

République Algérienne Démocratique et Populaire  
Ministère de l'enseignement Supérieur et de la Recherche Scientifique  
Université A. MIRA de Bejaia  
Faculté de Technologie  
Département de Génie Electrique



## *Mémoire de projet de fin d'études*

*En vue de l'obtention du Diplôme Master en Electronique*

*Option : Automatique*

*Thème*

# *Régulation d'une trémie par un pallier de trois vitesses*

**Présenté par**

M<sup>r</sup>. AIT SADI Nouredine

M<sup>r</sup>. DJERROUD M<sup>ed</sup> Ouali

**Dirigé par**

M<sup>me</sup>. BELLAHSENE Nora

M<sup>r</sup>. TOUAHRI Djamel

**Examineurs**

M<sup>r</sup>. Mendil.B

M<sup>r</sup>. Lehouche. H

*Promotion 2013-2014*

# ***Remerciements***

*Au terme de ce travail, nous tenons à exprimer nos vifs remerciements :*

*En premier lieu et surtout Dieu le tout puissant qui nous a donné la santé et guidé dans le chemin du savoir.*

*Notre enseignante et encadreur, Madame « Bellahsene Nora », pour nous avoir honorés en acceptant de diriger ce travail, pour ses orientations, son aide et ses conseils précieux.*

*Nos reconnaissances s'adressent également à toutes les personnes qui de près ou de loin ont soutenu nos efforts et ont contribué à ce que ce travail voit le jour et surtout notre encadreur au sein de la société DANONE DJURDJURA, Monsieur « Touahri Djamel »*

*Nous remercions, enfin, les membres de jury qui ont accepté d'évaluer ce travail.*

# ***Dédicaces***

*Je dédie ce travail*

*A mes très chers parents qui m'ont toujours poussé et motivé dans mes études. J'espère qu'ils trouveront dans ce travail toute ma reconnaissance et tout mon amour.*

*A mes chers frères et grande sœur pour leurs soutiens et leurs aides pendant mes études.*

*A mes grands parents paternelle et maternelle.*

*A mes tantes et oncles*

*A mes cousins et cousines*

*A mes amis, à mes copains de chambre et à tous ceux qui me sont chers.*

*Ait Sadi Noureddine*

# ***Dédicaces***

*Je dédie ce modeste travail*

*A mes très chers parents qui m'ont toujours poussé et motivé dans mes études. J'espère qu'ils trouveront dans ce travail toute ma reconnaissance et tout mon amour.*

*A mes chers frères pour leurs soutiens et leurs aides pendant mes études.*

*A mes grands parents.*

*A tous mes cousins et cousines.*

*A tous mes amis sans exception, à mes copains de chambre et à tous ceux qui me sont chers.*

***Djerroud Md Ouali***

# ***TABLE DES MATIERES***

Table de matières .....	i
Liste des figures .....	v
Liste des tableaux.....	vii
Abréviations .....	viii
Introduction générale.....	1
Problématique .....	2

## **Chapitre I : Description du procédé existant**

I-1 : Introduction .....	3
I-2 : Présentation de la machine de fabrication .....	3
I-3 : Fonctionnement de la machine.....	5
I-4 : Organigramme des opérations effectuées par la machine .....	6
I-4-1: Explication des différents éléments de l'organigramme.....	7
Dérouleuse bobine plastique .....	7
Tirage .....	7
Chauffage.....	8
Formage .....	9
La trémie .....	10
Dosage.....	10
Bobine opercule .....	11
Marquage et encrage .....	11
Soudage.....	12
Découpage.....	12
Tirage en coupe déchets .....	13
I-5 : Processus globale du mécanisme de production .....	13
I-6 : Conclusion .....	16

## **Chapitre II : Généralités sur PL7& InTouch**

II-1 : Présentation du logiciel de programmation PL7 _pro.....	17
II-1-1 : Définition.....	17
II-1-2 : Interface générale du logiciel .....	17
II-1-3 : Création d'une nouvelle application.....	17

II-1-4: Configuration matérielle et logicielle.....	19
1- Configuration matérielle.....	19
2- Configuration logicielle.....	19
II-1-5 : Gestion des adresses et des variables .....	20
II-1-6 : Programmation sous le PL7.....	21
1- Programmation des diagrammes à relais.....	21
2- Programmation des Grafcets.....	23
II-2-7: Transfert des programmes vers l'automate .....	24
II-2 : Présentation du logiciel de supervision InTouch .....	25
II-2-1: Introduction .....	25
II-2-2: Objectifs .....	25
II -2-3: Environnement D'InTouch.....	26
1- Application Manager (gestionnaire d'application).....	26
2- Window Maker.....	26
3- Window Viewer.....	26
II -2-4: Création d'une application .....	26
II -2-5: Création d'une fenêtre synoptique dans Window Maker.....	26
II -2-6: Création d'objets graphiques dans une fenêtre .....	27
1- Les outils de création d'objet.....	27
2- Outils "Wizard".....	28
II-2-7: Définition des tagnames (base de données des variables).....	28
II-2-8 : Paramétrage d'un élément graphique.....	29
II-2-9 : Window Viewer.....	31
II-3: Conclusion .....	31

### **Chapitre III : Etude de l'interface**

III-1 : Notion sur la régulation automatique.....	32
III-1-1: Définition .....	32
III-1-2: Objectifs de la régulation Automatique .....	32
III-1-3: Notion de Système .....	32
III-1-4: Notion de Boucle Ouverte/Fermée.....	33
1- Système boucle ouverte (Manuel).....	33
2- Système boucle fermée (Automatique).....	33
III-1-5 : Types de régulation automatique .....	33

1- Régulation tout ou rien (TOR).....	33
2- Régulation analogique.....	34
3- Régulation numérique.....	34
III-1-6: Critères de performance d'une régulation.....	34
III-2: Les éléments de la boucle de régulation.....	35
III-2-1: Le Silometre FMC 420(transmetteur).....	35
Fonctionnement.....	35
III-2-2: Le Variateur de vitesse EURO THERM 601 (convertisseur).....	36
Description.....	36
Paramétrage du variateur.....	36
III-2-3: Une pompe hydraulique.....	40
Principe de fonctionnement.....	40
III-2-4: La trémie.....	41
III-2-5: Capteur analogique.....	41
III-2-6: L'automate programmable Industriel API (régulateur).....	42
III-2-6-1: Définition.....	42
III-2-6-2: Structure d'un système automatisé.....	42
III-2-6-3: Domaines d'emploi des automates.....	43
III-2-6-4: Nature des informations traitées par l'automate.....	43
III-2-6-5: Architecture des automates.....	43
1- Aspect extérieur.....	43
2- Structure interne.....	45
III-2-6-6 : Fonctions réalisées par l'API.....	46
III-2-6-7: Traitement du programme par l'API.....	47
III-2-6-8 : Différentes langages de programmation.....	48
1- Les langages textuels.....	49
2- Les langages graphiques.....	49
III-2-6-9 : Critères de choix d'un automate.....	49
III-3: Conclusion.....	50

## **Chapitre IV : Résultat et simulation**

IV- 1 : Introduction.....	51
IV-2 : Solution apportée.....	51
IV-3 : Partie matérielle.....	52

IV-3-1 : Section électrique.....	52
IV-3-2 : Section paramétrage.....	53
IV-4: Partie logicielle.....	54
IV-4-1: Organigrammes.....	54
1-Organigramme N°1.....	54
2-Organigramme N°2.....	55
3- Organigramme N°3.....	56
IV-4-2: Grafcet et Programme sous PL7.....	57
IV-5: Supervision de la production .....	58
IV-6: Interprétation des résultats.....	59
1- Premier cas.....	59
2- Deuxième cas.....	59
3- Troisième cas.....	59
4 - Quatrième cas... ..	59
IV-7 : Conclusion.....	59
Conclusion générale .....	60
Bibliographie.....	I
Annexes .....	II
Annexe.A.....	II
Annexe.B.....	VII



# Liste des figures

## Chapitre I : Description du procédé existant

Figure I-1 : Schéma générale de la machine de conditionnement.....	4
Figure I-2 : Organigramme du fonctionnement de la conditionneuse thermoformage.....	6
Figure I-3 : Déroulement plastique .....	7
Figure I-4 : Tirage de la bobine .....	8
Figure I-5 : Boite de chauffage.....	9
Figure I-6 : Les étapes de formage.....	9
Figure I-7 : Forme de la trémie .....	10
Figure I-8 : Dosage des pots.....	11
Figure I-9 : Partie soudage .....	12
Figure I-10 : Partie découpage .....	13
Figure I-11 : Processus du mélange lait caillé et de la crème sucrée .....	14
Figure I-12 : Vue d'un seul tank .....	14
Figure I-13 : Aperçu global sur la circuiterie du processus .....	15

## Chapitre II : Généralités sur PL7 & Intouch

Figure II-1 : Interface générale du logiciel PL7_pro.....	17
Figure II-2 : Création d'une nouvelle application .....	17
Figure II-3: Fenêtre de navigation de l'application dans PL7-Pro.....	18
Figure II-4 : Présentation des modules de l'automate.....	19
Figure II-5 : Aperçu de la fenêtre de configuration logiciel.....	20
Figure II-6 : Répertoire des variables de la fenêtre de navigation de l'application .....	20
Figure II-7 : Fenêtre de programmation en diagramme à relais.....	21
Figure II-8 : Exemple de fenêtre de programmation en Grafcet .....	23
Figure II-9 : Demarrage InTouch .....	25
Figure II-10 : Fenêtre d'application .....	26
Figure II-11 : Création d'une nouvelle fenêtre dans Window maker.....	27
Figure II-12 : Fenêtre des utiles Wizard.....	28
Figure II-13 : Gestion des adresses « tagnames » .....	29
Figure II-14 : Paramètre des éléments graphique.....	29

### **Chapitre III : Etude de l'interface**

Figure III-1 : Notion de système .....	32
Figure III-2 : Système en BO .....	33
Figure III-3 : Système en BF.....	33
Figure III-4 : Schéma général de la boucle de régulation. ....	35
Figure III-5 : Le silometre FMC 420.....	35
Figure III-6 : Variateur de vitesse EURO THERM 601 .....	36
Figure III-7 : Vue du variateur sans le capot des borniers .....	37
Figure III-8 : Pompe hydraulique PCM.....	40
Figure III-9 : La forme de la trémie .....	41
Figure III-10 : Le capteur analogique .....	41
Figure III-11 : Structure d'un système automatisé.....	42
Figure III-12 : L'automate Schneider TSX 57452.....	44
Figure III-13 : Structure interne d'un API .....	45
Figure III-14 : Cycle d'un API.....	47

### **Chapitre IV : Résultat et simulation**

Figure IV-1 : Les intervalles de la trémie.....	51
Figure IV-2 : Partie commande du système.....	52
Figure IV-3 : Organigramme du circuit général.....	54
Figure IV-4 : Organigramme du système déjà existant.....	55
Figure IV-5 : Organigramme du système modifier.....	56
Figure IV-6 : La macro étapes avec Ladder .....	57
Figure IV-7 : Programme principale sous PL7 .....	57
Figure IV-8 : Vus du système à l'arrêt.....	58
Figure IV-9 : Vitesse pré réglée N°3.....	58
Figure IV-10 : Vitesse pré réglée N°2.....	58
Figure IV-11 : Vitesse pré réglée N°1 .....	58

# Liste des tableaux

## Chapitre II : Généralités sur PL7 & Intouch

Tableau II-1 : Les différents outils à la réalisation d'une programmation avec Ladder.....22

Tableau II-2 : Les différents outils à la réalisation d'une programmation avec Grafcet.....24

## Chapitre III : Etude de l'interface

Tableau III-1 : Les différents paramètres du variateur de vitesse.....39

## Chapitre IV : Résultat et simulation

Tableau IV-1 : Les différentes vitesses.....51

Tableau IV- 2 : Les adresses de l'automate.....52

Tableau IV-3 : Les différentes combinaisons.....52

Tableau IV-4 : Les vitesses préréglées.....53

# Abréviations

**NEP** : Nettoyage en place

**API** : Automate programmable industriel

**E/S** : Entrées /sorties

**IHM** : Interface homme-machine

**BO** : Boucle ouverte

**BF** : Boucle fermé

**TOR** : Tout ou rien

**μP** : Micro processeur

**IL**: Instruction List

**ST**: Structured Text

**LD**: Ladder Diagram

**FBD**: Function Bloc Diagram

**SFC** : Sequential Function Char

**Pr**: Vitesse Préréglée

# Introduction Générale

## **Introduction générale**

L'automatisme est une discipline importante et nécessaire dans tous les secteurs industriels. L'automatisation de toute unité de production augmente la productivité et améliore la qualité du produit. Les automates programmables industriels représentent l'élément important de la chaîne automatisée, car ils assurent une meilleure flexibilité et facilitent la maintenance. Une automatisation performante assure en plus d'un fonctionnement fiable de l'installation industrielle, la détection de toute anomalie éventuelle. La diversité des processus industriels nécessite des connaissances sur l'aspect processus et différentes technologies du domaine de l'automatisme. Les automatismes câblés qui ont précédé les automates programmables industriels sont en développement en même temps que ceux des systèmes de production. Ils sont compétitifs, toute fois leur structure présente des inconvénients relatifs à leur maintenance et leur configuration.

L'objectif de notre travail est de réaliser une régulation avec un pallier de trois vitesses en remplaçant de la commande tout ou rien (TOR) existante au niveau de l'entreprise DANONE-DJURDJURA ALGERIE (DDA).

Le mémoire est divisé en quatre chapitres, après une introduction générale.

Le premier chapitre est consacré à la présentation du système existant dans l'entreprise, le second chapitre porte sur les logiciels de programmation.

Dans le troisième chapitre, on s'intéresse à l'étude de l'interface et dans le dernier chapitre nous donnerons les différents Graficets de notre programme et les résultats de simulation, et nous terminerons par une conclusion.

Problématique

## **Problématique**

Le bon fonctionnement de la machine conditionnement ne se fait pas sans installer des équipements nécessaires qui assurent son fonctionnement normal.

N'importe quel équipements de cette machine joue un rôle bien déterminé et très important et c'est le cas de la pompe hydraulique qui assure non seulement l'acheminement du produit (pâte fraîche) vers la machine.

La commande de cette pompe se fait avec une régulation tout ou rien TOR mais cette commande fait en sorte de démarrer et d'arrêter la pompe chaque fois que la trémie atteint l'une de ses extrémités c'est-à-dire le maximum ou minimum.

Par conséquent la pompe démarre et s'arrête environ 300 fois par heure ce qui fait subir des contraintes mécaniques et électriques sur la pompe et réduit sa durée de vie et entrave ainsi la production.

Alors pour remédier à ce problème c'est-à-dire avoir une solution qui permet le bon fonctionnement de la pompe sans augmenter le coût, nous avons opté pour une modification qui consiste à modifier la régulation tout ou rien en régulation avec un palier de trois vitesses à l'aide de l'automate Schneider de telle sorte que la pompe fonctionne dans le régime permanent durant toute la production (éliminer la partie transitoire).



# Chapitre I

## Description du procédé existant

## **I-1 : Introduction**

La production des produits laitiers est en progression à grâce de l'augmentation de la consommation, celle-ci est due à une bonne qualité organoleptique et nutritionnelle qui caractérise le lait et ses dérivés.

Pour satisfaire cette forte consommation, les laiteries ont développé des nouvelles techniques en mettant à la disposition du consommateur des gammes des produits laitiers adaptés aux habitudes alimentaires actuelles

Parmi les leaders des industries laitières dans le monde entier, on cite le groupe français DANONE, fondé en 1966 par Antoine Riboud (voir l'annexe A), qui possède une unité de production à Akbou en partenariat avec DJURDJURA, et où on a effectué notre stage pratique afin d'enrichir nos connaissances en industrie et de découvrir le monde de travail.

## **I-2 : Présentation de la machine de fabrication**

La machine est une conditionneuse thermoformage (fabrication du fromage à pâte fraîche) de type **TREX** du leader international **ERCA** (Figure I-1) avec une capacité de **20800 pots** par heure elle est composée de certains éléments qui assurent le bon conditionnement du yaourt qui sont :

1. Un dérouleur bobine plastique
2. Une chaîne de tirage
3. Un aspirateur
4. Une boîte de chauffage
5. Une presse de fromage
6. Une trémie
7. Un doseur
8. Un couvercle (dérouleur opercule, marquage et encrage)
9. La presse de soudure
10. Une presse de découpe
11. Tapis de sortie
12. Tirage en coupe déchets

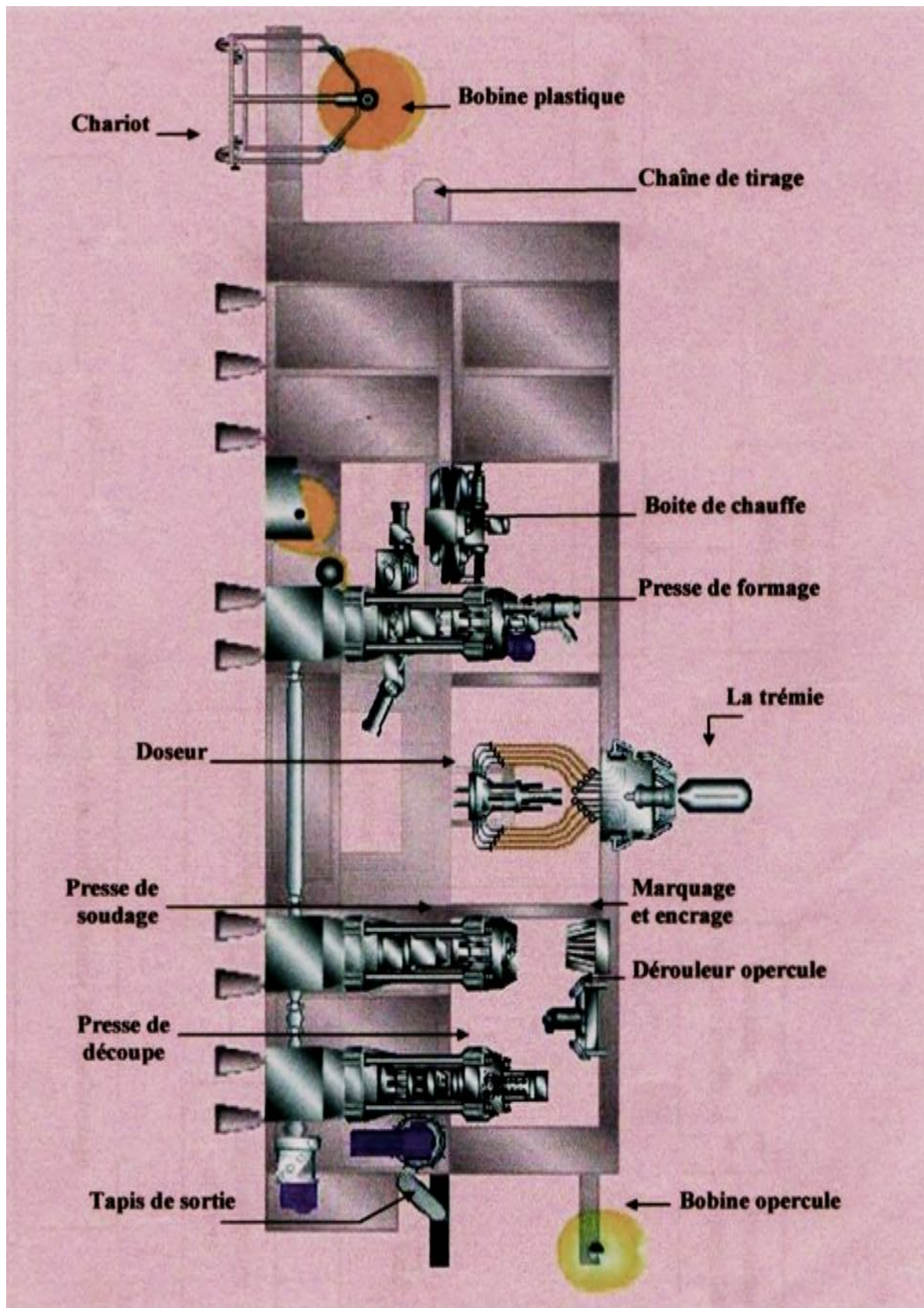


Figure I-1 : Schéma générale de la machine de conditionnement.

### **I-3 : Fonctionnement de la machine**

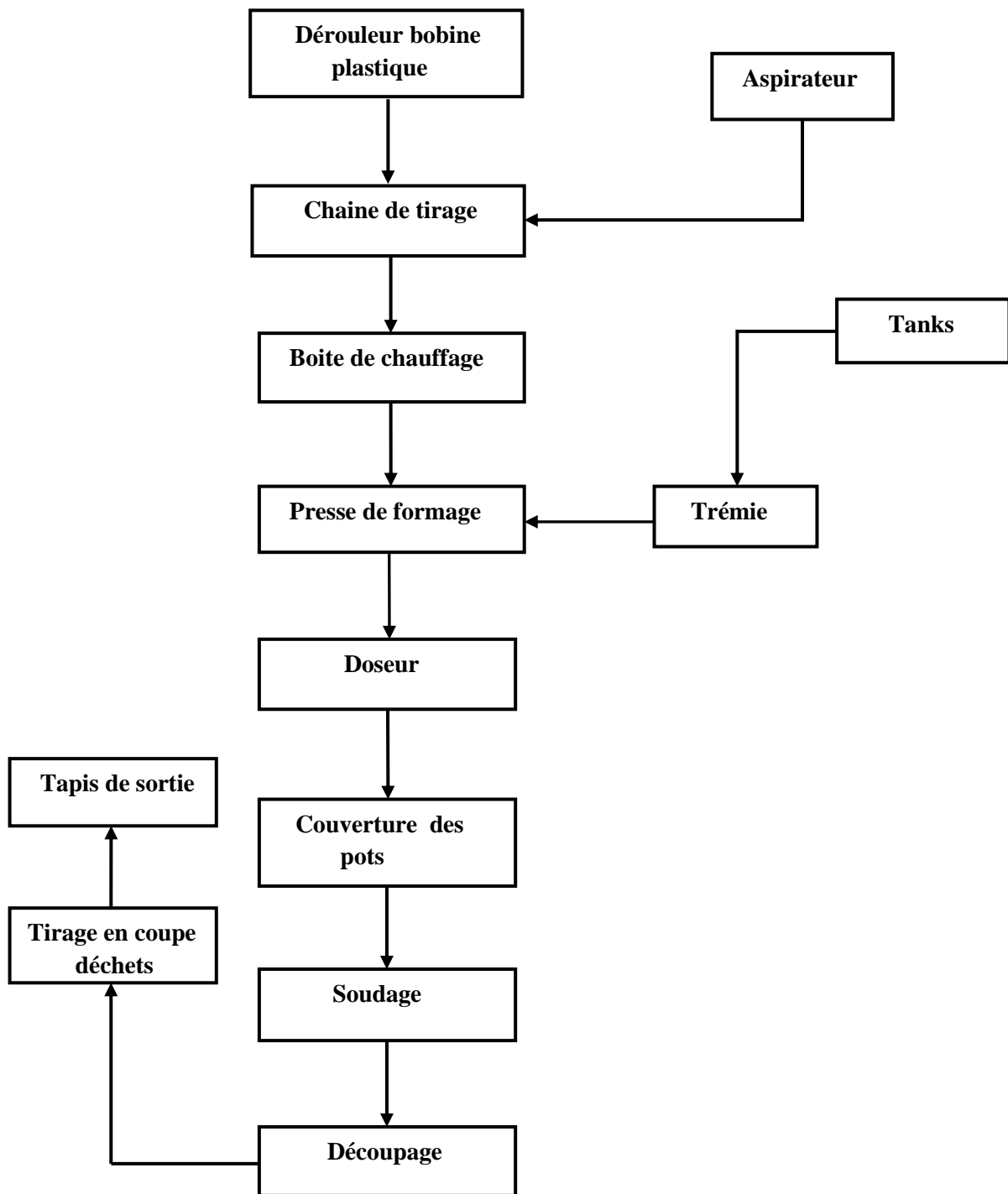
La bande plastique est transformée à l'entrée de la machine puis acheminé vers les différents outillages à l'aide des chaînes de tirages.

Après la stérilisation de la bande plastique par l'aspirateur, elle passe par la boîte de chauffage pour l'amener à sa température de thermoformage, ensuite la bande chauffée s'introduit dans la presse de formage. Les pots formés seront remplis par le doseur.

La bobine opercule est déroulée de telle façon à ce que le tirage fasse avancer les pots et l'opercule au même temps, pour une bonne opération de soudage, et ça après marquage et encrage (datage) de l'opercule, puis la presse découpe le plastique à l'aide des couteaux afin d'avoir à la sortie des séries à 8 pots. Et le tirage des déchets vers le recyclage [1].

Enfin, les pots découpés sont dégagés vers le tapis de sortie où un opérateur les attend pour les mettre dans des cagettes, ensuite l'amener à la chambre de stockage.

**I-4 : Organigramme des opérations effectuées par la machine**



**Figure I-2 :** Organigramme du fonctionnement de la conditionneuse thermoformage.

## I-4-1: Explication des différents éléments de l'organigramme

### Dérouleur bobine plastique

#### ✓ *Rôle*

Il déroule la bobine plastique d'une façon à ce que le tirage soit constant quelque soit le diamètre ou le poids de la bobine.

#### ✓ *Fonctionnement*

La bobine plastique est placée sur un chariot mis en position par un système de centrage et maintenue à l'aide du volant de verrouillage. Quand les pinces ont fini de tirer la bande plastique et que la presse de formage est fermée, le vérin pneumatique agit sur le bras pour dérouler la bobine.

