piston ce qui entraîne son déplacement dans un sens puis dans l'autre, comme observé la figure suivante. [4]

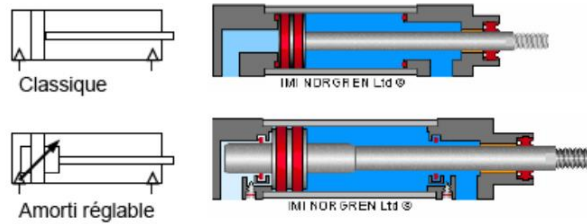


Figure 2.11: Vérin double effet.

Dans notre travail nous allons utiliser six vérins à double effet pour faire fonctionner notre projet.

### 2.3.2.2 Distributeur pneumatique

Le distributeur pneumatique est un composant électronique, appelé encore pré-actionneur électropneumatique. Il est associé à un vérin pneumatique, qui a pour fonction de délivrer un débit d'air comprimé à la réception d'un signal de commande (voir la Figure 2.12). [4]

Dans notre projet nous allons choisir les distributeurs pneumatiques par la commande électrique.

Il existe plusieurs modèles : 2/2,3/2,4/2,5/2 (orifices/position).



Figure 2.12 : Le distributeur.

Ce distributeur est donc destiné à diriger le fluide sous pression vers l'actionneur en sens inverse, et par une autre voie, il assure le retour sans pression à l'air libre (échappement). Les distributeurs possèdent les mêmes éléments de base (Voir la figure 2.13) : [4]

- ✓ Le corps.
- ✓ Le tiroir cylindrique en acier.
- ✓ Le pilotage électrique.
- ✓ Nombre d'orifice.
- ✓ Nombre de position

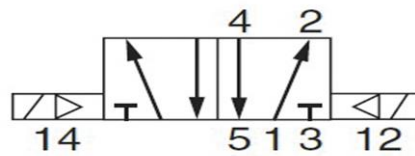


Figure 2.13 : Distributeur 5/2.

**Distributeurs 5/2** à cinq orifices (pression, sortie 1, sortie 2, échappement 1, échappement 2) et deux positions.

**Exemple** : pour alimenter les vérins en air comprimé, on utilise des pré-actionneurs appelés distributeurs. Ces derniers sont caractérisés par le nombre dont ils disposent et par leur nombre de positions de fonctionnement. Dans l'exemple ci-dessus, on alimenter un vérin double effet avec distributeur 5/2 à simple pilotage à commande électropneumatiques.

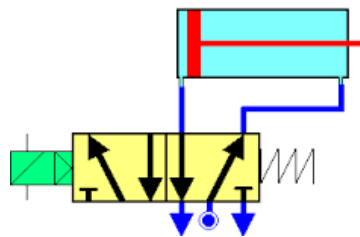


Figure 2.14 : Pilotage d'un vérin double effet par un distributeur 5/2.

### 2.3.2.3 Moteur pneumatique

Les moteurs pneumatiques, ou moteurs à air comprimé, convertissent l'énergie de l'air comprimé en travail mécanique par un mouvement rotatif.

Utilisé dans notre travail pour viser le bouchon de bouteille, le temps de lire un chut pression le pince alimente par le moteur est relâché.

### 2.3.2.4 Moteurs pas à pas

Un moteur pas à pas un actionneur qui produit une rotation par l'intermédiaire d'un mouvement angulaire, le pas, pour chaque impulsion numérique envoyée en entrée (voir la figure 2.15). Pour obtenir une révolution complète de 360°, deux cents impulsions sont nécessaires. [4]. Le moteur peut donc utiliser pour un positionnement angulaire précis

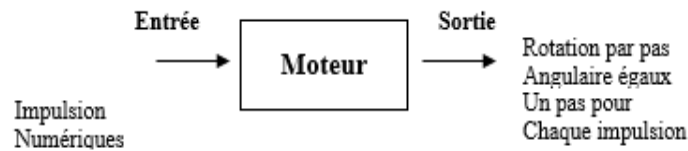


Figure 2.15 : Moteur pas à pas.

Il existe deux types de moteur pas à pas de base : à aimants permanents, avec un rotor à aimants permanents, et à réluctance variable, avec un rotor de fer doux. Il existe également un type hybride qui combine ces deux technologies. [3]

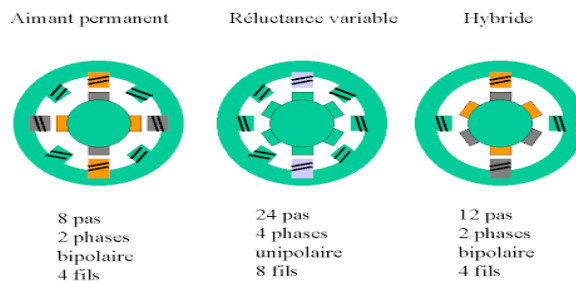


Figure 2.16: Principe de base d'un moteur pas à pas.

Un moteur pas à pas est utilisé dans ce projet, il est destiné pour actionner le convoyeur qui va déplacer les bouteilles.

### 2.3.2.5 Une pompe à eau

Dans notre projet nous avons utilisé une pompe comme actionneur dans la partie opérative d'un système automatisé à partir d'une commande électrique



Figure 2.17: Une pompe à eau

### 2.3.2.6 Relais électromagnétique

Un relais électromagnétique est un dispositif électronique, qui est utilisé dans domaine automatisme industrielle pour la protection d'automate programmable industriel (API) et le matérielle de la partie opérative.



Figure 2.18: Relais électromagnétique.

### 2.3.2.7 Le convoyeur

Dans notre étude nous avons utilisé deux convoyeurs, équipé d'un système de guidage qui transporte les bouteilles de volume 0.5l. Un convoyeur utilisé pour les chargements des bouteilles vides, et l'autre transporte les bouteilles vers les remplissages ensuite vers le bouchonnage.

Ce convoyeur commandé par un moteur pas à pas ou d'un vérin pneumatique.



Figure 2.19: Exemple d'un convoyeur.

### 2.3.2.8 Electrovanne

Une électrovanne est un dispositif électronique utilisé dans la partie opérative comme pré-actionneur électromagnétique TOR. [7]

Ces pré-actionneurs sont commandés par une tension continue de 24 volts (Voire la figure 2.21).

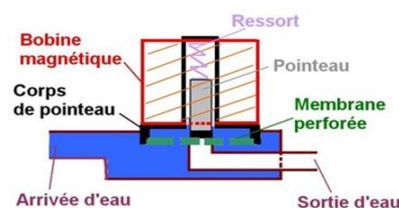


Figure 2.21: Schéma d'une électrovanne.

L'électrovanne est constituée principalement d'un corps de vanne est munie d'une bobine alimentée électriquement engendrant une force magnétique qui déplace le noyau mobile qui vient agir sur l'orifice de passage. En se déplaçant, le noyau permet ou non le passage d'air.[6]

## 2.4 Partie 2 Généralité d'automate S7-300 d'un SIMENS

### 2.4.1 Présentation de l'automate utilisée S7-300

L'automate programmable industriel S7-300 est un mini automate modulaire fabriqué par SIMENS pour les applications d'entrée et de milieu de gamme, qui fait partie de la gamme SIMATIC S7 est un automate destiné à des tâches d'automatisation moyennes et hautes gammes. [2]

Un automate S7-300 consiste en une CPU, un module d'alimentation PS, un module de signaux FM, un module de signaux SM et un processeur de communication. (Voir la figure 2.22)

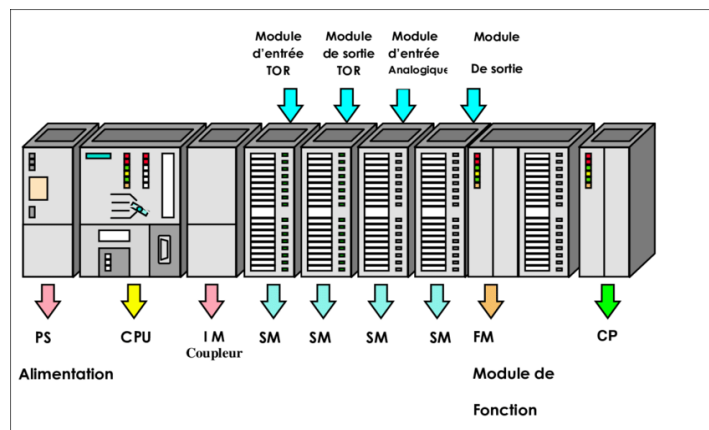


Figure 2.22: Constitution d'un automate programmable industriel S7-300.[2]

#### 2.4.1.1 Module d'alimentation (PS)

Le module d'alimentation (en anglais power supply) assure la conversion de la tension du secteur (ou du réseau) en tension de (24V, 48V, 120V ou 230V) pour l'alimentation de l'automate et des capteurs et actionneurs en (24V, 48V, 120V ou 230V), un courant de sortie assigné de 2A, 5A et 10A.

- Module d'alimentation PS utilisée dans notre projet est PS 307-5A
- N° de référence : 6ES7307-1EA00-0AA0.
- L'alimentation externe 120/230v c.a. : 24c.c/5A
- Peut servir de tension d'alimentation des capteurs et actionneurs.

Les tensions pour les capteurs, les actionneurs et voyants lumineux qui dépassent 24V sont fournies par les blocs d'alimentation ou transformateurs supplémentaires.

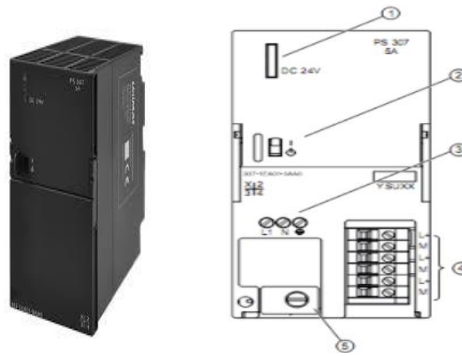


Figure 2.23: Schéma de raccordement PS 307-5. [9]

- (1). Signalisation de la présence d'une tension de sortie DC 24
- (2). Commutateur EN/HORS du 24 Vcc
- (3). Bornes pour la tension secteur et le conducteur de protection
- (4). Bornes pour la tension de sortie 24 Vcc
- (5). Arrêt de traction

#### 2.4.1.2 Module de CPU

La gamme S7-300 offre une grande variété de CPU tels que la CPU 314M, 315 ,315-2P, etc. Chacune possède certaines caractéristiques différentes des autres.

Le choix de la CPU 314-2PN/DP, pour un problème d'automatisation donné, est conditionné par les caractéristiques offertes par la CPU choisie. (Voir la figure 2.24).

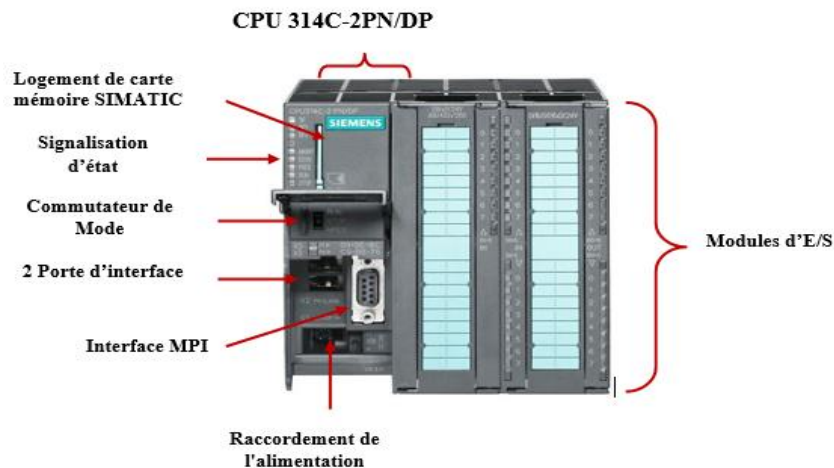


Figure 2.24: Le CPU 314-2 PN/DP de S7-300.

### ✚ Présentation de CPU choisie

-N° de référence : 6ES7 314-6EH04-0AB0

-Version : V3.3

-Description :Mémoire de travail de 192 ko ; 0,06 ms/kilo-instructions ; DI24/DO16 ; AI5/AO2 intégrées ; 4 sorties d'impulsions (2,5 kHz) ; 4 voies de comptage et de mesure avec codeurs incrémentaux 24 V (60 kHz) ; fonction de positionnement intégrée ; 2 ports ; MRP ; PROFINET CBA ; PROFINET Proxy CBA ; protocole de transport TCP/IP ; interface MPI/DP combiné (maître MPI ou DP ou bien esclave DP) ; configuration multirangée pouvant comporter jusqu'à 31 modules ; possibilité d'émission et de réception pour l'échange direct de données ; équidistance ; routage ; firmware V3.3.

#### 2.4.1.3 Le coupleurs (IM)

Les coupleurs IM360/361 ou IM365 permettent de réaliser des configurations à plusieurs châssis.

Généralement classer toujours le troisième dans le rack d'un SIMENS S7-300. Utilisé pour relier les différents racks.

#### 2.4.1.4 Modules de signaux (SM)

Ils servent d'interface entre le CPU et l'automate S7-300 SIMATIC. Ils existent des modules d'entrées TOR, des modules de sorties TOR ainsi que des modules d'entrées analogiques et des modules de sorties analogique. Les modules d'entrées/sorties sont des interfaces vers les capteurs et les actionneurs de la partie opérative. [9]

On a utilisé dans notre projet trois modules de signaux (SM) :

- DI 16x DC 24v : Module de 16 entrées TOR, 24 V, par groupes de 16, pas prévu pour une configuration avec modules de bus actifs.  
**-Adresses des entres** : E5.0 jusqu'un à E6.0.
- DO 16x DC 24v/0.5A : Module de 16 sorties TOR 24 V/0.5 A, par groupes de 8, pas prévu pour une configuration avec modules de bus actifs.  
**-Adresses des sorties** : A4.0 jusqu'un à A5.0.
- DI16/DO16x24V/0.5A : Module de 16 entrées et de 16 sorties TOR, 24 V/0.5 A.  
**-Adresses des entres** : E7.0 jusqu'un à E8.0.  
**-Adresses des sorties** : A7.0 jusqu'un à A8.0.
- AI4/AO2 : Module d'entrées /sorties analogiques AI4/12Bit+AO2/12Bit ;  
**-Adresses des entres** : E20.0 jusqu'un à E20.3.  
**-Adresses des sorties** : E20.0 jusqu'un à E20.3.



Figure 2.25: Modules Entrées/sorties d'un automate S7-300. [9]

#### 2.4.1.5 Modules de fonctions (FM)

Le modules de fonctions (FM) permet d'établir la tâche de traitement des signaux du CPU à un moment critique, et nécessitant une importante capacité de mémoire comme le comptage, positionnement et régulation. On peut citer les modules suivants : [2]

1. FM 354 et FM 357 : Module de commande d'axe pour servomoteur.
2. FM 353 : Module de positionnement pour moteur pas à pas.
3. FM 355 : Module de régulation.
4. FM 350 – 1 et FM 350 – 2 : Module de comptage.

### 2.4.1.6 Modules de communication (PC)

Les modules de communication permettent d'établir des liaisons homme-machine ; point à point, PROFIBUS, Industriel Ethernet.

Plusieurs automates S7-300 peuvent communiquer entre eux et avec d'autres automates SIMATIC S7 au moyen d'un câble-bus PROFIBUS. Le S7-300 est programmé à l'aide d'une console de programmation (PG). Cette PG est à relier à la CPU par un câble PG.

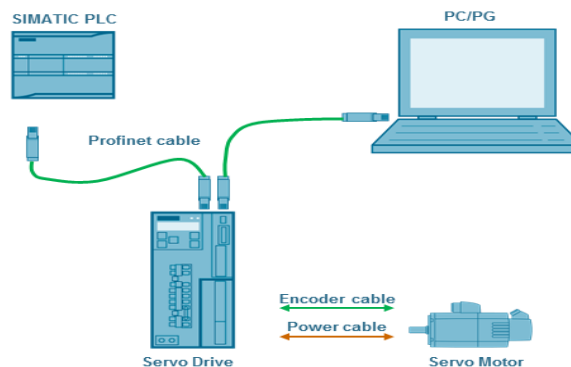


Figure 2.26: L'interface entre l'automate (API) et l'ordinateur (PC).

### 2.4.2 Caractéristique d'un automate S7-300

1. Mini-automate modulaire pour les applications d'entrée et de milieu de programme.
2. Gamme diversifiée de CPU.
3. Gamme complète de modules.
4. Possibilité d'extension jusqu'à 32 modules.
5. Bus de fond de panier intégré aux modules.
6. Possibilité de mise en réseau avec
  - l'interface multipoint (MPI)
  - PROFIBUS où
  - Industriel Ethernet
7. Raccordement central de la PG avec accès à tous les modules.
8. Liberté de montage aux différents emplacements.
9. Configuration et paramétrage à l'aide de l'outil « configuration matérielle ».

### 2.4.2 Interface (MPI)

L'interface d'une liaison MPI en anglais (Multi Point Interface), utilisé dans la programmation Steps 7 pour faire simulation, le contrôle-commande avec HMI (Interface Homme-Machine).

L'interface MPI est nécessaire pour programmer un SIMATIC S7-300 depuis le PC et l'échange de données entre CPU CIMATIC S7 jusqu'à 32 nœuds maximum. Chaque CPU du SIMATIC S7-300 est équipée d'une interface MPI intégré. [2]

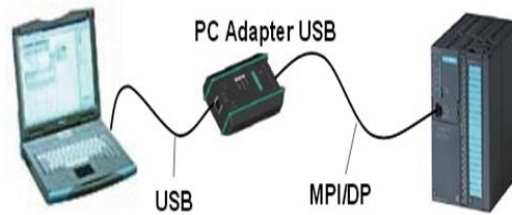


Figure 2.27: Présentation de l'interface (MPI).

## 2.5 Programmation via STEP 7

STEP7 est un logiciel de programmation qui fait partie de l'industrie SIMATIC de Siemens. Il permet de concevoir, configurer, programmer, tester, mettre en service et maintenir les systèmes d'automatisation SIMATIC.

Nom de Steps7 nommé pour la tâche de base travaillé par l'utilisateur, que l'on programme par ce logiciel sont :

- La création et la gestion de projet ;
- La configuration et le paramétrage du matériel et de la communication ;
- La gestion des mnémoniques ;
- La création des programmes ;
- Le chargement des programmes dans les systèmes cibles ;
- Le test de l'installation d'automatisation ;
- Le diagnostic lors des perturbations des installations [10].

### 2.5.1 Démarrage du logiciel STEP7


Après l'installation de logiciel Steps7 dans bureau d'ordinateur , on localise l'icône SIMATIC Manager sur l'écran de l'ordinateur puis avec un double clic sur cette icône, on se permet d'ouvrir sa fenêtre fonctionnelle (Voir la figure 2.28).



Figure 2.28: Assistant nouveau projet.

### 2.5.2 Création d'un projet STEP7

Le logiciel SIMATIC Manager étant maintenant ouvert, on clique sur l'item fichier puis assistant nouveau projet (clic sur suivant). Après la sélection du type de la CPU (pour notre projet, on a choisi une CPU 314-2 PN/DP) et l'insertion du bloc d'organisation OB1 et le langage à CONT, une fenêtre s'ouvre pour donner un nom de projet. Pour notre projet nommé (ELE M2) et on clique sur créer. Comme présenter dans a figure (2.29)

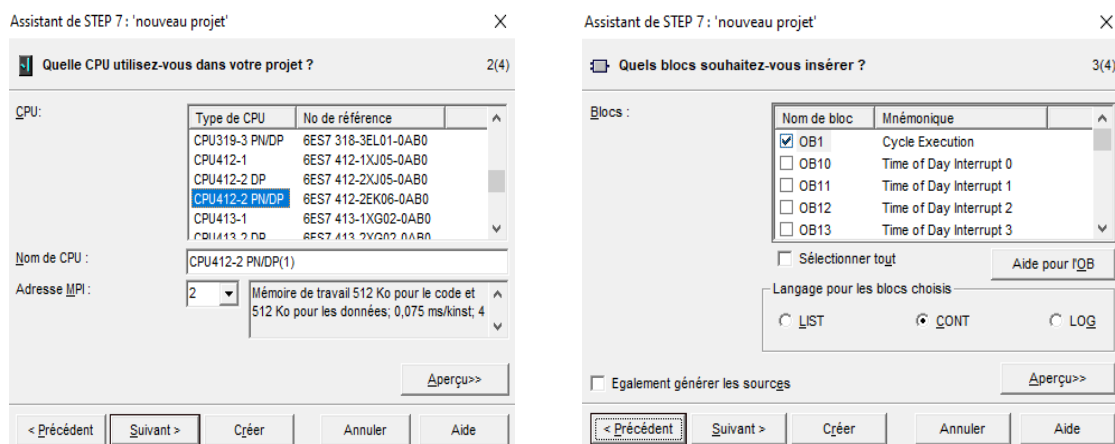


Figure 2.29: Choix de la CPU 412-2PN/DP et du bloc d'organisation OB1.

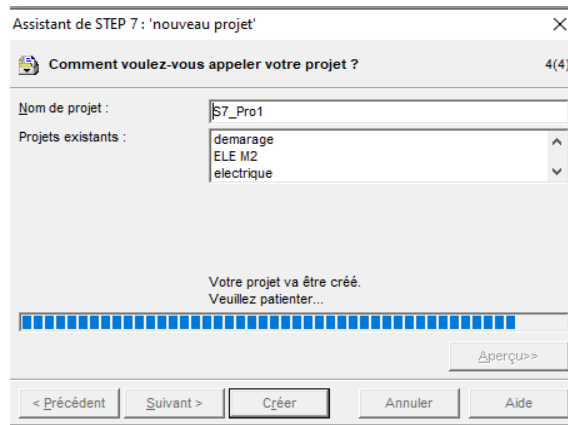


Figure 2.30: Nom et création du projet.

### 2.5.3 Configuration matérielle

Après avoir cliquer sur créés on va choisir la station SIMATIC 300, ensuite on va cliquer sur Matériel pour la configuration des modules nécessaire. Cette étape importante qui correspond à l'agencement des châssis et des modules , commence après le choix de RACK. (Voir la figure 2.31)

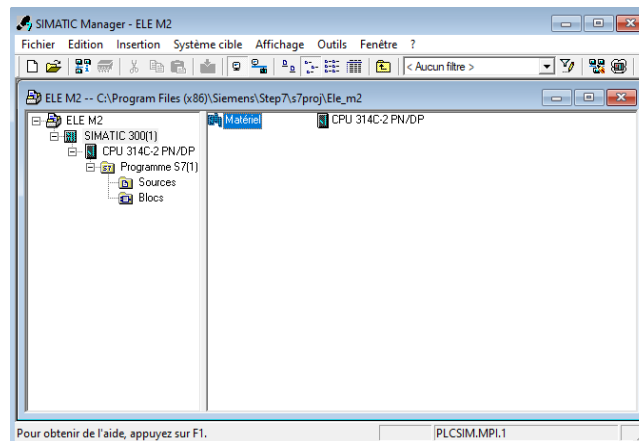


Figure 2.31 : Station SIMATIC S7-300.

Choix du matériels SIMATIC S7-300 avec une CPU 314-2 PN/DP, nous conduit à introduire la hiérarchie suivante :

- On commence par le choix du châssis selon la station choisie auparavant. Pour la station SIMATIC S7-300, on aura « RACK-300 » et on le glisse dans le châssis ;
- Ensuite on va choisir le module d'alimentation PS 307-5A et mettre dans l'emplacement N°1 ;
- Après avoir choisie le RACK, on lui glisse le CPU 314-2 PN/DP dans l'emplacement N°2 ;
- N°3 réservés comme adresse logique pour un coupleur dans une configuration multi-châssis (généralement pour relier plusieurs châssis).

- ✓ Les autres emplacements sont réservés pour les modules qui se trouvent dans le fichier SM-300. Dans notre projet nous avons choisi un entrer digital DI 16x DC 24v, et une sortie digital DO 16x DC 24v/0.5A.

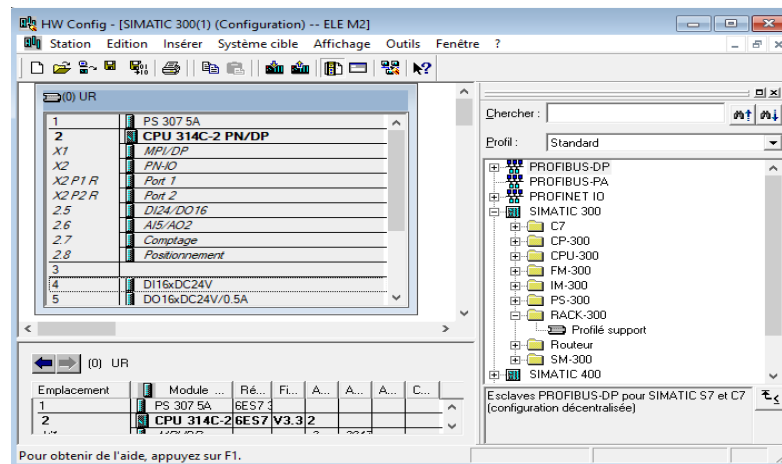


Figure 2.32: Configuration matériels.

## 2.5.4 Création du programme

Après la configuration du matériel, on revient à la fenêtre station SIMATIC S7-300, on clique sur programme ensuite on clique sur OB1, la fenêtre s'ouvre pour commencer le programme.

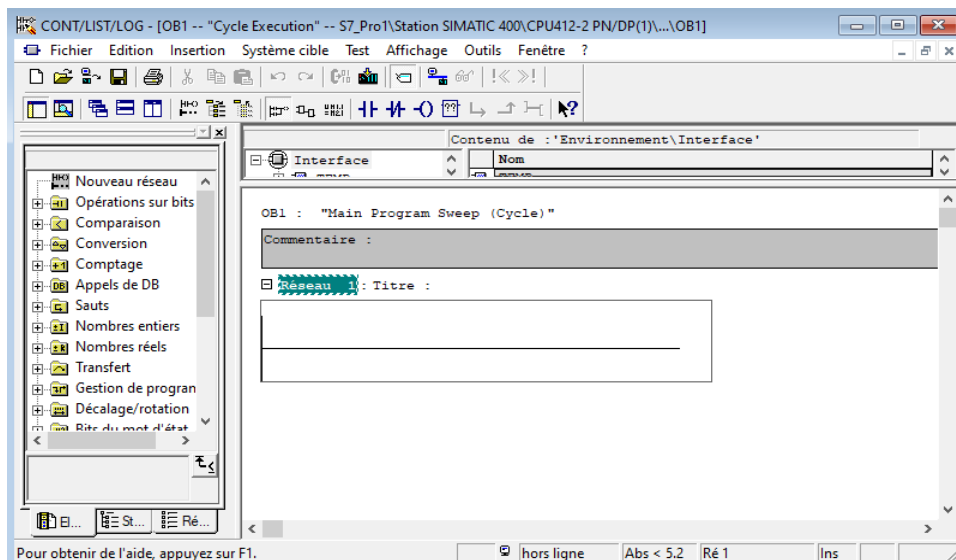


Figure 2.33: présentation la fenêtre de programmation.

