

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET  
POPULAIRE**  
**Ministère de l'Enseignement Supérieur et de la Recherche  
Scientifique**  
**Université Mouloud MAMMARI de Tizi-Ouzou**



**Faculté de Génie Electrique et d'Informatique**  
**Département d'Automatique**

**PROJET DE FIN D'ETUDES**

En vue de l'obtention du diplôme

*D'INGENIEUR D'ETAT EN AUTOMATIQUE*

*Thème*

**Automatisation est supervision d'une station de  
transport du sucre à l'usine Coca-Cola  
Rouiba**

Proposé par : Coca cola

Présenté par : Mr Akrou Sofiane  
Mr Arab Nourdine

Dirigé par : M<sup>elle</sup> O.Chilali

Co-Promoteur : F.Bachi

Soutenu le : 04/ 07 /2009

Devant le jury d'examen composé de :

Président : A.MAIDI

Examineur : M.HADOUCHE

Examineur : O.ADJMOUT

*Promotion 2009*

# Remerciements

Nous tenons remercier notre promotrice Melle CHILALI pour son aide et ses conseils durant l'élaboration de ce projet.

Ainsi notre Co-promoteur Mr BACHI FARID « chef de département de l'automatisme à Fruitall Coca-Cola » et monsieur ARKOUB RACHID « Ingénieur en automatique » au sein d'usine du ciment à Alger(SCAL).

Nous remercions également les membres de jury qui feront l'honneur de juger notre travail, d'apporter leurs réflexions et suggestions scientifiques.

Nos remerciements les plus chaleureux s'adressent à nos familles.

Nos derniers remerciements s'adressent à tous ceux qui ont contribué de près ou de loin à la réalisation de ce projet.

# SOMMAIRE

❖ INTRODUCTION GENERALE .....	01
❖ CHAPITRE I : Description et fonctionnement du transporteur	
1-Introduction .....	03
2-Présentation de la station .....	03
3- Description et représentation des éléments de la station.....	04
3-1- Trémie 1 .....	04
3.1.1. Présentation .....	04
3.1.2. Fonctionnement .....	04
3.2. Convoyeurs à vis sans fin .....	05
3.2.1. Présentation .....	05
3.2.2. Fonctionnement .....	06
3.3. Silo .....	06
3.3.1. Présentation .....	06
3.4. Trémie 2 .....	07
3.4.1. Présentation .....	07
3.4.2. Fonctionnement .....	08
3-5 Moteurs asynchrones .....	08
4- Fonctionnement générale de la station .....	11
5- Position de problème .....	11
6-Conclusion .....	12
❖ CHAPITRE II : Conception et Modélisation du nouveau système	
1-Introduction .....	13
2- Définition .....	13
3- Structure d'un système automatisé de production .....	13
3-1 La partie opérative .....	14
3-2 La partie commande .....	15
3-3 La partie dialogue .....	15
4- Les éléments de base du GRAFCET.....	15
4-1Etape .....	15
4-1-1-Etape initiale.....	16
4-1-2-Les actions associées à une étape .....	16
4-2 Transition .....	18
5-Règles d'évolution .....	19
6-Structure de base .....	19
6-1 Divergence en ET.....	19
6-2 Convergence en ET .....	20
6-2 Divergence en OU.....	20
6-3 Convergence en OU .....	21
7-Niveau de GRAFCET .....	21
8-Modifications apportés à la station du transport de sucre .....	22
8-1 Capteurs de niveau .....	22

8.1.1. Caractéristiques .....	23
8.1.2. Fonctionnement .....	23
8-2 Variateur de vitesses .....	23
8-3 Détecteur de rotation.....	24
8-4 Vanne .....	25
9- Modélisation du nouveau système .....	25
9-1 Graficet niveau 1 .....	25
9-2 Listes des actionneurs et pré-actionneurs et capteurs .....	27
9-3 Graficet niveau 2.....	27
10- Conclusion .....	29

### ❖ CHAPITRE III : Automatisation de l'installation de transport du sucre

1-Introduction .....	28
2- Définition d'un automate .....	28
3- Choix d'un automate .....	30
4- Présentation générale de l'automate S7-300 .....	30
4-1 Caractéristiques de l'automate S7-300 .....	31
4-2 Constitutions de l'automate S7-300 .....	31
4-2-1 Module d'alimentation .....	32
4-2-2 L'unité centrale (CPU) .....	32
4-2-3 Module de coupleur .....	33
4-2-4 Module d'entrées /sortie (TOR) .....	33
4-2-5 Module de fonction (FM) .....	34
4-2-6 Module de communication (CP) .....	34
4-2-7 Châssis d'extension .....	34
4-2-8 Console de programmation .....	34
5- Logiciel de programmation .....	34
5-1 Création du projet.....	34
5-2 Configurations et paramètres du matériel .....	36
5-3 Création du programme utilisateur .....	38
5-4 Structures de programme d'utilisateur.....	38
6- Exemple de notre programme .....	40
7-Simulation avec le logiciel S7-PLCSIM .....	42
8- Conclusion .....	44

### ❖ CHAPITRE IV : Supervision de la station de transport du sucre

1-Introduction .....	45
2- Place de la supervision .....	45
3-Constitution d'un système de supervision .....	45
3-1 Module de visualisation .....	45
3-2 Module d'archivage .....	45
3-3 Module de traitement .....	46
3-4 Module de communication .....	46
4- Logiciel de supervision WinCC.....	46
4-L'application disponible sous WinCC .....	47
4-1-1 Graphic designer.....	47
4-1-2 Tag Logging.....	47

4-1-3 Alarm Logging.....	47
4-1-4 Global script runtime .....	47
4-1-5 Report designer .....	47
4-1-6 User administrator .....	47
5- Application développée sous WinCC .....	48
6- Conclusion .....	52
<b>❖ CONCLUSION GENERALE.....</b>	<b>53</b>
<b>❖ ANNEXES</b>	
<b>❖ BIBLIOGRAPHIE</b>	

## **1 .Introduction**

Le transporteur du sucre joue un rôle prépondérant dans la gamme de fabrication de sirop simple, de telle sorte que toute la chaîne de production repose sur le bon fonctionnement de celui-ci. En réalité, le rôle principal de l'installation est le stockage du sucre, seulement ce stockage passe par plusieurs étapes. C'est-à-dire que lorsque le sucre est stocké : il est tamisé, débarrassé de ses débris, et prêt à être dissolu pour la préparation du sirop.

Dans ce chapitre, nous décrirons les éléments du transporteur ainsi que leurs fonctionnements.

## **2. Présentation de la station**

La station est placée au coin d'un local, qui occupe une surface de 200 m<sup>2</sup> environ, dont la station utilise un espace de 10 m<sup>2</sup>. Le reste de la surface est réservé pour le stockage du sucre dans des sacs de 1000 kg. Ces sacs sont posés sur des palettes, pour être déchargés, avec un Clark dans la trémie de la station.

La station « transporteur du sucre » (Figure I-1) est composée de :

- deux trémies (1) et (2) : une pour vider dedans le contenu des sacs du sucre et l'autre pour le stockage de sucre fini;
- deux convoyeurs à vis (3) et (4) : pour assurer le transport du sucre vers les autres éléments de stockage;
- silos de stockage du sucre (5);
- des moteurs (6) pour l'entraînement des convoyeurs à vis et les vis sans fin, de la trémie et du silo.