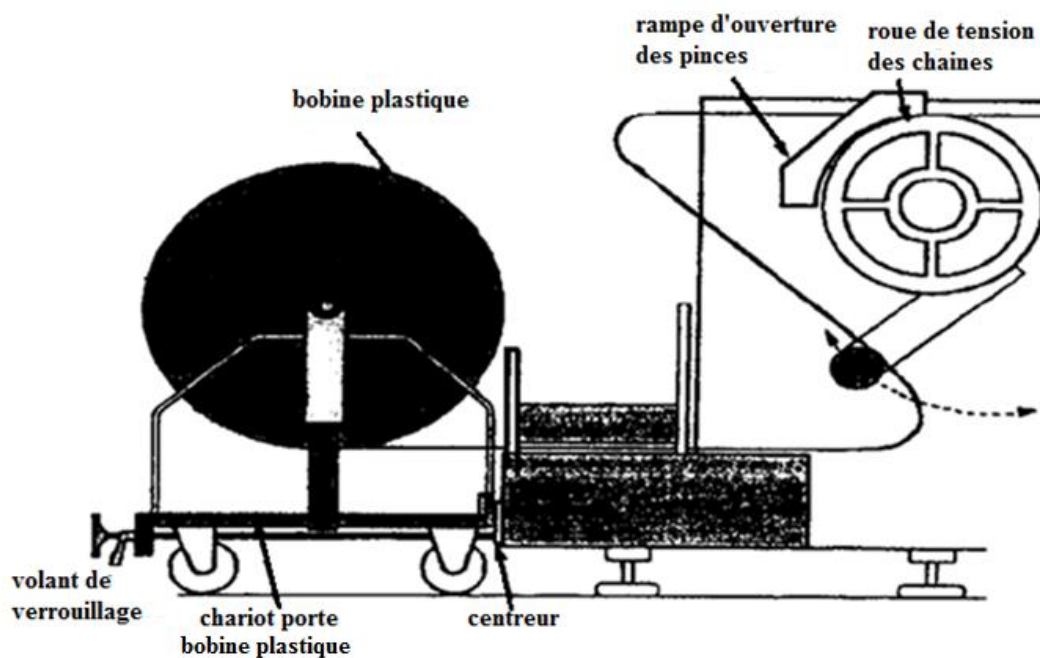


Figure I-3 : Dérouleur bobine plastique.

### Tirage

#### ✓ *Rôle*

Il permet de déplacé a chaque cycle la bonde plastique d'une valeur appelée le pas est ainsi la transfère de l'entrée vers les différents outillages.

### ✓ *Fonctionnement*

Un ensemble de pinces fixes et mobiles s'ouvrent et se ferment de manière alternative. Les pinces mobiles sont montées sur un chariot de tirage qui à chaque cycle effectuent un mouvement d'aller retour.

Dans le sens du défilé, les pinces mobiles sont fermées et entraînent ainsi la bonde plastique (les pinces fixes ouvertes) au retour les pinces fixes sont fermées permettant le maintien de la bonde plastique et la conservation du pas (les pinces mobiles sont ouvertes).

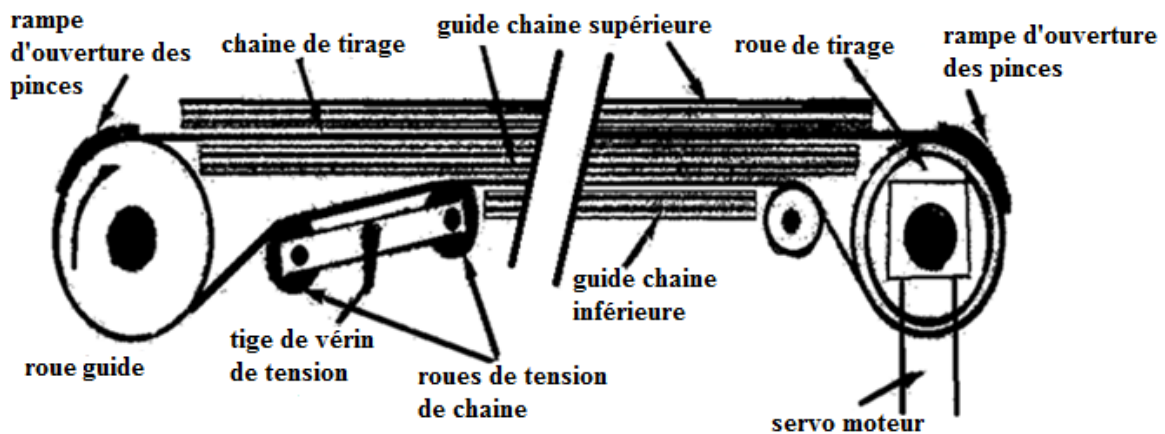


Figure I-4 : Chaîne de tirage.

## Chauffage

### ✓ *Rôle*

Il est assuré par une boîte de chauffage qui amène la bande plastique à sa température de thermoformage et seule la partie à forme est chauffée.

### ✓ *Fonctionnement*

Des le démarrage de la machine les vérins de commandes actionnent les parties supérieur et inférieur de la boîte. La pression de contact sur la bande est définie par la compression des ressorts montés sur les plaques de chauffe supérieures.



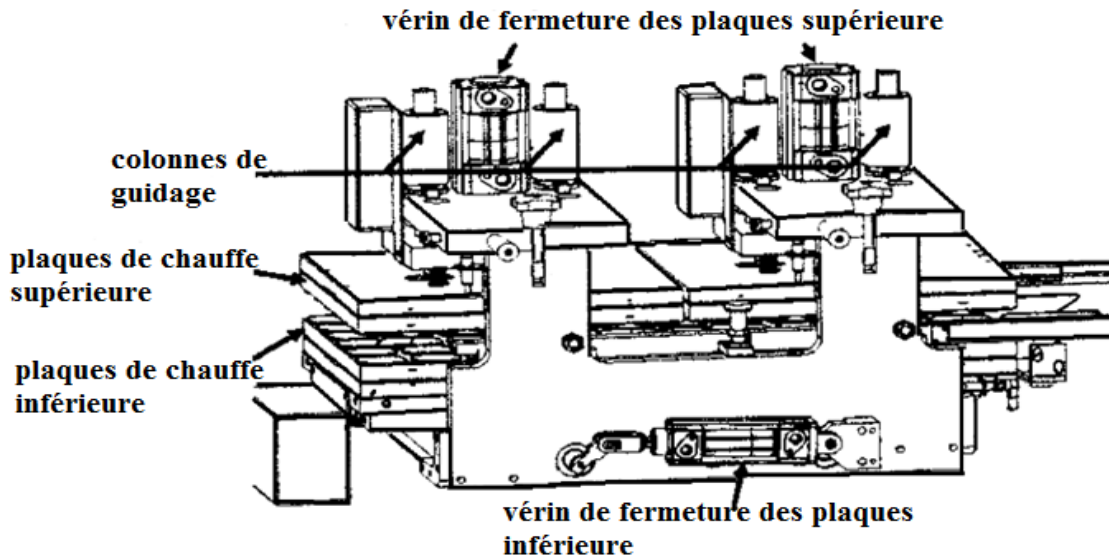


Figure I-5 : Boite de chauffage.

### Formage

✓ *Rôle*

Etire le plastique chaud afin de lui donne la forme des pots.

✓ *Fonctionnement*

Initialement la presse est ouvert, le plastique arrive grâce au tapis de déroulement, une fois le plastique pincé (pinces fixes fermée), la presse se ferme puis les poinçons descendent et éjectent l'air emprisonné par le moule. Une fois en place ils libèrent un air chaud de formage. Avant l'ouverture du moule les poinçons remontent forment ainsi les pots.

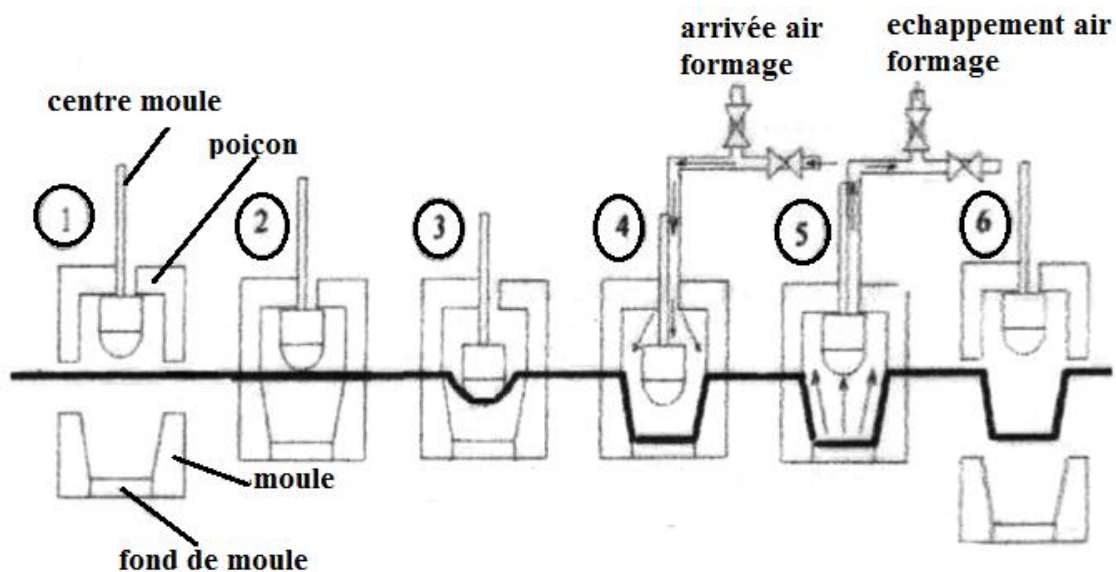


Figure I-6 : Les étapes de formage.



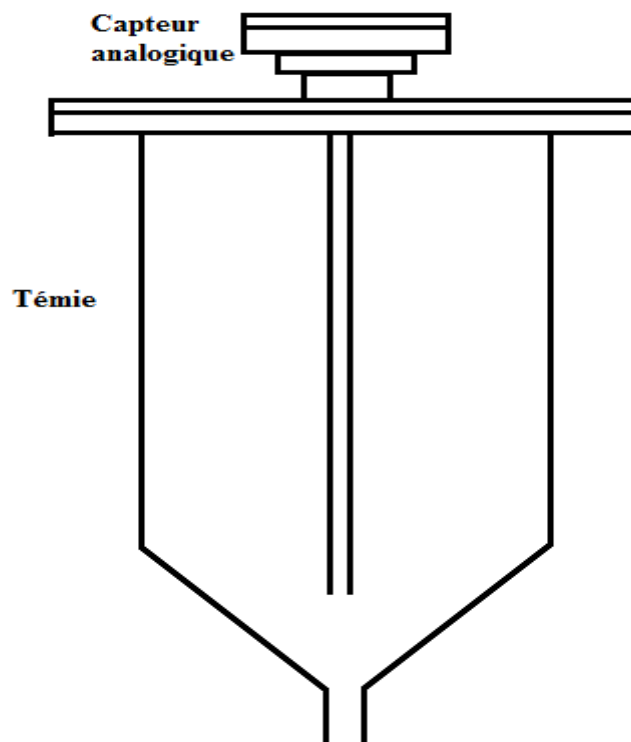
## La trémie

### ✓ *Rôle*

Après le stock de yaourt dans des tanks de stockages, il est transféré à la trémie via une pompe à travers une conduite. La trémie alimente le doseur de la machine qui fait des cycles de dosage de produit dans les pots à la quantité voulue, donc la disponibilité du produit dans la trémie est nécessaire pour avoir un bon dosage et pour avoir une cadence de production fixe.

### ✓ *Fonctionnement*

Le niveau de la trémie est détecté par un capteur analogique qui envoie un signal à l'automate. Si le niveau est bas la pompe démarre pour envoyer le produit et si le niveau est haut il arrête la pompe.



**Figure I-7** : Forme de la trémie.

## Dosage

### ✓ *Rôle*

Il fait des cycles de dosage du produit dans les pots à la quantité voulue. Réglable sur le bouton réglage dose sur le pupitre dose.

### ✓ *Fonctionnement*

La tête du doseur descend et les pistons sont en position basse alors que les membranes de sortie s'ouvrent, pendant la phase de chargement les membranes s'ouvrent pour remplir la chambre du doseur, et durant la phase d'injection du produit est injecté à travers la descente des pistons une fois cette étape terminée, la tête du doseur s'ouvre pour remplir les pots.

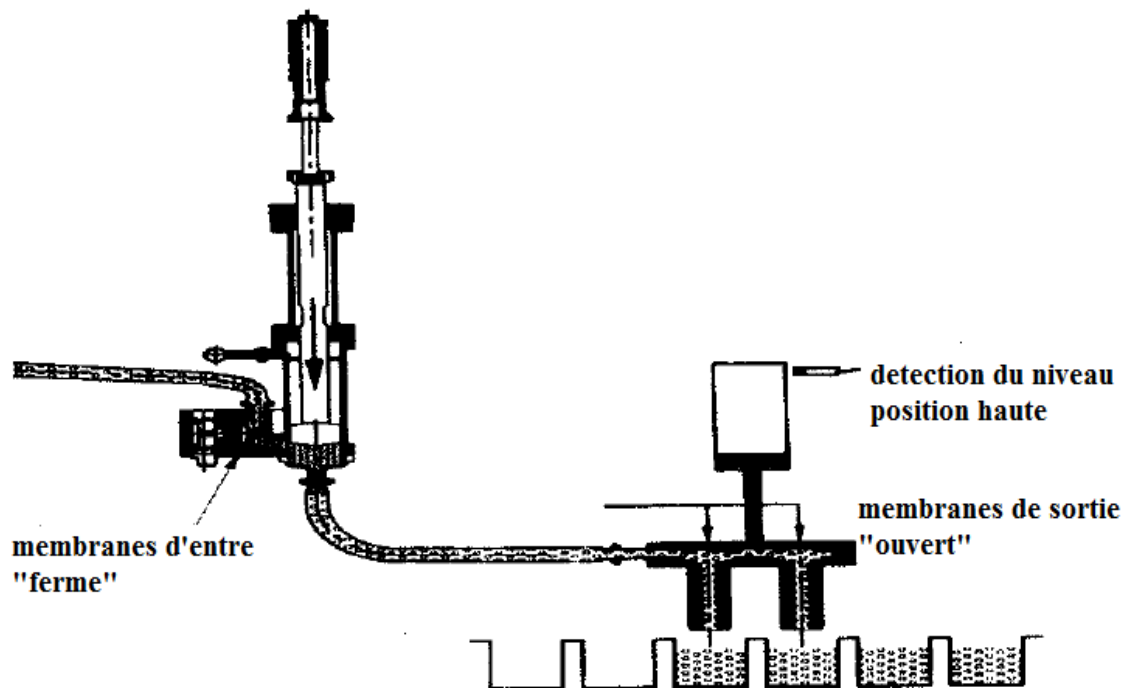


Figure I-8: Dosage des pots.

### **Bobine opercule**

#### ✓ *Rôle*

Déroule la bobine d'une façon à ce que l'effort de tirage du plastique soit toujours le même

#### ✓ *Fonctionnement*

Colle le début de la nouvelle bobine sur la moitié de scotch puis colle la fin de l'ancienne bobine en superposant les pans et à la fin rabattre les deux extrémités du scotch.

### **Marquage et encrage**

#### ✓ *Rôle*

Il réalise l'inscription de la date sur l'opercule.

#### ✓ *Fonctionnement*

Actionne par le vérin supérieur le rouleau encreur se déplace une fois vers la plaque porte caractère pour y déposer de l'encre.

Grace au vérin inférieur, la plaque porte caractères monte et pose ces inscriptions sur l'opercule puis redescend.

Le rouleau encreur revient sur sa position initiale tout en déposant de l'encre sur la plaque porte caractère.

## Soudage

### ✓ *Rôle*

Il est assuré par une presse de soudage qui soude l'opercule revêtu d'une laque thermocollante sur la bonde plastique plate ou formée au moyen d'un ensemble électrode contre électrode.

### ✓ *Fonctionnement*

L'électrode est amenée à sa température par la résistance la sonde de température permet la régulation thermique pour effectuer la thermo soudure.

A chaque cycle de la machine, l'électrode et la contre électrode sont formées, et la soudure est effectuée.

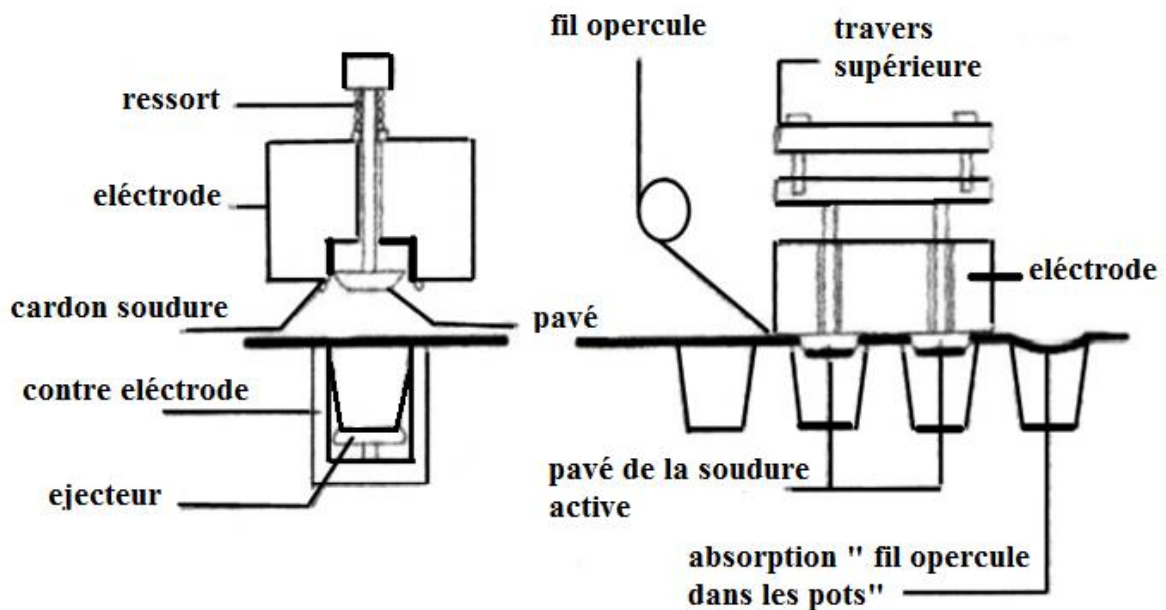


Figure I-9 : Partie soudage.

## Découpage

### ✓ *Rôle*

Il découpe les pots avec des outils (couteaux) montées sur la presse de découpe.

### ✓ *Fonctionnement*

Le mouvement de découpe rectiligne alternatif est donné par les quatre billes verticales qui sont obtenue par la rotation du moteur, transmis à l'aide d'un système bille manivelle. Le moteur transmet son mouvement au maneton qui parcourt une trajectoire circulaire avec un mouvement uniforme, et à l'aide de l'axe maneton la bille reçoit un mouvement rectiligne alternatif, cette dernière et par l'intermédiaire des deux axes et les quartes galets qui se déplacent suivant la forme des romps, transforme le mouvement transversale alternatif on un mouvement vertical alternatif qui fait monter la table et la met en contact avec la partie fixe (outils de découpe) pour avoir le découpage des pots.

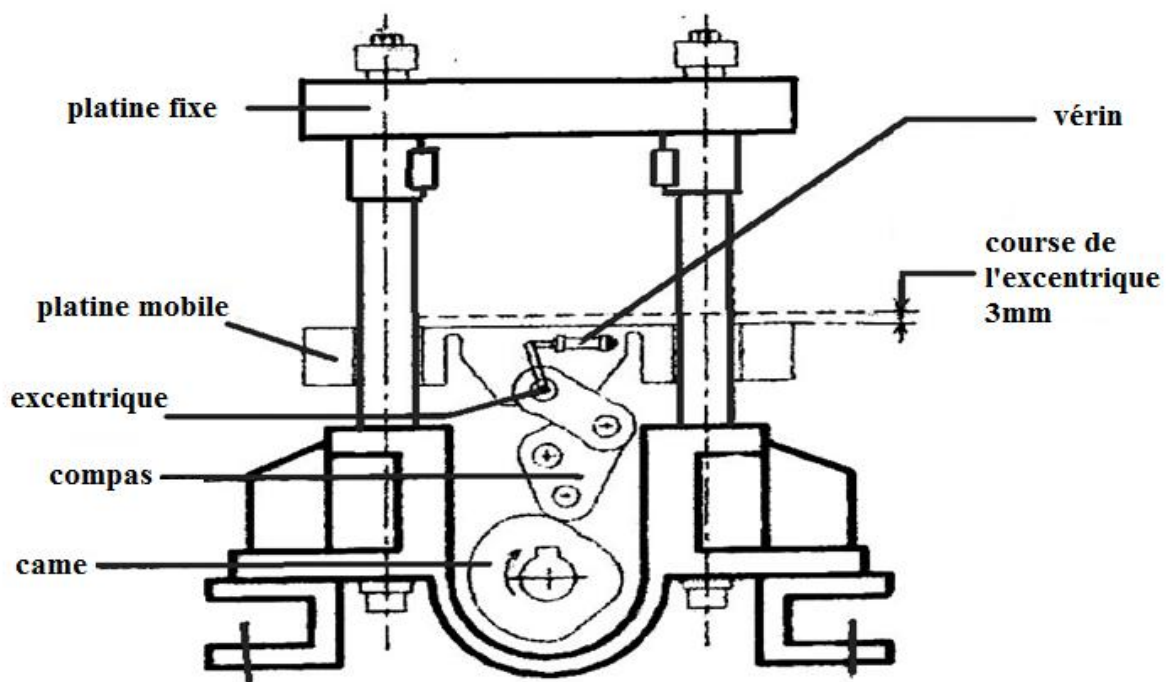


Figure I-10 : Partie découpage.

### Tirage en coupe déchets

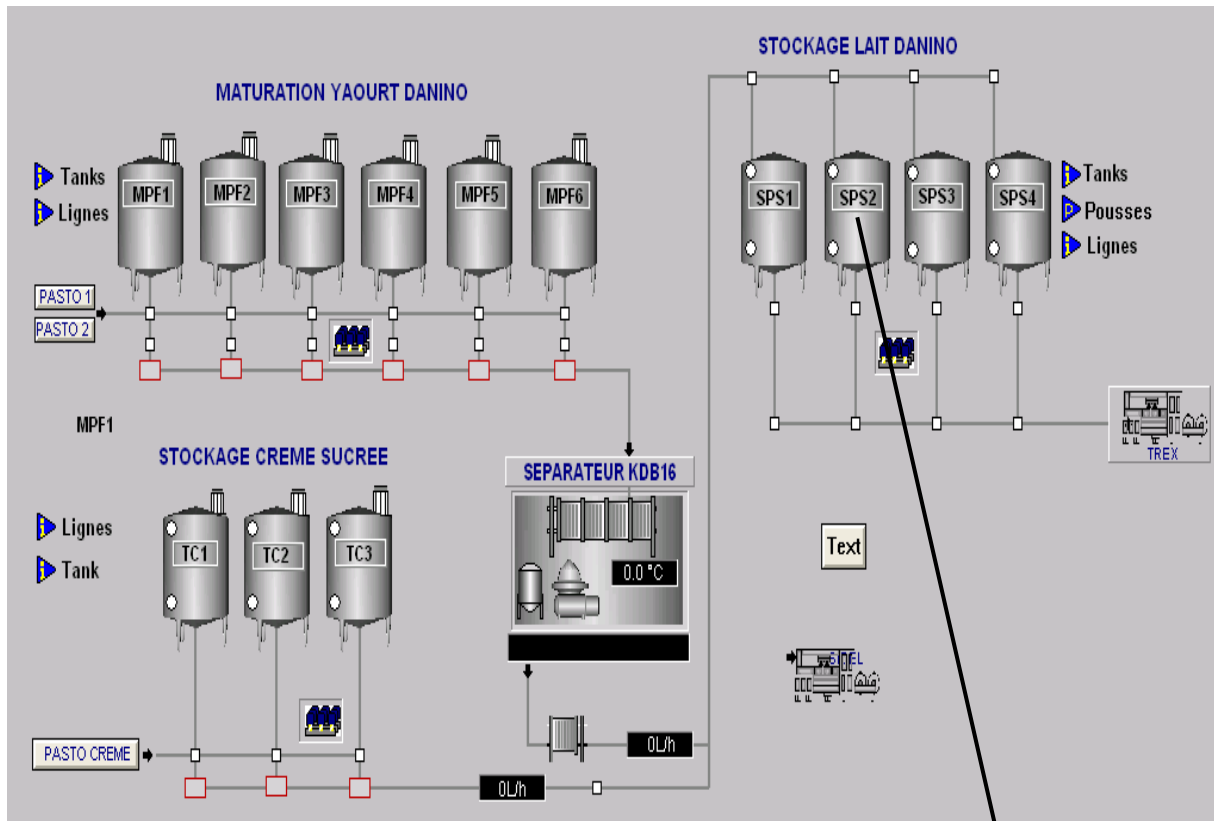
#### ✓ *Rôle*

Appréhende les déchets latéraux et les ramène au bac à déchets.

### I-5 : Processus globale du mécanisme de production

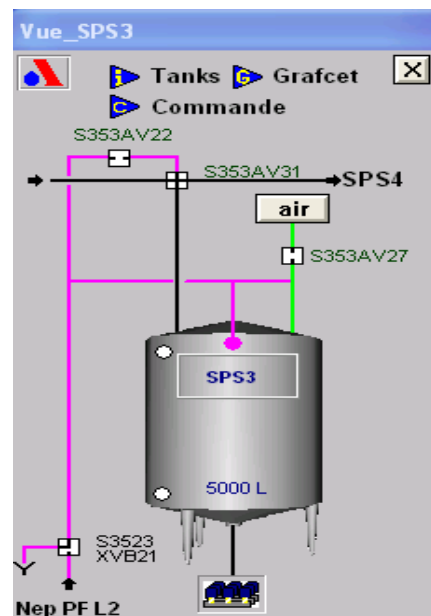
Dés l'arrivée du lait cru il passe par un séparateur qui le sépare en lait pur (caillé) et en sureau (eau). Cette opération à pour but d'avoir le caillé du lait qui est ensuite, grâce à une pompe hydraulique, injecté dans un refroidisseur pour le refroidir et le stocker dans quatre tanks différents (SPS) de capacité de 5000 litre prévus à cet effet.