Les blocs d'organisation (OB1) constituent l'interface entre le système OB d'exploitation et le programme d'utilisateur. L'ensemble du programme peut être concaténé dans un seul bloc OB1 (programme linéaire). Dans notre projet, le langage Ladder (CONT), est utilisé comme langage de programmation.

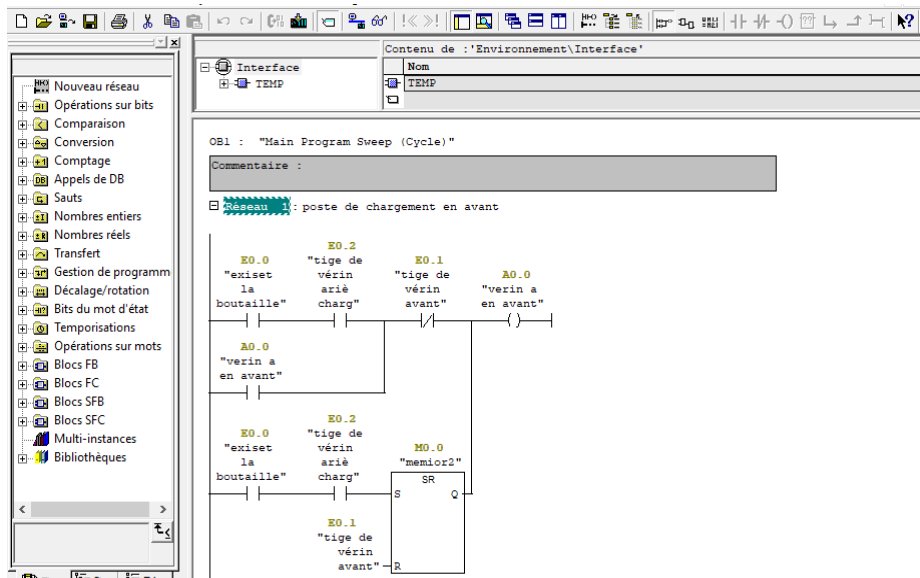


Figure 2.34: Exemple d'une partie de programme.

### 2.5.5 Structure du programme STEP7

La programmation structurée permet la rédaction claire et transparente de programmes. Elle permet la construction d'un programme complet à l'aide de modules qui peuvent être échangés et/ou modifiés à 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 [2].

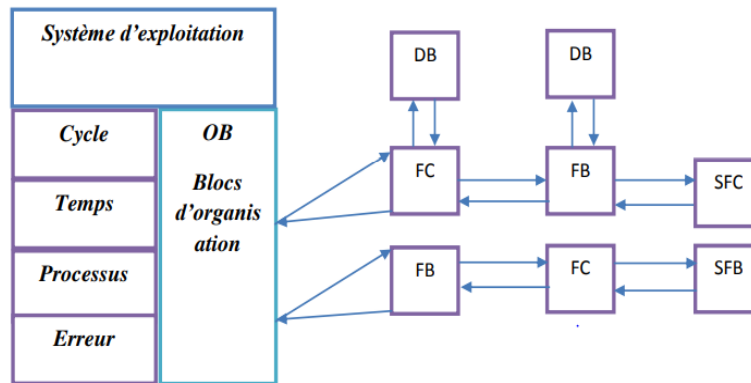


Figure 2.35: Présentation des Blocs de programmation Steps7. [2]

Le programme de commande de notre projet est élaboré en langage de programmation LADDER qui est le plus exploité en industrie et le plus utilisé par les électriciens. Dans notre travail nous allons utiliser deux blocs pour la partie programme et sont :

### ✚ Bloc d'organisation (OB)

Un bloc d'organisation est appelé cycliquement par le système d'exploitation et constitue donc, l'interface entre le programme utilisateur et le système d'exploitation. Nous allons utiliser dans la partie programme le bloc d'organisation L'OB, qui 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 [10].

Dans notre travail nous allons utiliser les blocs d'organisation suivant :

Bloc d'organisation OB	Description
OB1	Cycle libre
OB85	Erreur d'exécution du programme
OB121	Erreur du programme
OB122	Erreur d'accès aux E/S

Tableau 2.3: blocs d'organisations utilisées dans notre travail.

### ✚ Bloc Fonction (FC)

Une fonction FC ne possède pas une zone de mémoire propre. Les données locales d'une fonction sont perdues après l'exécution de la fonction. Il est également possible d'appeler d'autres FB et FC dans une fonction via des instructions d'appels de blocs [10].

**-Bloc fonction FC1 :** nous allons utiliser pour programmer la vanne d'eau dans le post de remplissage de notre travail, qui vous programmé par l'instruction mise à l'échelle FC105

SCALE convert.

#### **-Description de FC 105 :**

La fonction Mise à l'échelle (SCALE) généralement utiliser pour les entrer analogique, qui prend une valeur entière (IN) et la convertit selon l'équation ci-après en une valeur réelle exprimée en unités physiques, comprises entre une limite inférieure (LO\_LIM) et une limite supérieure (HI\_LIM).

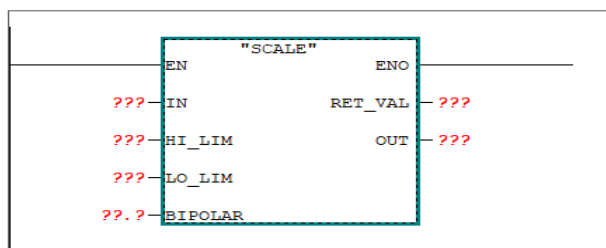


Figure 2.36: L'instruction mise à l'échelle FC105.

Dans notre projet nous allons commander le niveau du liquide (niveau de vanne 0 à 100%), qui déclenche une alarme lorsque le niveau atteint un niveau supérieur à 80% ou inférieur à 20%. (Voir la figure 2.37)

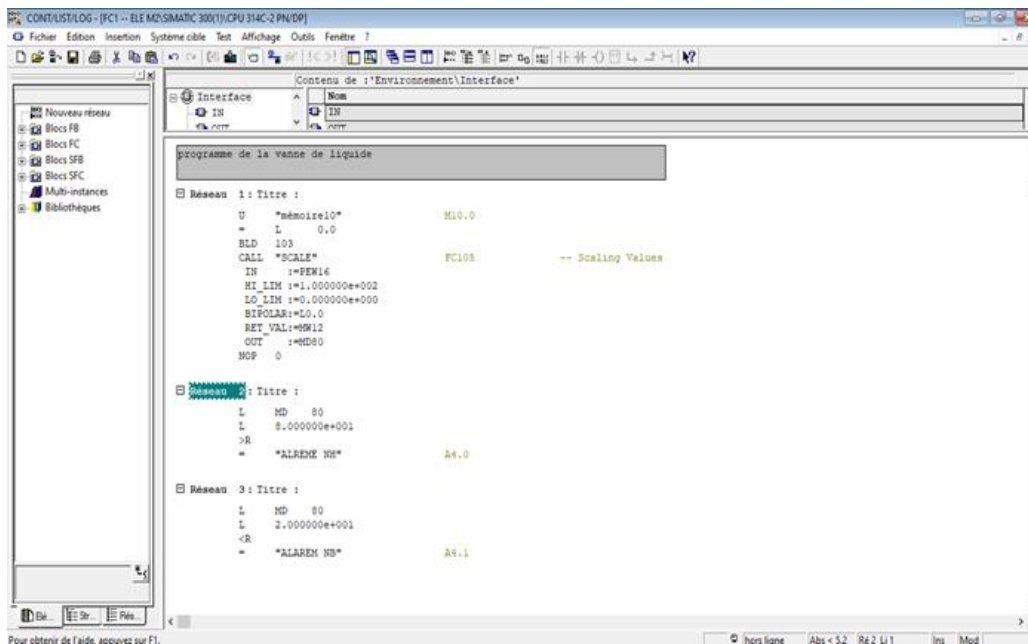


Figure 2.37: Programme de niveau d'eau dans la vanne.

**-Bloc fonction FC2 :** nous allons programmer le déplacement des bouteilles, qui est programmé par l'instruction de Transfert (L'instruction MOVE).

**-Description de l'instruction MOVE :**

Cette opération est activée par l'entrée de validation (EN). La valeur indiquée dans l'entrée (IN) est copiée à l'adresse précisée dans la sortie (OUT). L'état de signal de (ENO) est identifié par celui d'EN.

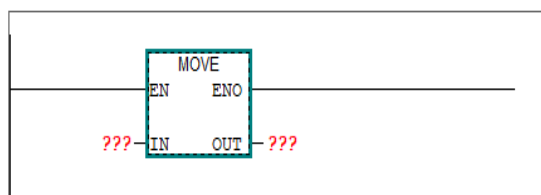


Figure 2.38: Présentation de l'instruction MOVE.

Dans la figure 2.39 nous allons présenter la partie de programme, de déplacement des bouteilles.

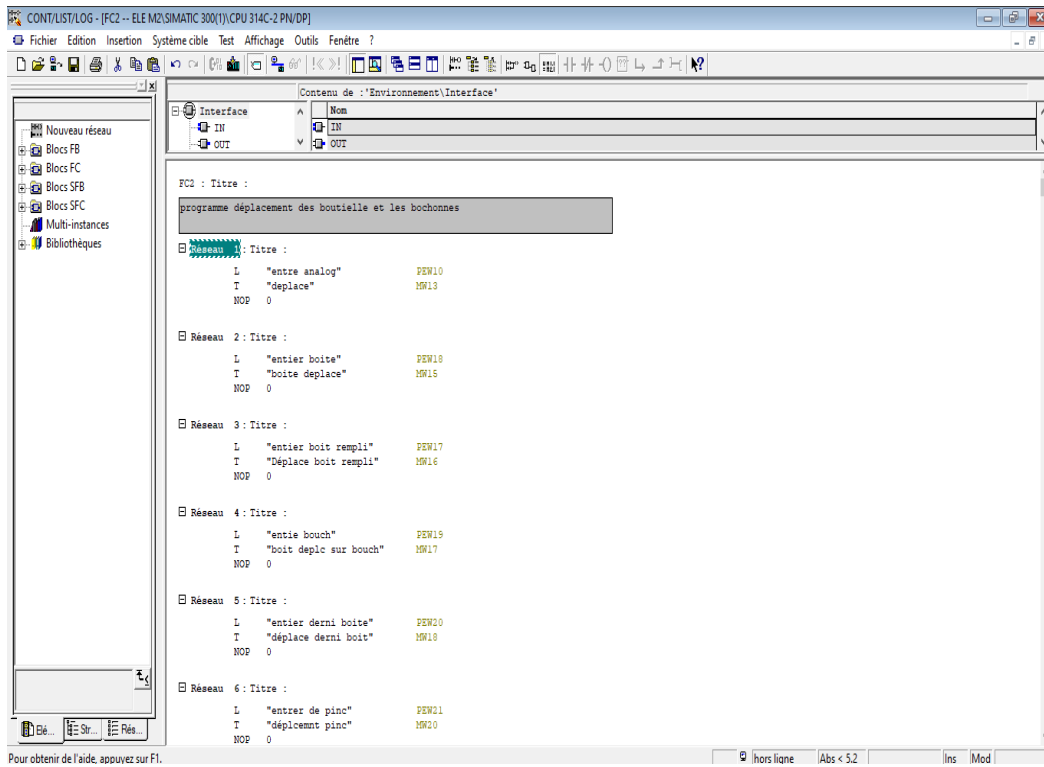


Figure 2.39: programme déplacement des bouteilles.

### 2.5.6 Table des mnémoniques

Editeur de mnémonique permet d'obtenir de toutes les variables globales dans notre projet. En effet il définit des désignations symboliques et des commentaires pour les signaux d'entrée-sortie les mémoires, les blocs de données etc.

The screenshot shows the 'Editeur de mnémoniques - Programme S7(1) (Mnémoniques)' window. The main window displays a table of mnemonics for 'Programme S7(1) (Mnémoniques) -- ELE M2/SIMATIC 300(1)/CPU 314C-2 PN/DP'. The table has the following columns: 'Etat', 'Mnémonique', 'Opérande', 'Type de do', and 'Commentaire'. The table contains 22 rows of data, each representing a mnemonic symbol, its address, and its data type. For example, row 1 has 'active moeteur de ts...' with address 'A 1.4' and type 'BOOL'. Row 2 has 'capteur d'exicet le b...' with address 'E 3.1' and type 'BOOL'. Row 3 has 'capteur d'exicte un b...' with address 'E 2.2' and type 'BOOL'. Row 4 has 'démarage le rempliss...' with address 'A 1.6' and type 'BOOL'. Row 5 has 'détaction de chut pre...' with address 'E 2.0' and type 'BOOL'. Row 6 has 'détaction de débimètre' with address 'E 1.0' and type 'BOOL'. Row 7 has 'esxicte un bouteille en' with address 'E 2.5' and type 'BOOL'. Row 8 has 'etat de remplissage' with address 'M 100.6' and type 'BOOL'. Row 9 has 'exicete bouteille vide' with address 'E 2.1' and type 'BOOL'. Row 10 has 'exicite bouteille en 3' with address 'E 1.1' and type 'BOOL'. Row 11 has 'exicite bpuchon en g' with address 'E 1.7' and type 'BOOL'. Row 12 has 'exicite la boutaitte en p' with address 'E 0.6' and type 'BOOL'. Row 13 has 'exicite un bouchon' with address 'E 1.2' and type 'BOOL'. Row 14 has 'exisete la bouteille' with address 'E 0.0' and type 'BOOL'. Row 15 has 'exisete le bouteille vid' with address 'E 0.4' and type 'BOOL'. Row 16 has 'fin de bouchage' with address 'M 11.0' and type 'BOOL'. Row 17 has 'fin de boucher' with address 'M 5.0' and type 'BOOL'. Row 18 has 'fin de remle' with address 'M 4.1' and type 'BOOL'. Row 19 has 'fin de rempli' with address 'A 2.0' and type 'BOOL'. Row 20 has 'fin de remple' with address 'M 4.0' and type 'BOOL'. Row 21 has 'fin de remplissage' with address 'M 100.5' and type 'BOOL'. Row 22 has 'fin de remplissage' with address 'M 10.0' and type 'BOOL'. The status bar at the bottom indicates 'Pour obtenir de l'aide, appuyez sur F1.' and 'NUM'.

Etat	Mnémonique	Opérande	Type de do	Commentaire
1	active moeteur de ts...	A 1.4	BOOL	
2	capteur d'exicet le b...	E 3.1	BOOL	
3	capteur d'exicte un b...	E 2.2	BOOL	
4	démarage le rempliss...	A 1.6	BOOL	
5	détaction de chut pre...	E 2.0	BOOL	
6	détaction de débimètre	E 1.0	BOOL	
7	esxicte un bouteille en	E 2.5	BOOL	
8	etat de remplissage	M 100.6	BOOL	
9	exicete bouteille vide	E 2.1	BOOL	
10	exicite bouteille en 3	E 1.1	BOOL	
11	exicite bpuchon en g	E 1.7	BOOL	
12	exicite la boutaitte en p	E 0.6	BOOL	
13	exicite un bouchon	E 1.2	BOOL	
14	exisete la bouteille	E 0.0	BOOL	
15	exisete le bouteille vid	E 0.4	BOOL	
16	fin de bouchage	M 11.0	BOOL	
17	fin de boucher	M 5.0	BOOL	
18	fin de remle	M 4.1	BOOL	
19	fin de rempli	A 2.0	BOOL	
20	fin de remple	M 4.0	BOOL	
21	fin de remplissage	M 100.5	BOOL	
22	fin de remplissage	M 10.0	BOOL	

Figure 2.40: La partie de table des mnémoniques du projet.

### 2.5.7 Présentation du PLCSIM

STEP7 pro contient L'application S7-PLCSIM de simulation permet d'exécuter, de tester et simuler notre programme dans l'automate programmable industriel (API). La simulation étant complètement réalisée au sien du logiciel Step7. Il est nécessaire qu'une liaison d'interface PG/PC et l'automate programmable pour activer la simulation.

S7-PLCSIM dispose d'une interface simple qui permet de visualiser et de forcer les différents paramètres utilisés par le programme (comme, par exemple, d'activer ou de désactiver des entrées ou de sortie).

Sélectionnez la commande de menu Démarrer > SIMATIC > STEP 7 > Paramétrage de l'interface PG/PC. Résultat : la boîte de dialogue de paramétrage de l'interface PG/PC s'affiche.

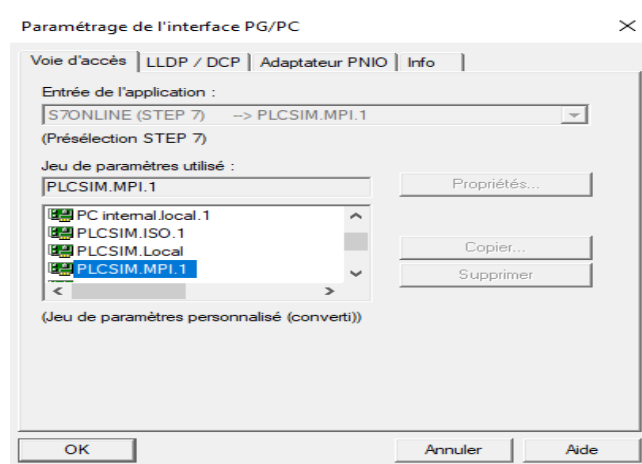



Figure 2.41: Paramétrage de l'interface PG/PC.

Dans notre projet nous avons utilisé l'interface de liaison MPI (Multi Point Interface), qui relie l'API avec le PC.

- En cliquant sur le bouton  qui se trouve dans la barre d'outils de se gestionnaire, comme le montre la figure suivante :

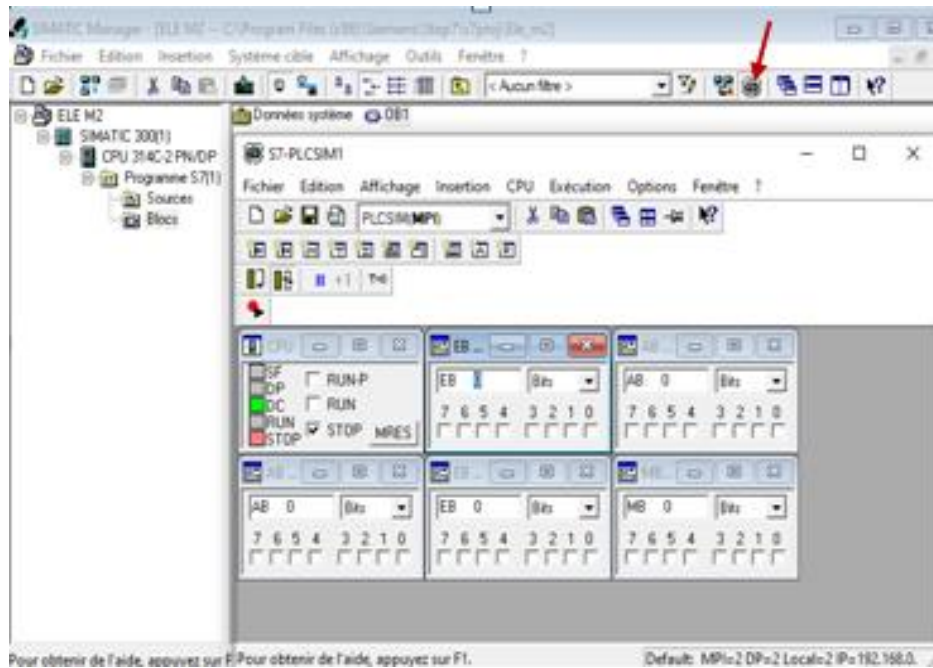


Figure 2.42 : Présentation de simulateur S7-PLCSIM.

A partir de la fenêtre du simulation PLCISIM, on peut commander les entres, sorties, et l'autre insertion utiliser dans notre projet et l'état de CPU.

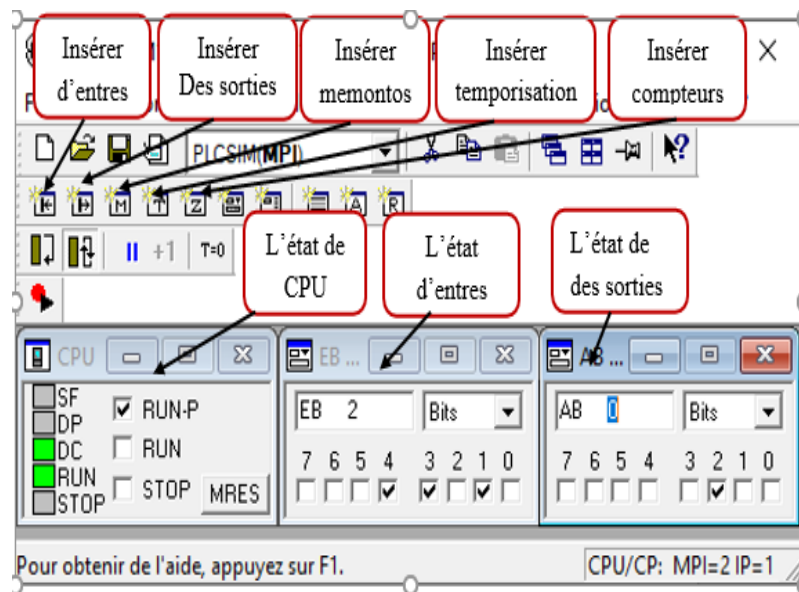


Figure 2.43: Présentation les éléments de S7 PLCSIM.

- **Types de liaison**

Dans les versions précédentes de S7-PLCSIM, vous ne pouviez pas simuler un automate programmable que via une liaison MPI. Avec S7-PLCSIM, vous pouvez utiliser chacun des paramétrages d'interface suivants pour établir une liaison.

- PLCSIM (ISO)
- PLCSIM (local)
- PLCSIM (MPI)
- PLCSIM (PROFIBUS)
- PLCSIM (TCP/IP)

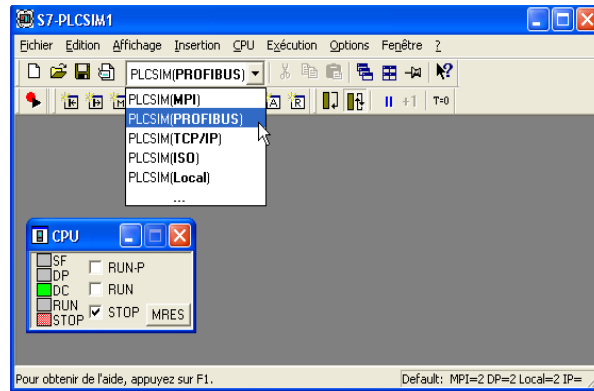


Figure 2.44: Présentation le type de la liaison S7 PLCISIM.

## 2.6 Conclusion

Ce chapitre est constitué de deux partie distinct et complémentaire d'une partie matérielle, utilisé dans la partie opérative d'un (API), Dans la seconde partis, notre travail détaillé l'utilisation de l'automate modulaire S7-300 de SIMENS ainsi sur le logiciel de programmation STEP7, qui représente la partie software.

### 3.1 Introduction

Afin d'analyser et valider le cahier de charge de notre système nous allons détailler le GRAFCET du système étudié, pour bien comprendre le fonctionnement des étapes de la chaîne étudiée.

Pour piloter l'automatisation du poste de remplissage et bouchonnage nous allons réaliser un programme à implanter dans l'automate S7-300 grâce au logiciel d'automatisation SIMATIC STEP7. Nous allons décrire le programme d'automatisation d'un notre système ensuite on a procédé à sa simulation grâce à l'application de simulation S7-PLCSIM. Ensuite nous allons exécuter et visualiser le programme.

Une autre partie de ce chapitre est consacrée au contrôle-commande d'un système de supervision avec le logiciel WinCC flexible pour visualiser l'état de fonctionnement de la chaîne de production pilote afin de surveiller et de détecter les anomalies qui peuvent survenir au cours du fonctionnement du procédé.

### 3.2 Cahier de charge

Etude d'une chaîne de poste de remplissage et bouchonnage des bouteilles d'eau.

#### 3.2.1 Principe de fonctionnement

Le système consiste à transférer des bouteilles vers le premier poste de chargement, à provoquer le déplacement de ces dernières vers le deuxième poste, pour le remplissage à base d'une pompe mu électrovane, après la fin du remplissage la bouteille se déplace vers le troisième poste où elle sera bouchonnée à base des vérins pneumatiques.

#### 3.2.2 Choix de technologie

##### -Circuit de puissance

Un circuit de puissance est un circuit pneumatique composé d'un nombre de vérin à double effet, on a choisi cette technologie à cause du prix des vérins, fiabilité, moins de panne...etc.

##### -Circuit de commande

Un circuit de commande est composé d'un nombre de relais électromagnétique piloter à travers un automate.

Les actionneurs sont équipés de capteurs de position de type magnétique qui permet de transférer la position exacte d'un actionneur vers les entrées de l'automate.

Les préactionneurs sont des distributeurs 5/2 équipés d'une électrovane excités par les relais électromagnétique.