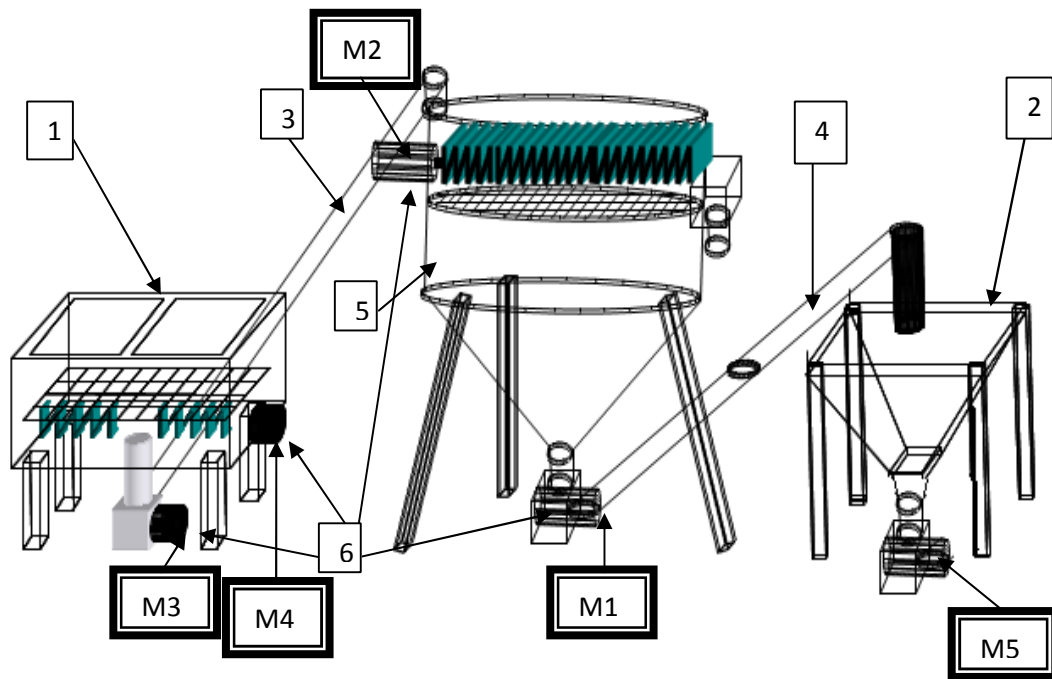


Figure I.1 : schéma de la station de transport du sucre.

### 3. Description et représentation des éléments de la station

#### 3.1. Trémie 1

##### 3.1.1. Présentation

C'est un réservoir sous forme rectangulaire (Figure I.2) : longueur 250cm, largeur 120cm et hauteur 130cm. Elle occupe un volume de 4000 litres. Elle est fabriquée en acier inoxydable. Elle possède trois ouvertures, deux en haut (1) pour le remplissage et une autre ouverture en bas(2) pour l'évacuation du sucre à l'aide d'un convoyeur à vis(3). Elle se pose sur 04 pieds(4) de 1m de longueur. La base de la trémie est légèrement concave.

##### 3.1.2. Fonctionnement

Le remplissage de la trémie s'effectue avec un Clark, en vidant des sacs de sucre de 1000 kg chacun, à travers deux portes de 1 m<sup>2</sup> chacune (1), qui s'ouvrent manuellement. Le sucre passe tout d'abord par un tamis (5). Ensuite, par deux vis sans fin (6) et (8), qui tournent dans deux sens opposés, et qui sont entraînées par un moteur (7). Le rôle des vis est d'écraser les blocs du sucre et le regrouper au canal d'évacuation (2). Celui-ci est équipé d'un moteur (9) qui entraîne la vis de convoyeur (3), pour transporter le sucre de la trémie 1 au silo.

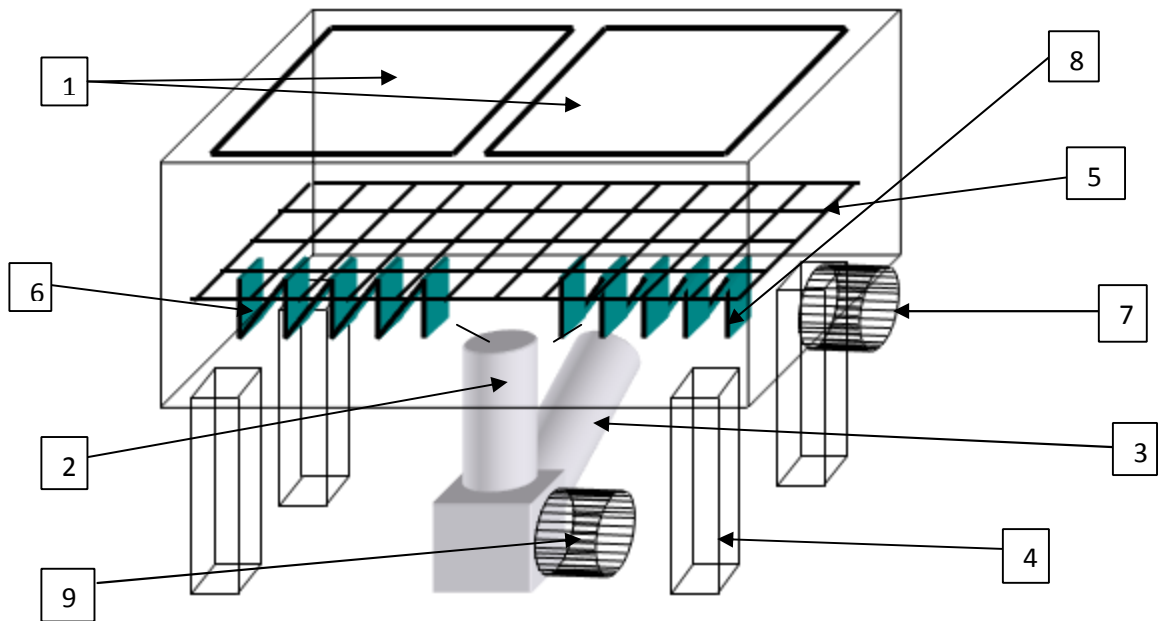


Figure I.2 : représentation schématique de la trémie 1.

### 3.2. Convoyeurs à vis sans fin

#### 3.2.1. Présentation

Le convoyeur a pour fonction d'assurer le transport du sucre, de la trémie 1 vers le silo de stockage, et du silo vers la trémie 2. Il est constitué d'une vis sans fin, qui est composée des spires (1) en acier spécial : son rôle est de transporter le sucre. La vis est installée dans un canal (2), de diamètre 25cm (de longueur 5m pour le premier convoyeur et de 2m pour le deuxième) et fabriquée en acier inoxydable (Figure I.3). Cette vis est entraînée par une motorisation (3).

La taille du convoyeur est choisie d'une façon à fonctionner à des vitesses, relativement faibles, afin d'augmenter sa durée de vie et d'atteindre un faible niveau sonore.

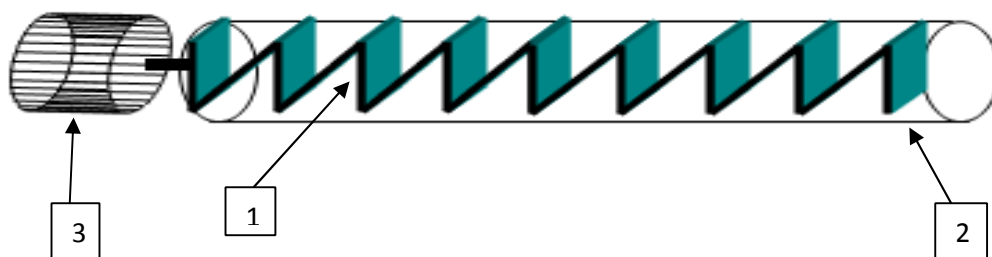


Figure I-3 : schéma de la vis sans fin.