Puis le lait pur (caillé) est mélangé avec une crème fraîche sucrée provenant des autres tanks de stockages comme le montre la figure I-11 suivante

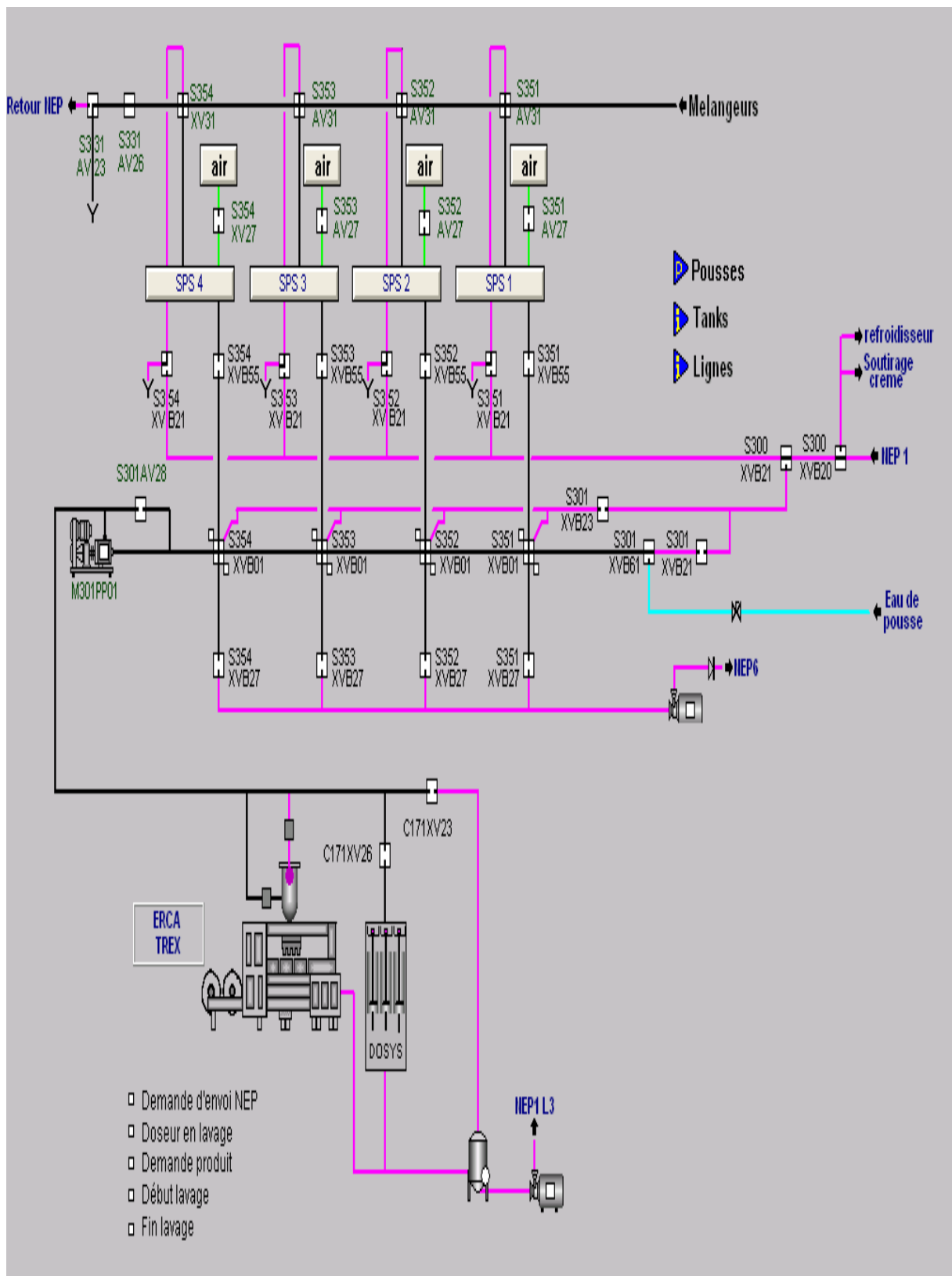


**Figure I-11 :** Processus du mélange lait caillé et de la crème sucrée.

Puis une machine qui porte le nom de **TREX** fait une demande de produit après que l'opérateur l'est mise en service. L'opérateur présélectionne l'un des tanks à l'avance par exemple le tanks 3(SPS 3) et dès que le signal demande produit parvient à l'automate il ouvre une vanne sous le tank en question puis une autre vanne de soutirage et une pompe hydraulique remplit la trémie qui se situe en dessus de la machine thermoformage.



**Figure I-12 :** Vue d'un seul tank.



**Figure I-13 :** Aperçu global sur la circuiterie du processus.

Et à la fin de la production (arrêt de la machine) l'opérateur procède à son lavage. premièrement en ouvrant une vanne qui permet le passage d'eau qui pousse notre pate fraiche qui reste dans les tuyaux d'acheminement vers la machine durant un temps calculé à l'avance (réduire les pertes) et de fermer une autre vanne et ceci grâce à un jeu entre les vannes (l'eau ne rentre pas dans la machine) puis à la fin de l'opération l'opérateur procède a un lavage en place (NEP) de toute la machine avec de l'acide ( $\text{HNO}_3$ ) et de la soude ( $\text{NAOH}$ ). Et à la fin avec de l'eau (rinçage). Le déroulement du nettoyage en place est assurée par des différents outils comme : une Cuve d'eau, Cuve de soude, Cuve d'acide, un Echangeur de chaleur des, vannes ...ets [12].

### **I-6 : Conclusion**

Dans ce chapitre nous avons exposé la machine conditionneuse Trex de Erca tout en donnant son mode de fonctionnement et en présentant le rôle et le fonctionnement de chacun de ses éléments.

Ensuite nous nous somme intéressé à la description du circuit existant dans l'unité en expliquant les différentes étapes du procédé de fabrication.

# Chapitre II

## Généralités sur PL7& InTouch



## II-1 : Présentation du logiciel de programmation PL7 pro

### II-1-1 : Définition

Comme plusieurs automates disponibles sur le marché, l'automate TSX-Premium de la société Télémécanique (maintenant Schneider automation) possède son propre logiciel de programmation. Comparativement à la majorité des autres logiciels, PL7-Pro fournit une interface graphique pour la programmation des diagrammes à relais (Ladder) et pour les diagrammes Grafcet

### II-1-2 : Interface générale du logiciel

Lors de son ouverture, le logiciel offre une interface semblable à celle de n'importe quel programme Windows. On retrouve une barre de menus, une barre d'outils et une fenêtre de travail [2].

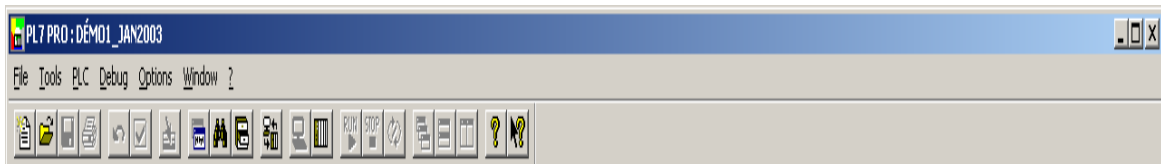


Figure II-1 : Interface générale du logiciel PL7\_pro.

### II-1-3 : Création d'une nouvelle application

Pour concevoir une nouvelle application, accéder au logiciel « PL7-PRO » puis ouvrir un nouveau fichier, à ce stade, il va falloir faire le choix du type de processeur associé à l'automate, ainsi que du type de programmation envisagé [3].

L'écran suivant apparaît :

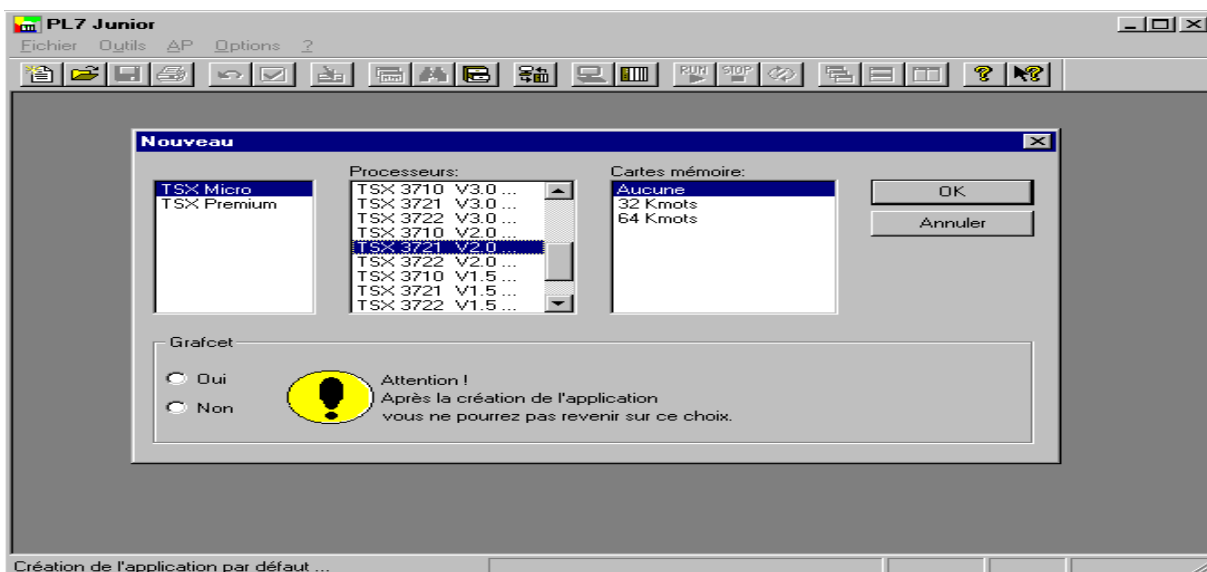


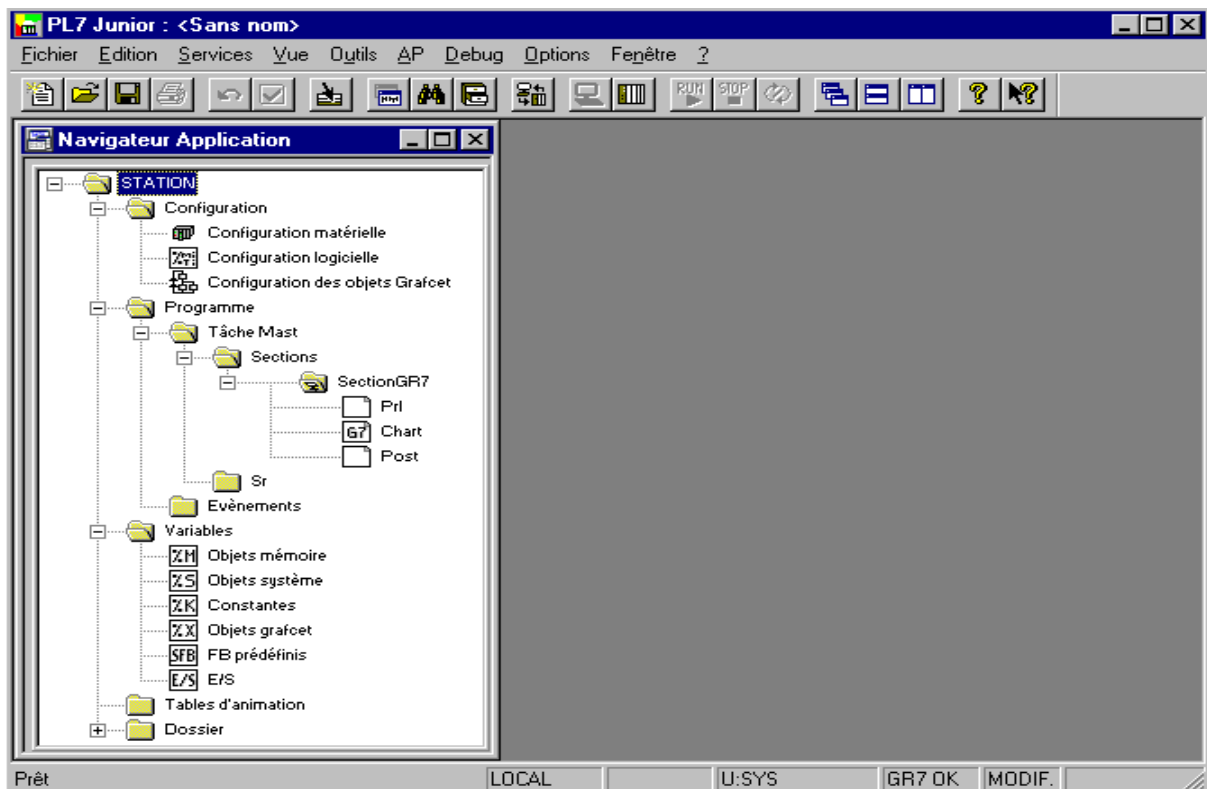
Figure II-2 : Création d'une nouvelle application.

- Choix du type de gamme d'A.P.I : **TSX Micro.**
- Choix du type de processeur inscrit sur une face de côté du rack de l'A.P.I : **TSX 3722 V2.0.**

**Attention** la version est importante pour la bonne exécution du programme futur.

- Choix de la carte mémoire additionnelle : **Aucune.**
- Choix du type de langage Grafcet ou pas, **Oui** pour pouvoir explorer le maximum de configuration.

Après avoir validé, Il apparaît la fenêtre du « Navigateur d'application » c'est le tableau de bord de l'application [3].



**Figure II-3:** Fenêtre de navigation de l'application dans PL7-Pro.

L'ensemble de l'arborescence est développé dans l'image ci-dessus. Nous allons procéder aux différentes configurations successives avant d'amorcer la programmation en tant que telle. Mais avant d'aller plus loin effectuons un premier enregistrement de l'application. Sélectionner dans le menu **Fichier** l'action **Enregistrer**. Et renseignez la fenêtre ci-dessous :

- Choisir le répertoire de sauvegarde : *disque dur D – répertoire de la section.*
- Donner un nom au fichier qui est enregistré avec l'extension **.STX**.
- Valider en appuyant sur **Enregistrer**.

## II-1-4 : Configuration matérielle et logicielle

### 1- Configuration matérielle

La configuration matérielle assure l'interface et la cohérence entre l'environnement externe et l'application logicielle. C'est dans cette partie que sont définies les cartes d'entrée/sortie, de communication ... [3].

- Se placer sur la rubrique **Configuration matérielle**.
- Cliquer droit avec la souris.
- Choisir l'action **Ouvrir**.

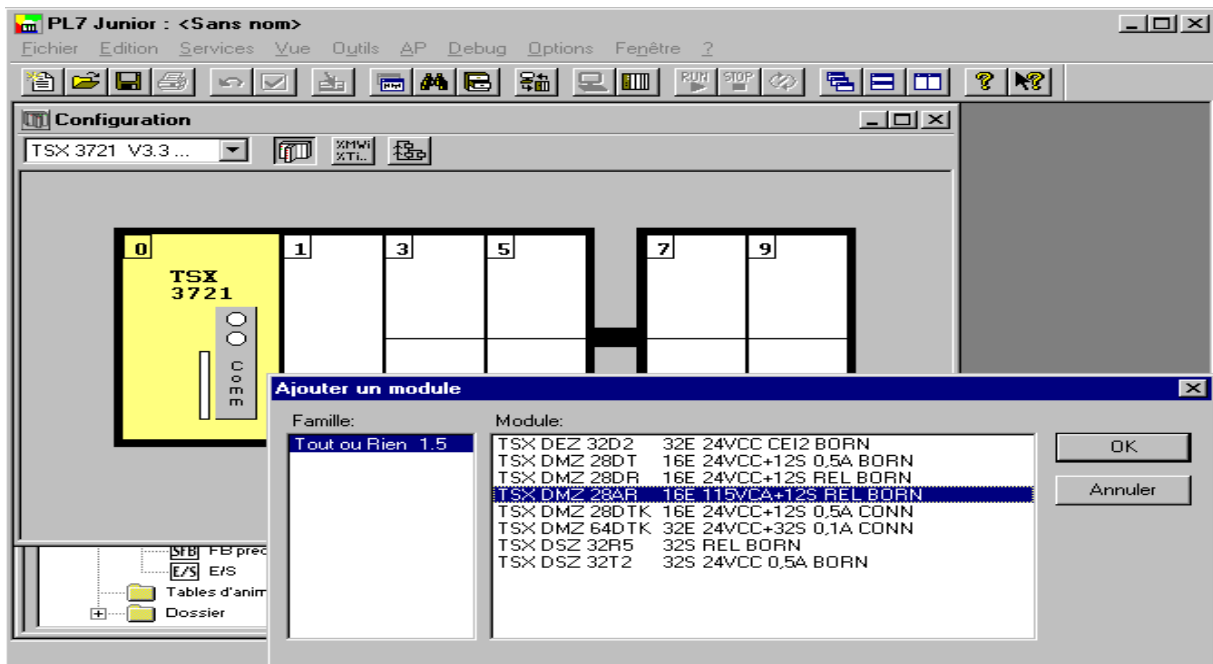


Figure II-4 : Présentation des modules de l'automate.

Pour ajouter un module à l'automate sélectionner l'emplacement

- Cliquer droit avec la souris.
- Choisir l'action Ajouter un module.
- Choisir dans la liste de carte d'entrée/sortie proposée celle qui correspond à votre application.

### 2- Configuration logicielle

La configuration logicielle permet d'ajuster les éléments logiciels tels que temporisateur, compteur... besoins de l'application [3].

Pour accéder à la configuration logicielle on se place sur la rubrique configuration logicielle puis cliquer sur le bouton droit de la souris Choisir l'action Ouvrir.

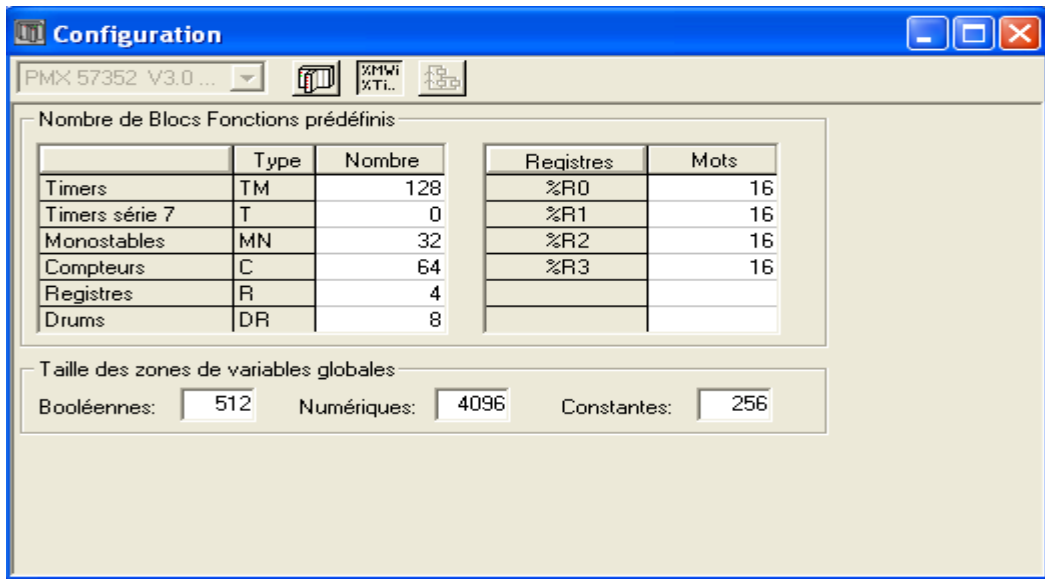


Figure II-5 : Aperçu de la fenêtre de configuration logiciel.

### II-1-5 : Gestion des adresses et des variables

Le but de ce logiciel étant la programmation de l'automate, il doit fournir une méthode plus simple pour accéder à ce dernier dans nos programmes. Cette méthode, c'est l'utilisation des adresses internes de la mémoire de l'automate. Tout objet de programmation sera associé à une adresse, que ce soit un contact, une sortie ou une variable interne [2].

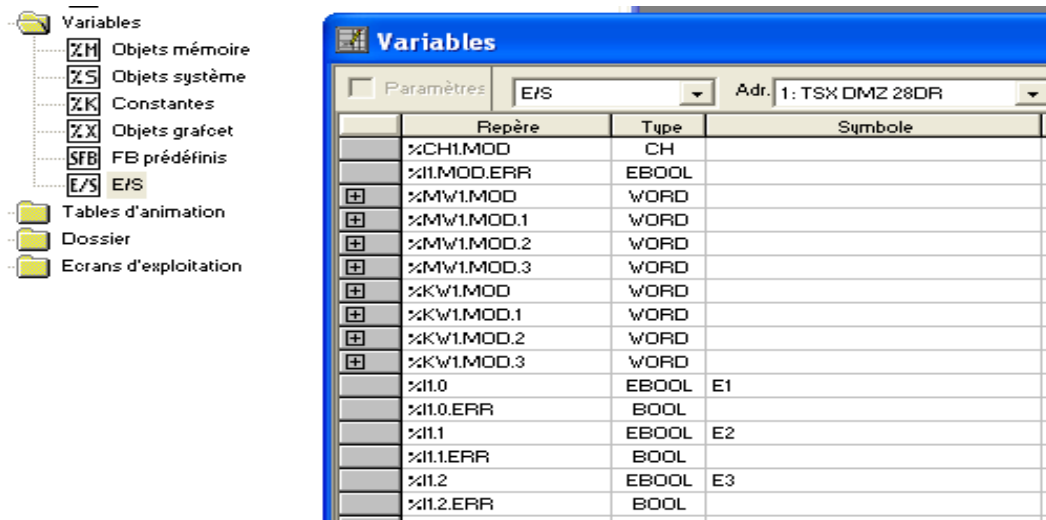


Figure II-6 : Répertoire des variables de la fenêtre de navigation de l'application.

- **Memory Objects**

Adresses des variables disponibles pour vos programmes.

- **System Objects**

Adresses des objets systèmes (ne pas modifier rien dans ce tableau).

- **Constants**

Adresses des constantes disponibles pour vos programmes.

- **Predefined FB**

Adresse des blocs fonctionnels disponibles pour les diagrammes à relais.

- **E/S**

Adresses des entrées / sorties des différents modules de l'automate.

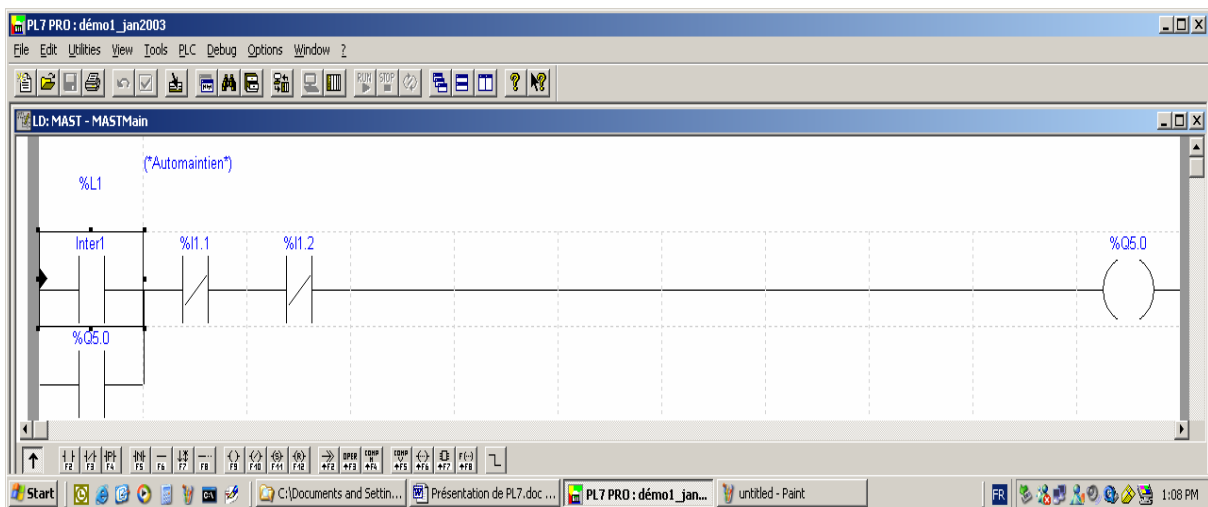
Il est possible de retrouver toutes les adresses internes de l'automate. Ceci peut nous permettre d'abord de connaître les adresses disponibles, mais surtout d'associer un symbole aux adresses couramment utilisées. Par exemple, si nous voulons nommer les adresses des interrupteurs, nous devons aller dans le tableau « I/O » et de choisir le module correspondant aux entrées (le module 1). Ceci affiche un tableau qui contient toutes les adresses disponibles dans ce module. Nous n'avons ensuite qu'à repérer les adresses %I1.0 à %I1.15 et d'entrer une valeur dans le champ « Symbole » pour chacune des adresses [2].

## II-1-6 : Programmation sous le PL7

### 1- Programmation des diagrammes à relais




















Une fois les adresses correctement gérées, nous pouvons traiter de la programmation comme telle d'une application dans PL7. Le premier type de programmation que nous utiliserons est le diagramme à relais.

Tout d'abord, nous devons ouvrir la fenêtre de programmation. À partir de la fenêtre de navigation de l'application, aller dans le répertoire « Program » et ouvrir le répertoire avec ses sous répertoires jusqu'à la fin de l'arborescence. Vous retrouverez alors un ou plusieurs éléments avec un icône « LD » [2].