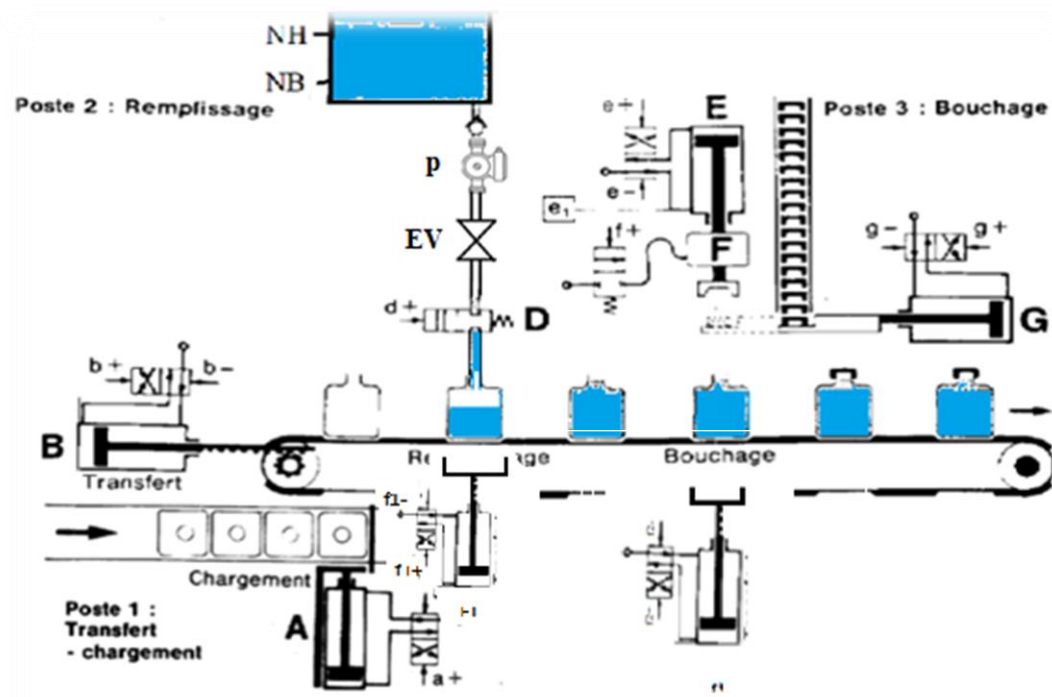


Figure 3.17: Présentation de poste remplissage et bouchonnage.

### 3.3 Organigramme du programme de notre étude

#### 3.3.1 Organigramme du programme du poste de chargement (poste 1)

Dans notre travail nous allons commencer un départ de cycle par exciter une bouteille sur le poste 1. (Voir la figure 3.2)

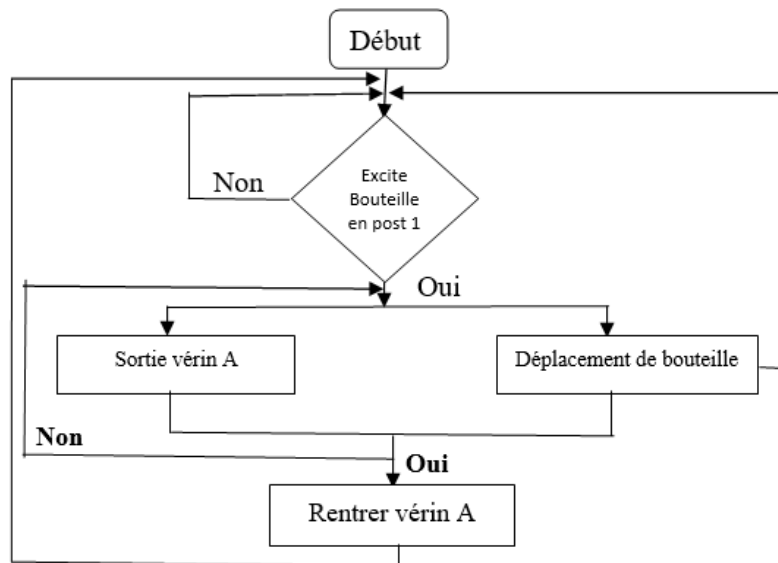


Figure 3.18: Organigramme du programme pour le post 1.

### 3.3.2 Organigramme du programme pour le post 2

Le deuxième poste est démarrer lorsque transfert de la bouteille vide sur le remplissage et fixé par le vérin F1. (Voir la figure 3.3)

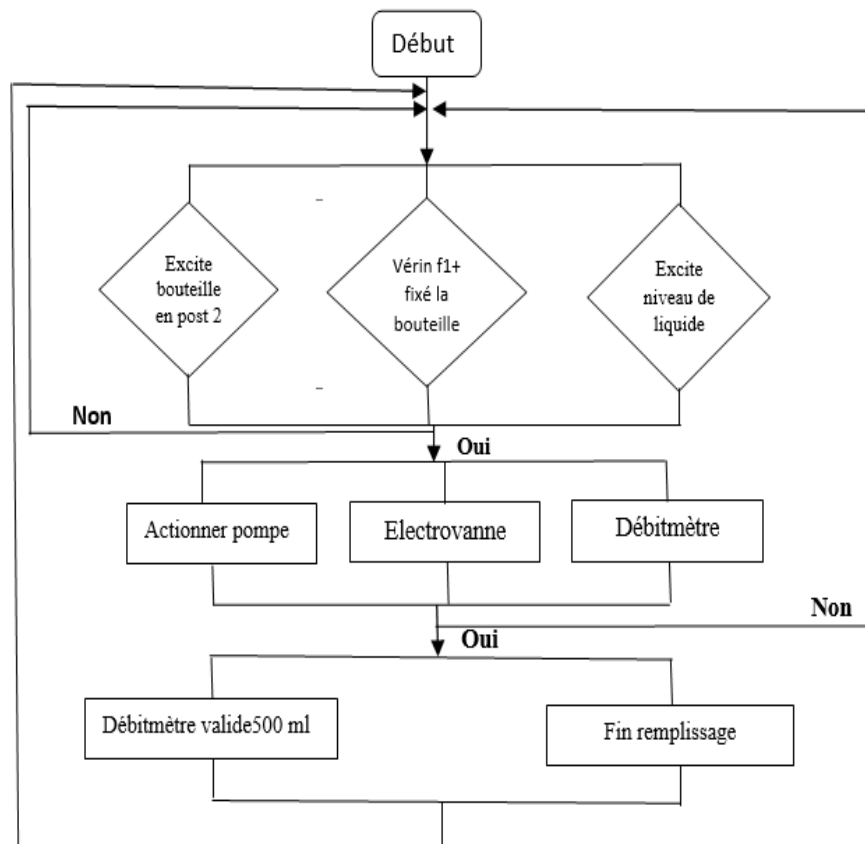


Figure 3.19: Organigramme du programme poste de remplissage.

### 3.3.3 Organigramme du programme de post 3

Dans le dernier post nous allons programmer trois vérins et un moteur pneumatique pour faire le bouchage des bouteilles. (Voir la figure 3.4)

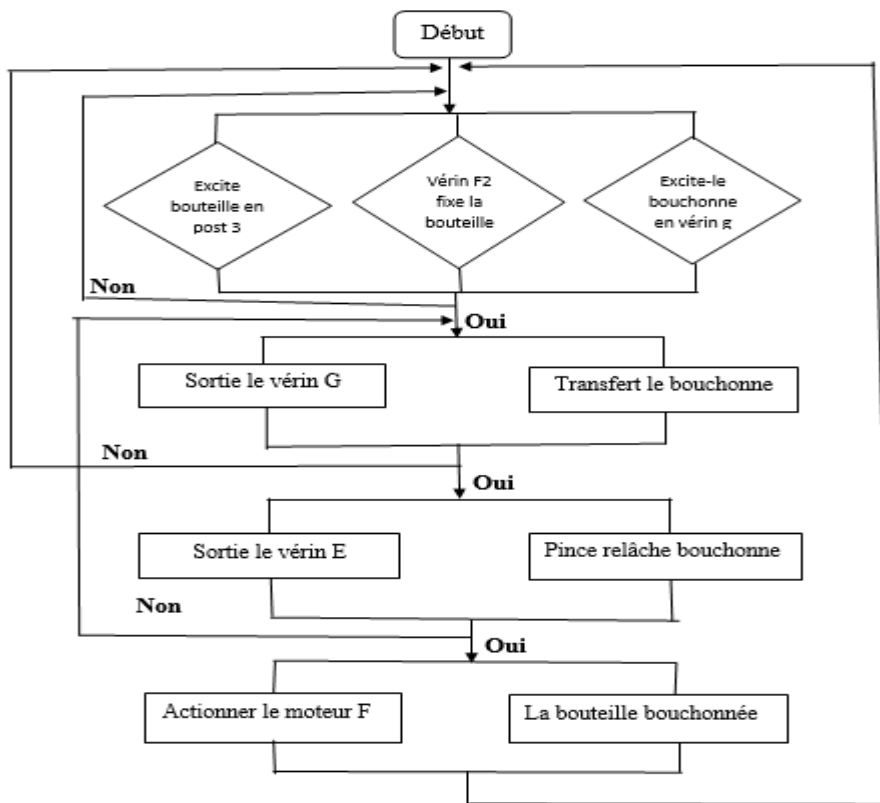


Figure 3.20 : Organigramme de programme pour le post 3.

### 3.3.4 Organigramme du programme du tapis roulant

Le déplacement de tapis pour transférer les bouteilles avec les deux postes de remplissage et bouchonnage, nous allons présider de méthode, par un vérin ou un moteur pneumatique.

- **Méthode 1 :**

Déplacement du tapis par un vérin, qui l'on existé une bouteille vide sur cette tapis. Comme montrer la figure III.5.

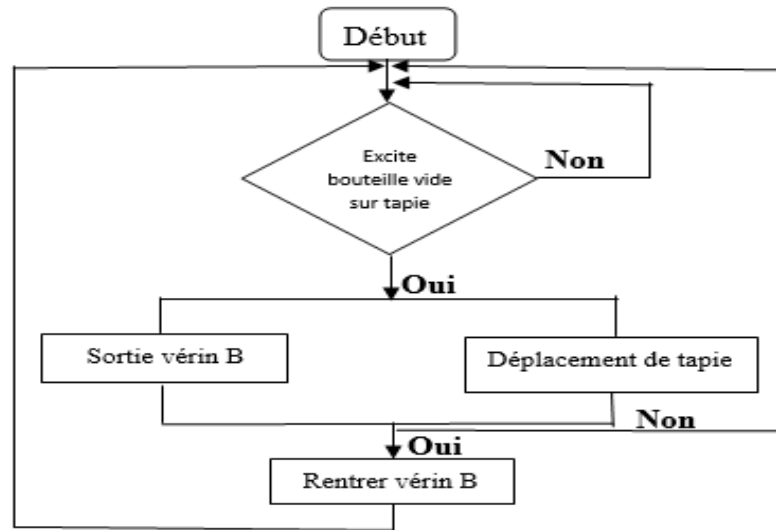


Figure 3.21: Organigramme de programme pour déplacement des bouteilles.

- **Méthode 2 :**

Dans cette méthode nous allons utiliser un moteur pas à pas pour déplacer le tapis roulant. (Voir la figure 3.6)

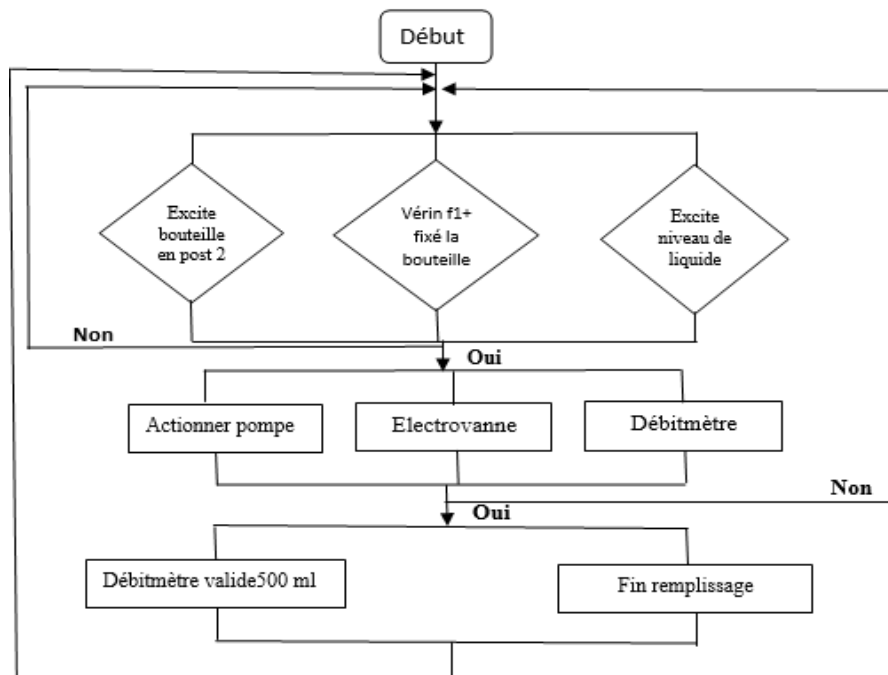


Figure 3.22: Organigramme du programme pour le tapis par un moteur pas à pas.

### 3.4 Présentation de GRAFCET d'un système

Les graphes de fonction séquentielle représentent une technique graphique puissante pour décrire le comportement séquentiel d'un programme. Les langages graphiques sont utilisés depuis plusieurs années, notamment le GRAFCET.

#### 3.4.1 GRAFCET niveau 01

C'est une description en termes de fonction des comportements que doit avoir la partie commande face aux informations provenant de la partie opérative. (Voir la figure 3.7)

Cette partie de grafect explique généralement le fonctionnement d'un système, qui permet démonstration les déférentes étapes réaliser dans les trois postes de la chine de production.

**Etape 01 :** Existence d'une bouteille dans le poste de chargement.

**Etape 02 :** Sortie du vérin A de chargement pour le transfert de la bouteille sur le tapis.

**Etape 03 :** Sortie du vérin B pour déplacer le tapis et le transfert de la bouteille vide pour le remplissage.

**Etape 04 :** Activer la pompe, l'électrovanne et le débitmètre qui remplit la bouteille.

**Etape 05 :** Déplacement la bouteille vers le poste 3 de bouchonnage.

**Etape 06 :** Sortie du vérin G pour le transfert du bouchon de la bouteille.

**Etape 07 :** Sortie du vérin E pour trouver le bouchon de la bouteille.

**Etape 08 :** Activer le moteur pneumatique pour bouchonner la bouteille.

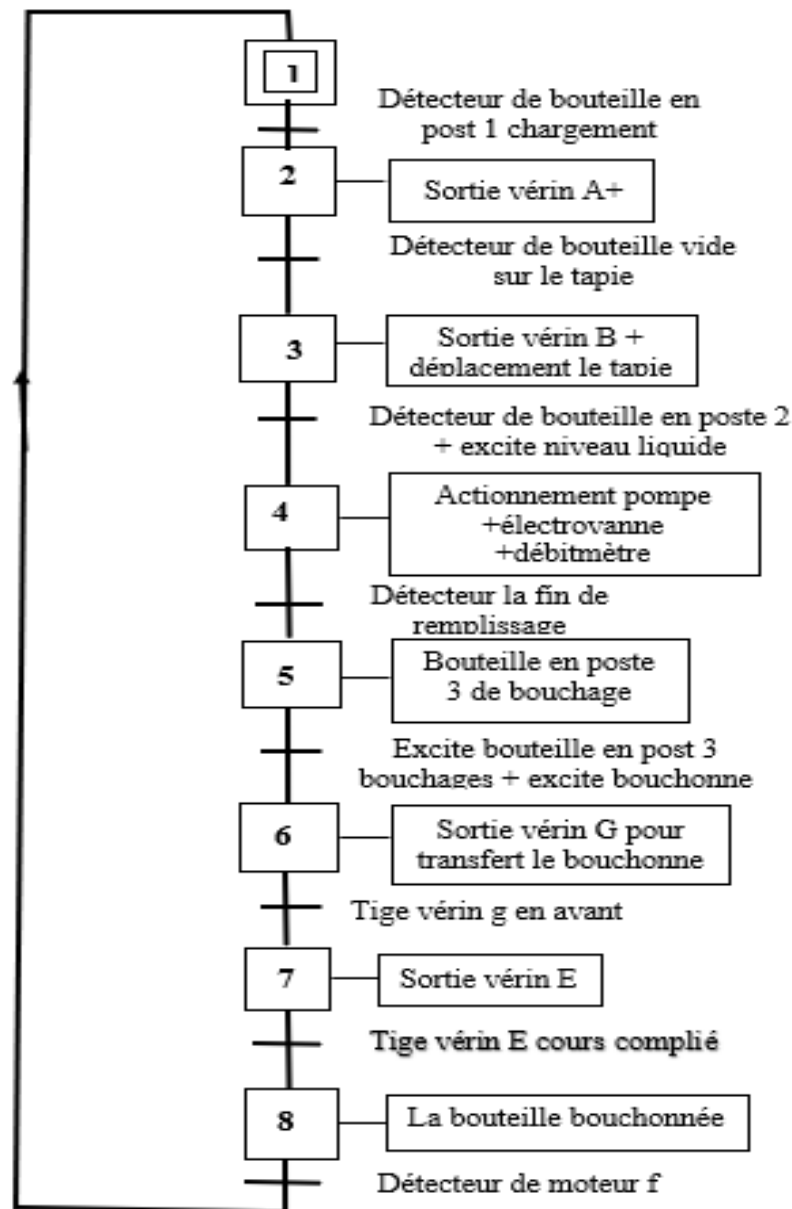


Figure 3.23: Présentation du GRAFCET niveau 01.

### 3.4.1.1 Liste d'actionneurs, capteurs d'un système :

#### 3.4.1.1.1 Actionneurs (Les sorties)

- **Vérins**

**-vérin A : Pour le post de chargement**

A0.0 : "sortie vérin A de chargement A+"

A0.1 : "rentrer vérin A de chargement A-"

**-vérin B : Pour le déplacement de tapis**

A0.2 : "sortie vérin b de tapis B+"

A0.4 : "rentrer vérin b de tapis B-"

**-vérin G : pour le poste de bouchonnage qui transfert le bouchon**

A0.7 : "sortie vérin G+"

A4.2 : "vérin g retour G-"

**-vérin E : Pour le poste de bouchonnage qui bouche la bouteille**

A0.1 : "sortie vérin E+"

A7.0 : "rentrer vérin E-"

**-vérin de fixation bouteille en post de remplissage**

A1.4 : "sortie vérin fixé post 2"

A1.5 : "rentrer vérin fixé post 2"

**-vérin de fixation bouteille en poste de bouchonnage**

A1.6 : "sortie vérin fixe post 3"

A1.7 : "rentrer vérin fixe post3"

- **Convoyeur**

A0.3 : "tapis roulant déplacer"

- **Pompe**

A4.5 : "pompe pneumatique d'injecte P"

- **Electrovanne**

A5.0 : "actionner l'électrovanne EV"

- **Débitmètre**

A0.6 : "débitmètre attache D"

- **Moteur pneumatique**

A1.3 : "moteur F attache"

A1.2 : "bouteille bouchonné"

### 3.4.1.1.2 Capteurs (Les entres)

- **Capteurs de position**

E2.0 : " Excité la bouteille versle poste de chargement CP1"

E2.1 : "Tige de vérin A0 en Arrée"

E2.2 : "Tige vérin A1 en avant"

E2.3 : "Existé la bouteille vide CP"

E2.4 : "tige vérin b en aérée b0"

E2.5 : "tige vérin b en avant b1"

E5.7 : "tige vérin fixe Arrée post 2"

E2.6 : " Existé la bouteille CP2"

E5.4 : "tige vérin fixé avant p2"

E3.1 : " Existé la bouteille post PC3"

E5.5 : "tige vérin fixe Arrée p3"

E5.6 : "tige vérin fixe avant p3"

E3.2 : " Existé le bochonne en vérin g"

E3.4 : "tige de vérin g0 Arrée"

E3.5 : "tige de vérin g1 avant"

E3.6 : " Existé bochonne en vérin g"

E3.7 : "tige de vérin e0 en Arrée"

E5.0 : "tige vérin E demi cours"

E5.1 : "bouchon dans le pince"

E7.1 : "tige vérin E cours complet"

E5.3 : "protection thermique"

E7.2 : "détecteur de serrage"

E7.3 : "détecteur moteur attache"

E7.4 : "Décteur de bouteille bouchonnée"

- **Capteur se chute pression qui relâche le moteur**

E7.5 : "détecteur de chute pression "

- **Capteur de niveau**

E2.7 : "détecteur de niveau"

- **Capteurs analogiques**

PEW 16 : capteur analogique de niveau de vanne d'eau de la chaine.

PEW 10 : Capteur de déplacement de bouteille vers le poste de chargement.

PEW 18 : Capteur de déplacement de bouteille vide pour le remplissage.

PEW 17 : Capteur de déplacement de bouteille remplie.

PEW 19 : Capteur de déplacement de bouteille sur le poste bouchonnage.

PEW 21 : Capteur de déplacement de bouchon par le vérin g.

PEW 22 : Capteur de déplacement de bouchon par le vérin E pour bouchonner la bouteille.

### 3.4.2 GRAFCET niveau 02

Le GRAFCET de niveau 2 appelé aussi niveau de la partie opérative, il tient compte de plus de détails des actionneurs des pré-actionneurs et des capteurs de notre projet. (Voir la figure 3.8)

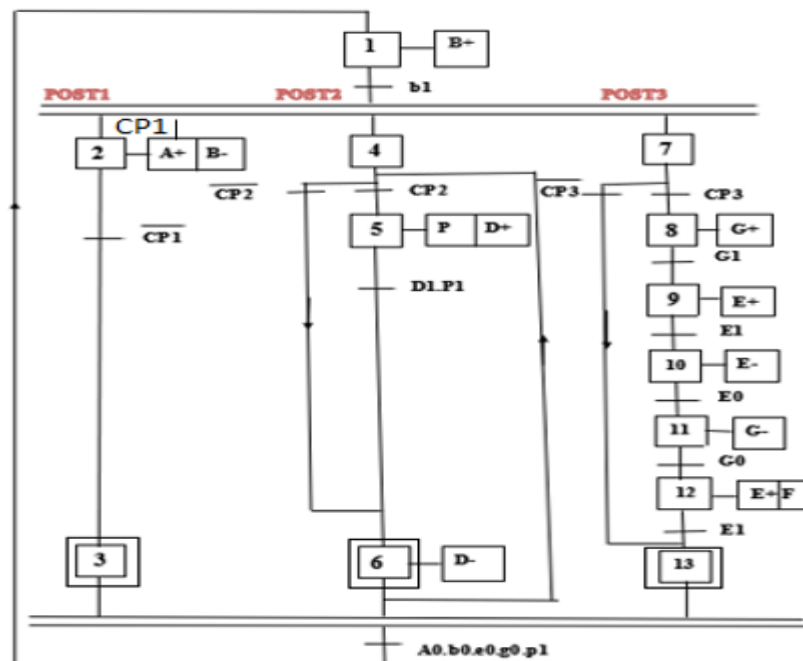




Figure 3.24: Présentation de GRAFCTE de notre projet.

### 3.5 Simulation du programme avec S7-PLCSIM

#### 3.5.1 Chargement du programme

##### ✚ Condition

- La simulation a été démarrée depuis le SIMATIC Manager de STEP 7.
- Le type de liaison approprié est configuré.
- L'adresse dans STEP 7 coïncide avec celle dans S7-PLCSIM ou le système cible est à l'état initial.

Après l'ouverture de la fenêtre de simulation S7 PLCISIM , nous allons charger tous les blocs de programme réalisés dans l'automate (API) en cliquant sur le , Ou en cliquant sur le bouton droit sur Blocks et choisir système cible, charger. Comme représenté dans la figure (3.9).

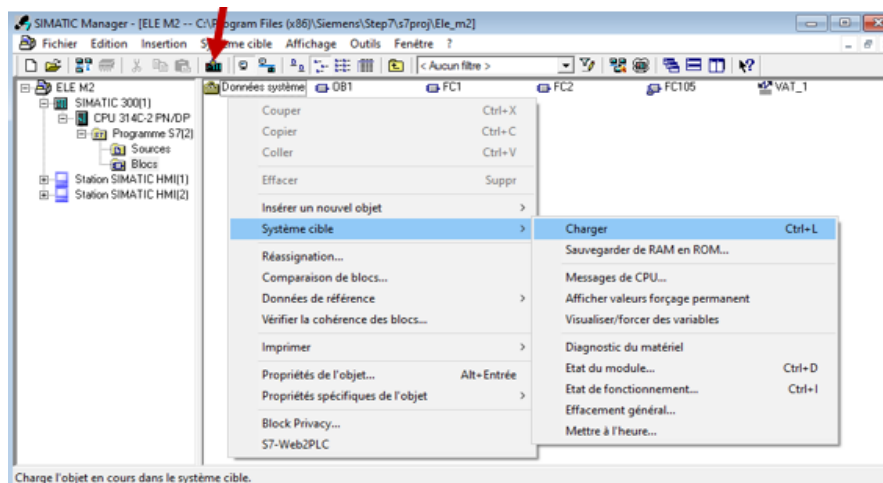



Figure 3.25 : Chargement de programme dans l'API de simulation.

#### 3.5.2 Exécution et visualisation du programme

Pour visualiser le programme chargé dans l'automate programmable S7 PLCISIM, nous allons procéder par deux manières

- **Sur S7 PLCISIM**

Après le chargement tous les blocs dans l'automate, nous allons cliquer directement sur la commande TEST-VISUALISATION , ensuite nous allons cocher la case **RUN-P** de la CPU,

Pour démarrer la simulation, on va choisir les bits correspondants dans les fenêtres des variables (entres, sorties, temporisation...etc.), Comme représenté dans la figure suivante.

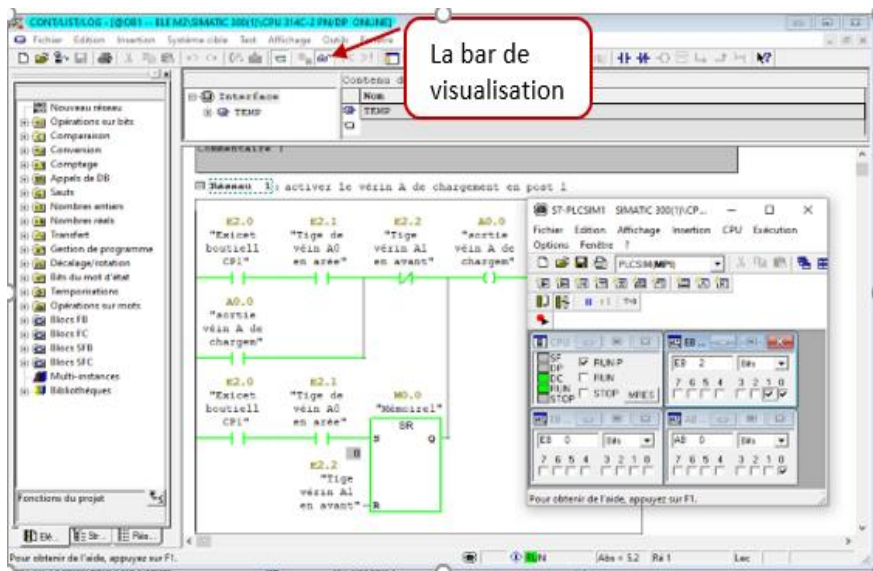


Figure 3.26: Simulation et visualisation du programme sur S7 PLCISIM.