### **3.2.2. Fonctionnement**

Le transport du sucre se fait à l'aide des spires de la vis sans fin, qui se situent à l'intérieur du canal du convoyeur. La vis sans fin est entraînée par un moteur asynchrone qui le fait tourner avec le sucre. Ce dernier, qui sort du canal de la trémie 1, vient avec les spires du convoyeur et se déplace ainsi le long du canal pour arriver à sa fin, c'est à dire au silo et de même manière pour le deuxième convoyeur.

### **3.3. Silo**

#### **3.3.1. Présentation**

C'est un réservoir de stockage de sucre. Il est sous forme d'entonnoir (Figure I.4). Il est composé de deux ouvertures : une pour la réception du sucre (1), qui est cheminé à l'aide de convoyeur à vis, et l'autre pour l'évacuation (2), vers le convoyeur à vis, qui est actionné par un moteur asynchrone (3).

Il est fabriqué avec un acier inoxydable et posé sur trois pieds, qui sont positionnés de façon symétrique. Il a une capacité de 18000 litres et formé de deux parties : une conique(5) (hauteur 150cm, diamètre de la base 350cm et le diamètre de canal 25cm), et l'autre cylindrique(4) (hauteur 155cm et diamètre 350cm). Il contient aussi une vis sans fin(6) qui est actionnée par un moteur(7) pour l'évacuation(8) des débris de sucre à l'extérieur vers des sacs, qui sont jetés par la suite.

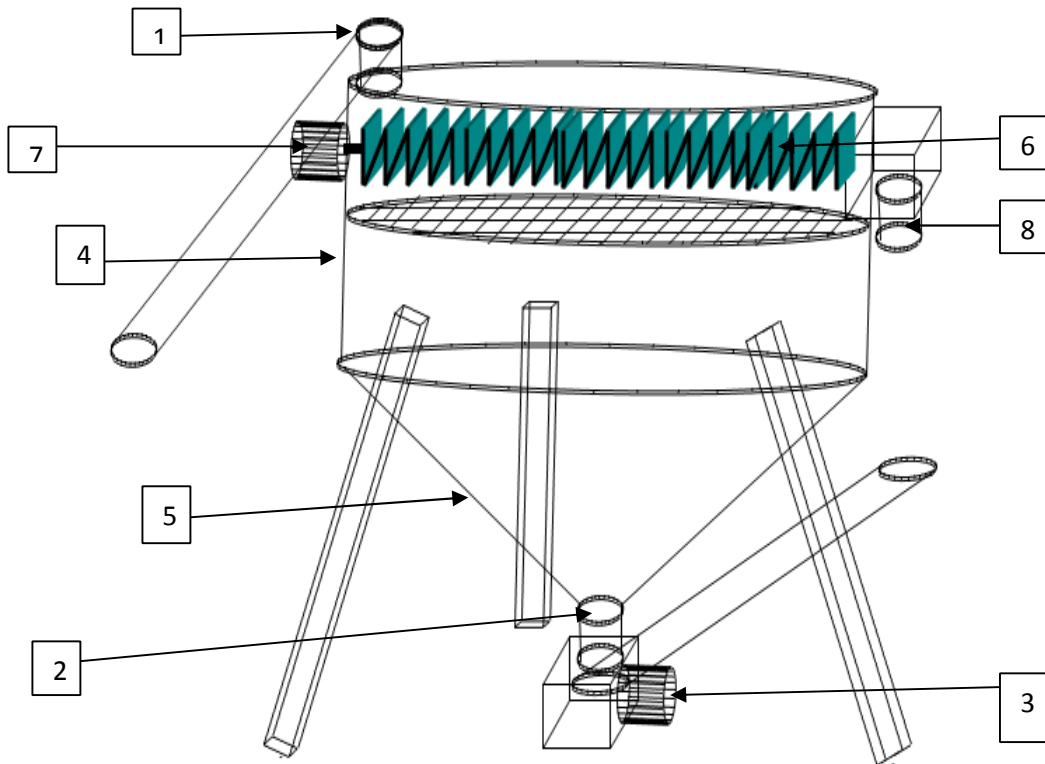


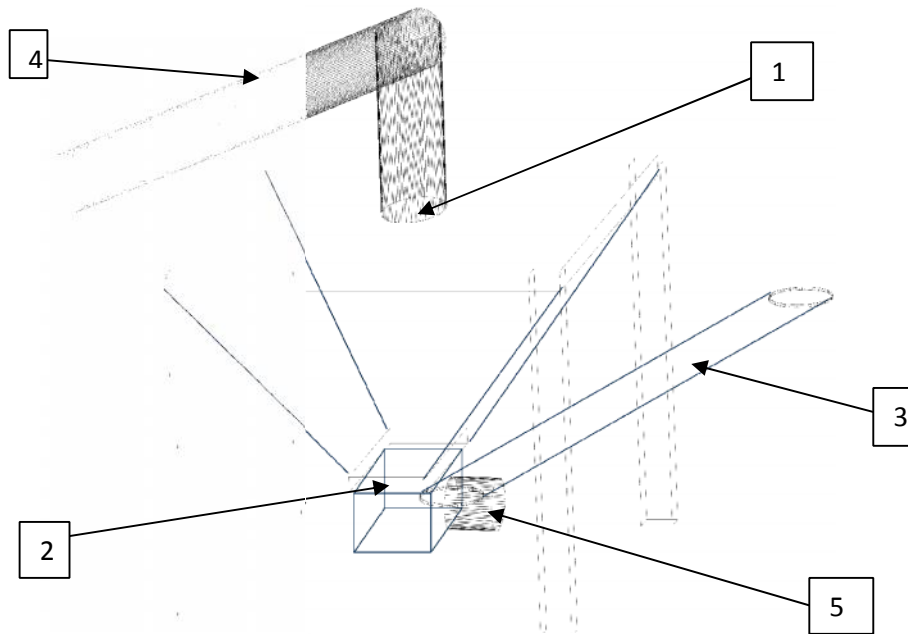
Figure I.4 : représentation schématique du silo.

### 3.4. Trémie 2

#### 3.4.1. Présentation

C'est un réservoir sous forme d'entonnoir (Figure I.5). Elle est fabriquée en acier inoxydable. Elle est placée contre le mur et posée sur quatre pieds. Elle est entre le silo de stockage du sucre et le silo désolateur, qui est installé dans le local voisin.

Elle est constituée de deux canaux : un pour la réception du sucre (1) qui est transporté à l'aide du convoyeur (4) (qui relie le silo et la trémie 2). L'autre canal (2) pour l'évacuation, qui s'effectue à l'aide du convoyeur (3) : sont rôle et de transporter le sucre a une autre station, pour la préparation du sirop simple.



**Figure I.5 :** représentation schématique de la trémie 2.

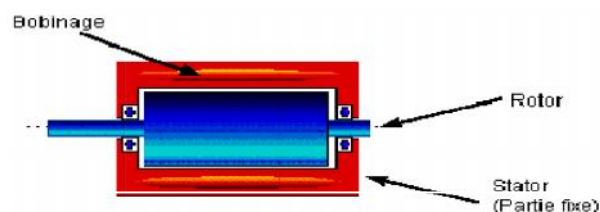
### 3.4.2. Fonctionnement

En réalité le rôle de cette trémie est le stockage du sucre. Le transit du sucre, vers le silo de l'autre local ne se fait que si le moteur (5) est lancé automatiquement, à partir de la station du local voisin, lors de la préparation du sirop simple.