**Figure II-7** : Fenêtre de programmation en diagramme à relais.

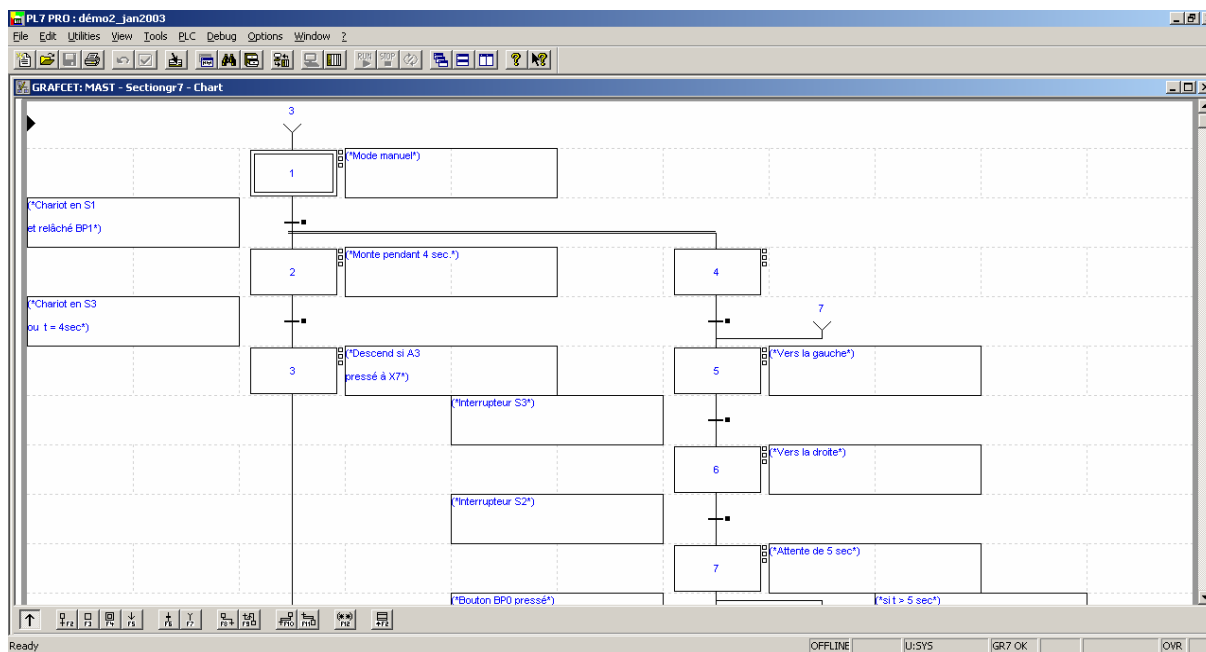
Le tableau II-1 illustre les différents outils à la réalisation de cette programmation [2].

Boutons	Description
	Contact normalement ouvert (vrai si contenu de l'adresse est vrai)
	Contact normalement fermé (vrai si contenu de l'adresse est faux)
	Contact à front montant (vrai si contenu de l'adresse passe de faux à vrai)
	Contact à front descendant (vrai si contenu de l'adresse passe de vrai à faux)
	Lien horizontal d'une case (pour effectuer les connexions)
	Lien vertical entre deux lignes (pour effectuer les connexions)
	Lien horizontal rapide (trace un lien jusqu'à une case du bout de la ligne)
	Bobine en logique positive (agit seulement si la ligne est vraie)
	Bobine en logique négative (agit tant que la ligne est fausse)
	Bobine d'enclenchement (active la bobine correspondante jusqu'à sa désactivation)
	Bobine de déclenchement (désactive la bobine correspondante)
	Lien de saut (saut vers un autre « rung » au cas où la ligne est trop courte pour la condition logique)
	Fonction « Operate » (pour effectuer une opération sur des variables)
	Fonction « Compare » (pour effectuer une comparaison entre des variables)
	Comme précédemment, mais avec plusieurs sorties (ne fonctionne qu'avec des variables de type « Mot » [« Word »])
	Fonctions spéciales d'appel de fonction (inutilisé dans nos programmes)
	Blocs fonctionnels (temporisateurs et autres)
	Fonction complexe (inutilisé dans nos programmes)
	Lien rapide (permet de tracer des liens entre les éléments en cliquant sur les cases à lier (plus utile que lien vertical présenté plus tôt)

**Tableau II-1 :** Les différents outils à la réalisation d'une programmation avec Ladder.

## 2- Programmation des Grafquets










Le deuxième type de programmation disponible est le Grafquet. Le but de cette section est d'illustrer l'implantation de ce genre de diagramme dans PL7 [2].






**Figure II-8 :** Exemple de fenêtre de programmation en Grafquet.

Cette fenêtre ressemble beaucoup à la fenêtre de programmation en diagramme à relais, mais les outils au bas de la fenêtre sont différents.

Le tableau qui suit illustre les différents outils est représenté ci-dessous

Boutons	Description
	Insérer une étape immédiatement suivie d'une transition
	Insérer une étape
	Insérer une étape initiale
	Insérer un lien sortant (pour effectuer un lien vers un entrant)
	Insérer une transition
	Insérer un lien entrant (pour effectuer un lien à partir d'un sortant)
	Insérer un lien étape transition (cliquer sur l'étape, puis sur la transition)
	Insérer un lien transition étape (cliquer sur la transition, puis sur l'étape)
	Insérer une convergence en « ET » (toujours placer la transition à gauche pour effectuer ce type de lien car vous devez cliquer sur l'étape en premier et faire

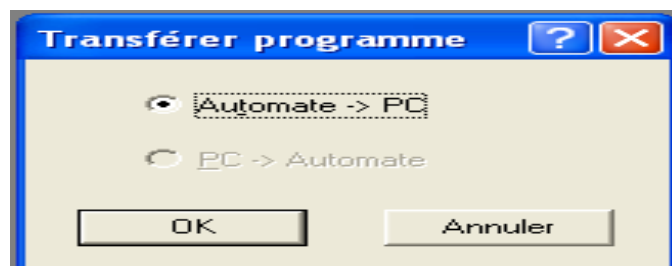
	le lien par la gauche ensuite...)
	Insérer une convergence en « ET » (toujours placer la transition à la gauche
	Insérer une case de commentaire (limiter ce genre de case au minimum car elles occupent de l'espace...)
	Insérer une macroétape (nous verrons le détail de ce type d'étape au cours)

**Tableau II-2 :** Les différents outils à la réalisation d'une programmation avec Grafset.

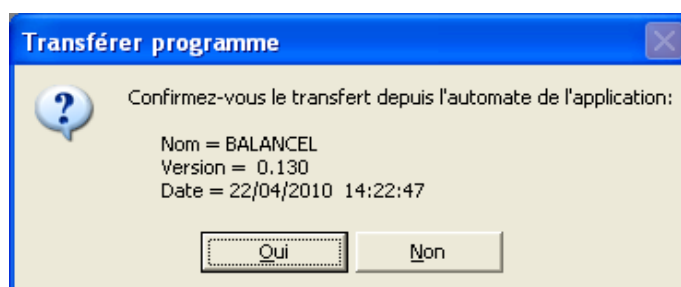
### II-2-7: Transfert des programmes vers l'automate

Le test des programmes passe nécessairement par le transfert du programme vers l'automate. Pour se faire, vous allez passer par le lien série qui relie l'ordinateur au module de contrôle de l'automate (module 0). La seule condition pour pouvoir transférer le programme en cours est qu'il soit entièrement validé, c'est-à-dire qu'il ne contienne pas de partie non validée (en rouge). Lorsque c'est le cas, simplement cliqué sur le bouton « Transfert » de la barre d'outils du logiciel.

Le logiciel demande alors si vous voulez effectuer un transfert PLC->PC ou PC->PLC. Sachant que PLC représente l'automate (Programmable Logic Controller) et que PC représente l'ordinateur (Personnal Computer).



Le logiciel nous demande une confirmation du transfert. Valider en cliquant sur Oui.






## II-2 : Présentation du logiciel de supervision InTouch

### II-2-1 : Introduction

InTouch, de Wonderware est une interface IHM graphique des données d'usine et du procédé. Il permet de mettre en œuvre et déployer des applications de dialogue homme-machine sous Windows, totalement distribuées et intégrées avec les autres applications de la suite Orchestra, automatisation, traçabilité, historisation des données, pilotage de Batch et visualisation via Internet [4].

Il saisit, affiche et archive des données de procédé et les présente à l'opérateur, idéalement dans un format facile à utiliser. C'est une puissante interface opérateur qui prend des données des dispositifs de contrôles/commande (API, capteurs intelligents, etc....) et les affiche pour qu'elles soient utiles à l'opérateur.

Pour ouvrir ce logiciel, il est suffisant de double-cliquer sur le raccourci de  ou en cliquant **démarrer** → **Tous les programmes** → **Wonderware Factory Suite** --> **InTouch** comme illustré dessous :

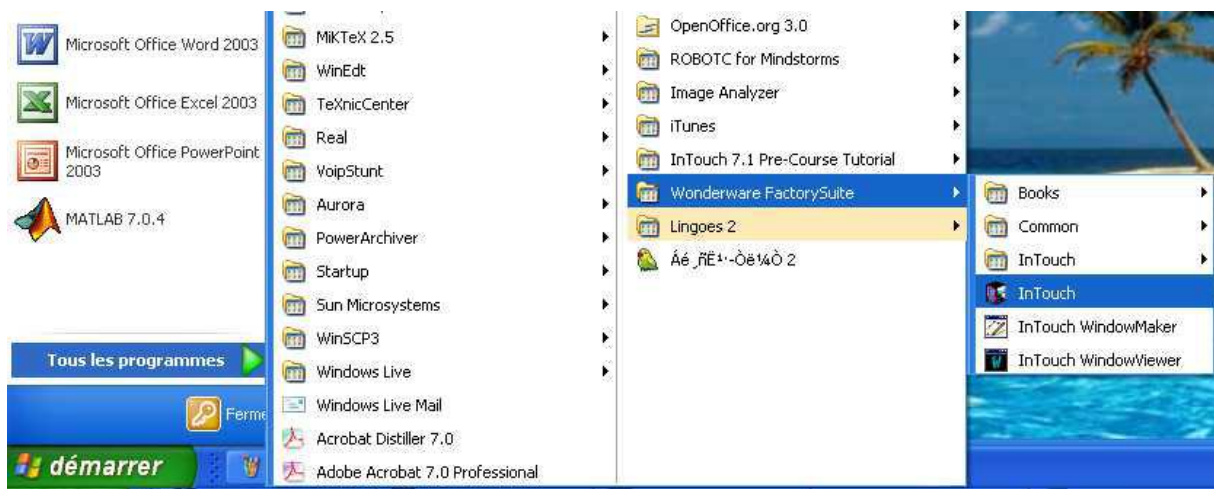


Figure II-9 : Démarrage Intouch.

### II-2-2: Objectifs

✓ Etre capable de mettre en œuvre les principales fonctionnalités du logiciel Intouch dans le contexte d'applications monopostes classiques.

✓ Acquérir un véritable savoir-faire et une méthodologie robuste permettant d'avoir suffisamment d'autonomie pour se voir confier le développement d'un projet de supervision au sein d'une architecture simple [4].

## II -2-3 : Environnement D'InTouch

InTouch est composé principalement de 3 parties : Application Manager, Window Maker et Windows Viewer

### 1- Application Manager (gestionnaire d'application)

Il nous permet de créer une nouvelle application ou d'organiser les applications créées (gestion des répertoires, des paramètres globaux...)

### 2- Window Maker

Il nous permet de développer l'application, la base de données temps-réel, les fenêtres d'animation en utilisant les objets graphiques et leur paramétrage.

### 3- Window Viewer

C'est l'environnement d'exécution, avec la visualisation dynamique des fenêtres graphiques créés dans Window Maker. Il exécute les Quick Scripts, l'historisation des datas, les liens client-serveur, les protocoles de communication Suite Link. Il lance les alarmes, gère les recettes et les Batch de procédé (fabrication par lot) [4]

## II -2-4 : Création d'une application

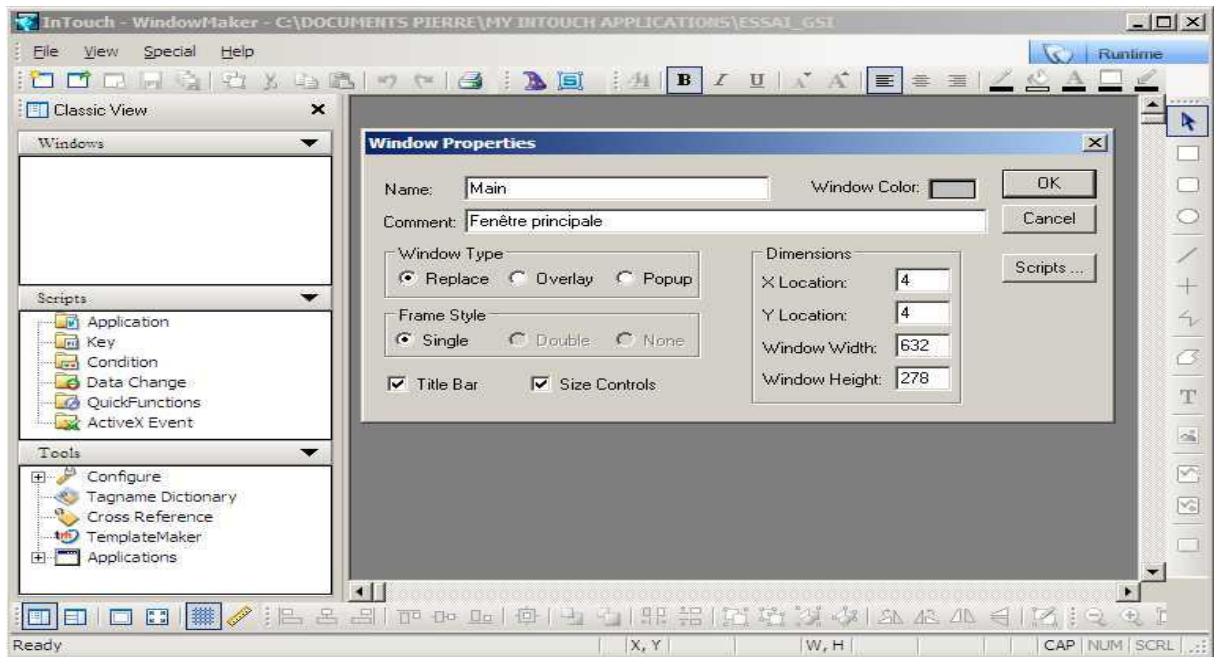
Dans l'application Wmanager il faut préalablement créer une application



Figure II-10 : Fenêtre d'application.

## II -2-5: Création d'une fenêtre synoptique dans Window Maker

InTouch est un outil graphique, il faut créer une modélisation graphique du processus à l'aide des outils de création d'objet. Pour ceci, on crée tout d'abord, un nouveau Windows à partir de Window Maker



**Figure II-11** : Création d'une nouvelle fenêtre dans Window maker.

Une application peut comprendre plusieurs fenêtres (appelées synoptiques de supervision).  
Le changement de fenêtre se fera par programmation (bouton de commande).

## II -2-6 : Création d'objets graphiques dans une fenêtre

### 1- Les outils de création d'objet

Ils sont illustrés comme suit :



Sélectionner les modes



Dessiner les formes



Texte.



Figure.



Buttons.



Aligner les positions.



Changer les plans ... espaces.




Faire ou défaire (grouper/dégrouper) un objet, une cellule.

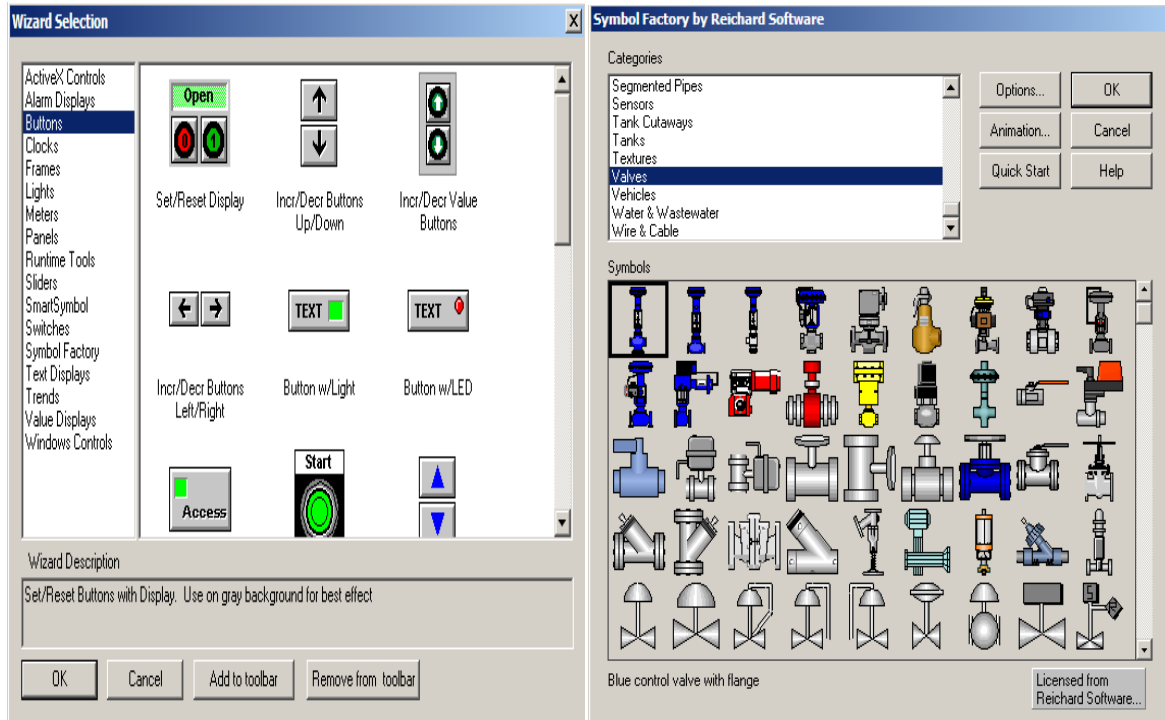


Changer les orientations.

## 2- Outils "Wizard"

De nombreux objets industriels ont une représentation graphique conventionnelle. Ces éléments pré-dessinés sont stockés dans une bibliothèque livrée avec le logiciel Cette

bibliothèque est accessible par le bouton Wizard  du menu général



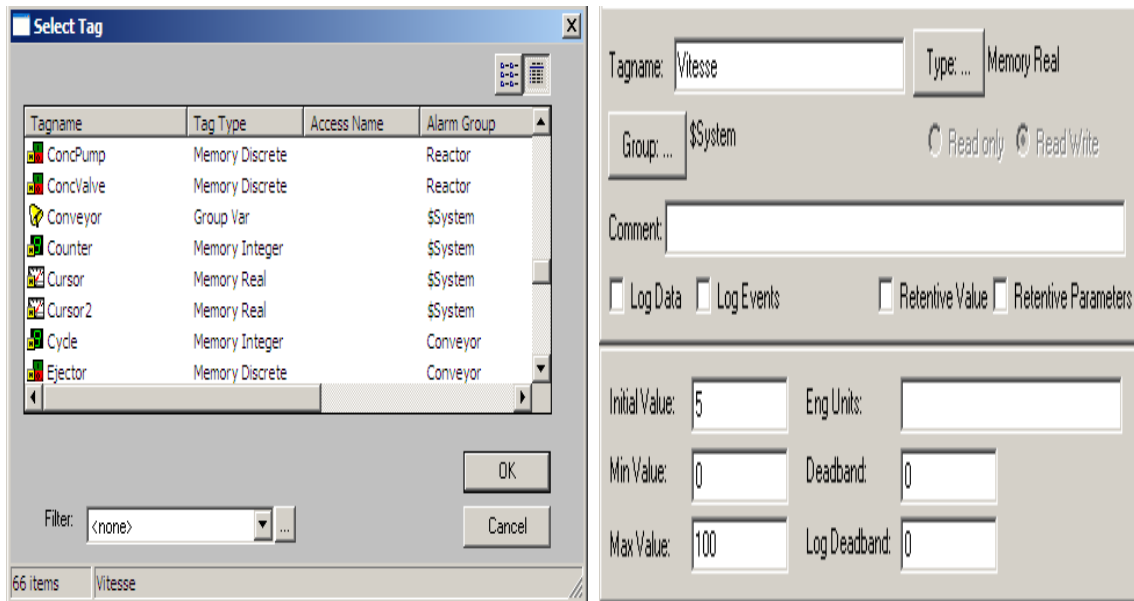
**Figure II-12** Fenêtre des utiles Wizard.

Il suffit de sélectionner un élément par double-clic pour le recopier dans la fenêtre en cours d'édition. L'objet est redimensionnable lors de sa mise en place. Une bibliothèque supplémentaire est contenue dans **Symbol Factory**.

Ces éléments graphiques permettront de donner un aspect professionnel à l'application.

### II -2-7: Définition des tagnames (base de données des variables)

Un tagname est une variable sous Intouch. Ces variables peuvent ensuite être appliquées à des objets pour définir leurs paramètres/propriétés mais aussi utilisées dans des scripts et des protocoles de communications [4].



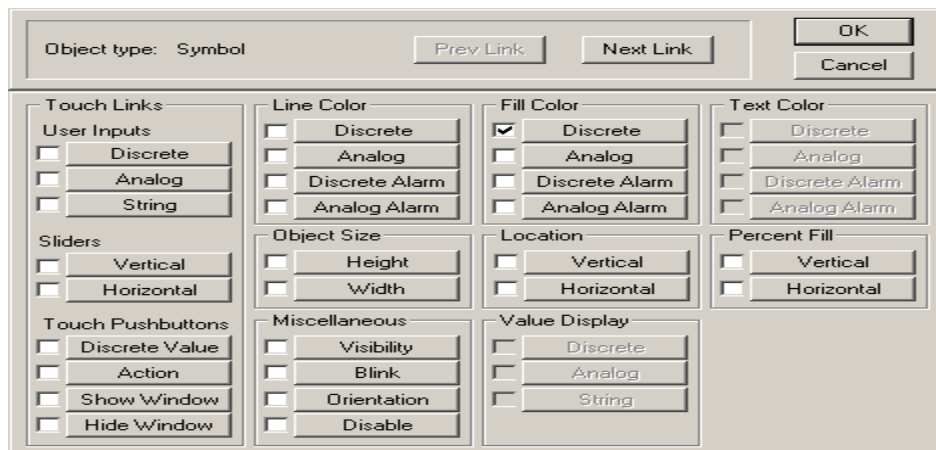
**Figure II-13 : Gestion des adresses tagnames.**

Chaque tagnome doit être défini selon un type (booléen, entier, réel, chaînes, etc.), ses propriétés spécifiques (variable interne, système, I/O, groupe d'appartenance, limites, unités).

Pour les tagnames I/O, il faut décrire le lien logique et/ou logique qui relie le tag au matériel ou à une application externe.

### II-2-8 : Paramétrage d'un élément graphique

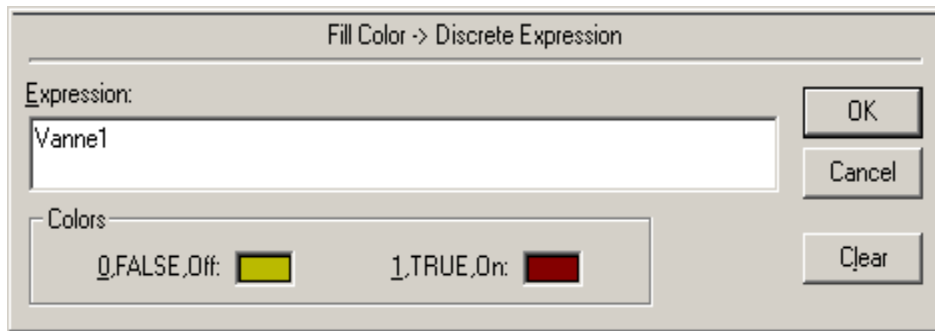
Pour attribuer une propriété dynamique à un élément graphique, il suffit d'un double-clic sur cet élément.



**Figure II-14 : Paramètre des éléments graphique.**

### Exemple 1

On sélectionne l'option « Fill Color » et on déclare la variable qui influencera ce changement de propriété.

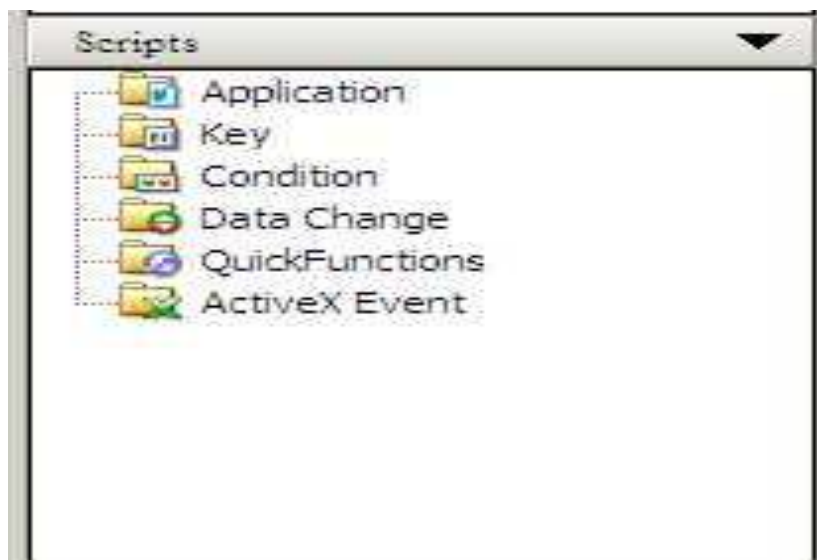


Affectation de la propriété la variable « Vanne1 » et lorsqu'elle sera à 0 la vanne sera de couleur verte et quand « Vanne1 » passera à 1 la vanne sera de couleur marron.