- **Sur la table de variable**

Après l’ouverture de la fenêtre de simulation S7 PLCISIM, nous allons cliquer sur le bouton droit puis sur BLOKS et choisir Insérer un nouvel objet\_ Table de variable pour effectuer la simulation du programme. On va commander (entres, sorties, memontos...etc.) par true ou false (0 ou 1) (Voir la figure 3.11).

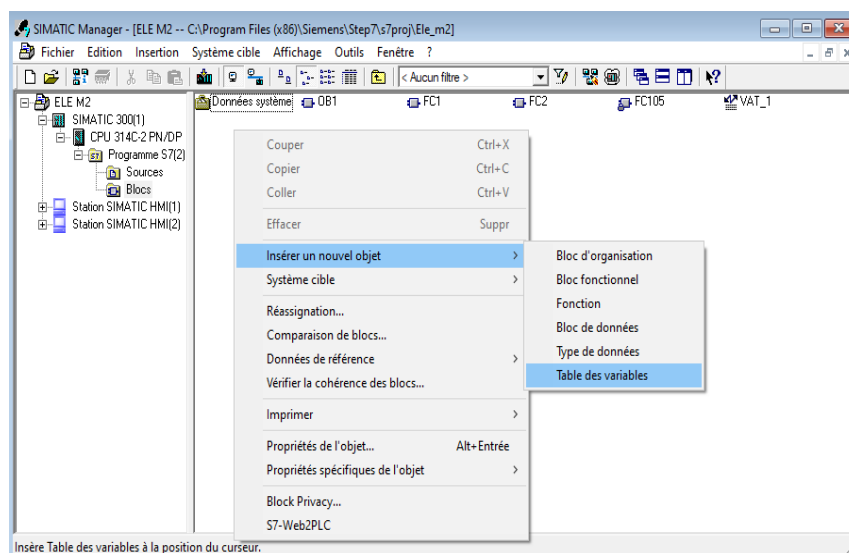



Figure 3.27: Présentation de Fenêtre de la table de variable.

Pour effectuer la simulation sur la table de variable nous allons cliquer deux fois sur la bar de . Comme présenter dans la figure suivant.

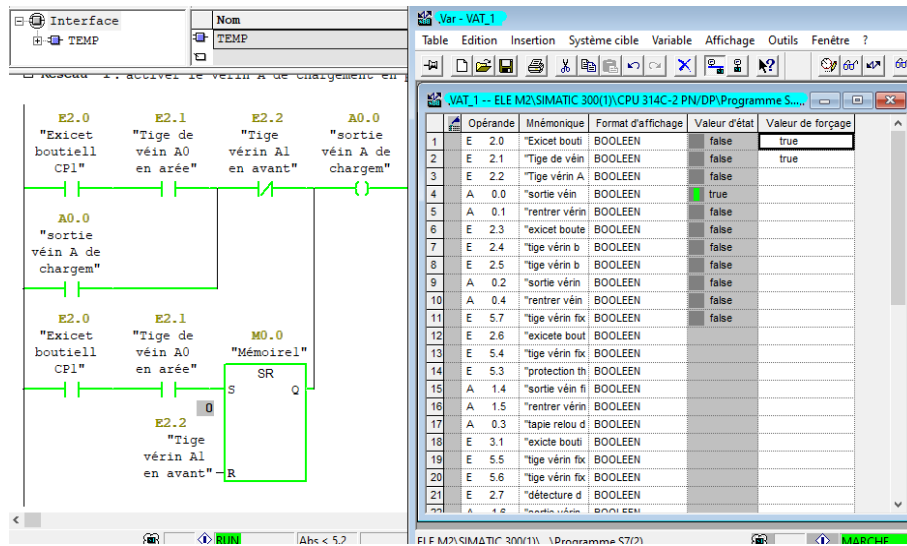



Figure 3.28: Simulation du programme par la TABLE DE VARIABLE.

### 3.5.1 Simulation et visualisation du programme de poste de chargement (Post 1)

Après le chargement du programme et l'activation de la simulation, nous allons visualiser l'état de chargement des bouteilles sur le post 1, par l'action de vérin A, grâce à l'icône  qui se trouve dans la barre d'outils de bloc d'organisation OB1. Dans la figure 3.14, on voit que la sortie vérin A (A0.0) et déplacement de bouteille (A14.0).

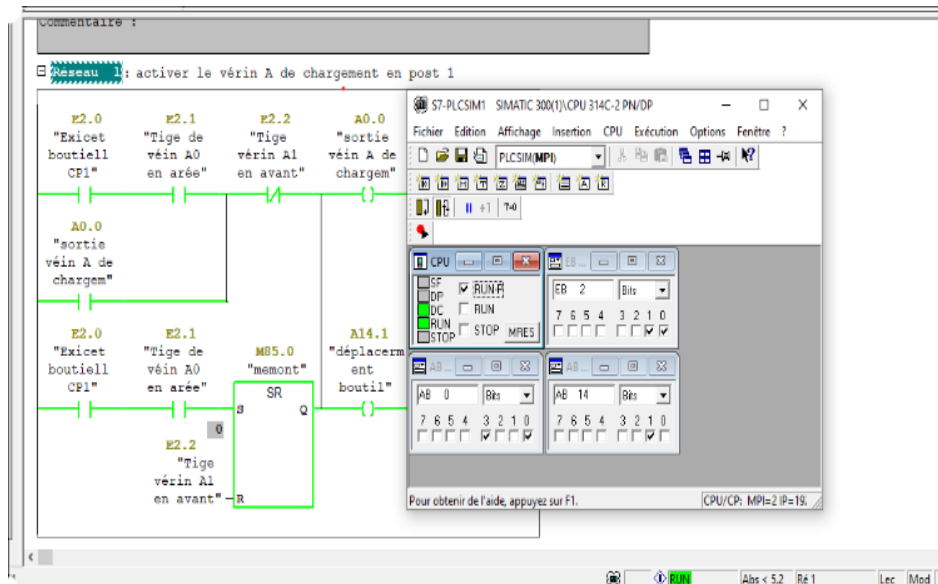



Figure 3.29: Simulation et visualisation de poste de chargement des bouteilles.

- **Visualisation de la vanne de liquide**

Dans la fonction FC1, on trouve l'icône  qui nous permet de visualiser l'activation et l'état (ouverture en %) de niveau de la vanne, comme présenter dans la figure 3.14.

Nous allons commander dans notre travail une alarme « niveau haut A4.0 » qui se declanche quand le niveau de la vanne est supérieur à 80%, et qui declanche une autre alarme « niveau bas A4.1 » quand le niveau de la vanne inférieur à 20%.

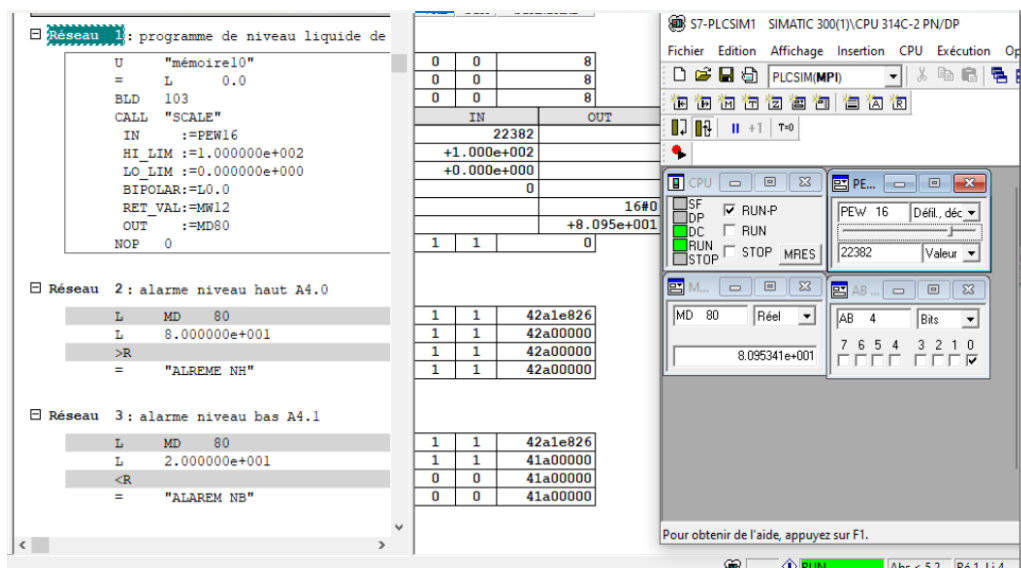


Figure 3.30: Simulation et visualisation de niveau de la vanne.

### 3.5.2 Simulation et visualisation du programme de poste de remplissage (Poste 2)

Dans le bloc principale OB1, nous allons simuler et visualiser l'état de remplissage des bouteilles par les trois actionneurs (pompe, électrovanne et débitmètre) des bases du poste 2. Dans la figure 3.15, on voit que les trois actionneurs sont actifs après la validation des conditions excitées.

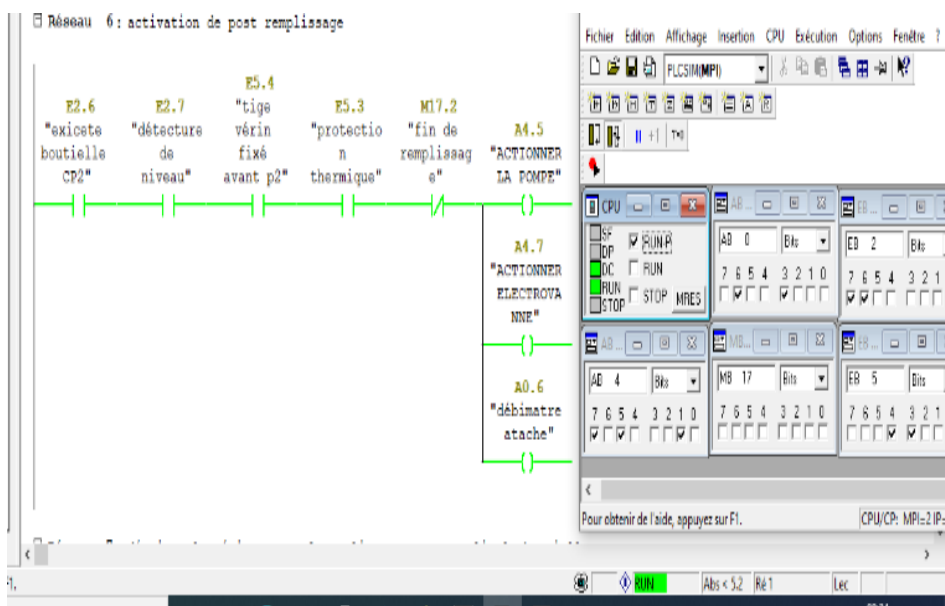


Figure 3.31:Simulation et visualisation de programme post 2.

Dans notre travail nous allons actionner le débitmètre pour calculer la quantité d'eau dans la bouteille, qui coupe le remplissage après l'affichage du volume 500ml. (Voir la figure 3.16)

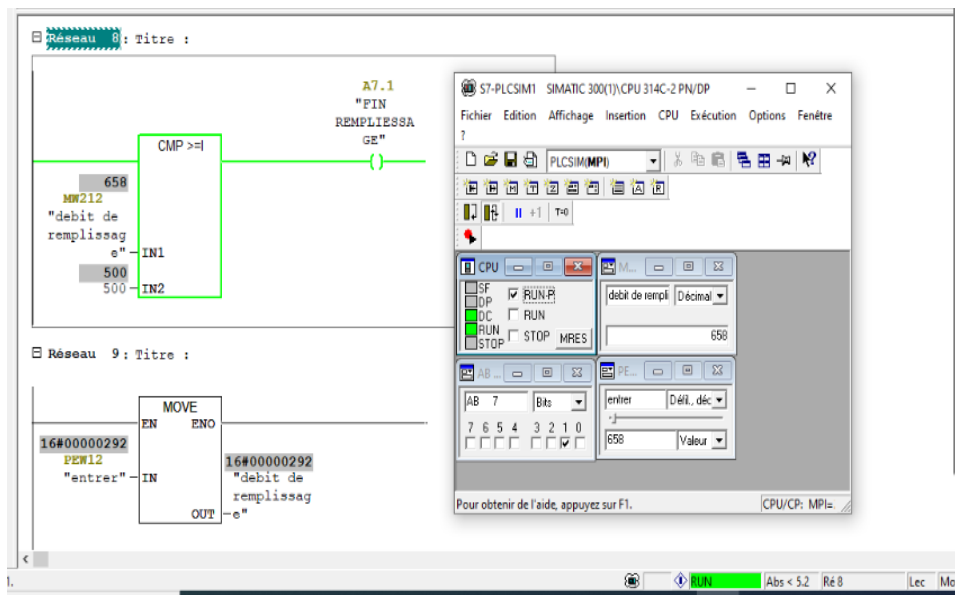


Figure 3.32: Simulation et visualisation du programme du débitmètre.

### 3.5.3 Simulation et visualisation de poste du bouchonnage des bouteille (Poste 3)

Dans notre travail nous allons présenter la simulation et la visualisation du dernier poste sur le bloc d'organisation OB1. Comme montrer sur la figure 3.17.

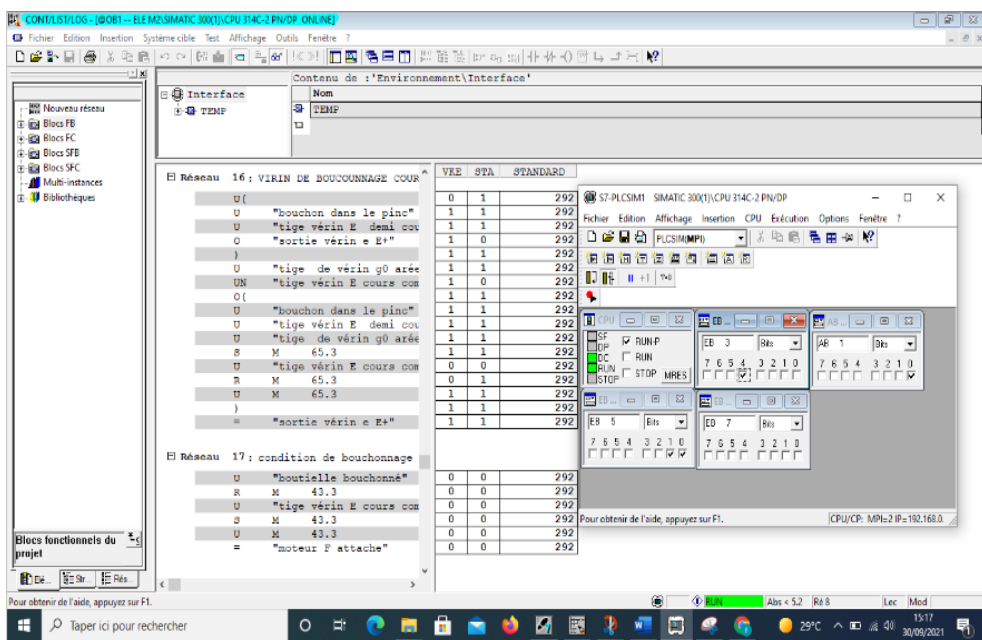


Figure 3.33: Présentation et visualisation du poste de bouchonnage des bouteilles.

### 3.6 La supervision du projet avec le WINCC flexible

#### 3.6.1 Définitions de la supervision

La supervision est une forme pour piloter et contrôler la partie opérative d'un système d'automatisme, constitue l'interface entre l'homme (opérateur) et la machine. Ainsi apporte un gestionnaire des alarme, d'archivage pour la maintenance, le traçage des courbes.

Ses avantages principaux de contrôle sont : la surveillance du processus à distance, la détection des défauts et le diagnostic et le traitement des alarmes.

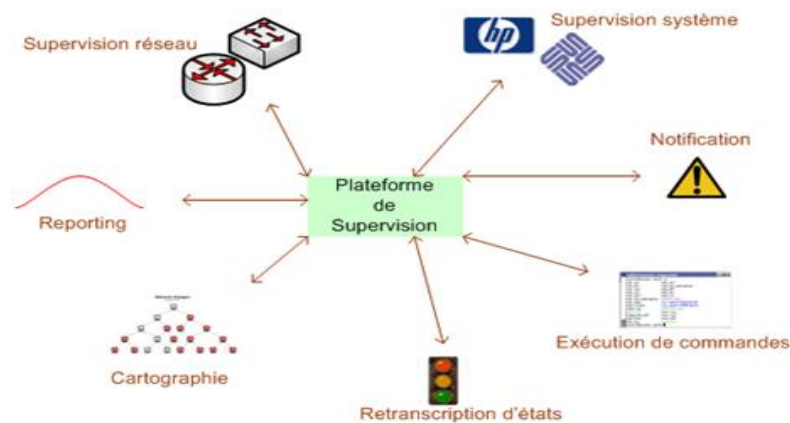


Figure 3.134:Schéma explicatif de introduction de la supervision.

#### 3.6.2 Création du projet avec WinCC flexible

Pour réaliser une interface graphique à l'aide du WinCC flexible on doit procéder par les deux méthodes suivantes :

**Méthode 01 :** Double cliquer sur l'icône de WinCC flexible qui se trouve dans le bureau. Cette méthode Professionnelle est présentée par la figure 3.19.



Figure 3.135: WinCC flexible 2008.

**Méthode 02 :** Cette méthode utilisée dans notre travail consiste à créer une station SIMATIC HMI à partir d'un logiciel steps7.

Après le chargement du programme crée dans STEPS7 , on teste la simulation sur l'application S7 PLCISIM, ainsi après l'installation du logiciel WinCC flexible nous allons lancer ce dernier sur le STEPS7 comme présenter dans la figure suivante.

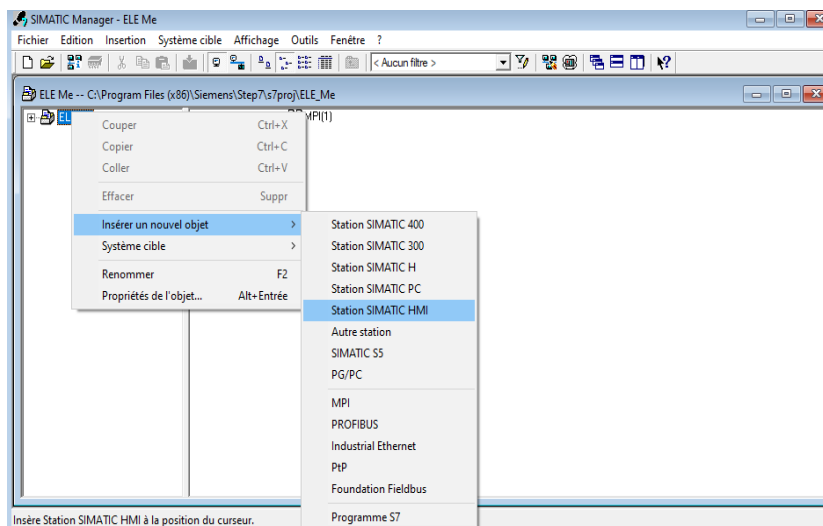


Figure 3.36:Création de station SIMATIC HMI de WinCC flexible.

### 3.6.3 Choix du panel du pupitre opérateur

Pour créer et visualiser votre projet dans le WinCC flexible, nous allons choisir une panel MP 270 10'' Touch, pour faire la partie supervision d'un système d'automatisation. (Voir la figure 3.21)

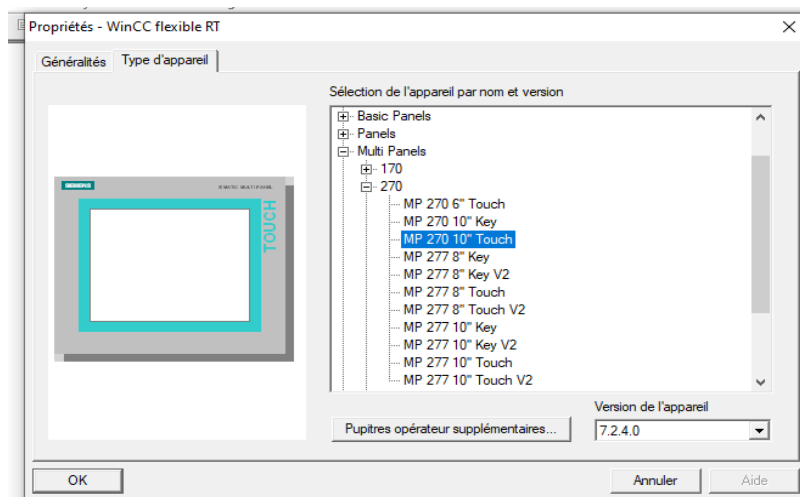



Figure 3.37:Présentation la fenêtre de choix du pupitre.

### 3.6.4 Démarrage de la station SIMATC HMI

Pour démarrer la station SIMATIC HMI on va cliquer après le choix de panel sur le bouton OK, ensuite nous allons cliquer deux fois sur le bar  pour faire la liaison entre la station SIMATC HMI et l'automate S7-300. (Voir la figure 3.22)

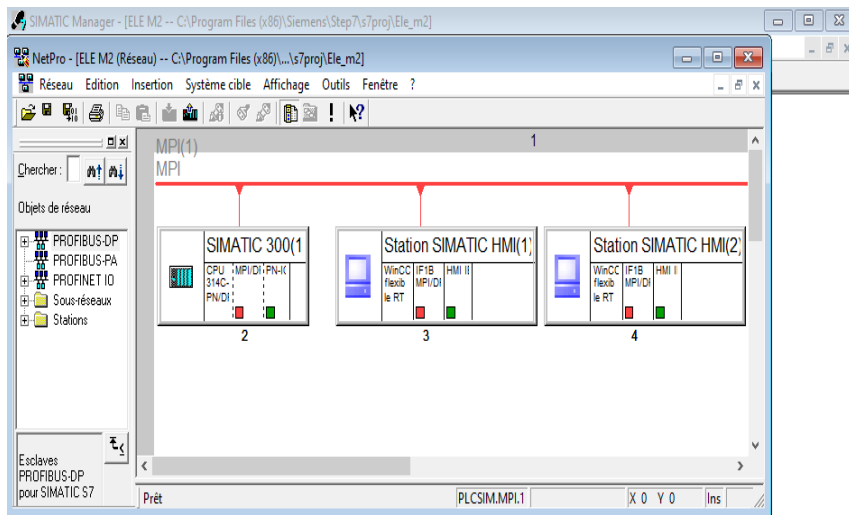



Figure 3.38: Liaison entre station SIMATC HMI et l'automate S7-300.

- Ensuite nous allons cliquer sur le bar  lancer dans STEPS 7, on va ouvrir la communication, après nous allons cliquer sur la liaison pour connecter notre panel choisi avec l'automate programmable industrielle.

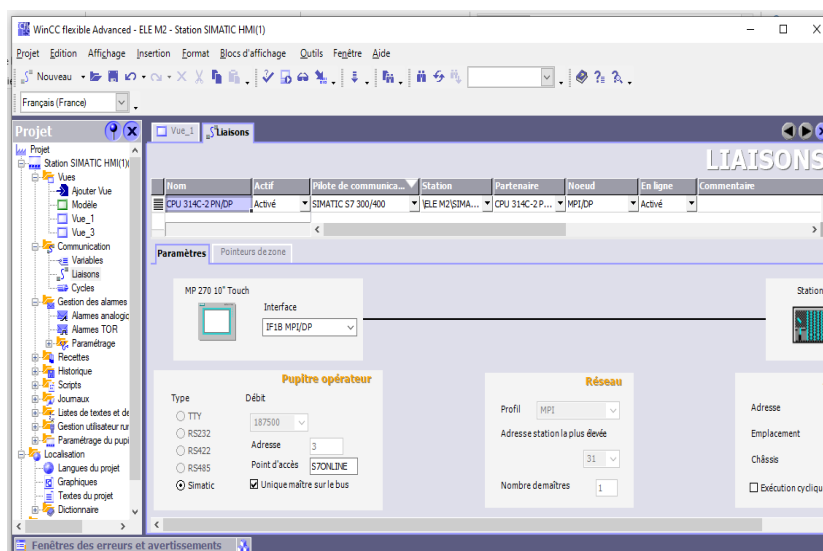


Figure 3.39 : Présentation du connecteur entre le panel et l'automate.

Après la connexion de notre panel PM 270 10 "Touch avec l'automate, nous allons créer la table de variable intégrée dans le programme assurer dans le STEP 7. (Voir la figure 3.24)

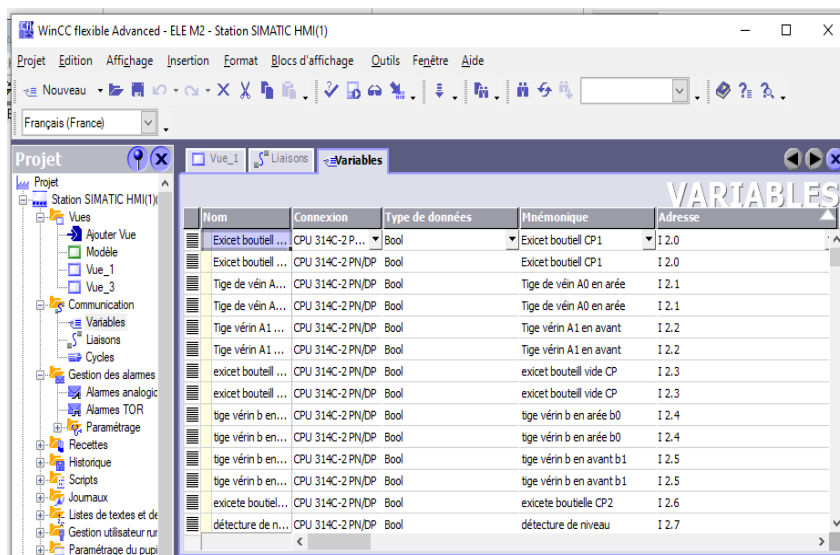


Figure 3.40: Présentation du tableau de variable dans le WinCC flexible.