### 3.5. Moteurs asynchrones

La machine asynchrone est une machine à courant alternatif sans connexion entre le stator et le rotor. Le terme asynchrone provient du fait que la vitesse de ces machines n'est pas forcément proportionnelle à la fréquence des courants qui les traversent.

Le moteur asynchrone triphasé est largement utilisé dans l'industrie (Figure I.6), sa simplicité de construction en fait un matériel très fiable et qui demande peu d'entretien. Il est constitué d'une partie fixe : le stator qui comporte le bobinage, et d'une partie rotative: le rotor qui est bobiné en cage d'écureuil



**Figure I.6 :** représentation schématique du moteur asynchrone.

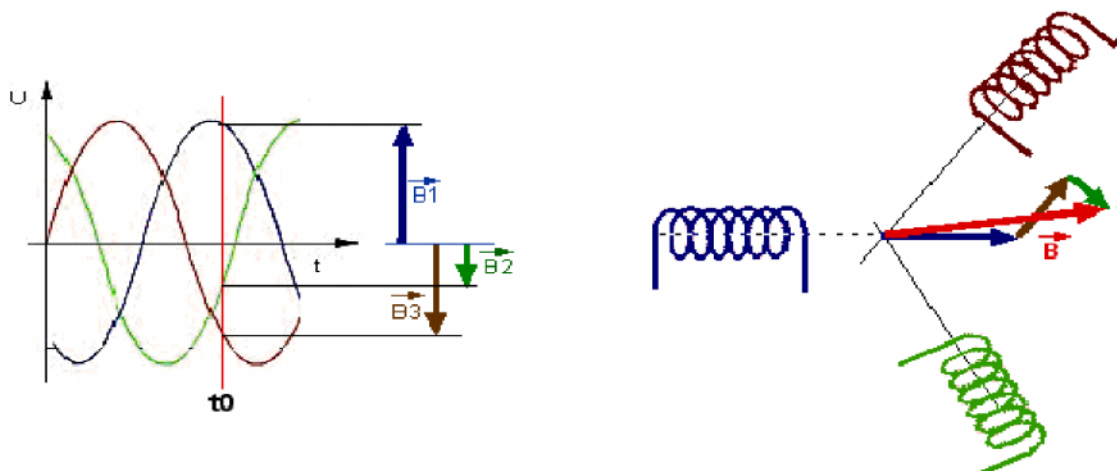
Le principe des moteurs à courant alternatifs réside dans l'utilisation d'un champ magnétique tournant, produit par des tensions alternatives.

La circulation d'un courant dans une bobine crée un champ magnétique  $B$  (Figure 1.7). Ce champ est dans l'axe de la bobine, sa direction et son intensité sont fonction du courant  $I$ .



**Figure I.7 :** représentation du champ et courant dans une bobine.

Dans le cas du moteur triphasé, les trois bobines sont disposées dans le stator à  $120^\circ$  les unes des autres, trois champs magnétiques sont ainsi créés (Figure I.8). Le champ magnétique résultant tourne à la même fréquence que le courant.



**Figure 1.8 :** le champ magnétique dans le cas triphasé

Notre station de transport du sucre est constituée de cinq moteurs triphasés asynchrones, dont la plaque signalétique et la suivante :

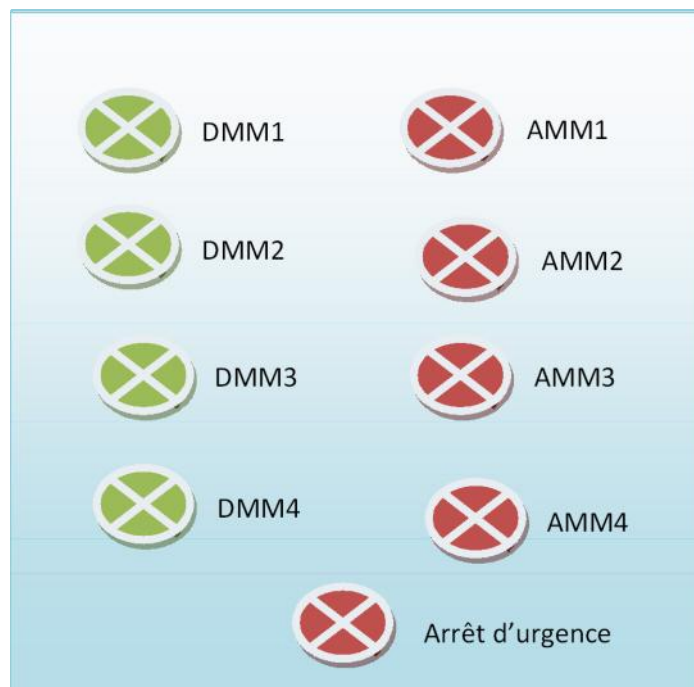
- Puissance : 4KW / 5.5 HP.
- Vitesse : 1430 tr/mn.
- Tension d'alimentation : 380V – 420V.
- Courant : 8.26 A.
- IP (indice de protection) : 5.5.
- Cos : 0.83.
- CI (classe d'enroulement) : F.

Ces cinq moteurs sont :

- **Moteur M1** : son rôle est de tourner la vis sans fin du convoyeur (4) (Figure I.1), pour transporter le sucre du silo à la trémie 2.
- **Moteur M2** : son rôle est de tourner la vis sans fin (6) (Figure I.4), pour éjecter les débris du sucre à l'extérieur et évacuer le sucre vers l'intérieur.
- **Moteur M3** : son rôle est de tourner la vis sans fin du convoyeur (1) (Figure I.4), pour transporter le sucre de la trémie 1 au silo.
- **Moteur M4** : son rôle est de tourner les deux vis sans fin qui sont dans la trémie 1 pour l'écrasement des blocs du sucre et remplissage du canal d'évacuation.
- **Moteur M5** : lié à la station voisine, qui sert à la préparation du sirop simple et qui est automatisé.

### 3.5 Une table de commande

Elle est installée à côté de la trémie 2, et comporte les boutons poussoirs de marche/arrêt des moteurs et d'arrêt d'urgence (Figure I-9).



Bouton démarrage



Bouton d'arrêts

**Figure I.9** : table de commande

**Remarque :** DMM<sub>i</sub>, avec  $i=1, \dots, 4$ , correspond au Démarrage Moteur M<sub>i</sub> (DMM<sub>i</sub>).

AMM<sub>i</sub>, avec  $i=1, \dots, 4$ , correspond au Arrêt Moteur M<sub>i</sub> (AMM<sub>i</sub>).

#### 4. Fonctionnement général de la station

Le fonctionnement général de la station de transporteur du sucre se fait selon les étapes successives suivantes :

- Le remplissage de la trémie 1 à l'aide d'un Clark, contrôlé par un opérateur.
- Le démarrage du M2, qui se fait par un bouton poussoir enclenché par l'opérateur, pour éjecter le reste des blocs du sucre, qui ne sont pas tamisés dans le silo, à l'extérieur.
- Après environ 15s, l'opérateur enclenche le moteur M3, pour faire transporter le sucre de la trémie 1 au silo.
- Après environ 15s, l'opérateur met en marche le moteur M4. Ce retard est pour donner un peu de temps au convoyeur 1, afin d'évacuer le reste du sucre.
- L'arrêt des moteurs, M2, M3 et M4, se fait dès que le silo est rempli. Cet arrêt est déclenché par l'opérateur par chaque bouton poussoir approprié.