La propriété "Fill Color" appliquée avec une variable analogique applique une fonctionnalité de remplissage. Elle est très utile pour simuler un réservoir.

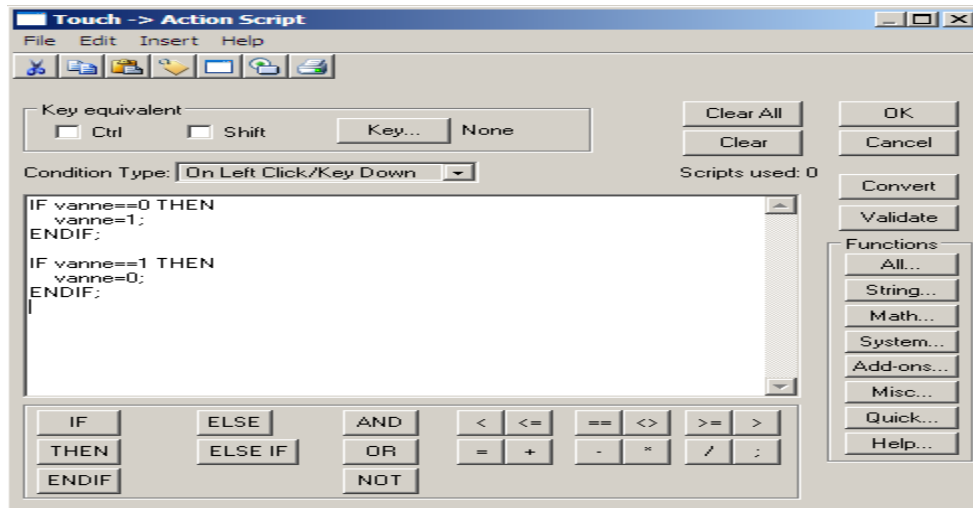
**Script** Intouch propose de programmer le séquencement des variables via son module "script". Un script est un élément de code (procédure) de type événementiel. Les principaux événements sont :

- ✓ ouverture et fermeture de l'application
- ✓ exécution périodique
- ✓ changement de valeur d'une variable
- ✓ réalisation d'une condition à partir d'une combinaison des variables



## Exemple 2

La vanne changera d'état toutes les 5 secondes. On peut ainsi simuler le fonctionnement d'un processus ou programmer certaines commandes.

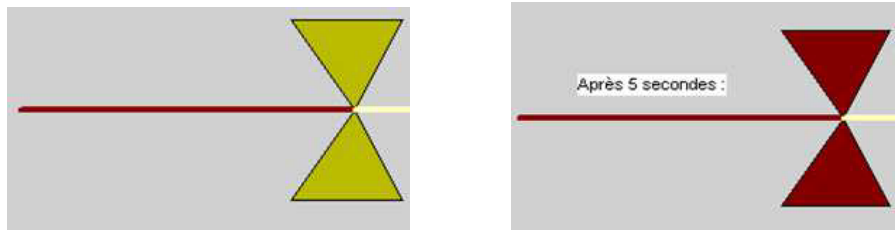


### II-2-9: Window Viewer

Pour lancer l'exécution d'animations d'objets, on clique sur le menu principal de Window Maker de la position droite : Runtime



La visualisation du processus ce fait avec l'application Viewer.



### II-3 : Conclusion

Dans ce chapitre on illustré le logiciel de programmation PL7 de Schneider, en présentant des généralités et des exemples sur son fonctionnement et sur ses différent langage de programmation.

Puis on à illustré le logiciel de supervision Intouch de Wonderware, nous avons également présenté ses généralités et des exemples sur son fonctionnement.

# Chapitre III

## Etude de l'interface



### III-1 : Notion sur la régulation automatique

#### III-1-1 : Définition

La régulation automatique regroupe l'ensemble des moyens matériels et techniques mis en œuvre pour maintenir automatiquement ( pas d'intervention manuelle) une grandeur physico-chimique d'un procédé (grandeur réglée), égale à une valeur désirée appelée consigne, quelles que soient les perturbations engendrées par le milieu extérieur sur ce procédé. Lorsque des perturbations (grandeurs d'entrée non commandable) ou un changement de consigne se produisent, la régulation automatique provoque une action correctrice sur une autre grandeur physique appelée grandeur réglant (grandeur d'entrée commandable), afin de ramener la grandeur réglée vers sa consigne initiale (cas de régulation) ou vers sa nouvelle consigne (cas d'asservissement ou changement de point de fonctionnement) [10].

#### III-1-2 : Objectifs de la régulation Automatique

- Synthétiser une loi de commande (un correcteur) afin d'obtenir un système performant: **précis, rapide** et **stable**, tout en s'affranchissant des influences néfastes des perturbations.
- Annuler (ou diminuer) les effets de la (ou des) perturbation(s).

##### *Remarque 1*

- ✓ Dans une régulation, on s'attachera à maintenir constante la grandeur réglée d'un système soumis à des perturbations.
- ✓ Dans un asservissement, la grandeur réglée devra suivre rapidement les variations de la consigne.

#### III-1-3 : Notion de Système

Un système est un ensemble d'éléments interconnectés pour accomplir une tâche prédéfinie. Il est affecté par une (ou plusieurs) variable(s) nommée les entrées du système. Le résultat de l'action des entrées est la réponse du système caractérisée par l'évolution d'une ou plusieurs variables nommée les sorties [5].

##### *Remarque 2*

Un système industriel est souvent appelé processus.

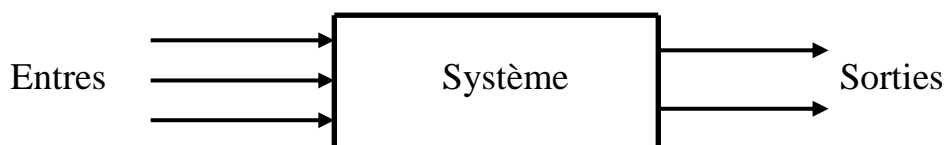


Figure III-1: Notion de système.

### III-1-4: Notion de Boucle Ouverte/Fermée

#### 1- Système boucle ouverte (Manuel)

On parle de fonctionnement en boucle ouverte quand c'est l'opérateur qui contrôle l'organe de réglage, ce n'est pas une régulation.

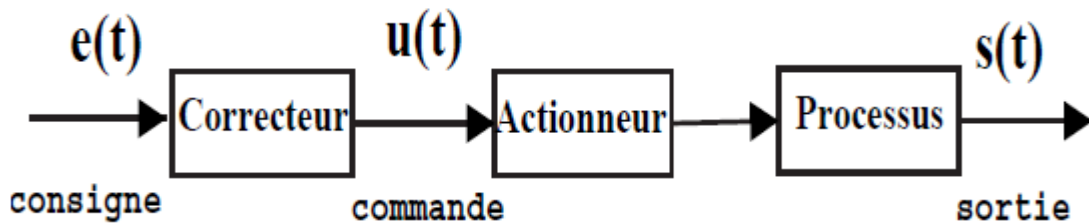


Figure III-2 : Système en BO.

#### 2- Système boucle fermée (Automatique)

C'est le fonctionnement normal d'une régulation. Le régulateur compare la mesure de la grandeur réglée et la consigne et agit en conséquence pour s'en rapprocher [5].

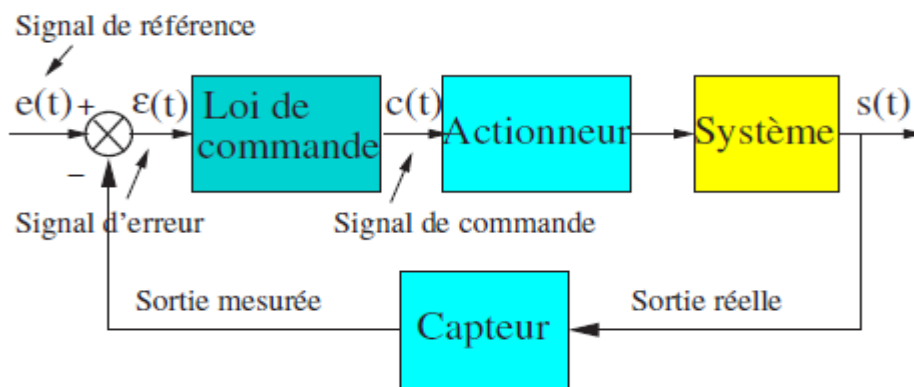


Figure III-3 : Système en BF

### III-1-5 : Types de régulation automatique

#### 1- Régulation tout ou rien (TOR)

Ce mode d'action est essentiellement discontinu. Sa réalisation impose de se fixer une limite inférieure et une limite supérieure lorsque la mesure atteint la limite inférieure l'actionneur prend une position particulier (ouverture ou fermeture pour une vanne) de façon analogue le fait d'atteindre la limite supérieure place l'actionneur dans la position contraire. La mesure oscille donc entre ces deux valeurs extrêmes et sa variation prend une allure en dents de scie. Ce réglage est simple ou le signal de commande ne prend que deux valeurs soit 0 ou 1 [11].

## ***2- Régulation analogique***

C'est le type de régulation où le signal du régulateur et la mesure variant d'une manière continue dans le temps. Le mode d'action analogique le plus simple est l'action proportionnelle qui est réalisée par un régulateur (p). Il convient en général aux installations ayant une grande inertie [1].

## ***3- Régulation numérique***

Le principe de la régulation numérique est que le régulateur prend la forme d'un algorithme programmé sur microprocesseur et exécuté en temps réel, i.e. impérativement à chaque période d'échantillonnage [8].

### **III-1-6 : Critères de performance d'une régulation**

Les performances d'une régulation peuvent se définir à partir de l'allure du signal de mesure suite à un échelon de consigne. Notons toutefois que les critères de performances classiques peuvent se résumer comme suit :

- **Stabilité**

Cette condition est impérative mais avec un certain degré de stabilité (marge de sécurité). En général on impose une marge de gain de 2 à 2,5. L'utilisateur parle en termes de pompage.

- **Précision**

L'exploitant demande à ce que le système possède une bonne précision en régime permanent d'où une nécessité de mettre un régulateur PI ou d'afficher un gain important dans le cas d'un régulateur P.

- **Rapidité**

On demande en pratique que le système soit capable rapidement de compenser les perturbations et de bien suivre la consigne.

- **Dépassement**

En général on recommande un système de régulation dont le régime transitoire soit bien amorti et dont le dépassement ne dépasse pas 5 à 10% de la valeur nominale [9].

### III-2 : Les éléments de la boucle de régulation

Nous allons présenter quelques généralités sur les différents composants de la boucle de régulation et leurs importances dans notre système.

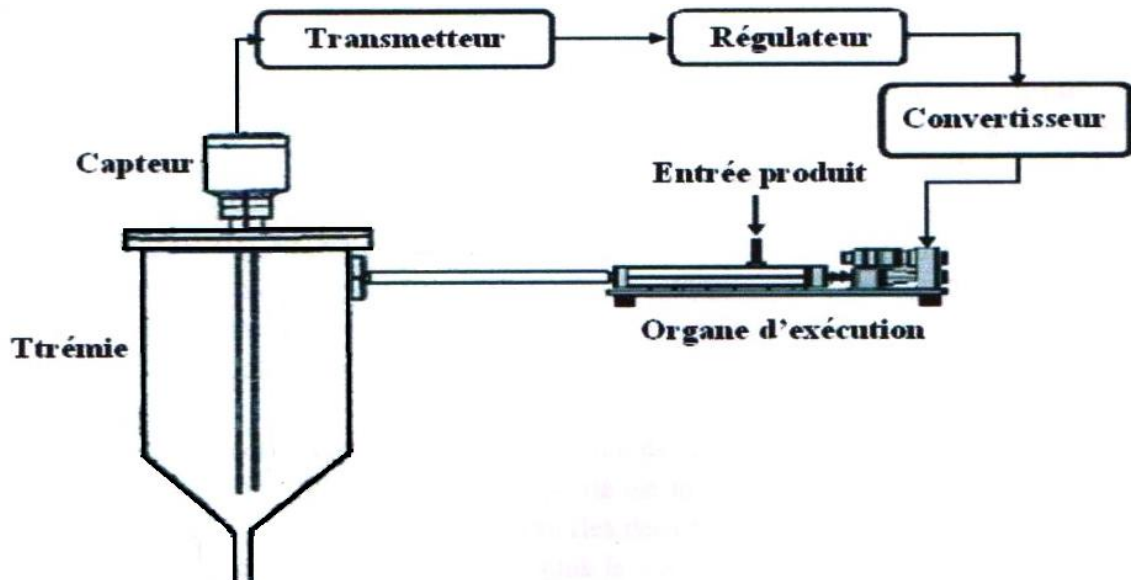


Figure: III-4 : Schéma général de la boucle de régulation.

#### III-2-1 : Le Silometre FMC 420(transmetteur)

Le FMC 420 est un convertisseur courant-tension qui a la forme représenté sur la Figure suivante :

##### ❖ *Fonctionnement*

Le Silometre FMC 420 alimente en courant continu les capteurs avec électroniques intégrées et reçoit en retour un signal proportionnel au niveau d'envoi. 0...4 mA.

Des signaux normés 0...10 V et 0...20 mA (ou 4...20 mA), pour l'affichage à distance du niveau sont disponibles en sortie Silometre. Une diode verte indique que l'appareil est prêt à fonctionner [14].

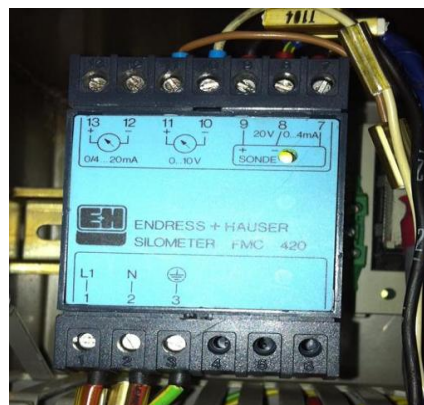


Figure III-5 : Silometre FMC 420.

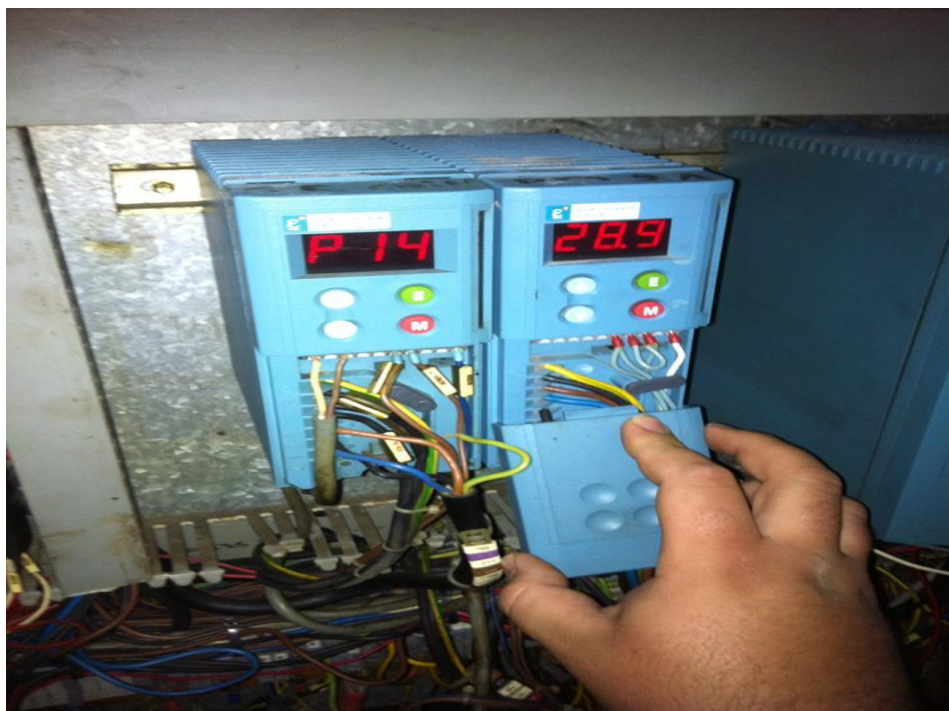
### III-2-2: Le Variateur de vitesse EUROTHERM 601 (convertisseur)

#### ❖ *Description*

La série 601 de Variateurs de Vitesse a été conçue pour le contrôle de vitesse des moteurs asynchrones triphasés standards. La série couvre la gamme de puissance du moteur entre 0.37kW et 2.2kW

Le 601 peut-être alimenté soit par une tension monophasée de 220/240 Volts (Modèle A et B), ou en triphasé 380/460V 50-60Hz pour le modèle C.

La version C alimentée en 400V triphasé comporte un module de freinage intégré et d'un connecteur permettant le raccordement d'une résistance extérieure pour le freinage dynamique [15].



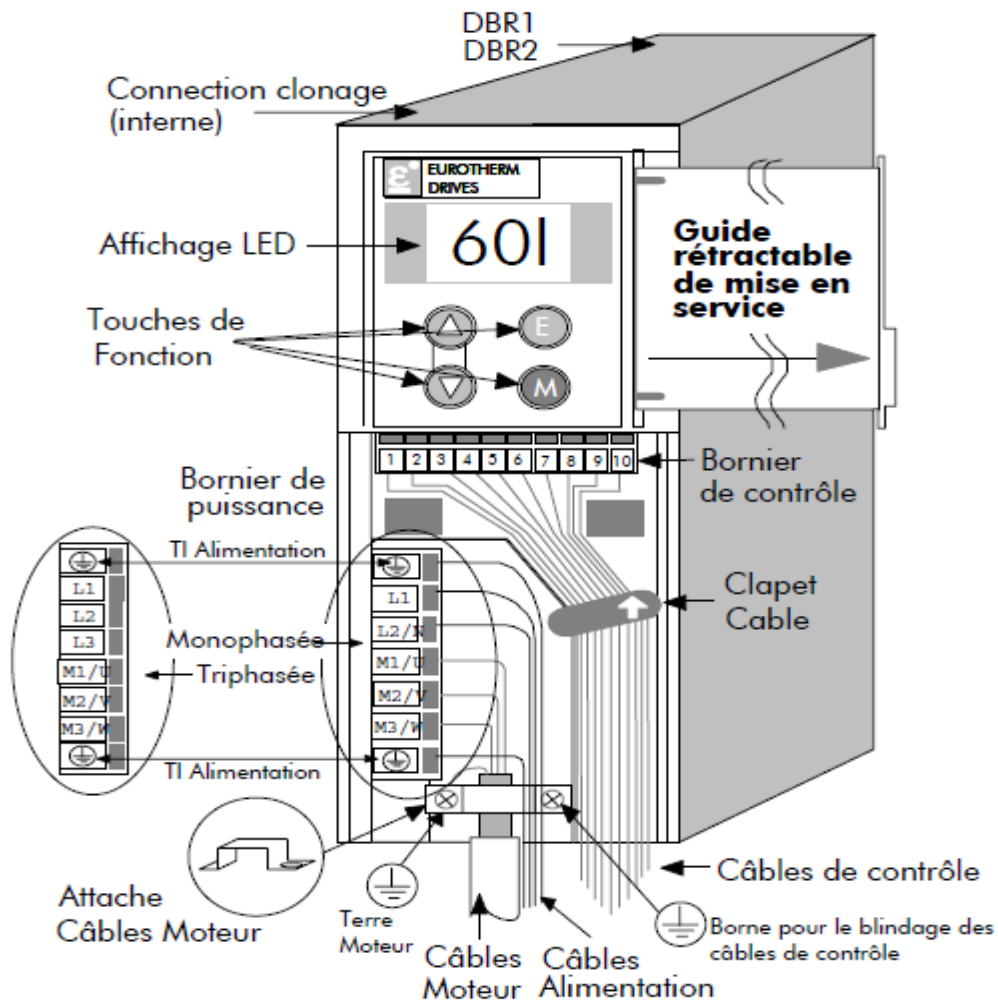
**Figure III-6 :** Variateur de vitesse EUROTHERM 601.

#### ❖ *Paramétrage du variateur*

Les borniers de contrôle du 601 sont du type SELV, c'est-à-dire à double isolation entre les circuits de puissance pour permettre une interconnexion simple et sûre.

Grâce à une stratégie de contrôle intelligente, le 601 est protégé contre les surcharges, tensions excessives et les courts-circuits entre phases et entre phase et terre. Ceci permet d'éviter des déclenchements intempestifs et garantit un fonctionnement fiable.

Des filtres RFI internes optionnels permettent une complète conformité à la directive CEM (compatibilité électromagnétique) pour la plupart des applications sans la nécessité d'addition de composants externes.



**Figure III-7 :** Vue du variateur sans le capot des borniers.

Trois afficheurs sept segments à LEDS permettent la programmation du variateur ainsi que la visualisation des paramètres de fonctionnement.

Les touches de fonction permettent de modifier les réglages du variateur et en mode LOCAL de le piloter. La structure “arborescente” de l’interface utilisateur et le fonctionnement des touches de fonction sont décrites dans le schéma suivant:



Cette touche est utilisée pour descendre du **niveau état** au **niveau titre** ou du **niveau titre** au **niveau valeur**. Cette touche de fonction est aussi utilisée pour **arrêter** le variateur en mode LOCAL.



### ECHAPPE

Cette touche est utilisée pour monter du **niveau valeur** au **niveau titre** ou du **niveau titre** au **niveau état**. Il est à noter que cette action valide le réglage effectué. Cette touche de fonction est aussi utilisée pour **démarrer** le variateur en mode LOCAL est sélectionne.



### HAUT

Cette touche est utilisée pour se déplacer dans le **niveau titre** ou incrémenter les valeurs des paramètres de réglage. Cette touche de fonction est aussi utilisée pour augmenter **la consigne de fréquence du variateur** en mode LOCAL.



### BAS

Cette touche est utilisée pour se déplacer dans le **niveau titre** ou décrémenter les valeurs des paramètres de réglage. Cette touche de fonction est aussi utilisée pour diminuer **la consigne de fréquence du variateur** en mode LOCAL.

Notre variateur contient 15 paramètres et 10 borniers de contrôle sur les quelle on va jouer afin de réaliser notre travail c'est-à-dire avoir un pallier de trois vitesses ces paramètres sont illustré dans le tableau suivant :

Description des borniers de contrôle			Réglage des paramètres		
1	Référence 0 V pour E/S analogique	0 V	P1	Vitesse mini ou vitesse pré-réglée 1	Fréquence à la quel le 601 fonctionne a consigne null
2	Entrée tension	Consigne	P2	Vitesse max ou vitesse pré-réglée 4	Fréquence a la quel le 601 fonctionne a consigne max
3	Entrée courant	Consigne	P3	Temps d'accélération	Temps pour passe de la fréquence null à la vitesse max

4	Alimentation 10 V	Pour E/S analogique	P4	Temps décélération	Temps pour passe de la vitesse max à la fréquence null
5	Sortie analogique	Consigne fréquence	P5	Limite courant	Limitation de courant donné en pourcentage
6	Alimentation 24 V	Pour E/S TOR	P6	Tension boost	Courige le flux du moteur
7	Entrée TOR	0 V =Arret 24V=Marche	P7	Fréquence de base	Fréquence de sortie à la quelle la tension max et délivrée au moteur
8	Entrée TOR	0V=Avant 24V=Arriere Ou vitesse préréglée	P8	Vitesse jog ou vitesse préréglée 2	La vitesse à la quelle le 601 fonctionne si le born9 =24v
9	Entrée TOR	0V= Arret 24V=jog Ou vitesse préréglée	P9	Vitesse préréglée 3	La vitesse à la quelle le 601 fonctionne quand P13=2
10	Sortie TOR	Sélection de la sortie TOR	P10	Mot de passe	interdire les modifications
P14	Choix de la sortie TOR	Choisir la sortie 0.1.2.3	P11	Mode d'arrêt	Choisir le mode d'arrêt
P15	Mode de copie	Mode 1 ou mode 2	P12	Loi U/F	Sélection la courbe tension/fréquence
			P13	Choix de la consigne	Permet Sélection de la source de la consigne vitesse

**Tableaux III-1** : Les différents paramètres du variateur de vitesse.



### III-2-3 : Une pompe hydraulique PCM

C'est une pompe PCM-MOINEAU à rotor excentré de type 13H6 avec une vitesse maximale de 575 tr/min, 50Hz et avec une capacité de 5000 L/h et un débit de 3.7m<sup>3</sup>/h. Elle est utilisée dans l'industrie laitière pour le transfert de lait concentré vers un évaporateur.

#### ❖ *Principe de fonctionnement*

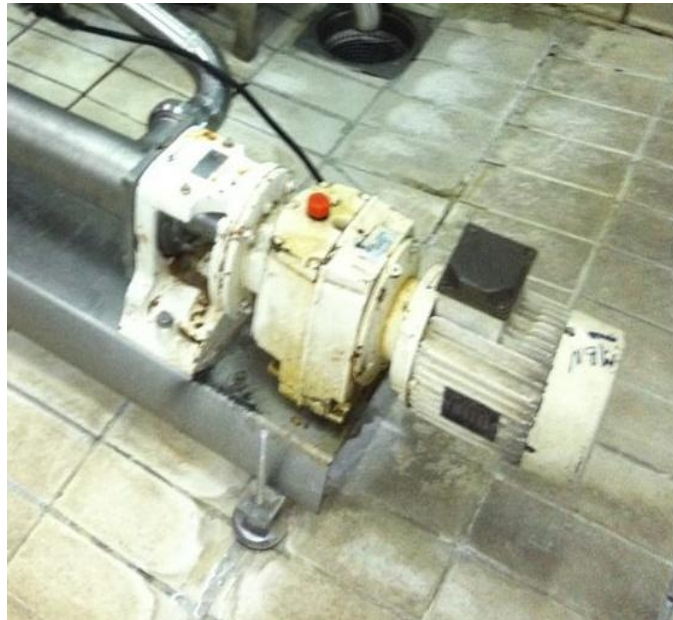
La pompe PCM à cavité progressive (ou à vis excentrée) de type MOINEAU se compose essentiellement de deux engrenages hélicoïdaux intérieurs, l'un et l'autre avec les particularités suivantes:

- Le stator, élément externe, a une dent de plus que le rotor, élément interne.
- Dans une section transversale quelconque, chaque section du rotor est en contact avec le stator.
- Les pas des hélices des deux éléments sont dans le rapport du nombre de dents.