### 3.6.5 Présentation de Vue dans le WinCC flexible

Après avoir lancé et configuré le logiciel de supervision WinCC flexible, nous allons ouvrir une vue de travail, qui contient les différents éléments pour la réalisation et le contrôle d'un projet, une bibliothèque, une zone de travail, une fenêtre de projet contenant l'ensemble des vues (accueil, paramètres, alarmes et courbes). (Voir la figure 3.25)

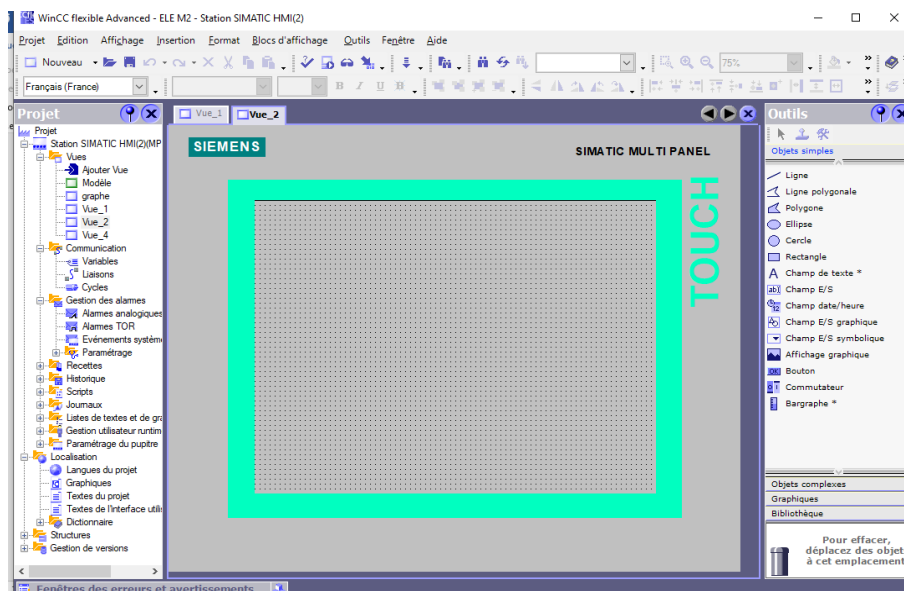


Figure 3.41: Présentation du Vue dans le WinCC flexible.

- Les éléments d'une vue du logiciel WinCC flexible :

**-Zone de travail :** La zone de travail sert à éditer les objets du projet. [2]

**-Boîte d'outils :** La fenêtre d'outils vous propose un choix d'objets que vous pouvez insérer dans vos vues, des objets graphiques et éléments de commande. [2]

**-Fenêtre du projet :** La fenêtre du projet est contenue tous les éléments et tous les éditeurs disponibles d'un projet réaliser dans le WinCC flexible, vous pouvez accéder aux paramètres du pupitre, à la localisation et à la gestion de version. [2]

**-Fenêtre des propriétés :** pour les paramétrages des objets dans la zone de travail, qui est composée de quatre éléments : « général, Propriétés, Animation, Evénement ». [2]

**-Fenêtre des erreurs et avertissements :** La fenêtre des erreurs et avertissements affiche les alarmes système générées p. ex. lors du test d'un projet. [2]

**-La fenêtre d'objets :** permettant la sélection d'objets déjà créés (et leur copie dans l'image par glisser-lâcher) ; [2]

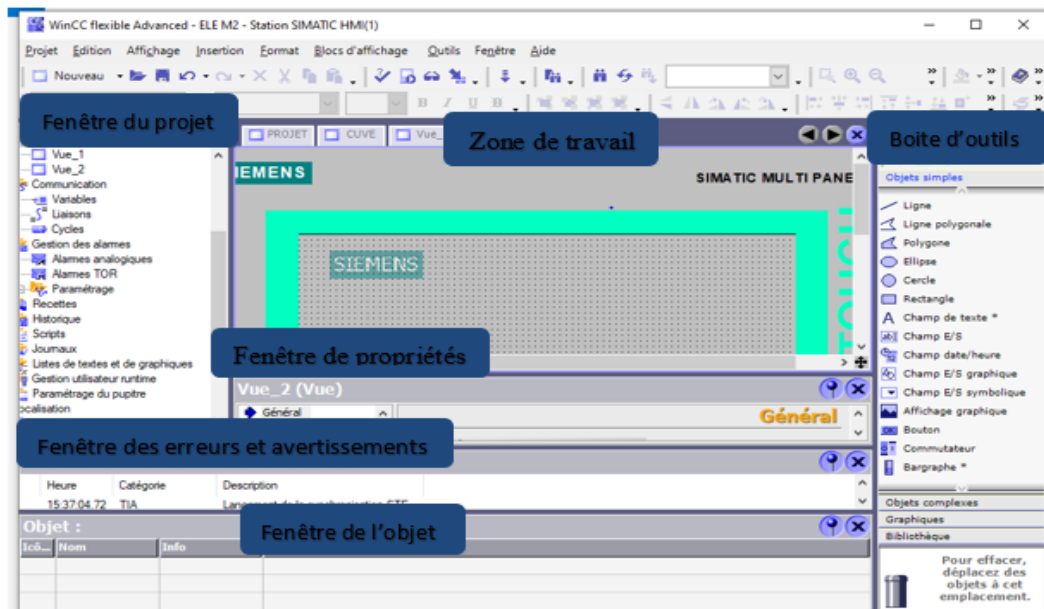


Figure 3.42: Les éléments d'une vue continue dans le WinCC flexible.

### 3.7 Création de vues du projet de poste de remplissage et bouchonnage

Dans notre projet, nous avons créer une vue modèle pour que les autres vues ressemblent à celle-ci, ensuite nous allons créer tout l'objet indispensable à la commande et au contrôle de l'installation de la chaine de production piloter sur WinCC flexible. (Voir la figure 3.27)

Dans notre projet on a configuré 4 Vues

1. 1ere vue : Accueil.
2. 2eme vue : cuve.
3. 3eme vue : alarme.

4. 4eme vue : PROJET.

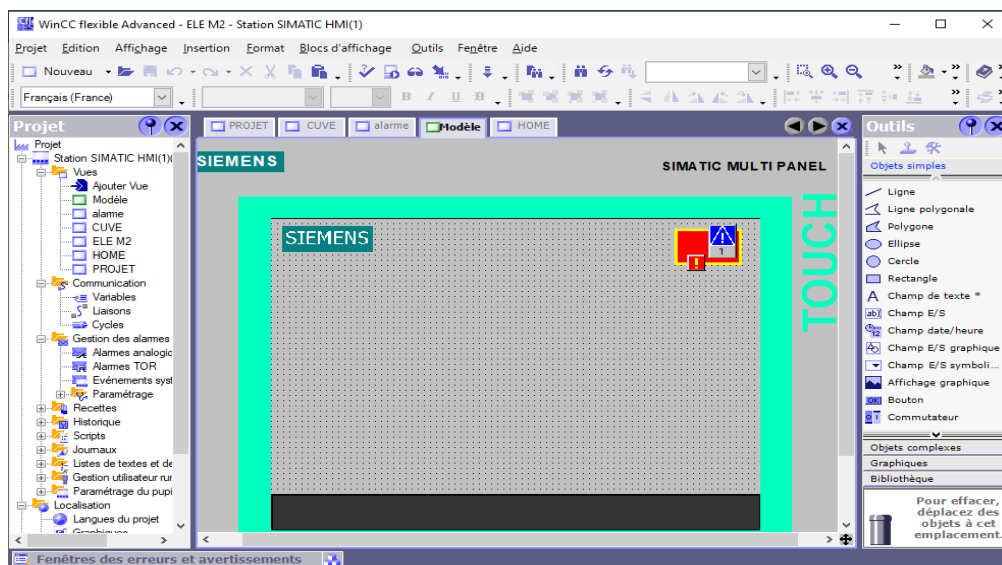


Figure 3.43: Présentation de la vue modèle de la chaine de production piloter.

Dans ce modèle, on trouve la plaque de **Siemens** pour vois dans toutes les vues utiliser, quitter d'alarme de vanne.

### 3.7.1 Création de la vue d'alarme

Les alarmes montrent les évènement ou l'états de fonctionnement qui se produisent sur l'installation de vanne de la chaine de la production pour les diagnostique des erreurs, comme présenter dans la figure 3.28.

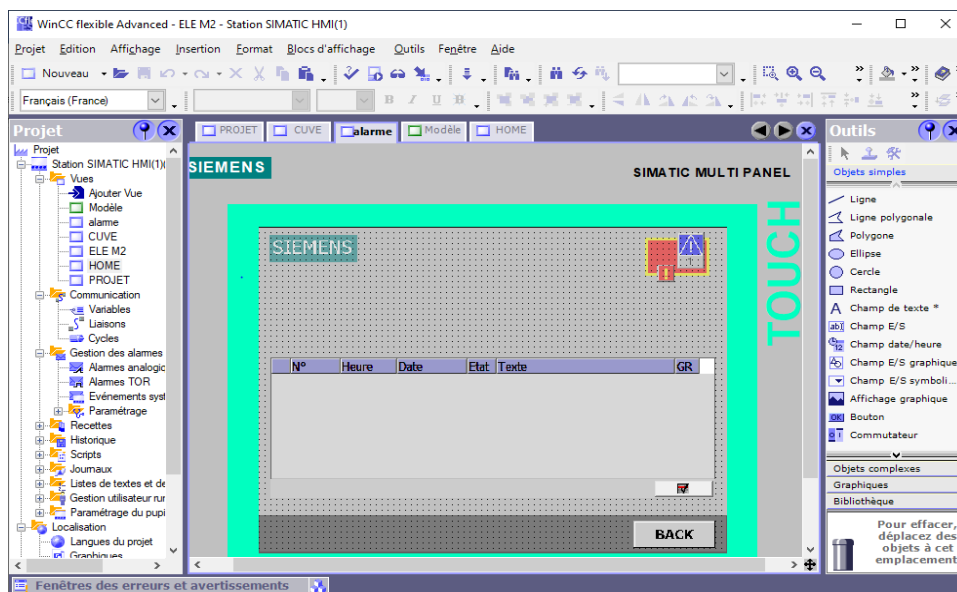


Figure 3.44: Vue d'alarme pour contrôler la vanne de remplissage.

Dans notre travail nous allons commander une alarme analogique indiquant les dépassements de limite de la vanne utilisée.

### ✚ Création de la vue de projet

Dans notre travail et sur la vue de PROJET nous allons expliquer et commander brièvement les éléments principaux d'une partie opérative (vérin, moteur, pompe, capteur de niveau...etc.), ensuite contrôler avec la station (HMI) de WinCC flexible Advanced, comme montrer dans la figure 3.29.

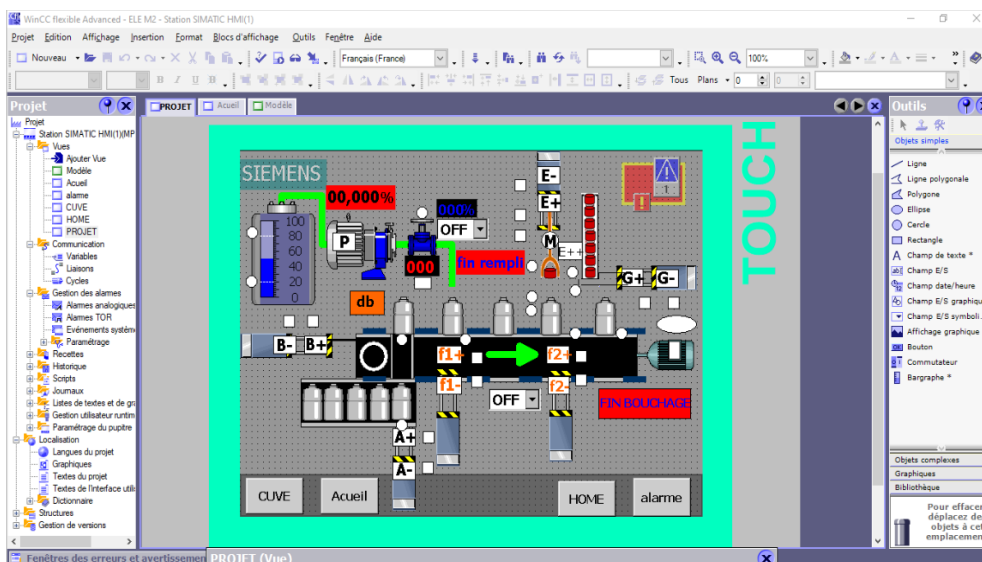



Figure 3.45: Présentation de la vue PROJET d'une chaine pilotée sur WinCC flexible.

## 3.8 Simulation avec le WinCC flexible Advanced

### ➤ Vue – Principal



Figure 3.30 :Vue principal.

Après la configuration de la liaison entre le panel utilisé et l'automate (WinCC flexible est intégré dans SIMATIC steps7), nous avons enregistré et compilé notre travail sur le logiciel WinCC flexible. Ensuite nous allons simuler et visualiser le programme sur Steps7, après on va lancer et tester la simulation par clic sur l'icône  qui se trouve dans la barre d'outils de WinCC flexible. Comme montré dans la figure 3.30.

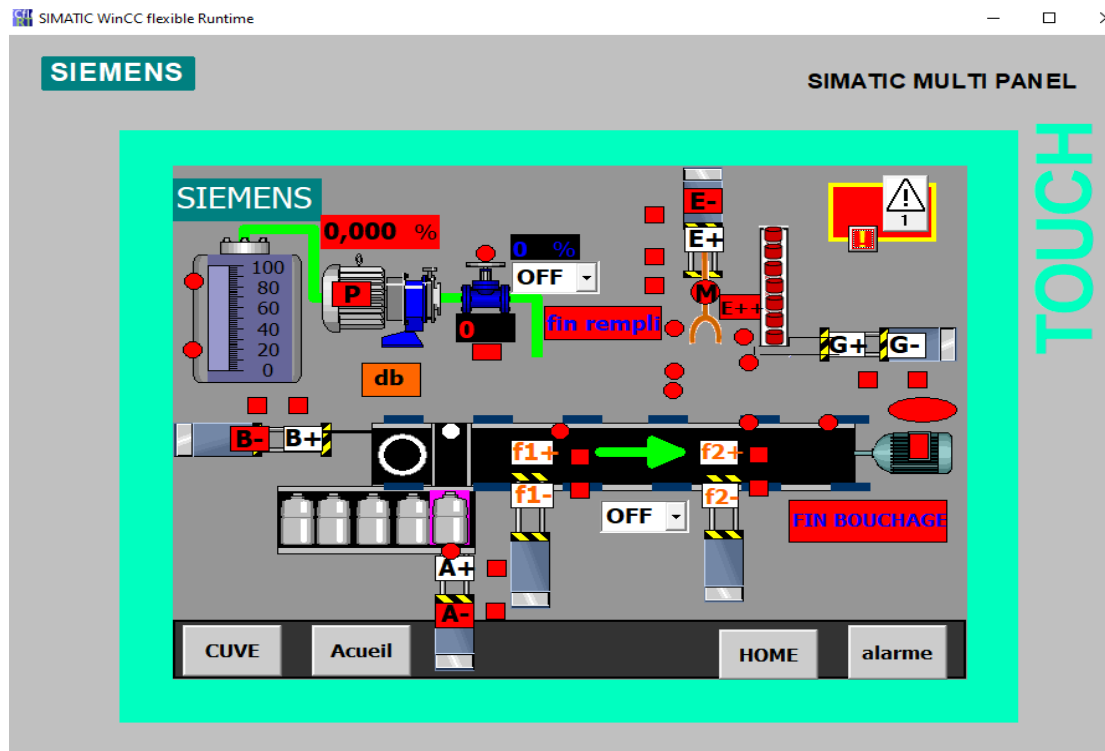


Figure 3.31 : Simulation de la chine de production piloter.

### 3.8.1 Visualisations d'alarme de la vanne

Lorsque la vanne de liquide est supérieure à 80% ou inférieure à 20%, l'alarme de déclanche (voir la figure 3.31).

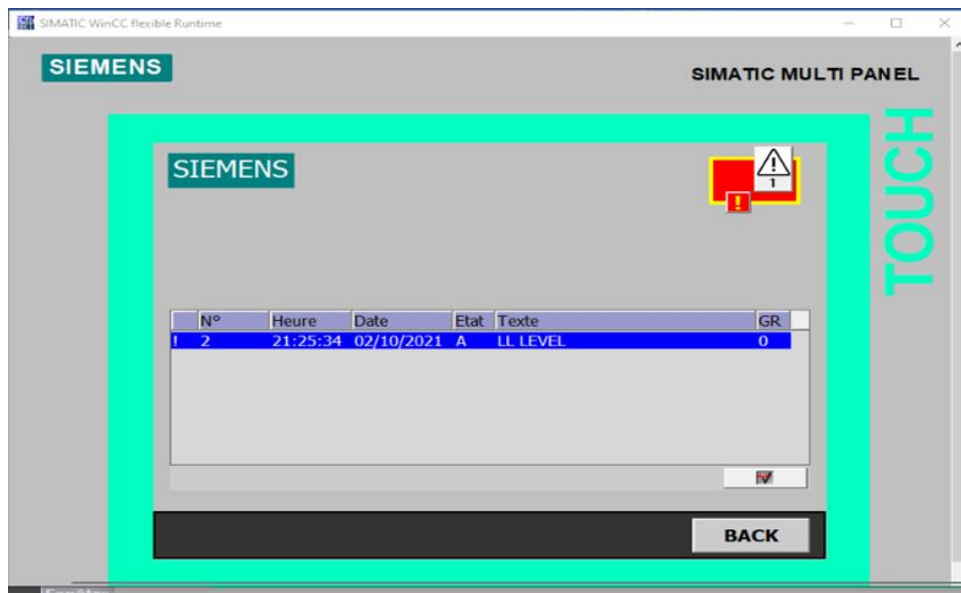


Figure 3.32 : Présentations de la simulation de vue d'alarme.

### 3.6 Conclusion

Ce chapitre est abordé par une présentation générale du cahier de charge et leur grafcet de projet étudié, nous avons procédé à la simulation du projet par deux methodes avec le logiciel STEP 7.

Dans une seconde partie on a communiqué le STEPS 7 avec le logiciel WinCC flexible par l'interface multipoint (MPI), que nous avons simulé et tester le programme réaliser dans STEPS 7, ainsi que son système de supervision, sous WinCC flexible.

## Conclusion générale

Notre travail de mémoire de fin d'étude consistait à étudier le fonctionnement de la chaîne de production de poste remplissage et bouchonnage à base d'un automate programmable industriel, et le contrôle du système avec une supervision homme-machine pour éviter tous les problèmes rencontrés avec la machine, et pour augmenter la productivité de production, quantité, qualité et apporter la sécurité aux personnels.

Notre projet nous a aidé à améliorer nos connaissances dans le domaine d'automatisation industrielle par la familiarisation avec l'automate S7-300 Siemens, et le logiciel de contrôle-supervision WinCC flexible.

Nous espérons enfin que ce travail sera une meilleure étude de fonctionnement d'une chaîne de production poste remplissage et bouchonnage avec automate SIMATIC de Siemens, et en souhaitant faire bonne supervision de ce projet dans le domaine d'automatisme industriel pour assurer le repos et la sécurité personnelle.

En tenant compte des limites de l'existant, il s'impose à nous, de trouver des moyens techniques pour pallier à ces limites. Ce qui résume par l'automatiser le poste remplissage et bouchonnage à l'aide d'un automate SIMATIC S7-300 qui permet de tester le bon fonctionnement du système avant de passer à la réalisation pratique.

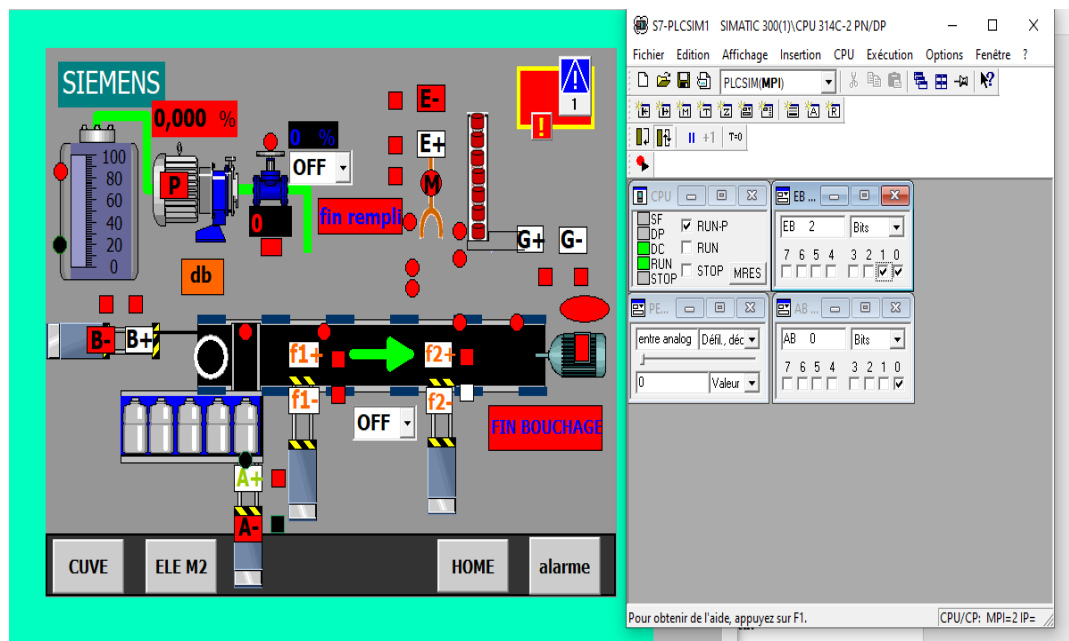
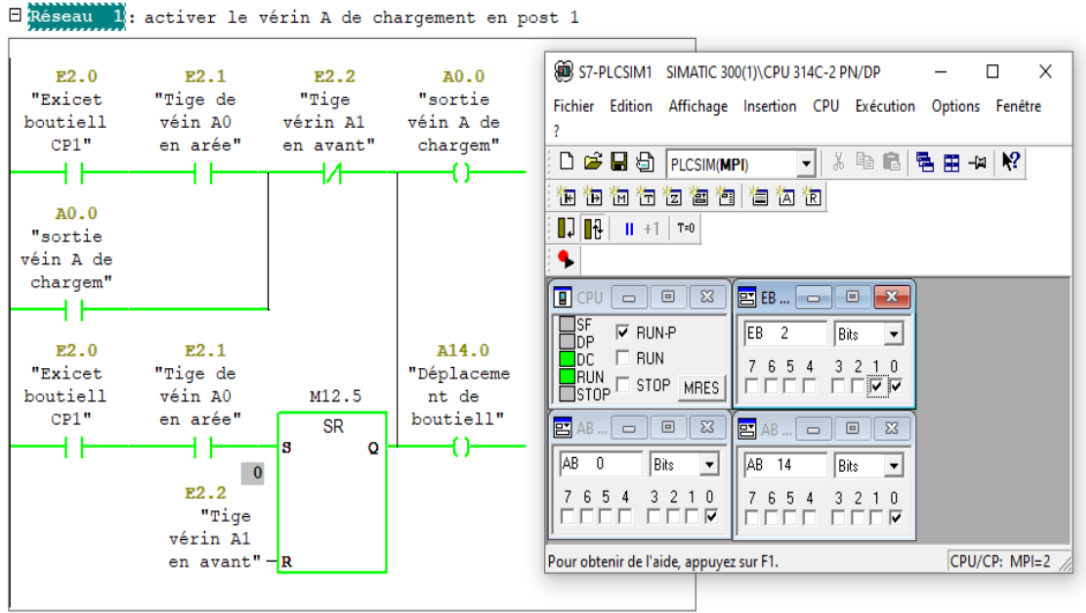
Le logiciel de programmation STEP7 qui fait partie de l'industrie logicielle SIMATIC, constitue le lien entre l'utilisateur et l'automate S7 – 300. Enfin, Les résultats de la simulation et la supervision ont montré que le cahier de charge de la chaîne de production automatisée du poste remplissage et bouchage est satisfaisant.