**Remarque :** sur la station trois opérateurs interviennent pour assurer le cycle du fonctionnement.

#### 5. Cahier des charge

Lors du cycle de transport du sucre, nous avons constaté, après étude de cette station, des problèmes qui influencent, négativement, sur le fonctionnement de la station. Ces problèmes sont dus au cycle de fonction de l'installation, qui est pratiquement manuel. C'est-à-dire dès le remplissage de la trémie 1, qui se fait manuellement, jusqu'à la fin (préparation du sirop simple), l'opérateur intervient d'une manière continue et sa présence est indispensable. Ce qui engendre, bien sûr, des pertes du temps lors du fonctionnement de cette station.

Le groupe Fruital Coca-Cola nous a confié la tâche consistante à automatiser cette installation en réduisant le plus possible l'intervention de l'opérateur.

## **6. Conclusion**

L'étude menée, dans ce chapitre, nous permet de conclure que l'automatisation de cette station (transporteur du sucre) est nécessaire pour accélérer la production et surmonter les différents problèmes rencontrés quotidiennement.

Ainsi, nous serons dans l'obligation d'apporter des modifications à la station pour la rendre complètement automatique.

## **1. Introduction**

L'automatisation d'un procédé (c'est-à-dire une machine, un ensemble de machines ou plus généralement un équipement industriel) consiste à en assurer la conduite par un dispositif technologique. Donc, une machine ou un système est dit automatisé lorsque le processus, qui permet de passer d'une situation initiale à une situation finale, se fait sans intervention humaine. Dans notre cas d'étude, c'est l'automatisation de la station de transport du sucre, qui est une machine purement manuelle. Afin d'atteindre cet objectif, des modifications (améliorations) du système s'avèrent nécessaires.

Cependant, l'automatisation d'un procédé industriel nécessite le passage par l'étape de modélisation. Ainsi, dans notre cas nous avons choisi l'utilisation de l'outil GRAFCET, qui est un outil simple à concevoir, à comprendre et à modifier pour d'autres fins.

## **2. Définition**

Le GRAFCET (Graphe Fonctionnel de Commande Etape / Transition) est un outil graphique de définition pour l'automatisme séquentiel, en tout ou rien. Mais il est également utilisé dans beaucoup de cas combinatoires, dans le cas où il y a une séquence à respecter mais où l'état des capteurs suffirait pour résoudre le problème en combinatoire. Il utilise une représentation graphique.

C'est un langage clair, strict mais sans ambiguïté, permettant des représentations fonctionnelles, opérationnelles et technologiques de la plupart des automatismes industriels. La grande force du GRAFCET est sa fiabilité de passer du modèle conceptuel à une implantation technologique dans un automate programmable.

## **3. Structure d'un système automatisé de production**

Un système de production est dit automatisé lorsqu'il peut gérer, de manière autonome, un cycle de travail préétabli, composé des séquences ou étapes.

Les systèmes automatisés, utilisés dans le secteur industriel, possèdent une structure de base identique (Figure II.1). Ils sont constitués de trois parties reliées entre elles : partie opérative, partie commande et partie dialogue.

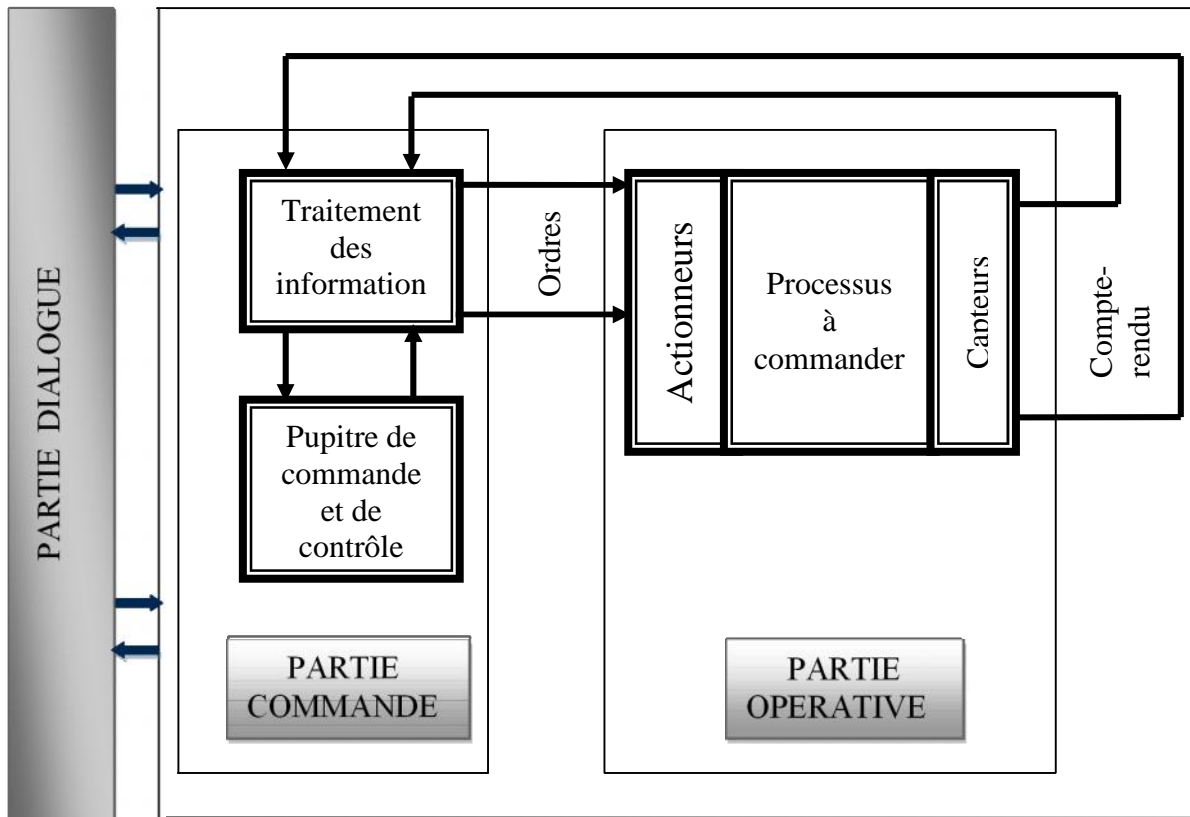


Figure II.1 : structure d'un système automatisé de production

### 3.1. La partie opérative

Elle est constituée de processus à commander, des actionneurs qui agissent sur ce processus et des capteurs permettant de mesurer son état.

- **Les actionneurs** : Se sont les éléments chargés de convertir l'énergie, afin de s'adapter au besoin de la partie opérative.
- **Les prés-actionneurs** : Se sont les composants de gestion de l'énergie d'alimentation des actionneurs. Ils reçoivent l'information de commande de la partie commande, puis délivrent en sortie la puissance nécessaire pour l'actionneur.
- **Les capteurs** : Ils fournissent à la partie commande des comptes rendus sur l'état du système. Ils convertissent les informations physiques de la partie opérative en grandeurs électriques, exploitables par la partie commande.