Le mouvement de rotation génère un déplacement axial des alvéoles closes délimitées par le rotor et le stator de l'aspiration vers le refoulement.

La pompe MOINEAU de PCM est une pompe volumétrique dont la cylindrée est égale au volume d'une alvéole.

Le système d'entraînement directement flasqué sur la pompe entraîne l'arbre de commande qui lui-même, par l'intermédiaire d'une barre flexible entraîne le rotor en rotation [15].



**Figure III-8 :** Pompe hydraulique PCM.

### III-2-4 : La trémie

La trémie sert à recevoir le produit qui est transféré des tanks de stockages via une pompe hydraulique. La trémie alimente le doseur de la machine qui fait des cycles de dosage de produit dans les pots à la quantité voulue, donc la disponibilité du produit dans la trémie est nécessaire pour avoir un bon dosage et que le cycle ne s'arrête pas en cours de production .



**Figure III-9** : La forme de la trémie.

### III-2-5 : Capteur analogique

Un capteur est un élément de mesure qui sert à la prise d'informations relatives à la grandeur à régler. C'est un élément capital d'une chaîne de mesure, il a pour rôle de saisir et de transformer la grandeur physique à régler et le contenu de son information en une autre grandeur physique accessible aux sens humains par un signal normalisé de l'information originelle [14].



**Figure III-10** : Le capteur analogique.

### III- 2-6 : L'automate programmable Industriel API (régulateur)

#### III-2-6-1:Définition

L'Automate Programmable Industriel (API) est un appareil électronique programmable, adapté à l'environnement industriel, qui réalise des fonctions d'automatisme pour assurer la commande de préactionneurs et d'actionneurs à partir d'informations logique, analogique ou numérique.

#### III-2-6-2: Structure d'un système automatisé

Tout système automatisé peut se décomposer selon le schéma ci-dessous :

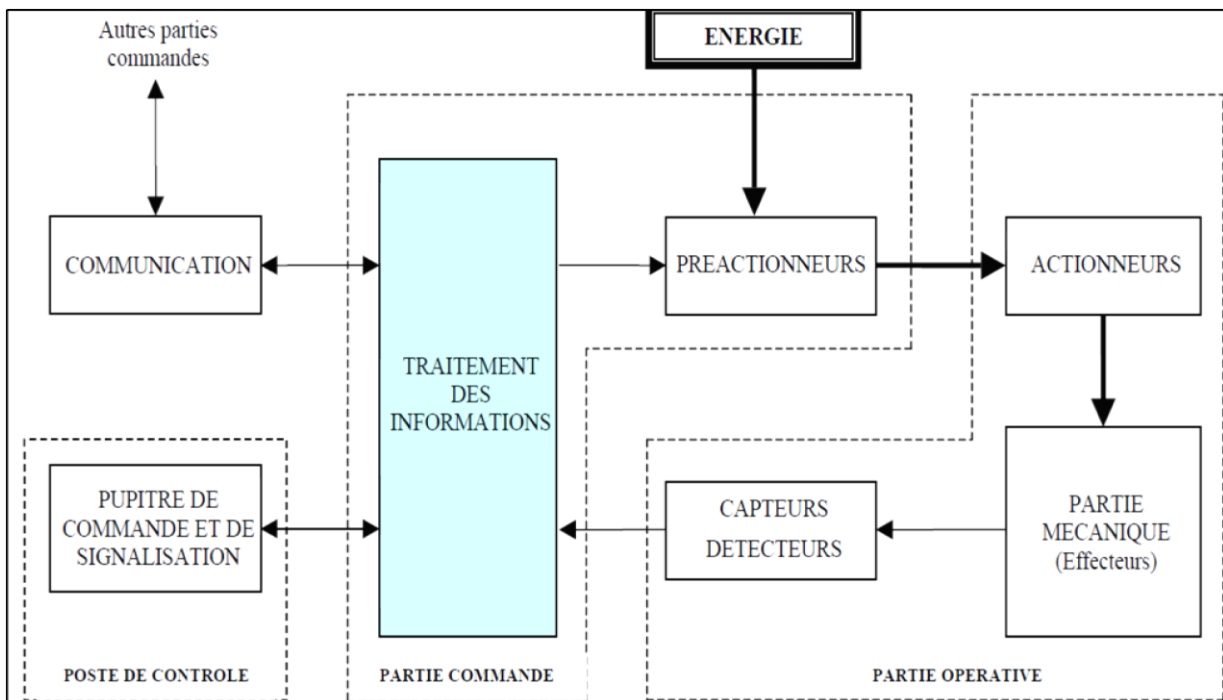


Figure III-11 : Structure d'un système automatisé.

#### ✓ Partie opérative

Elle agit sur la matière d'œuvre afin de lui donner sa valeur ajoutée. Les **actionneurs** (moteurs, vérins) agissent sur la partie mécanique du système qui agit à son tour sur la matière d'œuvre. Les **capteurs / détecteurs** permettent d'acquérir les divers états du système.

#### ✓ Partie commande

Elle donne les ordres de fonctionnement à la partie opérative. Les **préactionneurs** permettent de commander les actionneurs ; ils assurent le **transfert d'énergie** entre la source de puissance (réseau électrique, pneumatique ...) et les actionneurs.

Exemple : contacteur, distributeur ... Ces préactionneurs sont commandés à leur tour par le bloc **traitement des informations**. Celui-ci reçoit les consignes du **pupitre de commande** (opérateur) et les informations de la partie opérative transmises par les capteurs / détecteurs.

En fonction de ces consignes et de son programme de gestion des tâches (implanté dans un automate programmable ou réalisé par des relais (on parle de logique câblée)), elle va commander les préactionneurs et renvoyer des informations au **pupitre de signalisation** ou à d'autres systèmes de commande et/ou de supervision en utilisant un **réseau** et un **protocole de communication**.

✓ **Poste de contrôle**

Composé des **pupitres de commande et de signalisation**, il permet à l'opérateur de commander le système (marche, arrêt, départ cycle ...).

Il permet également de visualiser les différents états du système à l'aide de voyants, de terminal de dialogue ou d'interface homme-machine (IHM).

**III-2-6-3 : Domaines d'emploi des automates**

On utilise les API dans tous les secteurs industriels pour la commande des machines (convoyage, emballage ...) ou des chaînes de production (automobile, agroalimentaire ...) ou il peut également assurer des fonctions de régulation de processus (métallurgie, chimie ...).

Il est de plus en plus utilisé dans le domaine du bâtiment (tertiaire et industriel) pour le contrôle du chauffage, de l'éclairage, de la sécurité ou des alarmes [11].

**III-2-6-4 : Nature des informations traitées par l'automate**

Les informations peuvent être de type :

✓ **Tout ou rien (T.O.R.)**

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 ...

✓ **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 ...)

✓ **Numérique**

L'information est contenue dans des mots codés sous forme binaire ou bien hexadécimale. C'est le type d'information délivrée par un ordinateur ou un module intelligent [6].

**III-2-6-5: Architecture des automates**

**1- Aspect extérieur**

Les automates peuvent être de type **compact** ou **modulaire**.

De type **compact**, on distinguera les *modules de programmation* (LOGO de Siemens, ZELIO de Schneider, MILLENIUM de Crouzet ...) des *microautomates*. Il intègre le processeur, l'alimentation, les entrées et les sorties. Selon les modèles et les fabricants, il pourra réaliser

certaines fonctions supplémentaires (comptage rapide, E/S analogiques ...) et recevoir des extensions en nombre limité.

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

De type **modulaire**, le processeur, l'alimentation et les interfaces d'entrées / sorties résident dans des unités séparées (**modules**) et sont fixées sur un ou plusieurs **racks** contenant le "fond de panier" (bus plus connecteurs). Ces automates sont intégrés dans les automatismes complexes où puissance, capacité de traitement et flexibilité sont nécessaire [11].

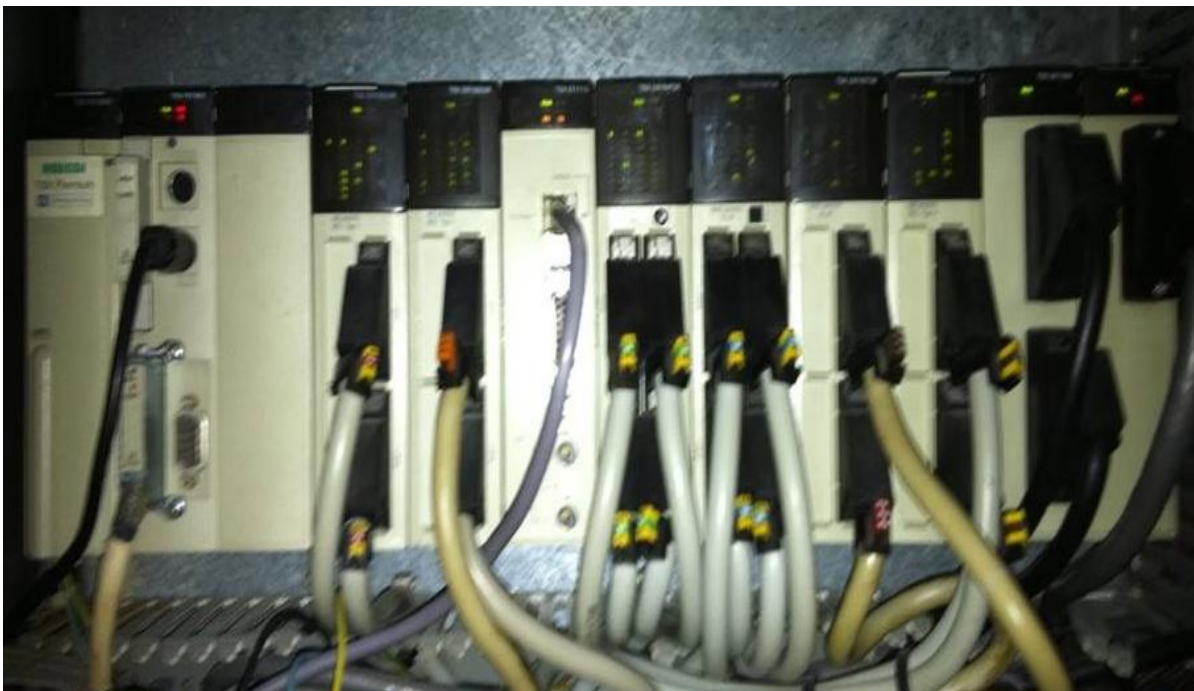


**Automate modulaire**



**Automate compact**

Exemple d'un automate programmable Schneider TSX premium 57452 modulaire qui est utilise dans notre application (régulation d'une trémie).



**Figure: III-12 : L'automate Schneider TSX 57452.**

## 2- Structure interne

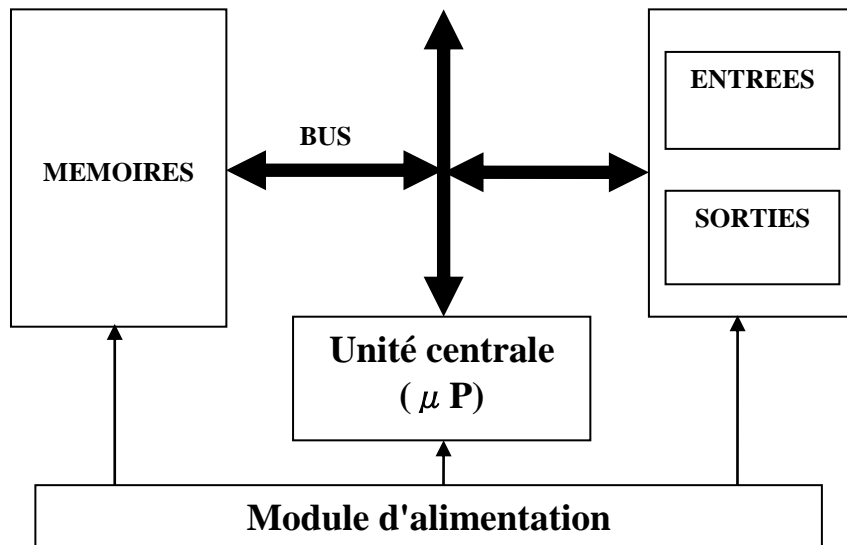


Figure III-13 : Structure interne d'un API.

### ➤ Module d'alimentation

Il assure la distribution d'énergie aux différents modules.

### ➤ Unité centrale

A base de microprocesseur, elle réalise toutes les fonctions logiques, arithmétiques et de traitement numérique (transfert, comptage, temporisation ...).

### ➤ Le bus interne

Il permet la communication de l'ensemble des blocs de l'automate et des éventuelles extensions.

### ➤ Mémoires

Elles permettent de stocker le système d'exploitation (ROM ou PROM), le programme (EEPROM) et les données système lors du fonctionnement (RAM). Cette dernière est généralement secourue par pile ou batterie. On peut, en règle générale, augmenter la capacité mémoire par adjonction de barrettes mémoires type PCMCIA.

### ➤ Interfaces d'entrées / sorties

#### • Interface d'entrée

Elle permet de recevoir les informations du S.A.P. ou du pupitre et de mettre en forme (filtrage, ...) ce signal tout en l'isolant électriquement.

#### • Interface de sortie

Elle permet de commander les divers préactionneurs et éléments de signalisation du S.A.P. tout en assurant l'isolement électrique [11].

### ***III-2-6-6 : Fonctions réalisées par l'API***

Les automates compacts permettent de commander des sorties en T.O.R et gèrent parfois des fonctions de comptage et de traitement analogique.

Les automates modulaires permettent de réaliser de nombreuses autres fonctions grâce à des modules intelligents que l'on dispose sur un ou plusieurs racks. Ces modules ont l'avantage de ne pas surcharger le travail de la CPU car ils disposent bien souvent de leur propre processeur.

#### **❖ Principales fonctions**

##### **1- Cartes d'entrées / sorties**

Au nombre de 4, 8, 16 ou 32, elles peuvent aussi bien réaliser des fonctions d'entrées, de sorties ou les deux.

Ce sont les plus utilisées et les tensions disponibles sont normalisées (24, 48, 110 ou 230V continu ou alternatif ...).

Les voies peuvent être indépendantes ou posséder des "communs".

Les cartes d'entrées permettent de recueillir l'information des capteurs, boutons ... qui lui sont raccordés et de la matérialiser par un **bit image** de l'état du capteur. Les cartes de sorties offrent deux types de technologies : les sorties à relais électromagnétiques (bobine plus contact) et les sorties statiques (à base de transistors ou de triacs).

##### **2- Cartes de comptage rapide**

Elles permettent d'acquérir des informations de fréquences élevées incompatibles avec le temps de traitement de l'automate.

Exemple : signal issu d'un codeur de position.

##### **3- Cartes de commande d'axe**

Elles permettent d'assurer le positionnement avec précision d'élément mécanique selon un ou plusieurs axes. La carte permet par exemple de piloter un servomoteur et de recevoir les informations de positionnement par un codeur. L'asservissement de position pouvant être réalisé en boucle fermée.

##### **4- Cartes d'entrées / sorties analogiques**

Elles permettent de réaliser l'acquisition d'un signal analogique et sa conversion numérique (CAN) indispensable pour assurer un traitement par le microprocesseur. La fonction inverse (sortie analogique) est également réalisée. Les grandeurs analogiques sont normalisées : 0-10V ou 4-20mA.

## 5- Autres cartes :

- Cartes de régulation PID
- Cartes de pesage
- Cartes de communication (Ethernet ...)
- Cartes d'entrées / sorties déportées

### III-2-6-7: Traitement du programme par l'API

Tous les automates fonctionnent selon le même mode opératoire [6].

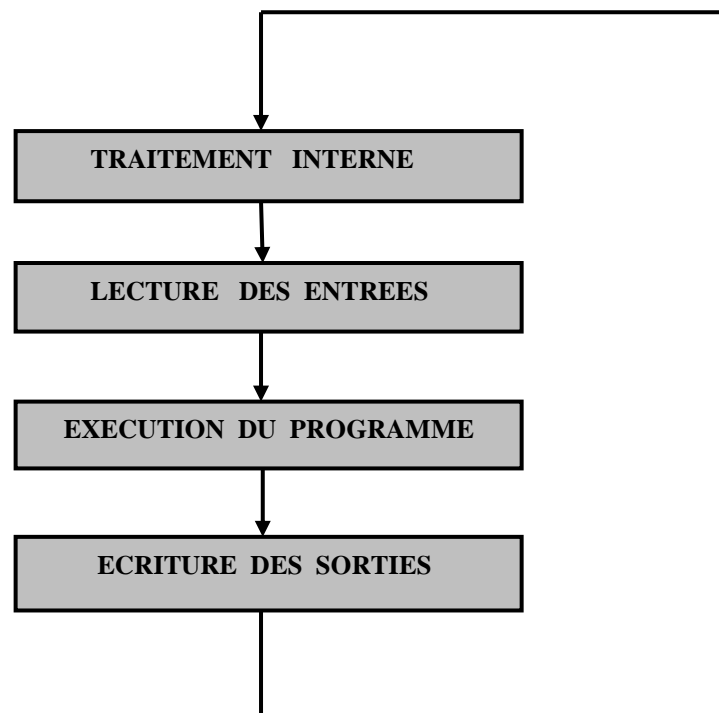


Figure III-14 : Cycle d'un API.

#### 1- Traitement interne

L'automate effectue des opérations de contrôle et met à jour certains paramètres systèmes (détection des passages en RUN / STOP, mises à jour des valeurs de l'horodateur, ...).

#### 2- Lecture des entrées

L'automate lit les entrées (de façon synchrone) et les recopie dans la mémoire image des entrées.

#### 3- Exécution du programme

L'automate exécute le programme instruction par instruction et écrit les sorties dans la mémoire image des sorties.

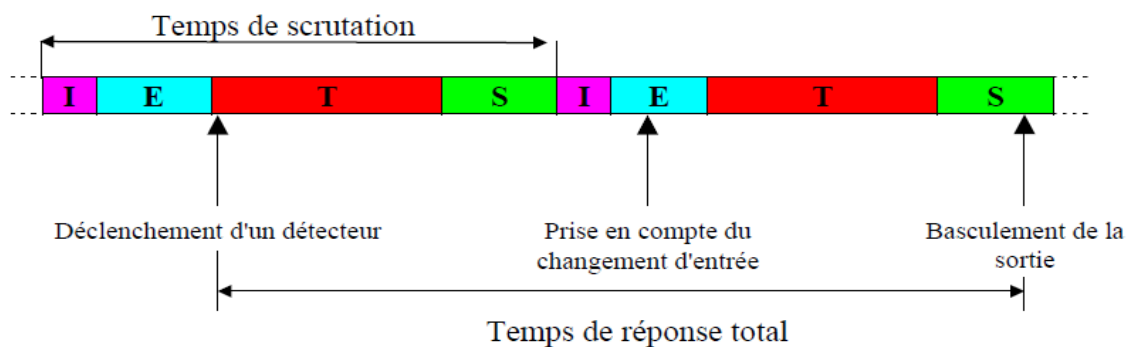


#### 4- Ecriture des sorties

L'automate bascule les différentes sorties (de façon synchrone) aux positions définies dans la mémoire image des sorties.

Ces quatre opérations sont effectuées continuellement par l'automate (fonctionnement cyclique). On appelle **scrutation** l'ensemble des quatre opérations réalisées par l'automate et le **temps de scrutation** est le temps mis par l'automate pour traiter la même partie de programme. Ce temps est de l'ordre de la dizaine de millisecondes pour les applications standards.

**Le temps de réponse total (TRT)** est le temps qui s'écoule entre le changement d'état d'une entrée et le changement d'état de la sortie correspondante :



Le temps de réponse total est au plus égal à deux fois le temps de scrutation (sans traitement particulier).

Le temps de scrutation est directement lié au programme implanté. Ce temps peut être fixé à une valeur précise (fonctionnement périodique), le système indiquera alors tout dépassement de période.

Dans certains cas, on ne peut admettre un temps de réponse aussi long pour certaines entrées, ces entrées pourront alors être traitées par l'automate comme des événements (traitement événementiel) et prises en compte *en priorité* (exemples : problème de sécurité, coupure d'alimentation ...).

Certains automates sont également pourvus d'entrées rapides qui sont prises en compte avant le traitement séquentiel mais le traitement événementiel reste prioritaire [6].

#### III-2-6-8 : Différentes langages de programmation

Chaque automate se programme via une console de programmation propriétaire ou par un ordinateur équipé du logiciel constructeur spécifique [11].

## 1- Les langages textuels

### ✓ Liste d'instructions (IL : Instruction List) :

Langage textuel de même nature que l'assembleur (Programmation des microcontrôleurs).  
Très peu utilisé par les automaticiens

### ✓ littéral structuré (ST : Structured Text):

Langage informatique de même nature que le Pascal, il utilise les fonctions comme *if ... then ... else ...* (si ... alors ... sinon ...) Peut être utilisé par les automaticiens.

## 2- Les langages graphiques

### ✓ Langage à contacts (LD : Ladder Diagram)

Langage graphique développé pour les électriciens. Il utilise les symboles tels que contacts, relais et blocs fonctionnels et s'organise en réseaux (ladder). C'est le plus utilisé.

### ✓ Blocs Fonctionnels (Function Block Diagram)

Langage graphique où des fonctions sont représentées par des rectangles avec les entrées à gauche et les sorties à droite. Les blocs sont programmés (bibliothèque) ou programmables. Utilisé par les automaticiens.

### ✓ Langage SFC (Sequential Function Chart)

Le GRAFCET, langage de spécification, est utilisé par certains constructeurs d'automate (Schneider, Siemens) pour la programmation. Parfois associé à un langage de programmation, il permet une programmation aisée des systèmes séquentiels tout en facilitant la mise au point des programmes ainsi que le dépannage des systèmes. On peut également traduire un Grafcet en langage à contacts et l'implanter sur tout type d'automate. Certains logiciels permettent une programmation totale en langage GRAFCET et permettent de s'adapter à la plupart des automates existants (logiciels CADEPA ou AUTOMGEN).

### **III-2-6-9 : Critères de choix d'un automate**

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

Le personnel de maintenance doit toutefois être formé sur ces matériels et une trop grande diversité des matériels peut avoir de graves répercussions. Un automate utilisant des langages de programmation de type GRAFCET est également préférable pour assurer les mises au point et dépannages dans les meilleures conditions. Il faut ensuite quantifier les besoins [6].

### **Nombre d'entrées / sorties**

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

### **Type de processeur**

La taille mémoire, la vitesse de traitement et les fonctions spéciales offertes par le processeur permettront le choix dans la gamme souvent très étendue.

### **Fonctions ou modules spéciaux**

Certaines cartes (commande d'axe, pesage ...) permettront de "soulager" le processeur et devront offrir les caractéristiques souhaitées (résolution, ...).

### **Fonctions de communication**

L'automate doit pouvoir communiquer avec les autres systèmes de commande (API, supervision ...) et offrir des possibilités de communication avec des standards normalisés (Profibus ...).

## **III-3 : Conclusion**

Nous avons entamé ce chapitre par une présentation des notions sur la régulation automatique et une présentation de ses différents types et différents critères et performances.

En suite nous avons présenté les éléments de la boucle de régulation (notre application) tout en donnant les généralités et en expliquant le rôle de chacun d'entre eux.

# Chapitre IV

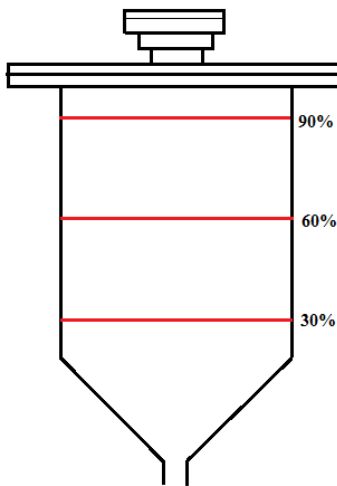
## Résultats et simulation

## IV- 1 : Introduction

Dans ce chapitre, nous allons présenter une solution à la problématique du projet, et présenter les résultats obtenus dans la partie réalisation et la partie programmation.

## IV-2 : Solution apportée

Nous avons divisé le niveau de la trémie sous forme d'intervalles comme suit 0-30%, 30-60%, 60-90% comme le montre la figure (IV-1) de telle sorte à avoir une régulation avec un pallier de trois vitesses c'est-à-dire que dans l'intervalle 0-30% la vitesse de la pompe (organe d'exécution) fonctionne avec la vitesse maximale 500 tr/min 50 Hz et que dans l'intervalle 30-60% la pompe fonctionne avec une vitesse réduite 300 tr/min 30 Hz et enfin dans le dernier intervalle 60-90% la pompe fonctionne avec la vitesse minimale 100 tr/min, 15Hz afin d'avoir, dans le dernier cas (troisième vitesse), le débit entrant dans la trémie équivalent au débit sortant vers le doseur de la machine conditionneuse.



Niveau trémie	Vitesses du variateur
0 --- 30%	500 tr/min 50 Hz
30 --- 60%	300 tr/min 25 Hz
60 --- 90%	100 tr/min 10 Hz
>90%	Arrêt de la pompe

**Tableau IV-1:** Les différentes vitesses.

**Figure IV-1:** Les intervalles de la trémie.

Le tableau suivant résume les différentes entrées/sorties, et les objets mémoire utilisés dans notre programme avec leur symbole attribué.