## Références bibliographiques

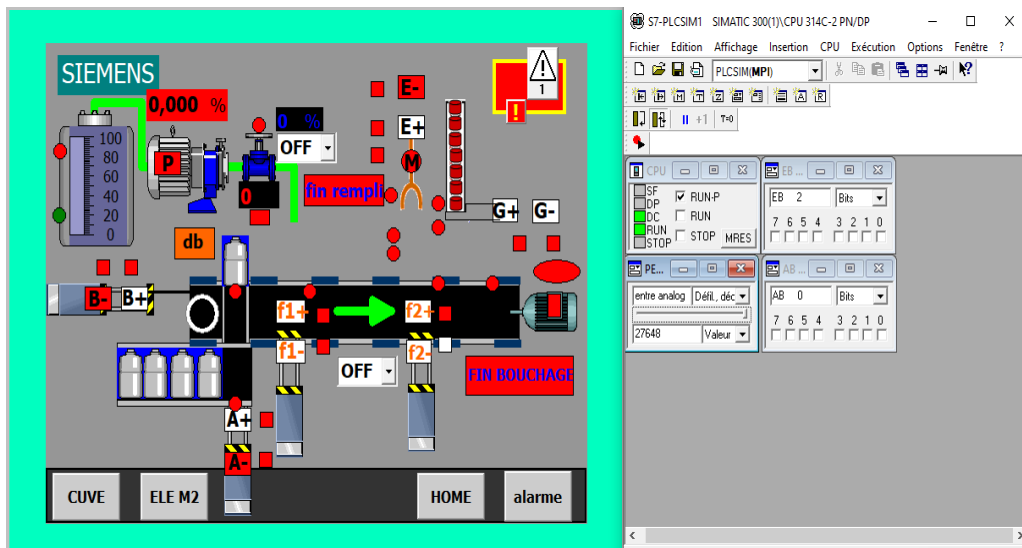
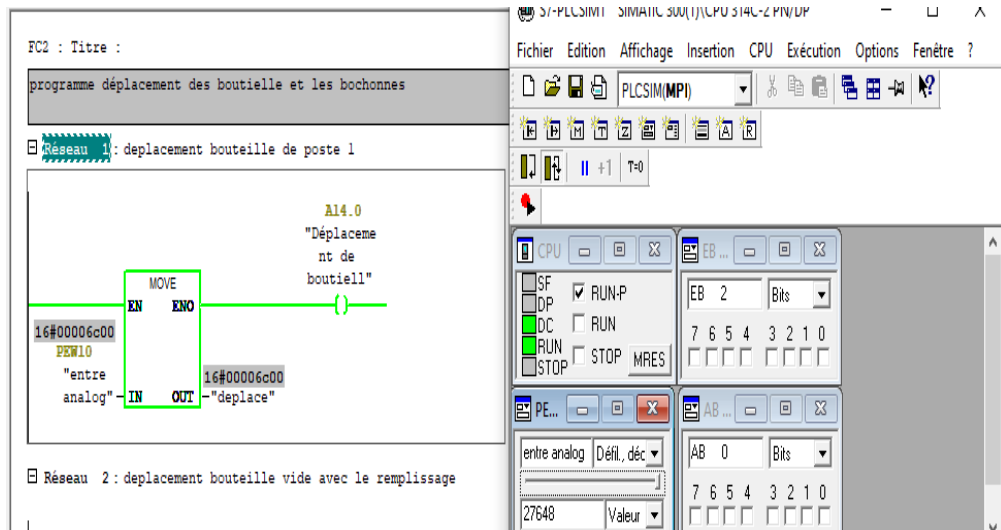
- [1] ANDRE. S, Automates programmable, programmation, automatisation, et logique programmée, Edition L'ELAN, 1983.
- [2] CHERCHOUR, Hamza, CHAHBOUNE, Mohamed Lamine, MENDIL, Boubekour, et al. Commande et supervision d'un processus de margarine via un automate programmable chaine pilote. 2015. Thèse de doctorat. Université abderrahmane mira béjaia.
- [3] William, bolton. AUTOMATES PROGRAMMABLES INDUSTRIELS. 2éd. Paris : DUNOD, 2015, nombre de page 430, (l'usine nouvelle). ISBN 978-2-10-074033-8.
- [4] M. SADOUN, Dimensionnement et automatisation d'une machine cellophaneuse. Nombre de pages 43 page. Discipline : Année 2016/2017. mémoire fin d'étude. université de bouira.
- [5] N. BENHADDA, Modélisation des Capteurs Inductifs à Courants de Foucault. Nombre de page 80 page. Discipline : Septembre 2006. Mémoire fin étude. Université El Hadj Lakhdar Batna
- [6] RAZIKA, Zedek et SAMIA, Meghnez. Automatisation et supervision de la machine à garnir les encoches" Encocheuse" à base d'un API S7-200 et une HMI TP 177A6". 2010. Thèse de doctorat. Université Mouloud Mammeri.
- [7] <https://capteur-pneumatique.weebly.com>.
- [8] [http://staff.univ-batna2.dz/sites/default/files/bendjerad\\_adel/files/cour\\_capteur\\_l3](http://staff.univ-batna2.dz/sites/default/files/bendjerad_adel/files/cour_capteur_l3).
- [9] A. BELKADI, M. TOUAT. Remplacement de la logique câblée par la logique programmée (API S7-300) du gratteur portique. Nombre de page 87 page. Discipline : 2017/2018. mémoire fin d'étude. université de bouira.
- [10] Siemens, " logiciel step7 manuel ", 2004.

## Simulation de programme complies avec le logiciel STEPS7 et la supervision dans WinCC flexible

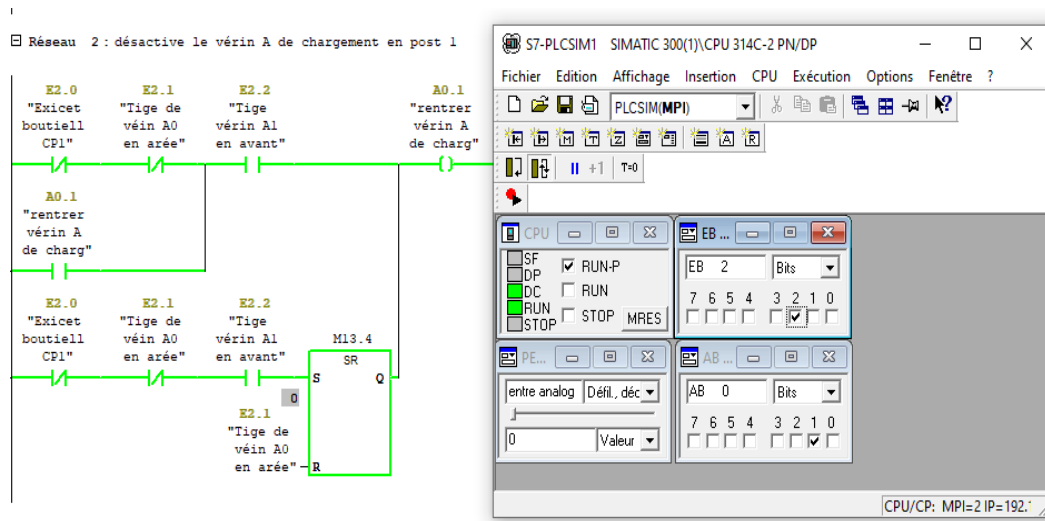
### A.1 sortie vérin Ade chargement

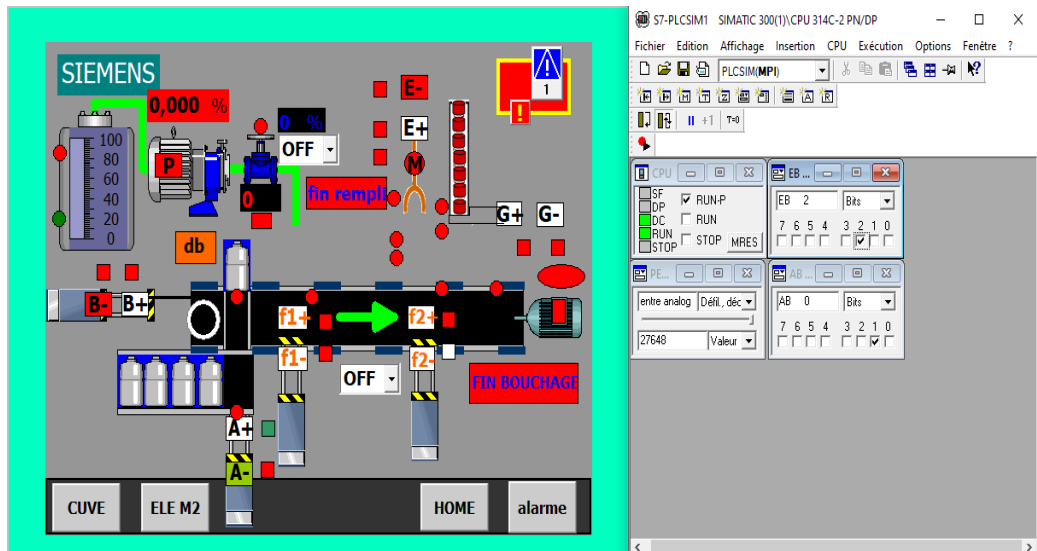


## A.2 Déplacement la bouteille de la post de chargement

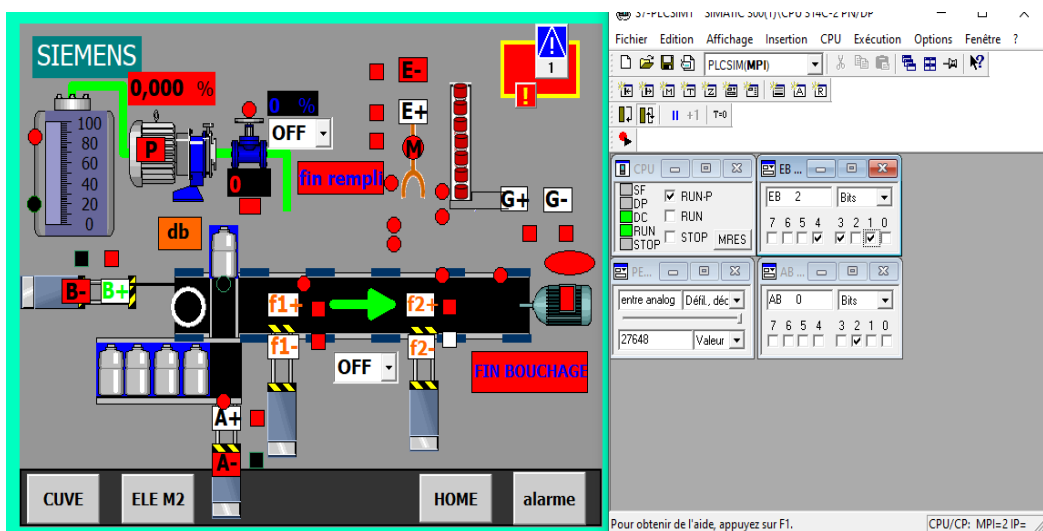
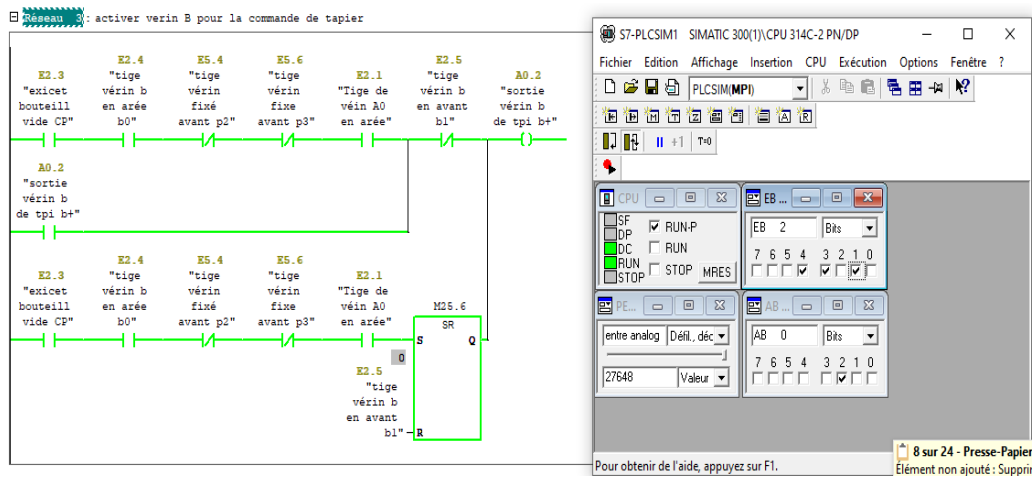


## A.3 Rentrer le vérin de chargement





### A.4 Sortie vérin B pour déplacer le tapier



### A.5 Déplacement la bouteille vide avec le remplissage et rentrer le vérin B

PLCSIM(MPI)

Réseau 2: déplacement bouteille vide avec le rempli

MOVE

EN ENO

16#00006892 PEW18 "entier boîte" IN OUT

16#00006892 "boîte -déplace"

A14.0 "Déplacement de bouteille"

CPU: SF, DP, DC, RUN, STOP, MRES. entier boîte: Défil., déc. Valeur: 26770. AB 0 Bits: 7 6 5 4 3 2 1 0

S7-PLCSIM1 SIMATIC 300(1) CPU 314C-2 PN/DP

Réseau 4: désactiver le vérin B pour le papier

E2.3 "exicet bouteill vide CP" E2.4 "tige vérin b en arée b0" E2.5 "tige vérin b en avant b1"

A0.4 "rentrer vérin b de tpi b-"

E2.3 "exicet bouteill vide CP" E2.4 "tige vérin b en arée b0" E2.5 "tige vérin b en avant b1"

M14.6 SR

E2.4 "tige vérin b en arée b0" R

CPU: SF, DP, DC, RUN, STOP, MRES. EB 2 Bits: 7 6 5 4 3 2 1 0. entre analog: Défil., déc. Valeur: 25673. entier boîte: Défil., déc. Valeur: 0. AB 0 Bits: 7 6 5 4 3 2 1 0

S7-PLCSIM1 SIMATIC 300(1) CPU 314C-2 PN/DP

SIEMENS

0,000 %

db

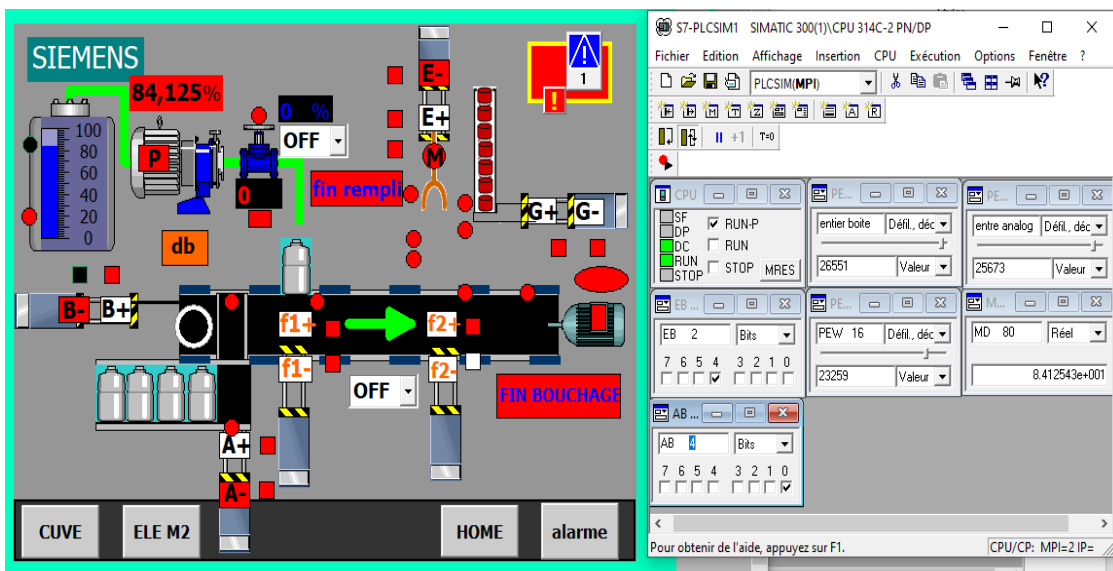
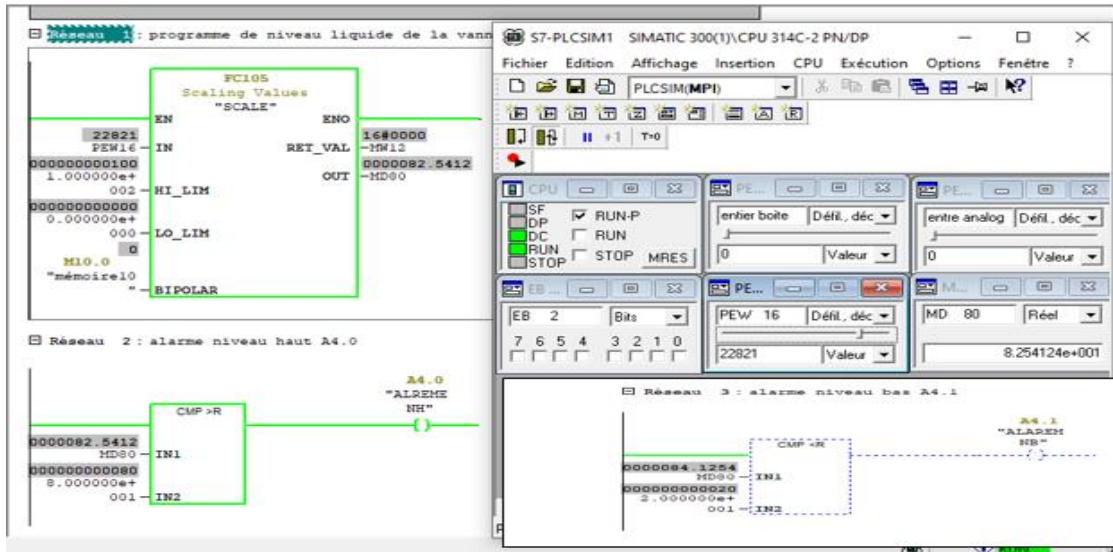
fin rempli

FIN BOUCHAGE

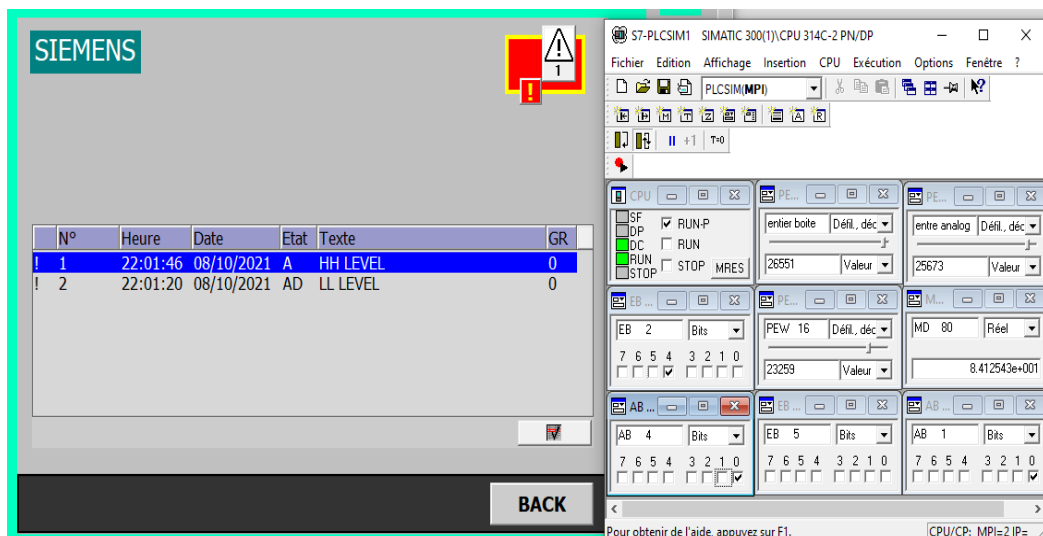
CUVE ELE M2 HOME alarme

CPU: SF, DP, DC, RUN, STOP, MRES. entier boîte: Défil., déc. Valeur: 26770. AB 0 Bits: 7 6 5 4 3 2 1 0. EB 2 Bits: 7 6 5 4 3 2 1 0. EB 0 Bits: 7 6 5 4 3 2 1 0

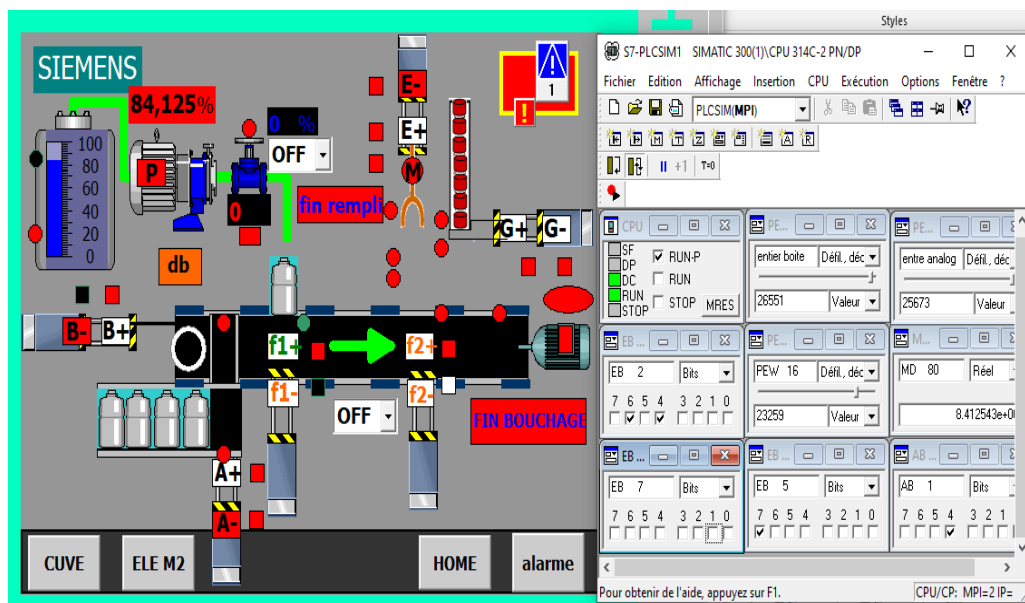
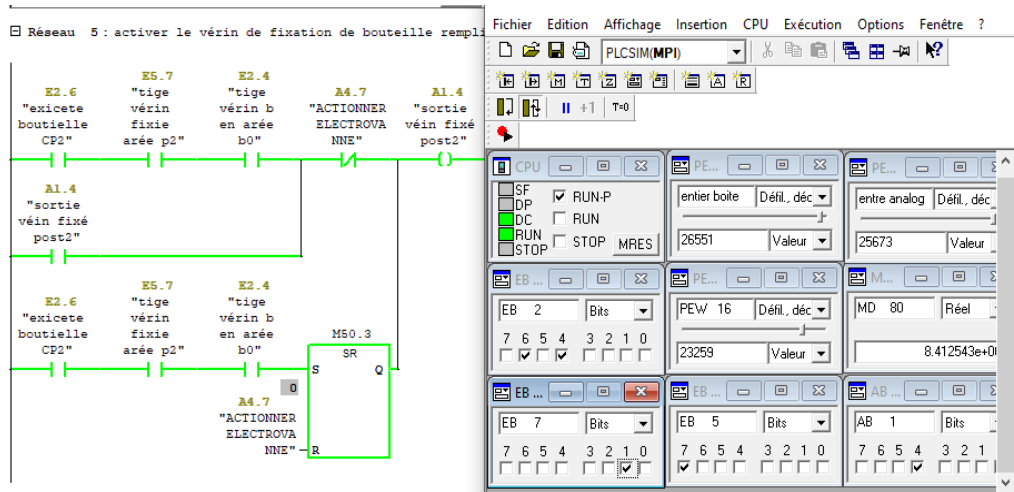
### A.6 Contrôle le niveau de la vanne de remplissage



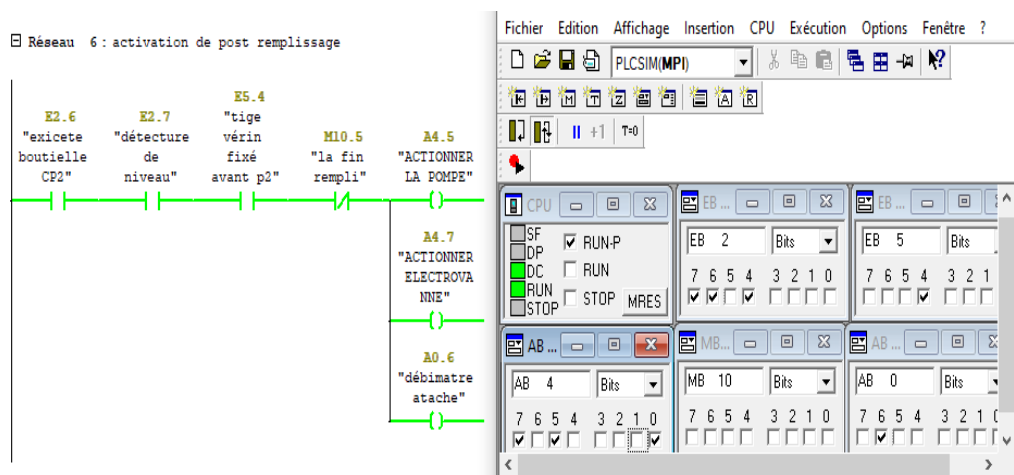
### A.7 affichage sur vue alarme niveau bas et niveau haut de la vanne

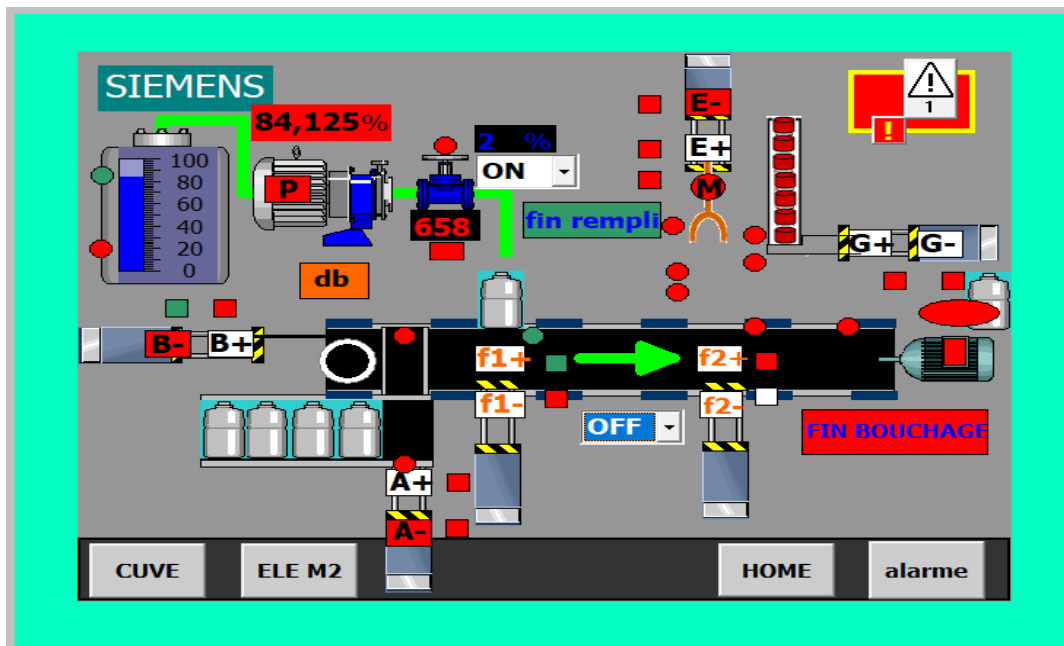
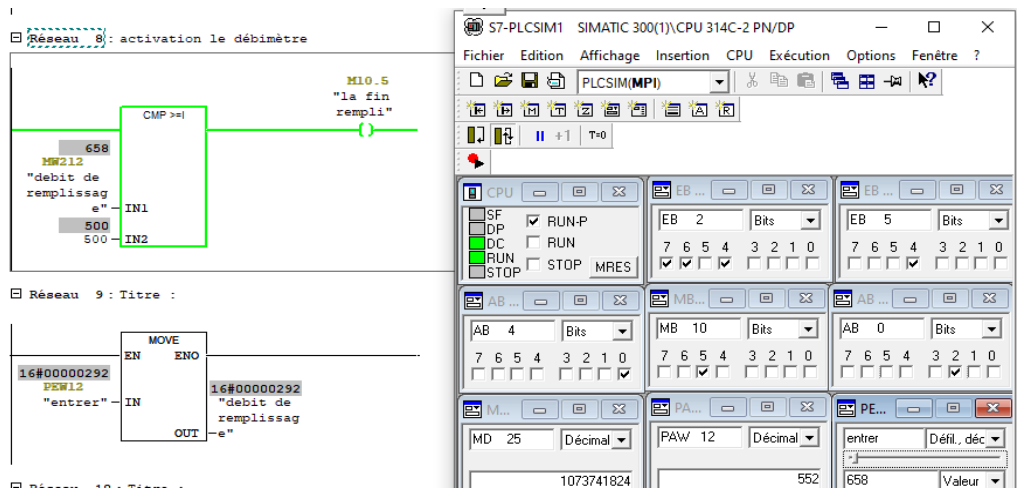