### 3.2. La partie commande

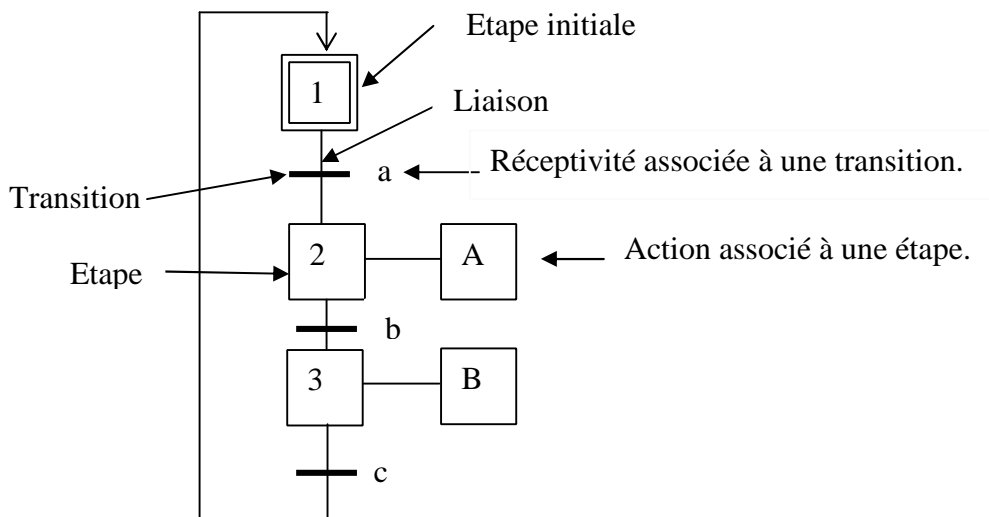
Constituée de l'automatisme qui élabore les ordres destinés au processus et aux sorties externes (visualisation), à partir des comptes rendus de la partie opérative, des entrées externes (consignes) et de l'état du système.

### 3.3 Partie dialogue

Elle comporte le pupitre de dialogue homme/machine, équipé des organes de commande permettant la marche et l'arrêt de l'installation, la commande manuelle des actionneurs, le démarrage du cycle, l'arrêt d'urgence, etc.

## 4. Les éléments de base du GRAFCET

Un GRAFCET est composé d'éléments qui forment sa structure graphique (Figure II.2) : les étapes, les transitions et les liaisons.



**Figure II-2** : les éléments d'un GRAFCET.

### 4.1. Etape

Une étape correspond à une phase durant laquelle on effectue une action, pendant une certaine durée. L'action doit être stable, c'est à dire que l'on fait la même chose pendant toute la durée de l'étape.

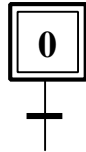
On représente chaque étape par un carré. On numérote chaque étape par un entier positif. Chaque étape porte son propre numéro et lorsqu'on la désigne, on précède ce numéro de la lettre X.

$X_i=1$  lorsque l'étape  $i$  est activée (elle sera précisée par un point).

$X_i=0$  lorsque l'étape  $i$  est désactivé.

#### 4.1.1. Etape initiale

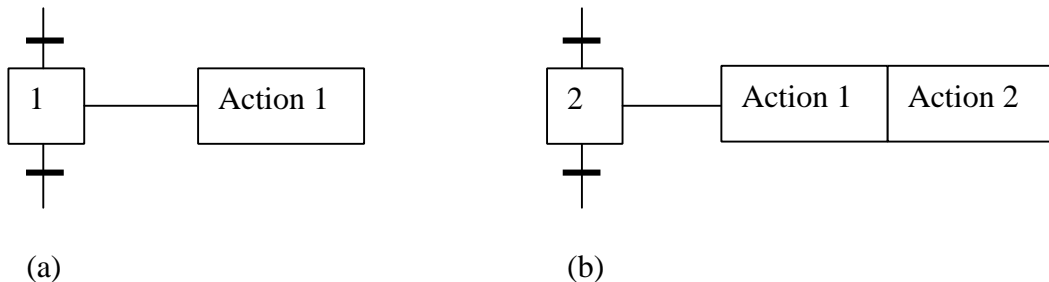
Elle représente l'étape qui est activé au début du fonctionnement. Elle se différencie en doublant les cotées du carré (Figure II.3).



**Figure II.3:** représentation graphique d'une étape initiale

#### 4.1.2. Les actions associées à une étape

A chaque étape sont associées une ou plusieurs actions (Figure II.4). Une action, qui est représentée dans un rectangle, peut être associée à plusieurs étapes.

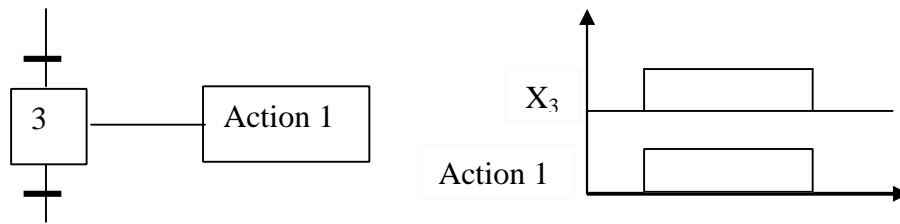


**Figure II.4 :** représentation graphique des actions associées à une étape :  
(a):étape simple, (b) : étape à plusieurs actions.

On peut définir plusieurs types d'actions :

##### a. Action continue

L'action est exécutée dès que l'étape associée est active, sans autre condition particulière (Figure II.5).



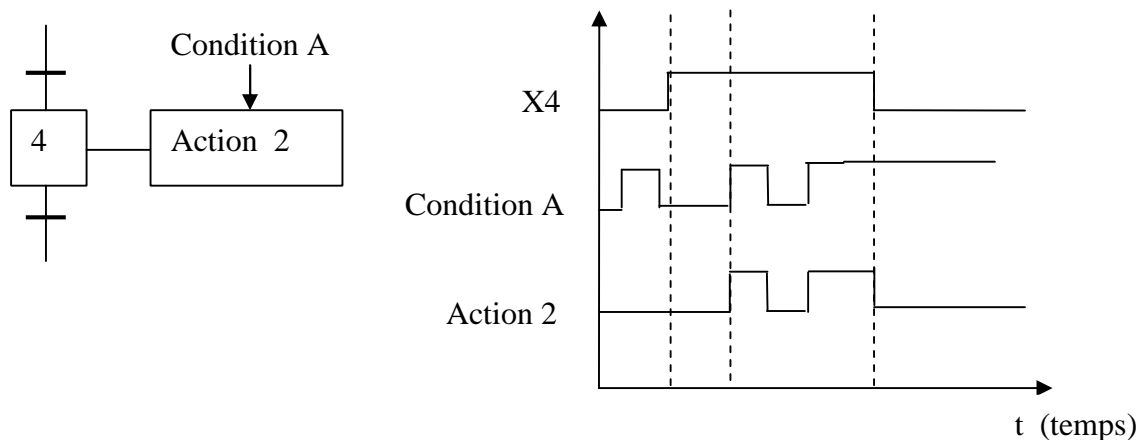
**Figure II.5 :** représentation graphique d'une action continue et son chronogramme.

L'action (1) dure tant que l'étape (3), à laquelle elle est associée, reste activée.

### b. Action conditionnelle

Une action normale est exécutée si l'étape à laquelle elle est associée est active. L'action conditionnelle sera réalisée si, en plus, la condition logique, qui est précisée en haut du cadre de l'action, est vraie. La condition est une équation booléenne au même titre qu'une réceptivité.

L'action démarre lorsque l'étape est active ET la condition est réalisée. Elle s'arrête si l'étape est désactivée OU si la condition n'est pas vérifiée (Figure II.6).



**Figure II.6:** action conditionnelle est son chronogramme.

### c. Action retardée ou limitée

Les actions retardées ou limitées sont des cas particuliers d'actions conditionnelles, où le temps intervient comme condition logique. La variable de temporisation est l'activité  $X$  de l'étape en cours.

L'action débute 3 secondes après l'activation de l'étape 2 (Figure II. 7).