Adresse	Symbole
%IW2.0 Entrée de l'automate	Niveau_trémie
%Q3.0 Sortie de l'automate	Marche
%Q3.1 Sortie de l'automate	Bit_0
%Q3.2 Sortie de l'automate	Bit_1

%M0 Objet mémoire	Toujour_0
%M1 Objet mémoire	Toujour_1

**Tableau IV-2 :** Les adresses de l'automate.

Les vitesses sont nommées respectivement vitesse pré-réglée N°1 Pr1 et vitesse pré-réglée N°2 Pr2 et enfin la vitesse pré-réglée N°3 Pr3.

Pour ce faire nous allons prendre une combinaison de deux bits (sortie de l'automate) le bit\_0 qui est équivalent à la sortie Q3.1 et le bit\_1 est équivalent à la sortie Q3.2. Donc pour la première combinaison qui est **bit\_0 = 0** et **bit\_1 = 0** elle représente la vitesse pré-réglée 1 **Pr1**, et la combinaison **bit\_0= 0** et **bit\_1 = 1** elle représente la vitesse pré-réglée 2 **Pr2**, et la dernière vitesse pré-réglée **Pr3**, est représentée par la combinaison **bit\_0 = 1** et **bit\_1 = 0** et la combinaison bit\_0 = 1 et bit\_1 = 1 n'est pas utilisée.

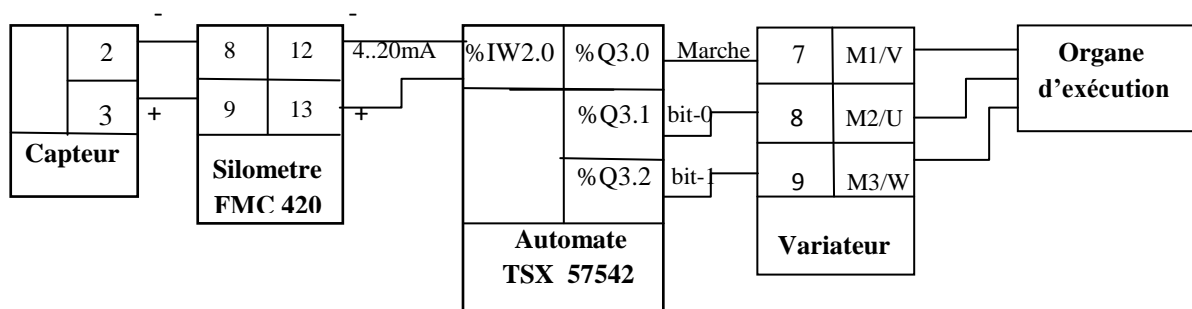
Bit_0	Bit_1	Vitesses
0	0	Pr1
0	1	Pr2
1	0	Pr3
1	1	X

**Tableau IV-3 :** Les différentes combinaisons.

### IV-3 : Partie matérielle

#### IV-3-1 : Section électrique

Dans cette partie nous allons présenter le câblage des modules d'entrées et sorties de l'automate ainsi que tous les éléments du système sous forme d'un circuit de commande.



**Figure IV-2 :** Partie commande du système.

### IV-3-2 : Section paramétrage

Dans cette partie nous avons paramétré le variateur de vitesse afin d'avoir les vitesses pré-réglées.

Premièrement on a câblé les deux borniers **8** et **9** du variateur de vitesses avec un câble provenant des sorties Q3.1 et Q3.2 de l'automate. Ces sorties sont des sorties booléennes et chaque sortie varie entre 0 et 24 (v).

Le paramétrage se fait de la façon suivante :

On appuie sur le bouton menu puis grâce aux touches de directions on sélectionne, avec le même bouton, le paramètre **P13** qui est CHOIX CONSIGNE grâce aux touches de direction on choisit la consigne vitesses pré-réglées puis de la même façon on choisit **P1** qui est la vitesse pré-réglée N°1 puis **P8** qui représente la vitesse pré-réglée N°2 et en fin **P9** qui est la vitesse pré-réglée N°3.

La borne utilisée pour régler la consigne de vitesse du moteur dépend de la valeur du paramètre P13 comme le montre le tableau IV-4 ci-dessous [14].

Réglage P13	Borne 8	Borne 9	Source Consigne
<b>0</b>	0V	0V	Borne 2 (0-10V) - avant
	0V	24V	Vitesse Jog (fixée par paramètre P8) - avant
	24V	0V	Borne 2 (0-10V) - arrière
	24V	24V	Vitesse Jog (fixée par P8)- arrière
<b>1</b>	0V	0V	Borne 3 (4-20mA) - avant
	0V	24V	Vitesse Jog (fixée par paramètre P8) - avant
	24V	0V	Borne 3 (4-20mA) - arrière
	24V	24V	Vitesse Jog (fixe par P8)- arrière
<b>2</b>	0V	0V	Vitesse Pré-réglée 1 (fixée par P1)
	24V	0V	Vitesse Pré-réglée 2 (fixée par P8)
	0V	24V	Vitesse Pré-réglée 3 (fixée par P9)
	24V	24V	Vitesse Pré-réglée 4 (fixée par P2)

**Tableau IV-4** : Les vitesses pré-réglées.

#### IV-4 : Partie logicielle

Dans cette partie nous allons présenter les Graficets et le programme principal sous le logiciel PL7.

##### IV-4-1 : Organigrammes

###### 1- Organigramme N°1

Cet organigramme représente le système général.

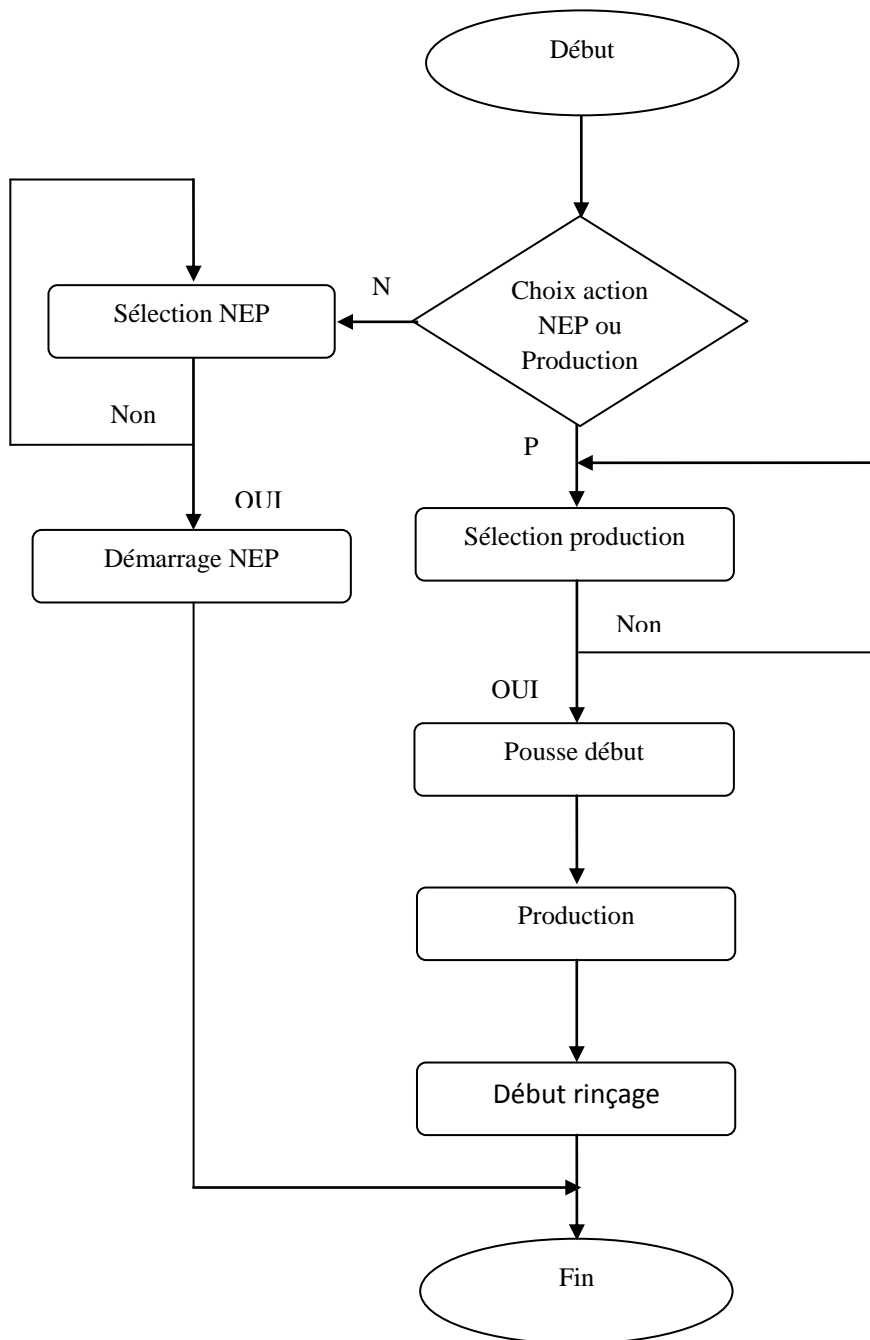


Figure IV-3 : Organigramme du circuit général.



## 2- Organigramme N°2

Dans cet organigramme on représente le système déjà existant avant notre modification.

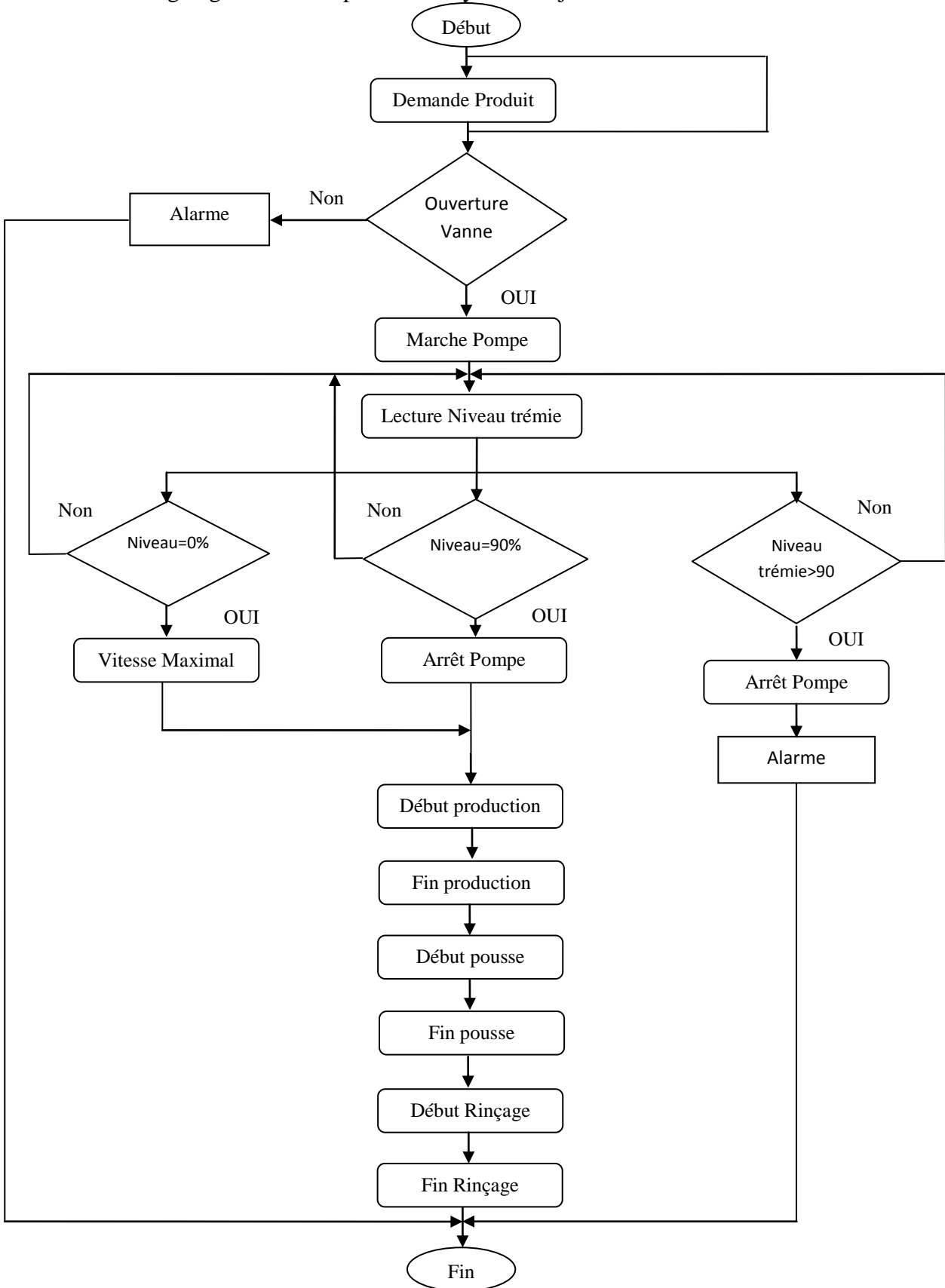


Figure IV-4 : Organigramme du système déjà existant.

### 3- Organigramme N°3

Dans cet organigramme nous allons représenter la partie nous avons élaboré.

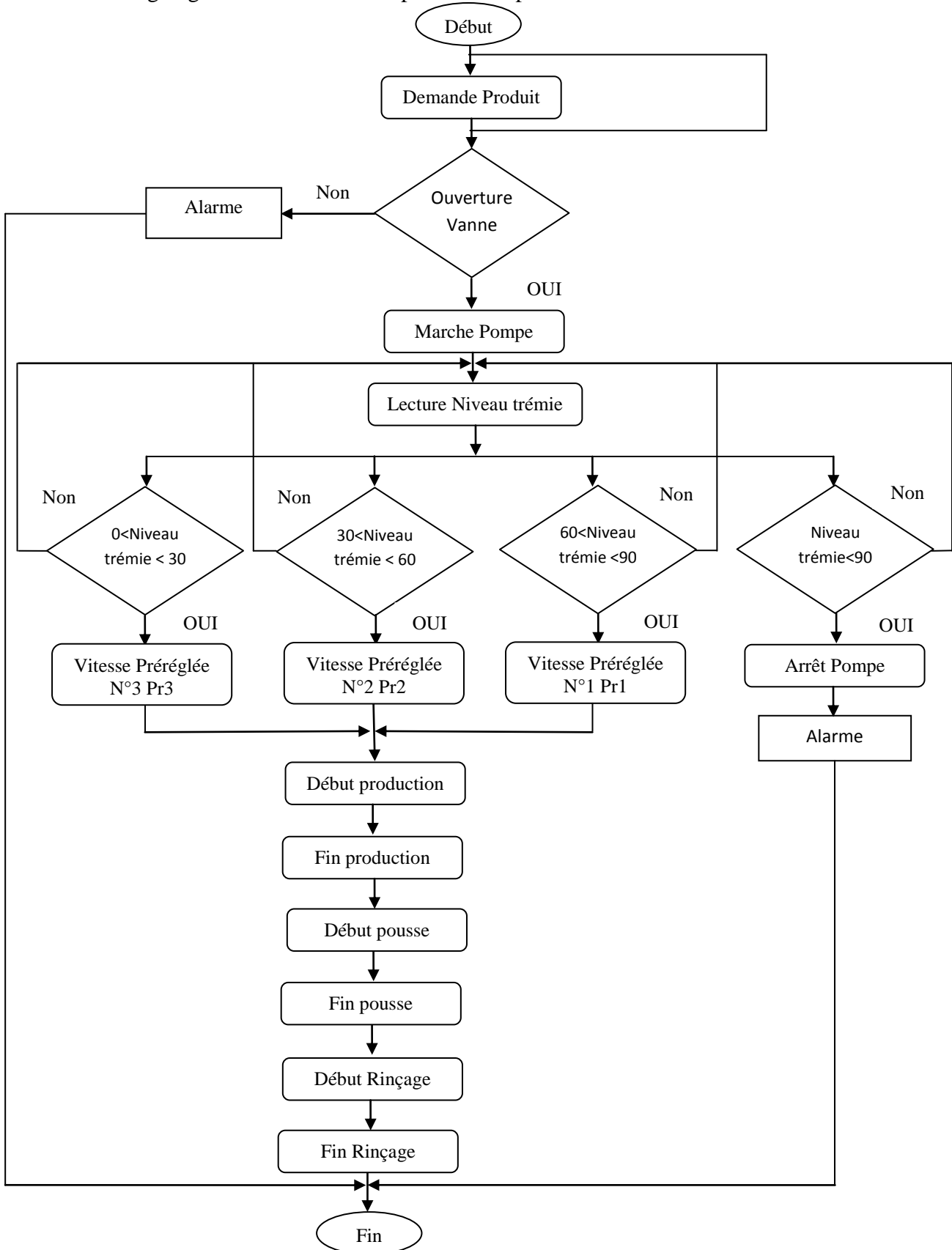


Figure IV-5 : Organigramme de système modifié.

#### IV-4-2 : Grafcet et Programme sous PL7

Dans le programme générale, on trouve deux programmes, une macro étape avec le langage Ladder qui force **%M0** ou l'objet mémoire (toujours\_0) à zéro (reset) et **%M1** ou l'objet mémoire (toujours\_1) à 1 (set), voir la Figure IV-6 et un programme principal, voir la figure IV-7 qui gère les différentes vitesses du variateur qui sont les étapes **11, 12 et 13**.

Pour les étapes **4** et **14** elles gèrent seulement la marche et l'arrêt du variateur une fois que le niveau de la trémie est supérieur à 90%.

L'entrée analogique de l'automate **%IW2.0** est symbolisée (**niveau trémie**). Les sorties **%Q3.0,%Q3.1**et **%Q3.2** sont symbolisées successivement (**marche, bit\_0, bit\_1**)

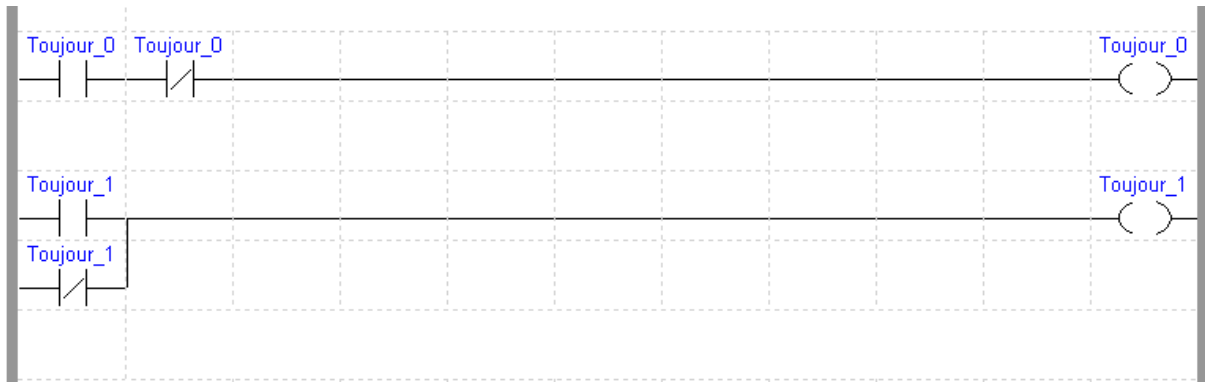


Figure IV-6 : La macro étapes avec Ladder.

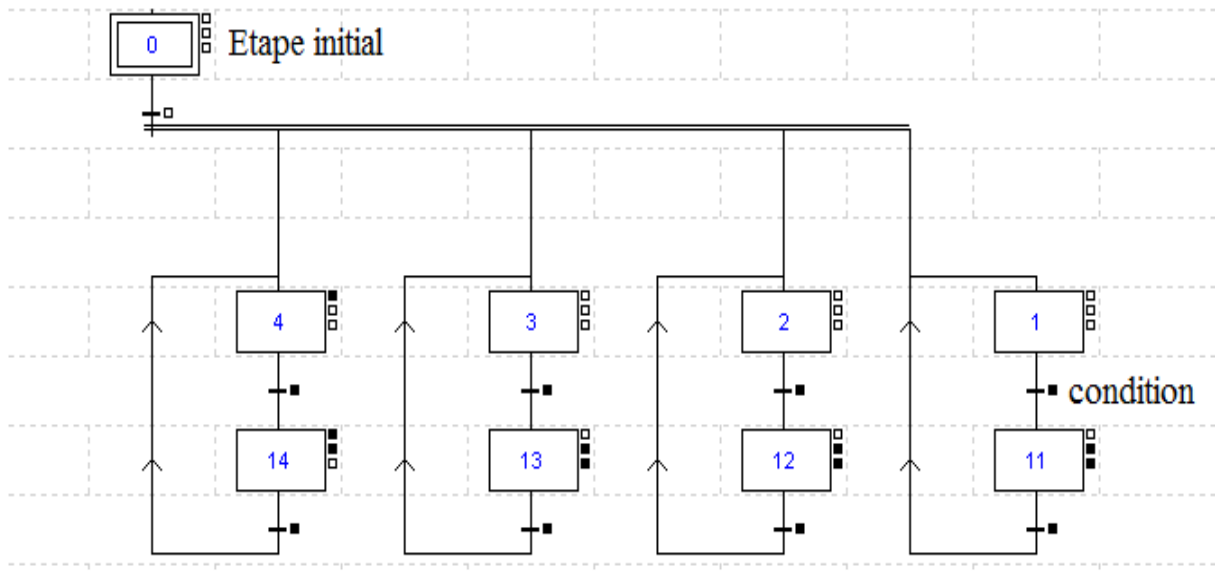


Figure IV-7 : Grafcet principale sous PL7.

❖ Pour plus de détails sur le Grafcet voir **annexe B**

## IV-5 : Supervision de la production

Grace au logiciel de supervision InTouch on a réalisé une vus du système.

### ✓ Premier cas

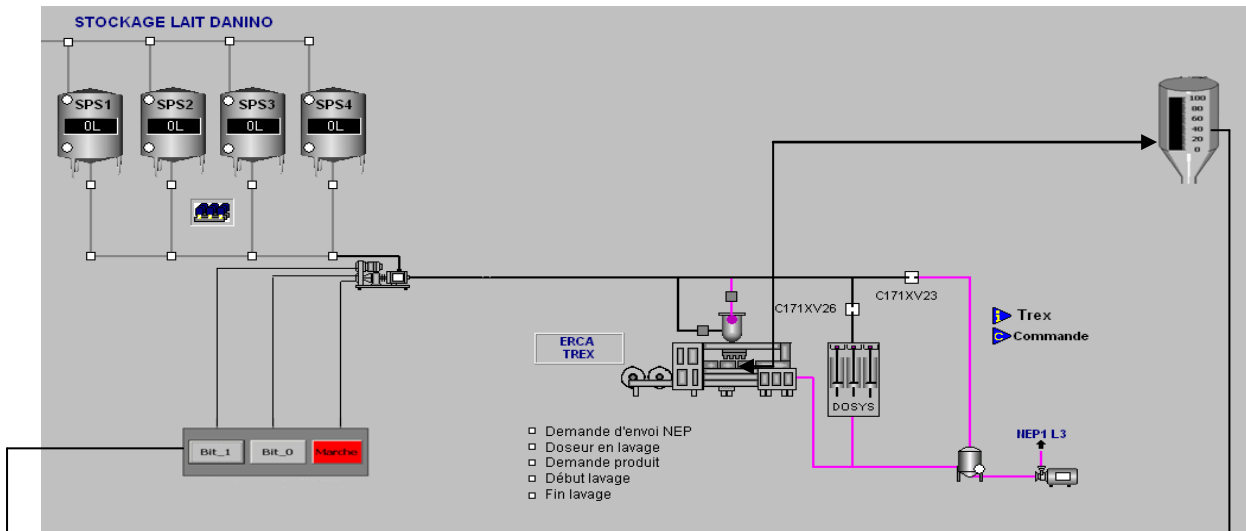


Figure IV-8 : Vue du système à l'arrêt.

### ✓ Deuxième cas

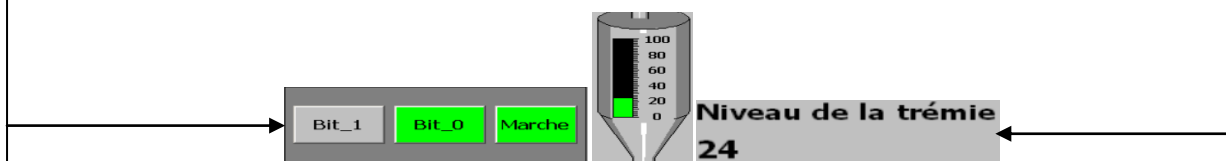


Figure IV-9 : Vitesse pré réglée N°3.

### ✓ Troisième cas

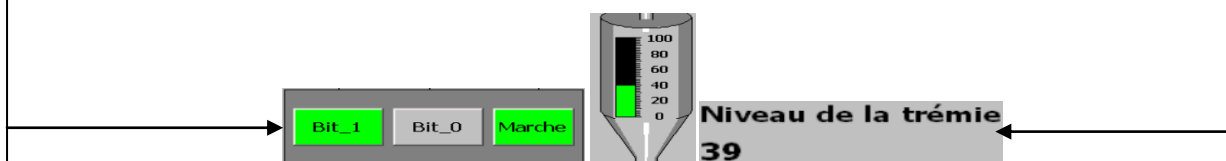


Figure IV-10 : Vitesse pré réglée N°2.