### A.8 Sortie vérin F1 pour fixation la bouteille

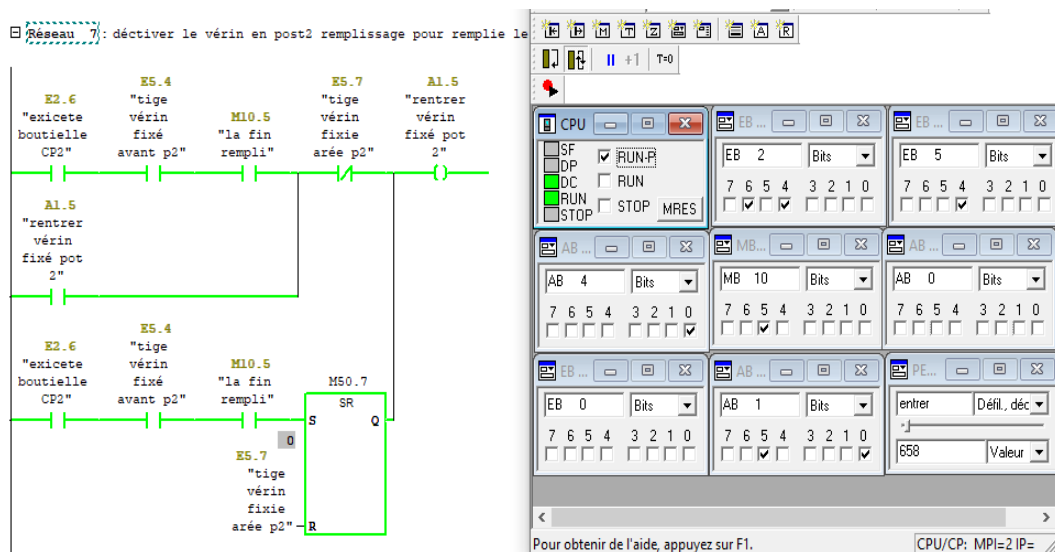


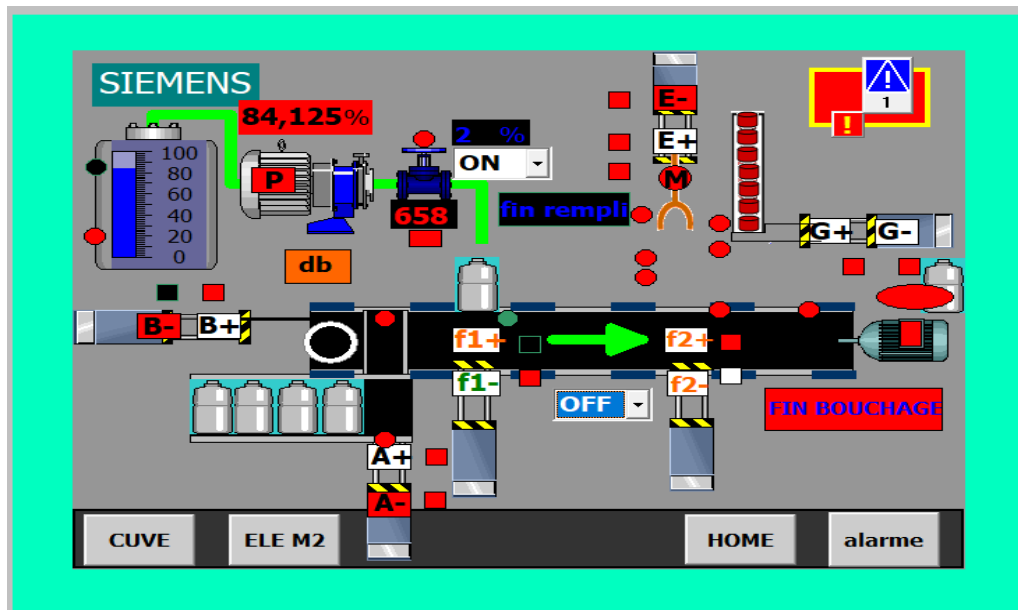
### A.9 L'activation de la pompe, électrovanne, décimètre



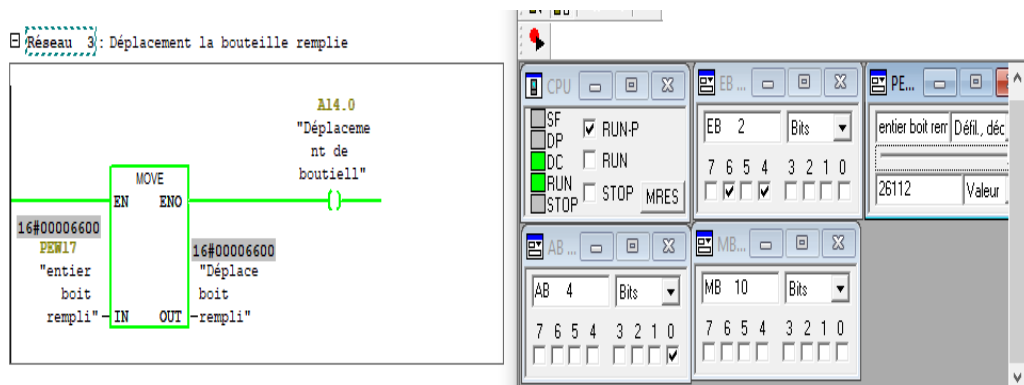


### A.10 Rentrer le vérin F1 de fixation

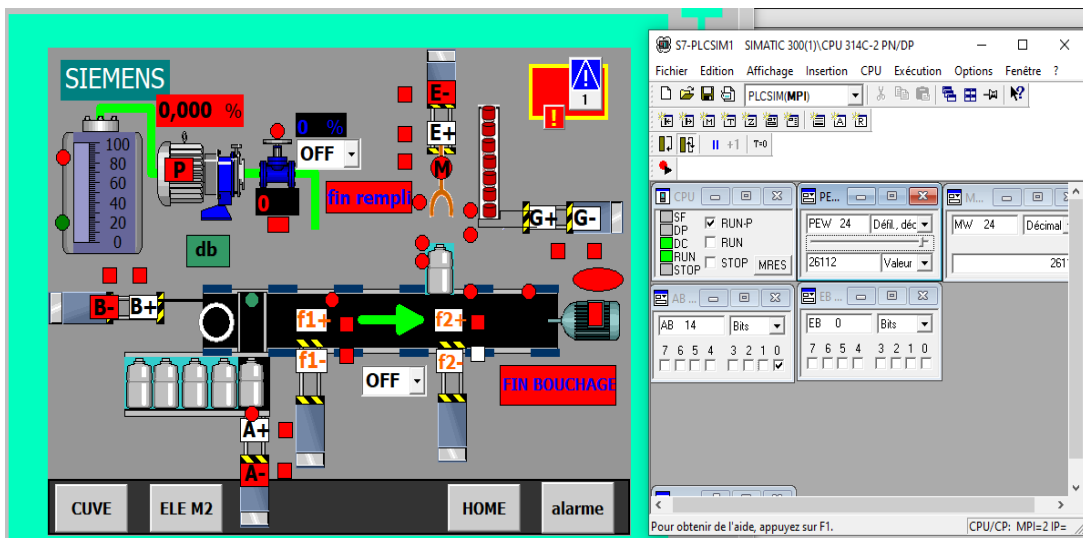
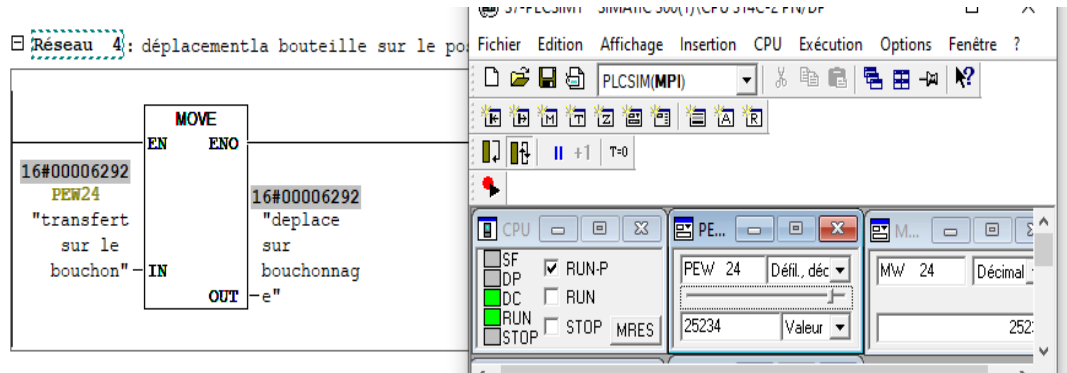




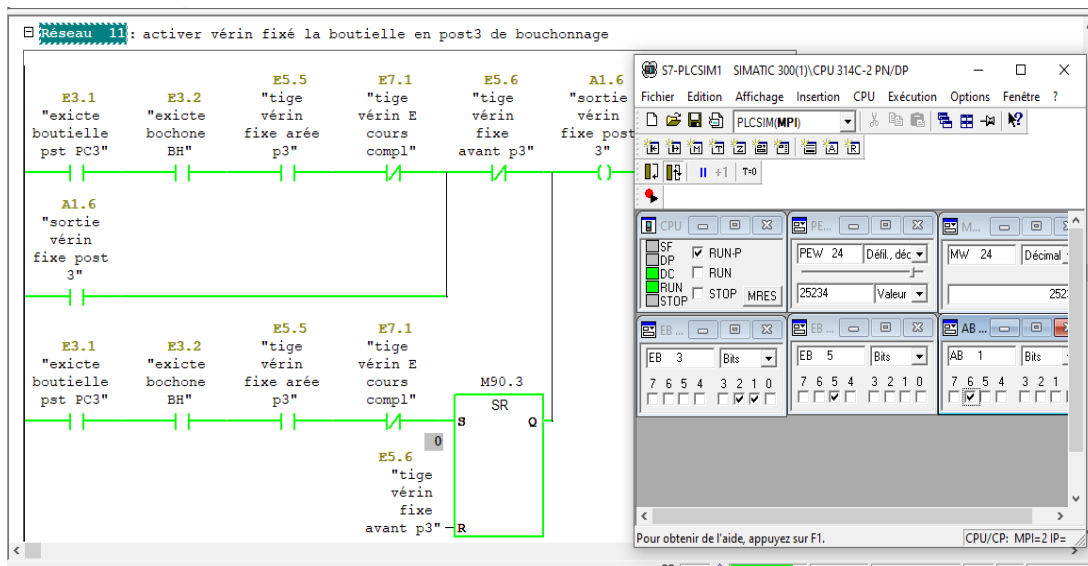
### A.11 Déplacement la bouteille remplie

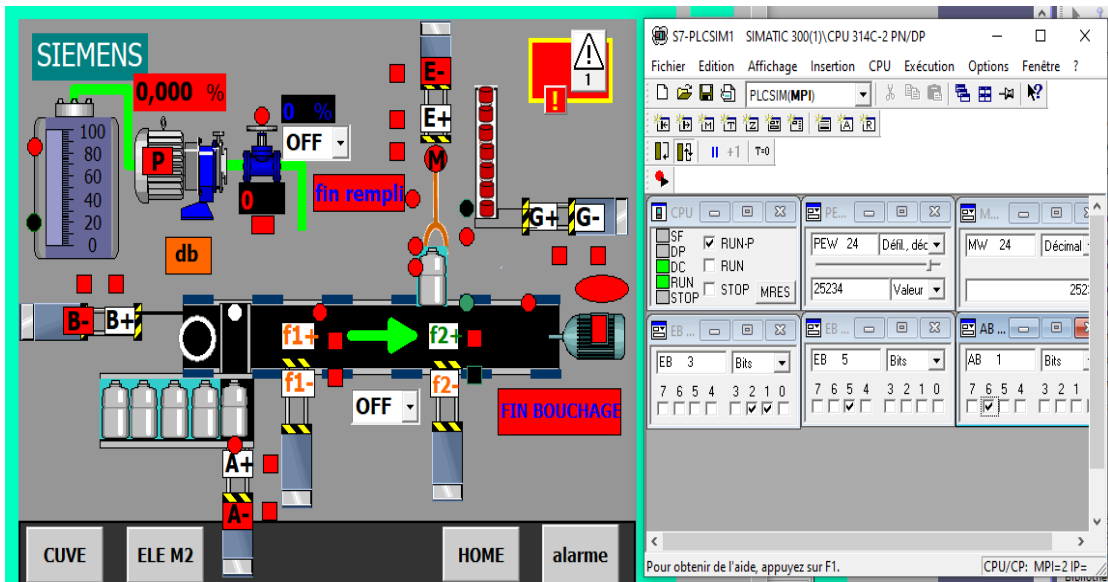


### A.12 Déplacement la bouteille sur le post de bouchage

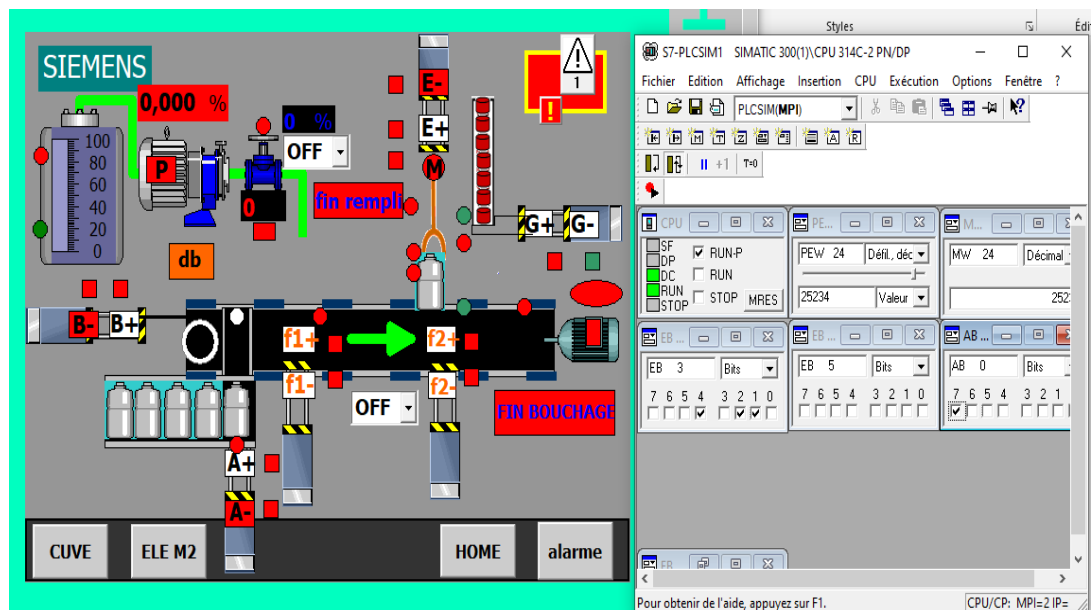
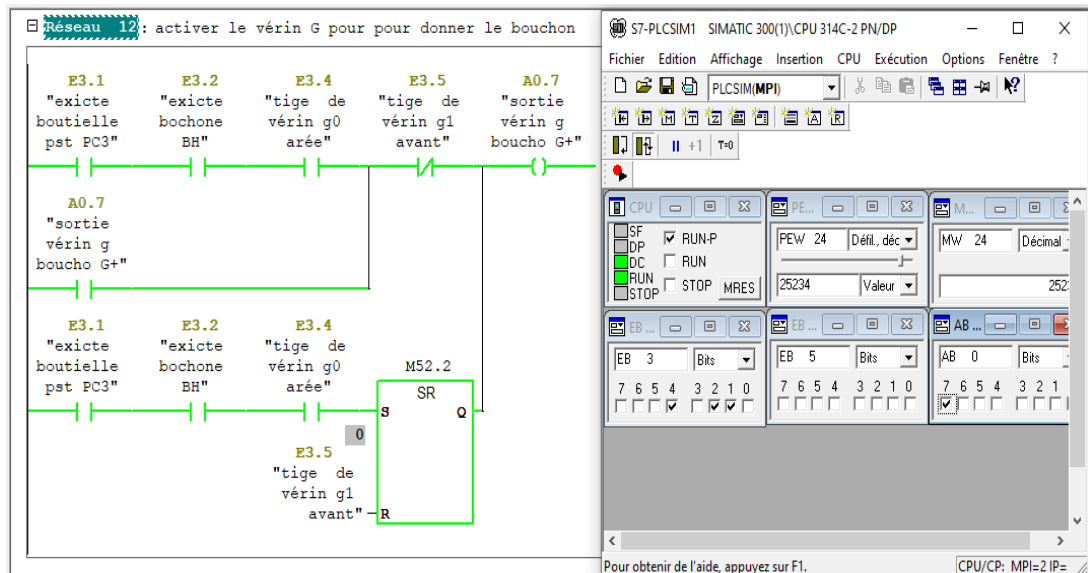


### A.13 Sortie vérin F2 de fixation la bouteille dans le poste bouchage (poste 3)

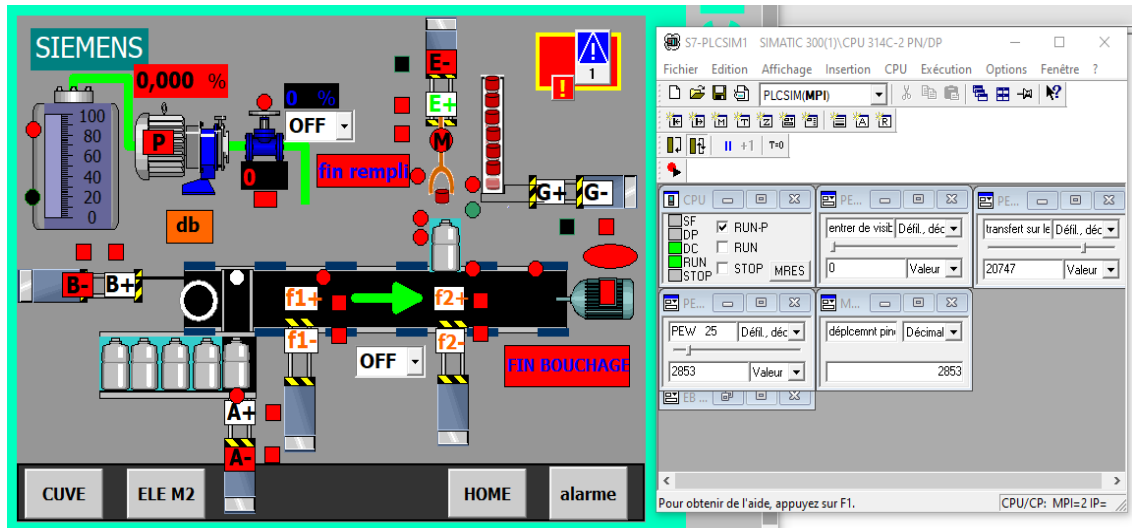
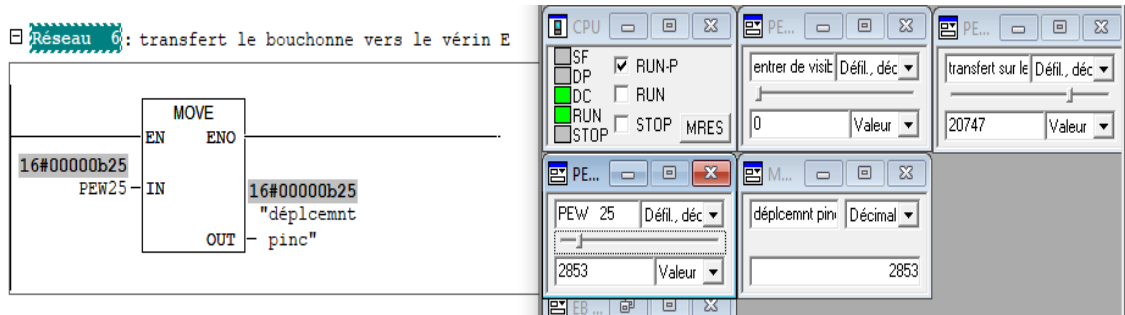




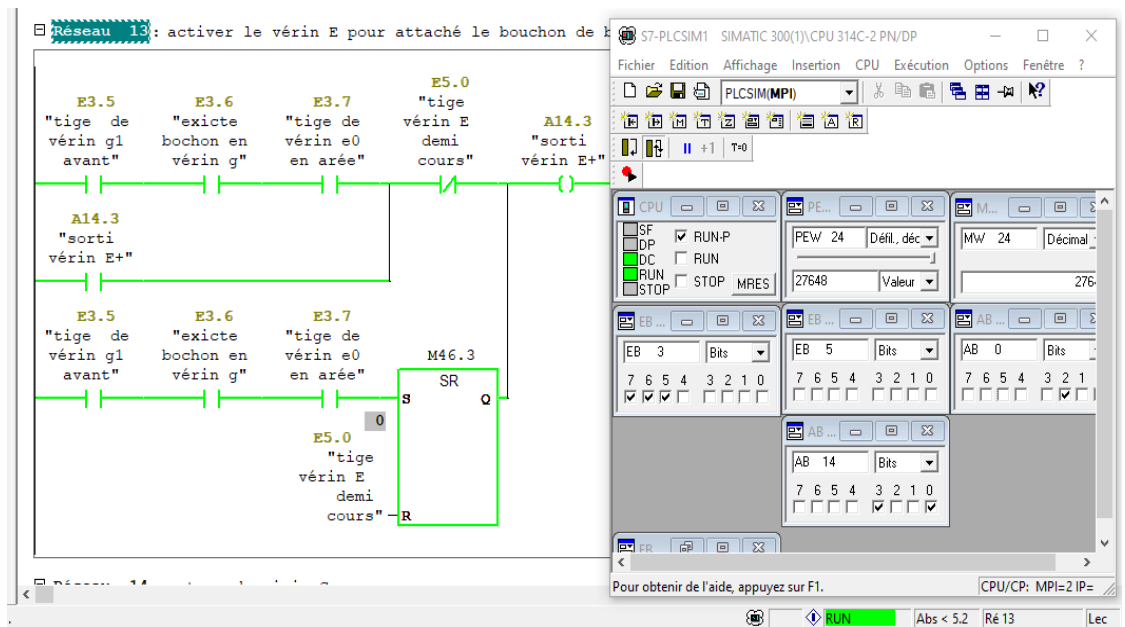
### A.14 Sortie vérin G pour transfert le bouchonne ver le vérin E