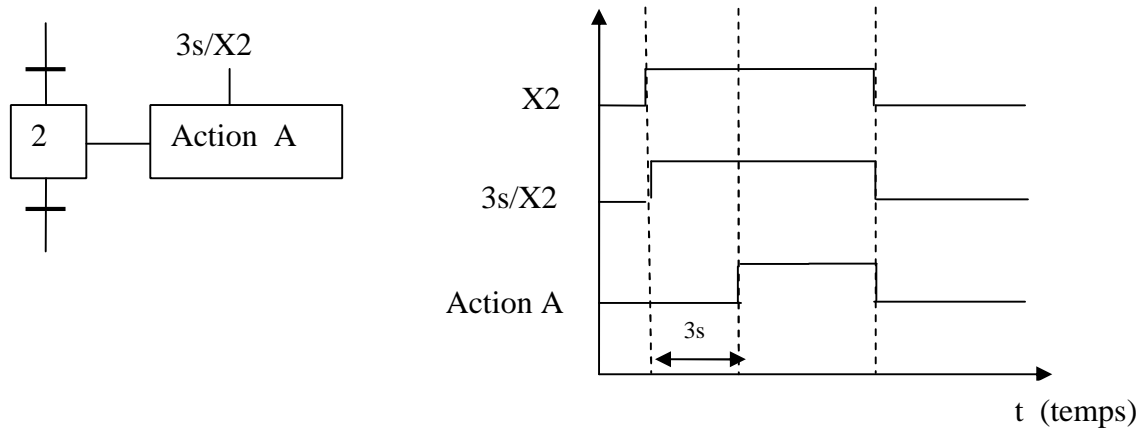


Figure II.7 : action retardée et son chronogramme

d. Action maintenue

Une action maintenue c'est une action qui est associée à plusieurs étapes (Figure II.9).

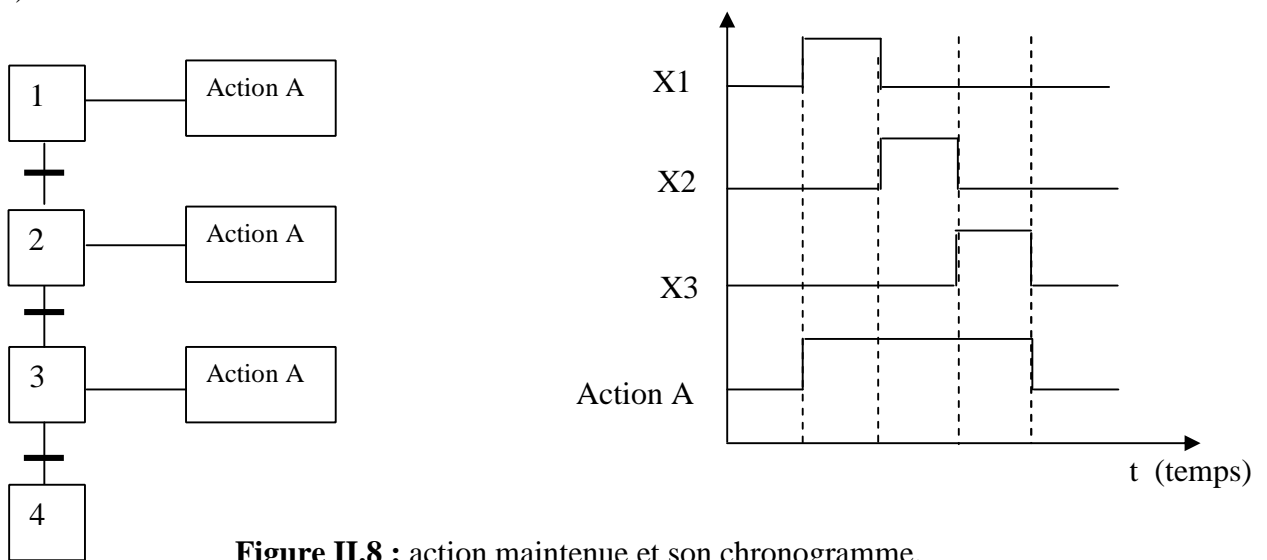


Figure II.8 : action maintenue et son chronogramme.

4.2. Transition

Une transition est une condition de passage d'une étape à une autre. Elle n'est que logique (dans son sens Vrai ou Faux), sans notion de durée. La condition est définie par une réceptivité qui est, généralement, une expression booléenne (c'est-à-dire avec des ET et des OU) de l'état des capteurs.

Une réceptivité est associée à chaque transition. C'est une condition qui détermine la possibilité ou non d'évolution du système par cette transition (Figure II.9).

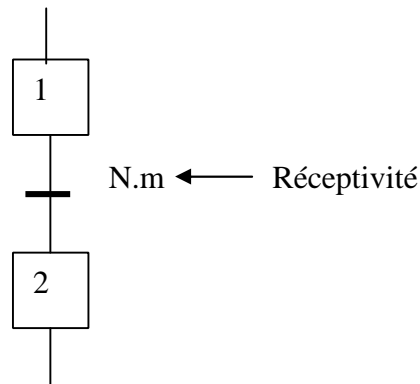


Figure II.9 : réceptivité associée à l'étape 1.

## 5. Règles d'évolution

La modification de l'état de l'automatisme est appelée évolution, et est régie par 5 règles :

**Règle 1 :** Les étapes initiales sont celles qui sont actives au début du fonctionnement.

**Règle 2 :** Une transition est validée lorsque toutes les étapes précédents cette transition sont activées. Une transition validée est franchie si la réceptivité associée, à cette transition, est vraie.

**Règle 3 :** Le franchissement d'une transition entraîne l'activation de toutes les étapes, immédiatement, suivantes et la désactivation de toutes les étapes, immédiatement, précédentes.

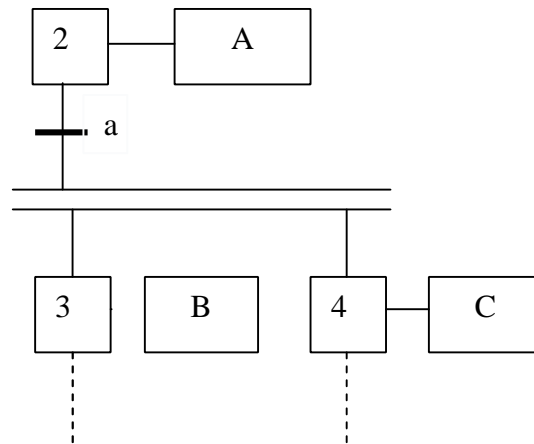
**Règle 4 :** Plusieurs transitions, simultanément franchissables, sont simultanément franchies.

**Règle 5 :** Si une étape doit être à la fois activée et désactivée, elle reste active.

## 6. Structures de base

### 6.1. Divergence en ET

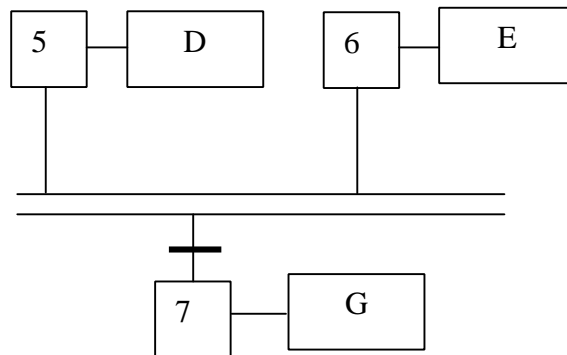
Lorsque l'action A est activée, la transition 2-(3,4) est validée. Lors du franchissement de la transition, il y a désactivation de l'étape 2 et activation simultanée des étapes 3 et 4 (Figure II.10).



**Figure II.10 :** divergence en ET.

### 6.2. Convergence en ET

Lorsque les étapes 5 et 6 sont activées, la transition (5,6)-7 est validée. Le franchissement de cette transition implique la désactivation simultanée des étapes 5 et 6 et l'activation de l'étape 7 (Figure II.11).