### ✓ Quatrième cas

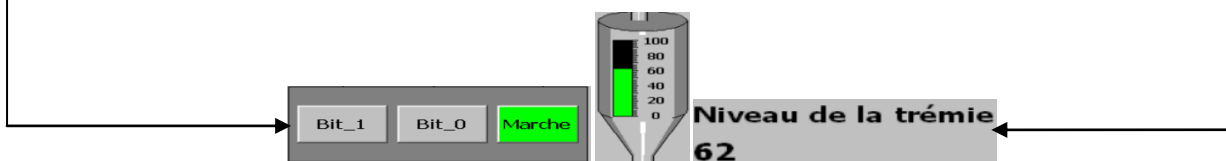


Figure IV-11 : Vitesse pré réglée N°1.

## **IV-6 : Interprétation des résultats**

### **1- Premier cas**

Dans ce cas, Figure IV-8, le système est à l'arrêt et on voit au bouton marche qui est au rouge, ainsi qu'à la trémie qui est vide.

### **2- Deuxième cas**

Dans la figure IV-9, on voit que le système est en production (Marche est en vert) et que le niveau de la trémie est à 24%. La combinaison est la suivante : **Bit\_0 = 1** et **Bit\_1 = 0** ce qui veut dire que la vitesse de la pompe est maximale Pr3, 500 tr/min 50 Hz.

### **3- Troisième cas**

Dans la figure IV-10, on voit que le système est en production (Marche est en vert) et que le niveau de la trémie est à 39%. La combinaison est la suivante : **Bit\_0 = 0** et **Bit\_1 = 1** ce qui veut dire que la vitesse de la pompe est réduite Pr2 300 tr/min 30Hz.

### **4- Quatrième cas**

Dans la figure IV-11, on voit que le système est en production (Marche est en vert) et que le niveau de la trémie est à 62%. La combinaison est la suivante : **Bit\_0 = 0** et **Bit\_1 = 0** ce qui veut dire que la vitesse de la pompe est minimal Pr1 100 tr/min 10 Hz.

## **IV-7 : Conclusion**

Dans ce chapitre nous avons présenté la solution apportée en deux parties, la partie matérielle et la partie logicielle. A la fin on a présenté la simulation de la modification effectuée sous formes d'images dans le logiciel InTouch.

Les résultats que nous avons trouvés sont concluants, à savoir une meilleure régulation de la trémie évitant ainsi tous les inconvénients de la régulation tout ou rien TOR.

# Conclusion Générale

## **Conclusion générale**

Le travail que nous avons effectué au sein de l'unité Danone Djurdjura Algérie nous a permis d'acquérir des connaissances techniques et pratiques ce qui nous a permis de nous familiariser avec le milieu industriel.

L'objectif de notre travail était de commander une trémie avec un pallier de trois vitesses.

Nous avons entamé notre travail par la présentation du circuit existant et son mode de fonctionnement.

La commande proposée est donnée par un Grafcet, ensuite nous avons élaboré le programme avec PL7, remplissant toutes les conditions nécessaires au bon fonctionnement du processus et puis nous avons paramétré le variateur de vitesses et enfin élaboré une interface graphique avec le InTouch qui permet la visualisation de la simulation.

Nous espérons avoir répondu à notre cahier de charge et ce présent travail sera utile aux productions à suivre.

# Bibliographie



# Bibliographie

- [1] OUAZZAR Idir « Etude et conception d'un système de régulation de niveau de la trémie » Université de Boumerdes, mémoire de fin d'étude de master II, 2011/2012.
- [2] Cédric. DEMERS-ROY « Présentation du logiciel PL7-Pro » Laboratoires du cours ELE 4202 Septembre, 2003.
- [3] Louis COUFFIGNAL « Introduction PL7 Pro » Lycée polyvalent Strasbourg 2005-2006.
- [4] Université de Lille I « Informatique Industrielle II – Supervision » Master GSI 2011/2012.
- [5] Eric Magarotto « Cour de régulation » Université de Caen, 2004/2005.
- [6] Philippe LE BRUN « Automates programmables industriels » Lycée Louis ARMAND Strasbourg, 2001.
- [7] Michel Itique « Régulation numérique » Edit. EYROLLES 61 Boulevard Saint-Germain, 75005 Paris. Yverdon-les-Bains, 2013
- [8] Michel BENSOAM « Notions d'asservissements et de Régulations », Edit. Patrique Gounidou. 2000-2001.
- [9] M. Rabi « Régulation analogique industrielle » ESTF- G.Thermique- G.Procédés ,1986.
- [10] Mathieu Auclerc « Régulation et automatisme des systèmes » Edit. DUNOD Paris. René Prigent, 2013
- [11] Alain GONZAGA « Les automates programmables industriels » Edit. Ellipses Marketing S.A Novembre, 2004
- [12] TOUAT Sahra « Conception et Réalisation d'une Armoire Electrique Automatisée » Université de Bejaia Mémoire de projet de fin d'études, 2012/2013

## Site web

- [13] [WWW.Endress+Hauser.com/FMC420](http://WWW.Endress+Hauser.com/FMC420)
- [14] [WWW.Eurotherm.com/601](http://WWW.Eurotherm.com/601)
- [15] [WWW.Pcmpompe.com](http://WWW.Pcmpompe.com)

# Annexes

# Annexe A

## I: Description de l'entreprise

### I-1: Aperçu historique

Les origines de groupe DANONE remonte à 1966, lors de la fusion de deux sociétés verrières françaises, glace de Boussois et verrières Souchon Newesel «BSN».

En 1973, le groupe «BSN» et Gervais DANONE réalisent un chiffre d'affaire très important dans les produits laitiers et les pâtes, et devient le premier groupe français.

En 1989, le groupe «BSN» était le troisième groupe agroalimentaire européen et le premier en France, en Italie et en Espagne.

En 1990 le groupe a adopté une stratégie de consolidation des positions acquises, il a acquis Volvic en France afin de renforcer sa position les activités d'eau en bouteille.

A partir de 1997 il a engagé un programme de recentrage sur trois mesures (produits laitiers frais, boissons et biscuits, snacks céréaliers)

En octobre 2001 le leader mondial DANONE a conclu un partenariat avec la laiterie DJURDJURA.

### I-2: Situation géographique

Située dans la zone industrielle TAHARACHT commune D'AKBOU sur la route national n° 26 à 70 km du centre de la wilaya de BEJAIA et à 170 à l'est de la capitale d'ALGER.

### I-3 : Description de quelque structure de l'entreprise

#### ➤ Direction générale

Dirigé par un directeur générale (DG) qui assure l'application des décisions lors des différents conseils d'administration et les travaux entre les différents services

#### ➤ Direction des ressources humaines (DRH)

Sa mission est d'élaborer une politique sociale et la gestion du personnel de l'entreprise (recrutement, formation ...)

➤ **Direction comptabilité et finances**

La direction de comptabilité et de finances a pour objectif la conception et la gestion des outils et les méthodes comptable et finances sur tous les niveaux de l'entreprise

➤ **Direction d'achats**

Ce service est muni de deux section d'achats locaux et étrange ou il prend charge la gestion des achats et assure le suivi des commandes jusqu'à leur satisfaction en assurent le délai compatible.

➤ **Direction marketing**

Assure les meilleures conditions de vente des produits

➤ **Direction d'usine**

Elle est subdivisées en deux départements :

❖ **Département technique**

Son rôle est la maintenance des équipements de la production on

- Veillant au bon fonctionnement des équipements de production, réglage de la machine
- Assurant la maintenance et l'entretien des machines a production
- Prévoir les pannes éventuelles

❖ **Département de production**

Il est charge de manufacturer les différents produits commercialises par l'entreprise

**I-4 : produits de l'unité**

La laiterie produit les dérivés suivants

- ✓ Yaourt étuvé aromatisé
- ✓ Yaourt nature
- ✓ Crème dessert (Danette)
- ✓ Fromage à pâte fraîche (PGF)
- ✓ Yaourt probiotique (Bioactivia au Bifidus Regularis)
- ✓ Danao
- ✓ Yaourt brassé (Dun' up)

**II : Processus de fabrication de yaourt**

**II-1 : Définition Yaourt**

Le yaourt ou yoghourt est à la fois le lait fermenté le plus consommé et le mieux connu. Cette dénomination est réservée au produit laitier coagulé obtenu par fermentation lactique

grâce à l'action de deux bactéries lactiques thermophiles spécifiques (*streptococcus thermophilus* et *lactobacillus delbrueckii ssp bulgaricus*) du lait pasteurisé avec ou sans addition (lait en poudre ...) .les bactéries dans le produit fini doivent être vivantes et présentes en abondance (FAO 1975) .la réglementation française fixe le nombre minimal à 10 millions de bactéries par gramme

## **II-2: Approvisionnement de la matière première**

Pour la fabrication de ces produits, l'unité importe les matières premières nécessaires (le lait cru, le lait en poudre, le sucre, les arômes, les ferments lactiques, l'eau, etc.)

### **II-2-3 : Etapes de production**

#### **1- Standardisation**

La composition du lait varie selon l'âge de l'animal, sa race, la saison, le moment de la traite. Le taux de matière grasse peut s'élever de 30a 40 g/litre notamment.la standardisation permet d'harmoniser la composition du lait de diverse exploitations elle se fait dans une cuve qu'on appelle écrémeuse

#### **2- Dégazage**

Le but est de débarrasser le lait des mauvaises odeurs. Il se fait à 60° C dans une cuve dans laquelle on crée un vide par baisse de pression. A cette température le lait bouilli et les gazes s'échappent par évaporation

#### **3- Homogénéisation**

C'est un traitement physique du lait de vache standardisé et dégazé il se fait de manière suivante : on fait passer le lait sous pression élevée à travers des orifices étroits afin de réduire la taille des globules gras du lait. Cette opération améliore la consistance du lait accroît sa blancheur et rend les lipides plus digestibles en outre elle empêche la matière grasse de se déposer à la surface et sur l'emballage

#### **4- Pre-pasteurisation**

Le lait homogénéisé est stocké dans des cuves verticales à double paroi pour être soumis à la pre-pasteurisation qui vise à détruire les bactéries pathogènes ou non pathogènes elle se fait

instantanément c'est-à-dire en quelle que secondes en chauffant a 75°C ensuite a 5°C et en garde réfrigère

### **5- Enrichissement**

Le lait homogénéise et pre-pasteurise est renforce en poudre de lait si sa composition n'est pas conforme aux normes internationales de fabrication du yaourt le mélange lait cru + poudre de lait se fait dans des grandes cuves on ajoute ensuite du sucre et en laisse reposer le mélange une demi-heure

### **6- Pasteurisation**

On procède a la pasteurisation finale du lait enrichi et homogénéise en chauffant a 93°C pendant 5 mn puis au refroidissement ce lait qu'on appelle lait de yaourt est maintenant prêt a être fermenté

### **7- Fermentation**

Le lait de yaourt refroidit est ensemencé avec des ferments lactiques sélectionnés afin d'accélérer la fermentation. Le lait fermenté est destiné à la fabrication du yaourt ferme ou du yaourt brassé dans ce qui suit nous allons expliquer comment on obtient le yaourt ferme et le yaourt brassé

#### **➤ Yaourt ferme :**

On refroidit le yaourt à 4° C et on ensemencé avec les ferments lactiques. On chauffe à l'aide d'échangeurs à plaques à 42° C ou 45° C pour l'activité de ces ferments, puis on procède à la mise en pots et à la romanisation

Les pots ensuite destinés à l'incubation dans des chambres froides pour la maturation qui peut durer 3 à 4 heures à une température de 43° C

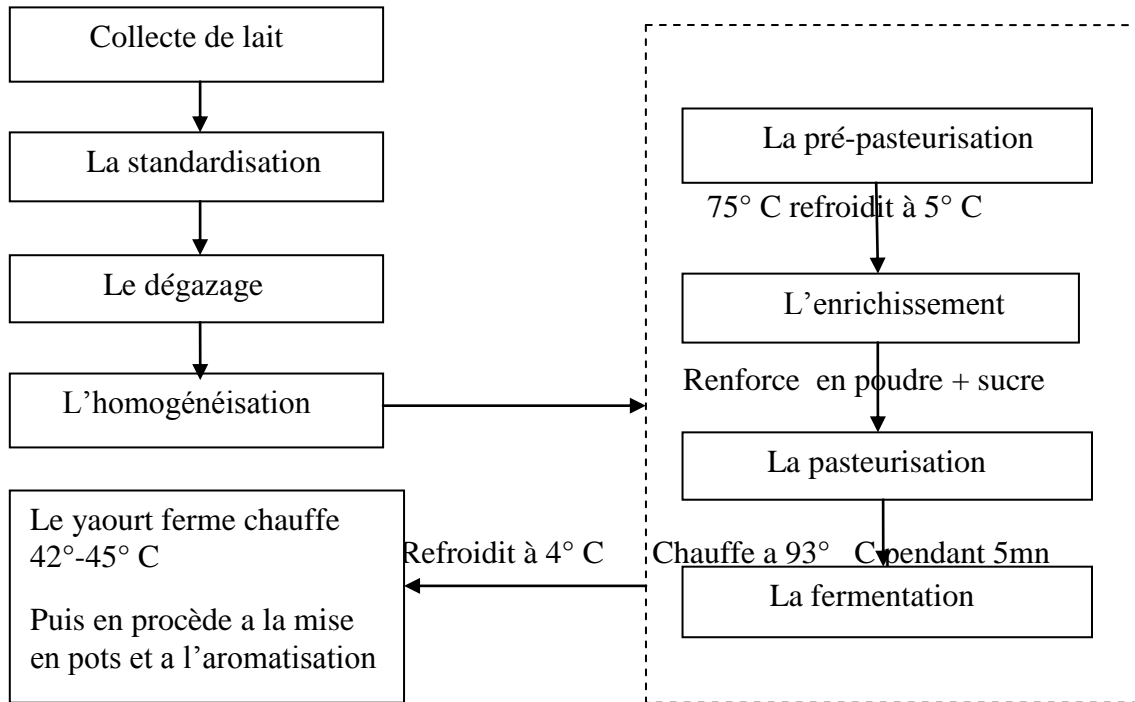
Un refroidisseur énergique et rapide en dessous de 5° C en chambre froide, bloque l'acidité et maintient la coagulation dans l'état de consistance souhaité

#### **➤ Yaourt brassé :**

Le lait de yaourt est réchauffé à 42° C ou 45° C par son passage dans des échangeurs à plaques il est ensuite ensemencé avec les ferments lactiques dans des cuves. Après 3 à 4 heures il est refroidit à 4° C puis conditionné, c'est-à-dire mis en pots ou bouteilles avec ou sans fruit, par une pompe doseuse

## 8- Conditionnement

C'est la dernière étape de fabrication. Elle nous permet de doser le yaourt dans des pots avec une conditionneuse



**Figure 1** : Schéma asymptotique de fabrication de yaourt

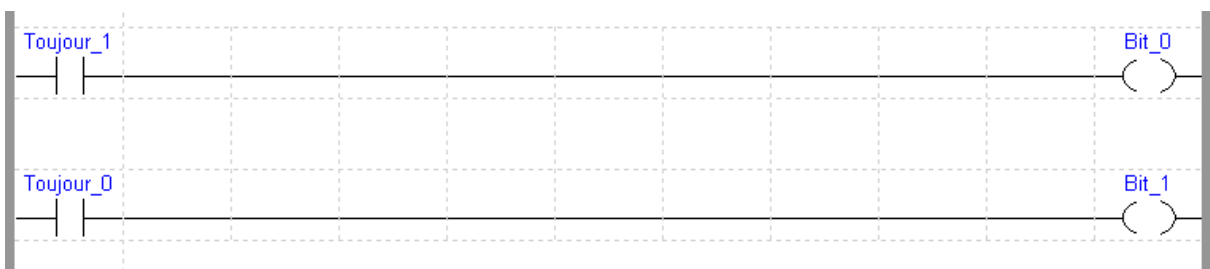
# Annexe B

## 1-Première vitesse (Pr 3)

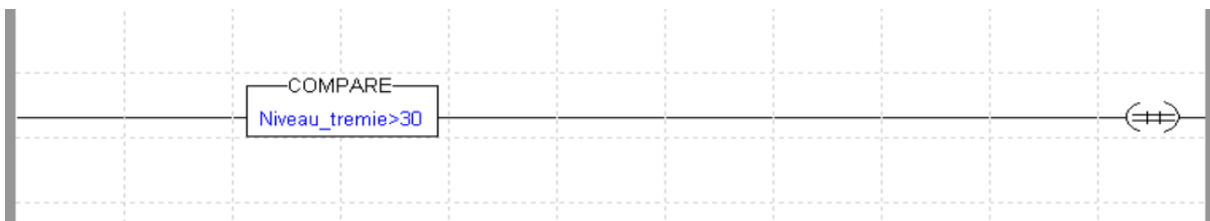
- La condition d'activation



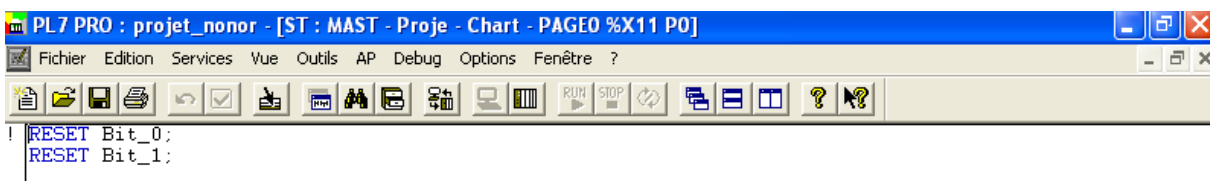
- L'action continue dans l'étape



- LA condition de désactivation

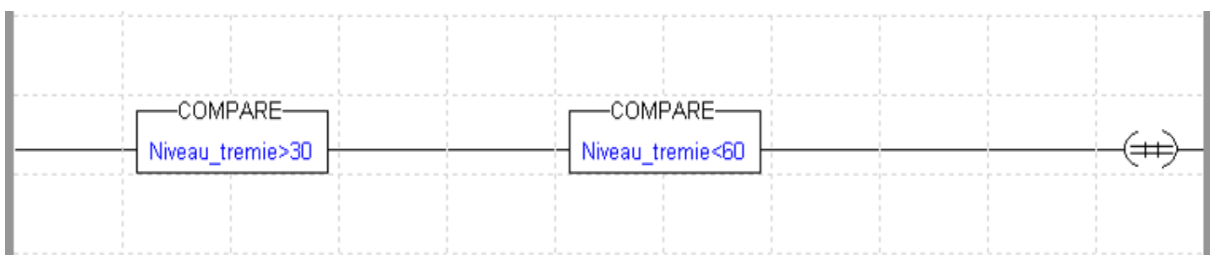


- Reset des bit\_0 et bit\_1 ou %q3.1 et %q3.2



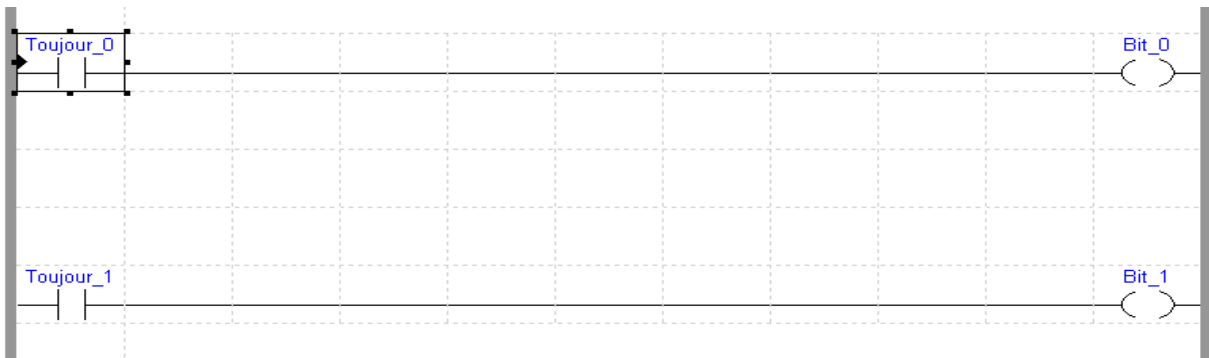
## 2- Deuxième vitesse (Pr 2)

- Condition d'activation

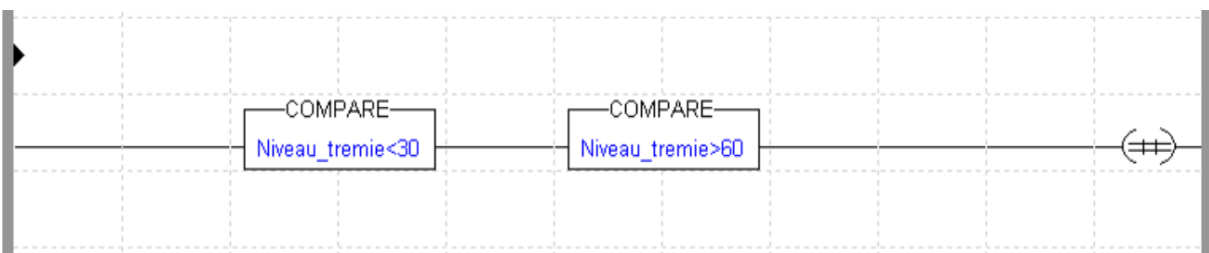




- L'action continue dans l'étape

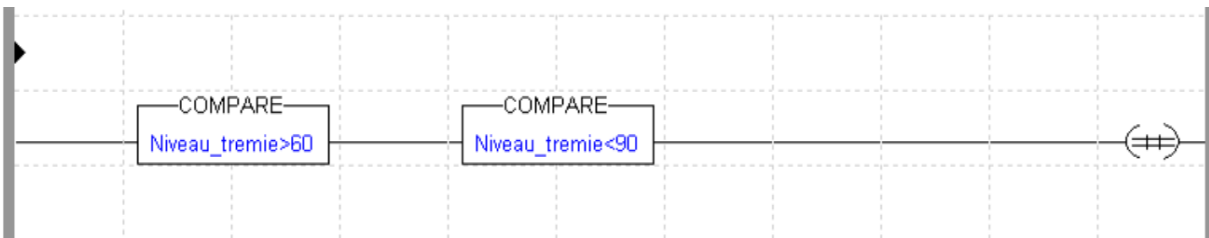


- Condition de désactivation

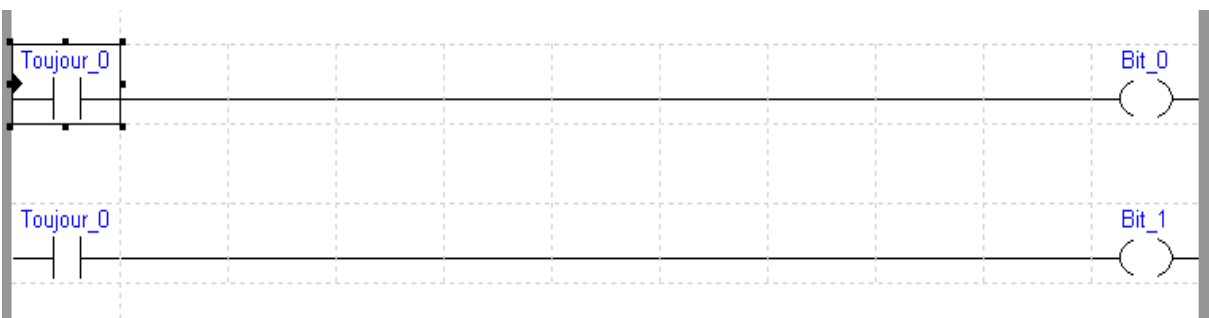


### 3-Troisième vitesse (Pr 1)

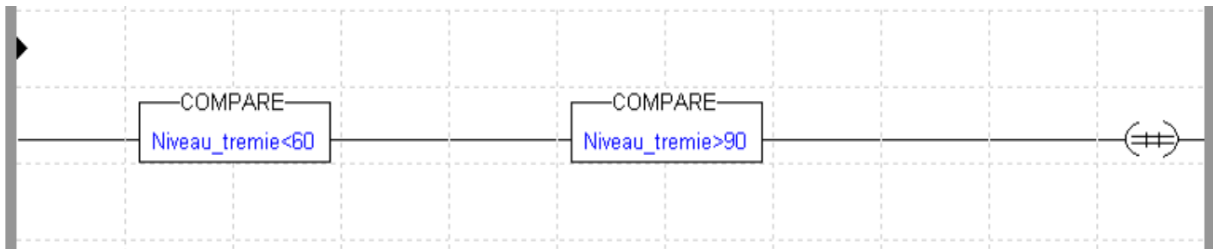
- Condition d'activation



- L'action continue dans l'étape

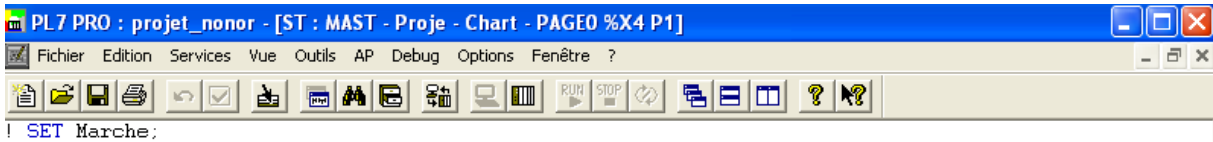


- Condition de désactivation



#### 4-marche et l'arrêt du variateur

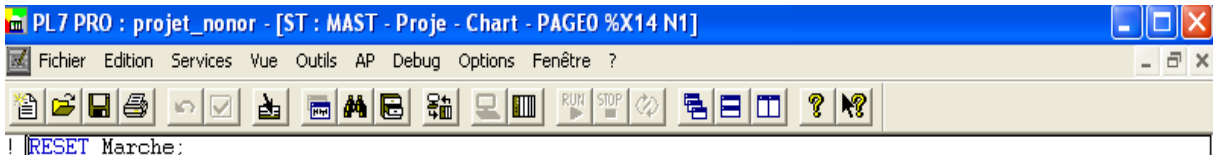
- Marche étape 4



- Condition de l'Arrêt



- L'action continue dans l'étape



- Condition de redémarrage