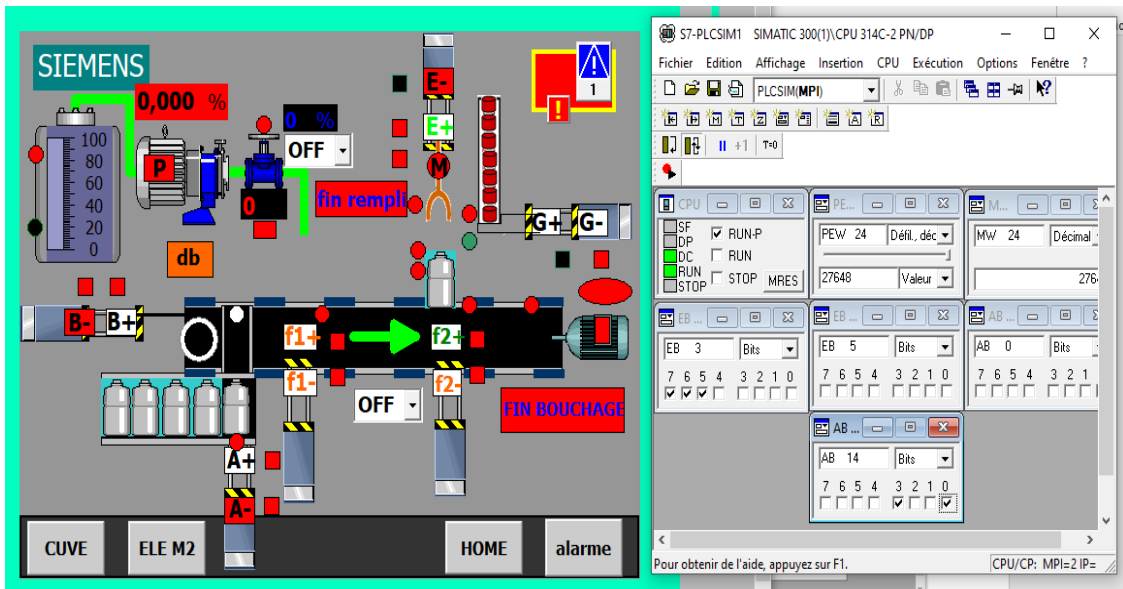


### A.15 transfert le bouchonne

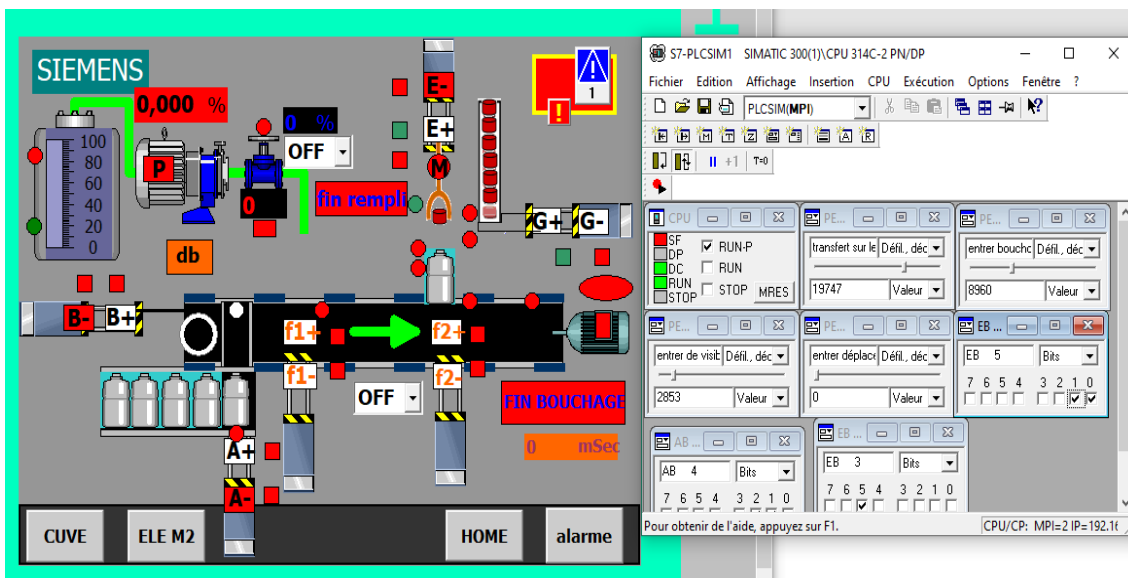
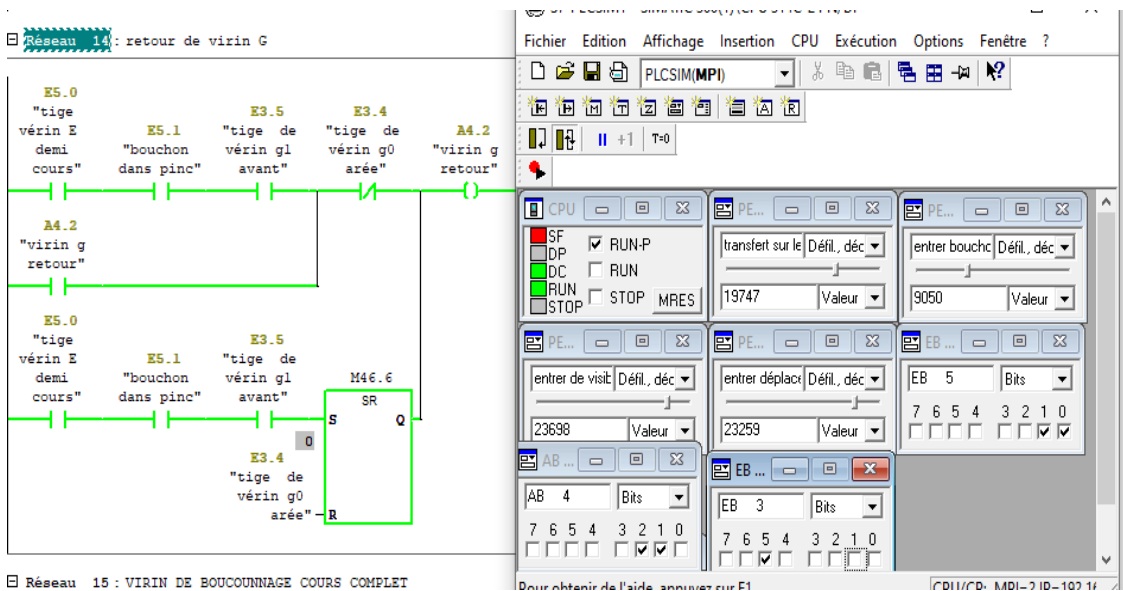


### A.16 Sortie vérin E pour donner le bouchonne

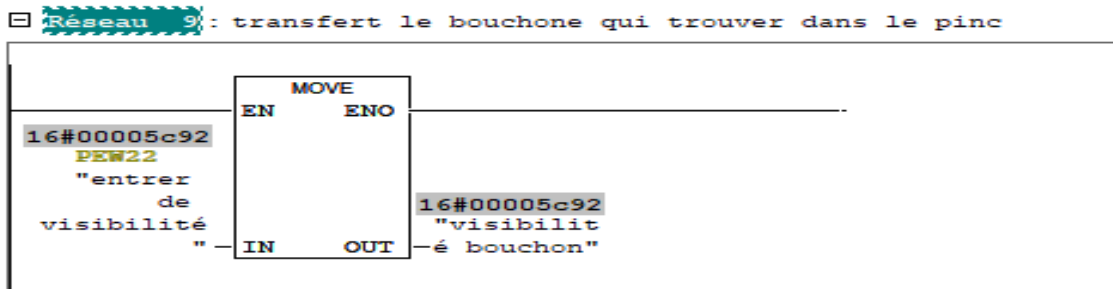
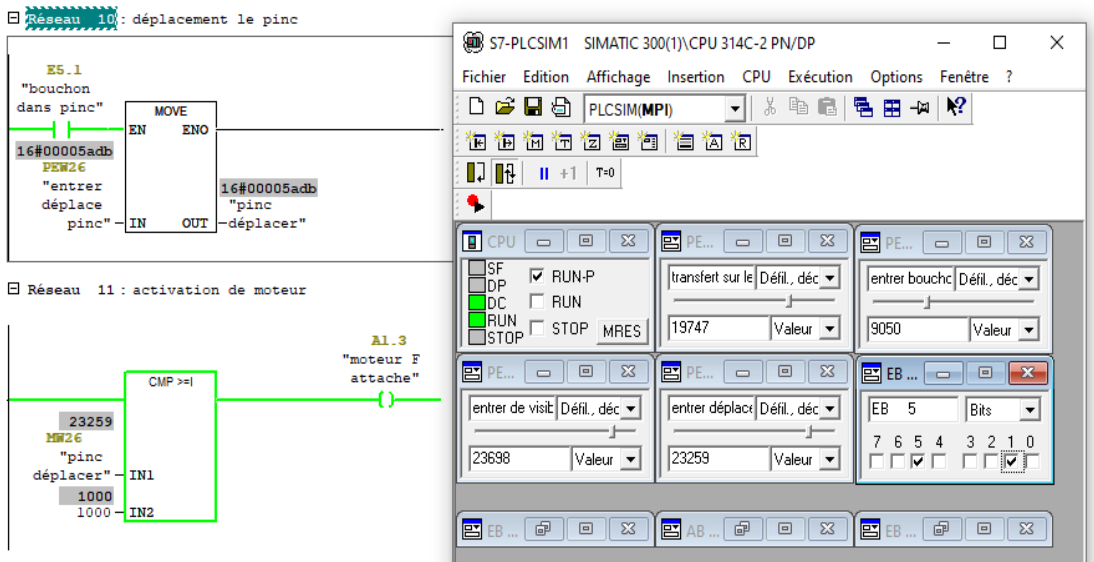
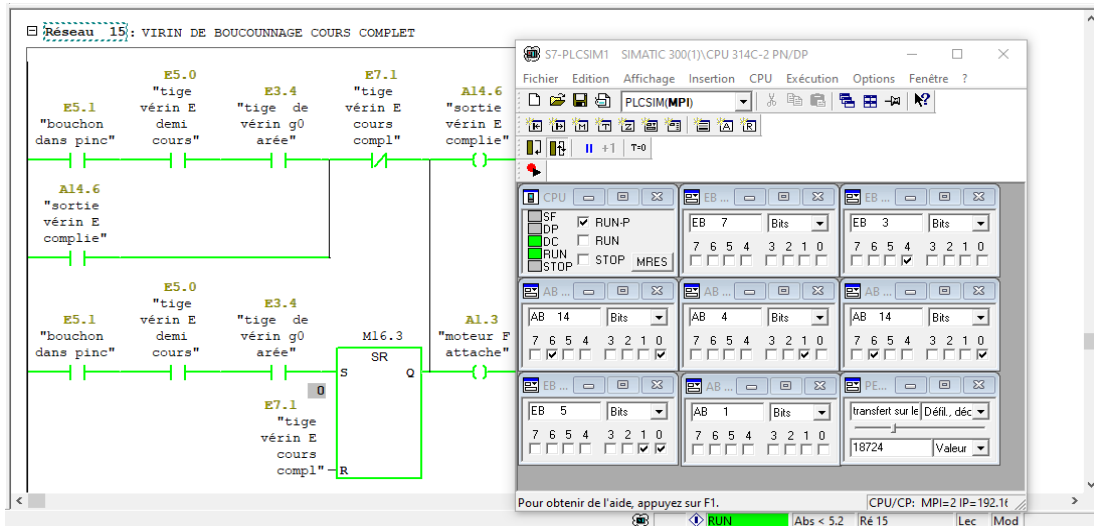


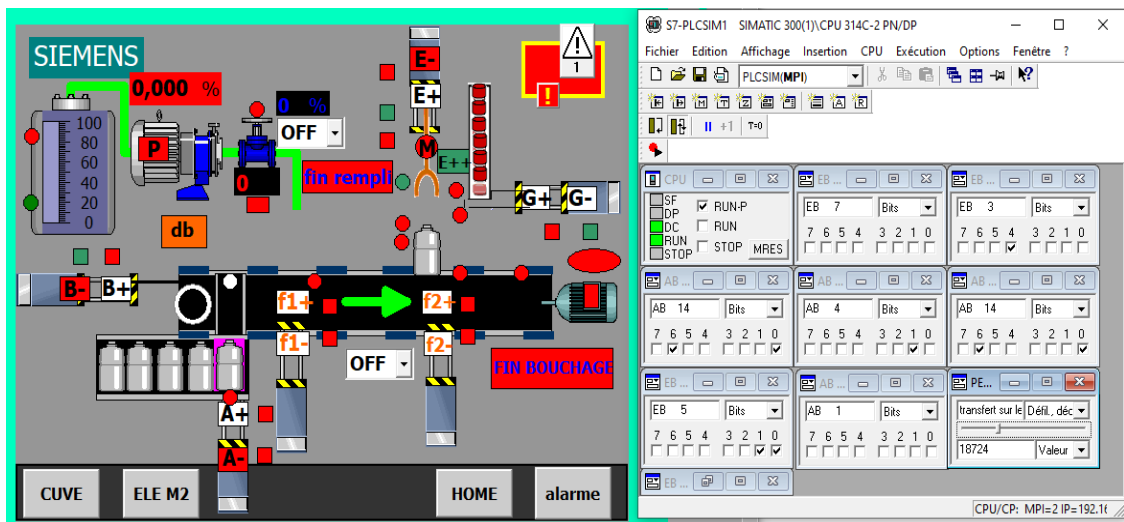


### A.17 Rentrer le vérin G

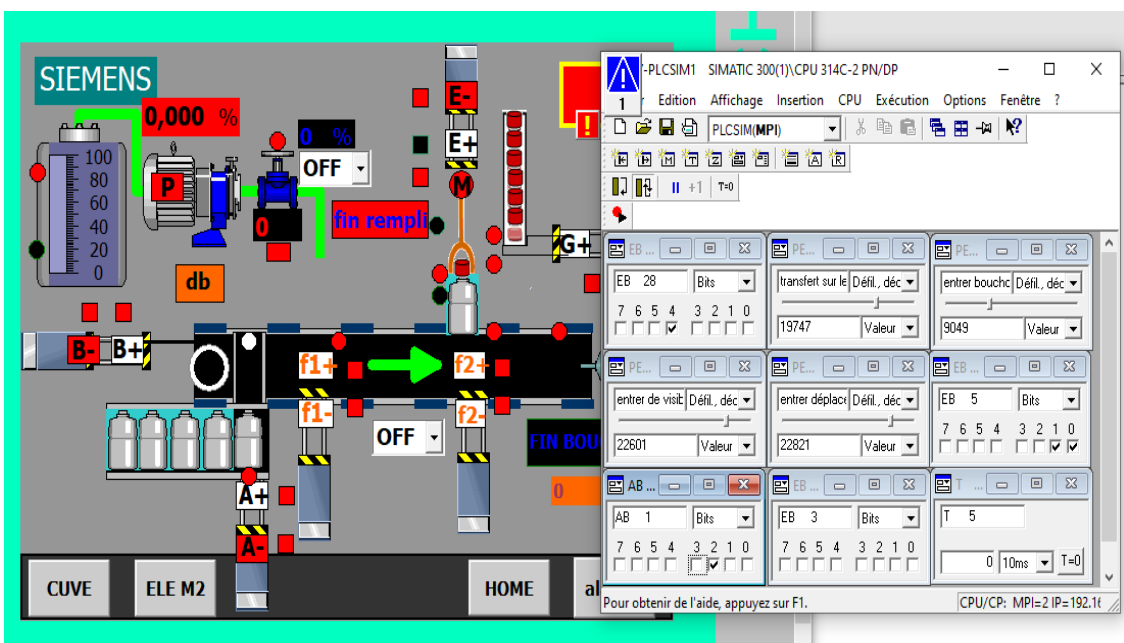
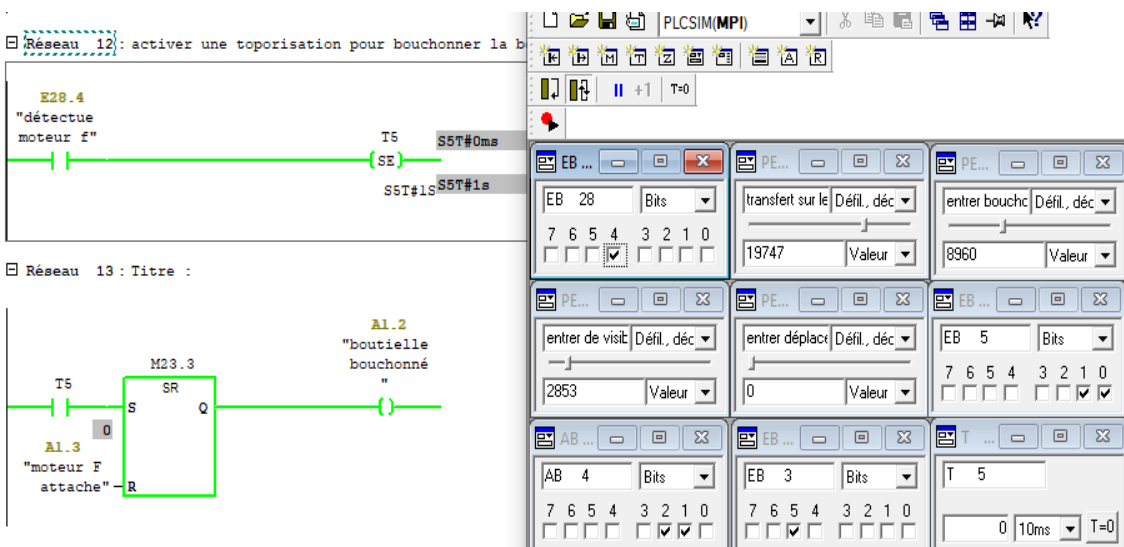


### A.18 Sortie vérin E complié et l'activation de moteur F

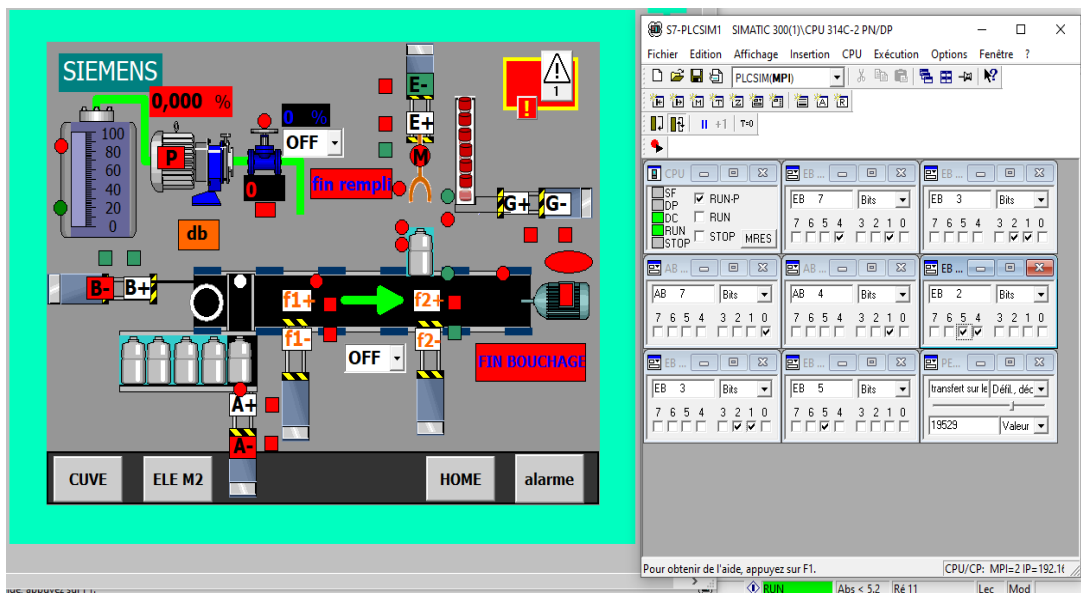
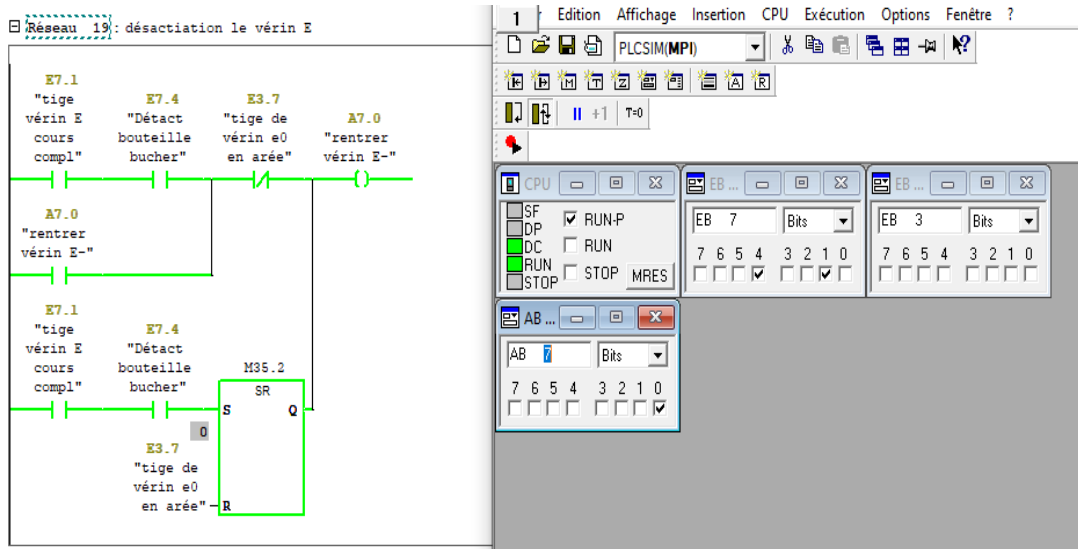




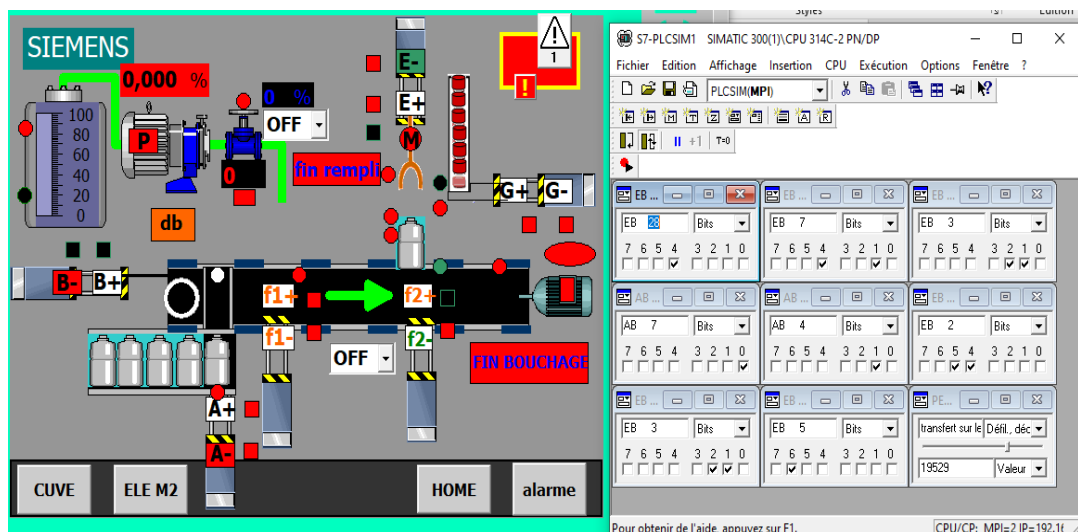
### A.19 A Certains temps à l'activation de moteur la bouteille bouchonnée



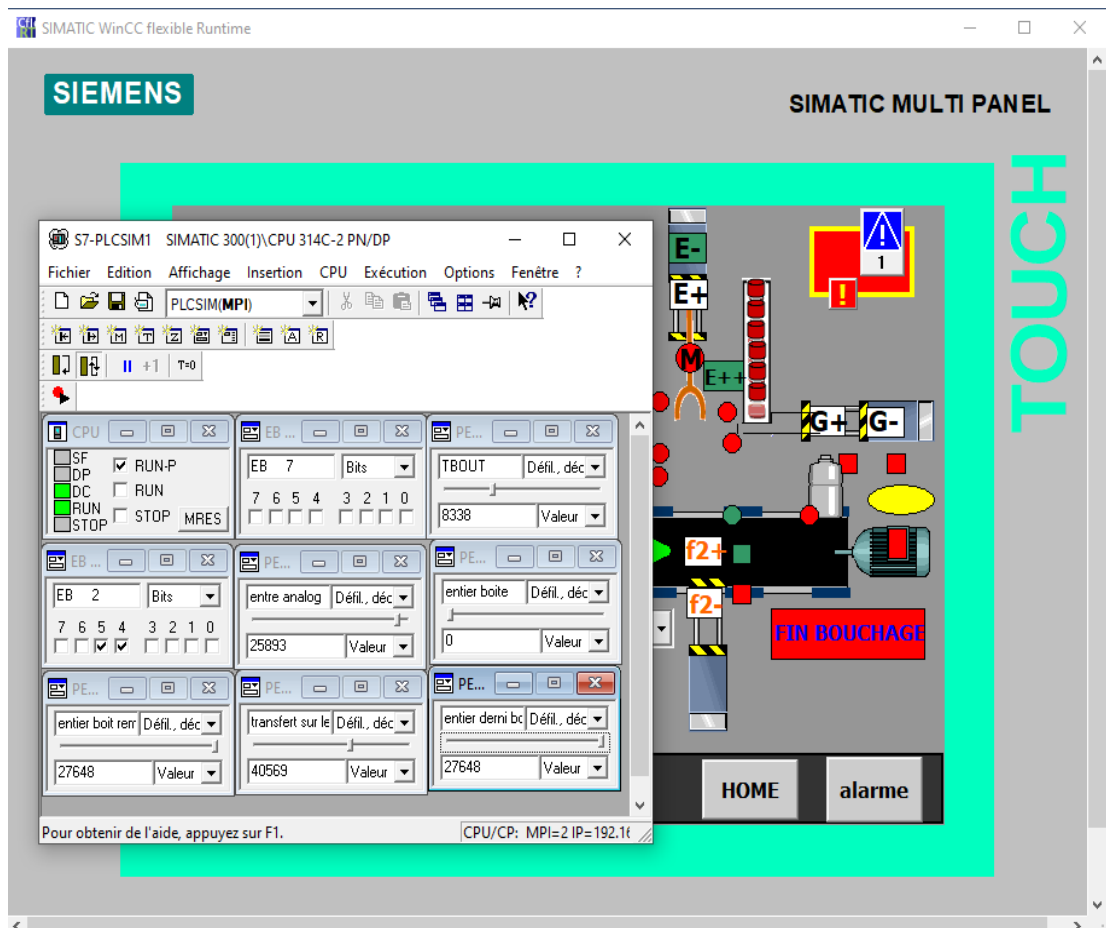
### A.20 Rentrer le vérin E



### A.21 Après le bouchonnage de bouteille rentrer le vérin F2 (Sortie f2-)







## ملخص

يتمثل العمل المقدم في هذه الرسالة في أتمتة وتشغيل خط إنتاج محطة تعبئة وتغطية تحتوي على ثلاث محطات. يتم التحكم في الأخير من وحدة تحكم PLC ، ولا سيما وحدة Simens S7-300 المعيارية.

يتم الحصول على بيانات العملية ومعالجتها بواسطة S7 SIMATIC 300 المدمج في برنامج steps7. كما نقوم أيضًا بإنشاء واجهة آلة بشرية (HMI) وعرض إنذار باستخدام برنامج WinCC flexible 2008 للمراقبة، لتسهيل مهمة التحكم والمراقبة والإنتاجية من قبل العامل.  
الكلمات المفتاحية:

النظام التلقائي ، برنامج الإشراف ، PLC ، Simens modulaire S7-300 PLC ، برنامج STEPS7 ، WinCC مرن

## Résumé

Le travail présenté dans ce mémoire, à automatiser, fonctionner une chaîne de production poste de remplissage et bouchonnage qui contient trois postes. Ce dernier est commandé à partir d'un automate API, en particulier le Simens modulaire S7-300.

Les données de processus sont obtenues et traitées par le SIMATIC S7-300 intégré à logiciel steps7. Nous créons encore une interface homme-machine (HMI) et une vue d'alarme avec un logiciel de supervision WinCC flexible 2008, pour faciliter la tâche de contrôle, de surveillance et la productivité de la chaîne de production par le travailleur.

**Mots clés :** Système automatique, Logiciel de supervision, API, automate Simens modulaire S7-300, logiciel STEPS7, WinCC flexible.

## Abstract

The work presented in this thesis, to automate, operate a filling and capping station production line which contains three stations. The latter is controlled from a PLC controller, in particular the modular Simens S7-300.

The process data is obtained and processed by the SIMATIC S7-300 integrated in steps7 software. We also create a human machine interface (HMI) and an alarm view with a WinCC flexible 2008 monitoring software, to facilitate the task of control, monitoring and productivity of china production by the worker.

**Keywords:** Automatic system, supervision software, PLC, Simens modular S7-300 PLC, STEPS7 software, WinCC flexible