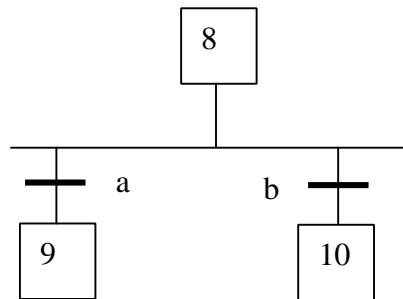


**Figure II.11 :** convergence en ET.

### 6.3. Divergence en OU

Si l'étape 8 est activée, alors désactivation de l'étape 8 et activation de l'étape 9. L'étape 10 reste inchangée.

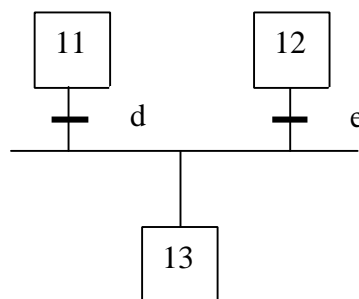
Si a et b sont vraies, si l'étape 8 est activée, alors désactivation de cette étape et activation de l'étape 2 et l'étape 3 en même temps (Figure II-12).



**Figure II.12** : divergence en OU.

#### 6.4. Convergence en OU

Si l'étape 11 est activée et la réceptivité d est vraie sans e, alors activation de l'étape 13 et désactivation de l'étape 11 et l'étape 12 en même temps (Figure II-13).



**Figure II.13** : Convergence en OU.

### 7. Niveaux du GRAFCET

Le GRAFCET est réalisé selon deux niveaux de représentation, qui sont définis comme suit :

- **Niveau 1** : Appelé aussi GRAFCET fonctionnel. Il décrit sous forme de spécifications fonctionnelles, le comportement de la partie commande vis-à-vis de la partie opérative et du monde extérieur. Les réceptivités sont décrites sous forme littérale par des mots et non pas par des abréviations. A ce niveau, on ne définit pas les actionneurs ni les capteurs mais uniquement les actions à effectuer et leur enchaînement, pour permettre de comprendre l'évolution de l'automatisme.
- **Niveau 2** : Ce GRAFCET ajoute aux exigences fonctionnelles, les précisions indispensables aux conditions de fonctionnement, grâce aux spécifications technologiques et opérationnelles, compte tenu de la technologie de la partie commande et de la partie opérative, ainsi que la prise en compte de la technologie

des actionneurs et des capteurs de l'automatisme. Il est utilisé pour la réalisation et l'exploitation des systèmes automatisés. La description des actions et des réceptivités est par abréviations.

## 8. Modifications apportées à la station de transport du sucre

Après l'étude de cette station de transport du sucre, nous avons constaté qu'aucune nouvelle technologie n'est introduite. Ainsi donc, pour son automatisation nous sommes contraints de rajouter des éléments indispensables à cette automatisation. Pour cela nous allons :

- ✓ Equiper le silo de stockage du sucre de deux capteurs (capteurs de niveaux), pour nous indiquer les niveaux bas et haut, afin de contrôler les moteurs.
- ✓ Pour la trémie 1, ajouter un capteur (capteur de niveaux) qui indique le niveau bas, afin d'arrêter le cycle ou de remplir la trémie.
- ✓ Deux capteurs (capteur de niveaux) pour la trémie 2 pour le contrôle de démarrage et l'arrêt des moteurs M1 et M5.
- ✓ Des variateurs de vitesses pour les moteurs. Qui assurent le transport du sucre vers le silo, afin d'assurer la présence du sucre tous le temps dans le silo et le contrôle des moteurs.
- ✓ Des détecteurs de rotation pour chaque convoyeur, pour la sécurité des moteurs et des convoyeurs.
- ✓ Une vanne manuelle pour le canal d'évacuation du silo, pour faciliter la tâche de réparation des éléments en amont.

### 8.1. Capteurs de niveau

Un capteur est un dispositif qui transforme l'état d'une grandeur physique, observée, en une grandeur utilisable, exemple : une tension électrique, une hauteur de mercure, un niveau, la déviation d'une aiguille, etc.



**Figure II.14 :** capteur de niveau.

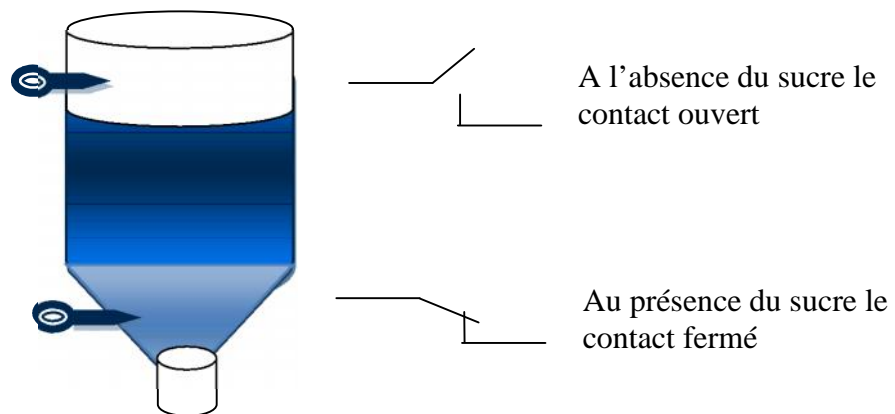
### 8.1.1. Caractéristiques

- Nomenclature Nivo compact FTC 731.
- Température de fonctionnement dans le silo  $-20\text{ C}^{\circ}$  à  $100\text{ C}^{\circ}$ .
- Pression de fonctionnement max 10 bar.
- Charge maximale admissible sur la sonde 4000n.

### 8.1.2. Fonctionnement

Quand le bout de la sonde est dans l'air, la constante diélectrique  $\epsilon_r=1$ , en présence d'un produit (sucre), la capacité du condensateur augmente sous l'effet de produits qui possèdent une constante diélectrique supérieur à 1. Cette variation de capacité est traitée pour actionner la fermeture et l'ouverture du contact du capteur.

Dans la figure II.15 nous présenterons le positionnement du capteur.



**Figure II.15** : schéma synoptique du positionnement de capteur.

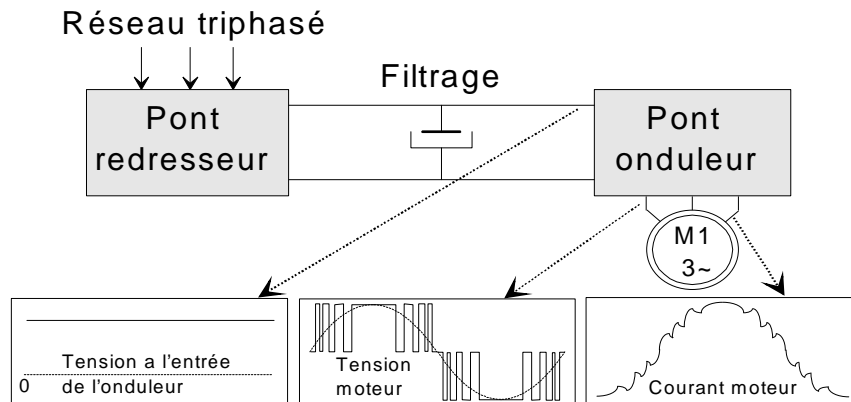
### 8.2. Variateur de vitesses

Un variateur de vitesses est un équipement électrotechnique alimentant un moteur électronique, de façon à pouvoir faire varier sa vitesse de manière continue, de l'arrêt jusqu'à sa vitesse nominale (Figure II.16).



**Figure II.16** : variateur de vitesses.

Un variateur de vitesses est constitué d'un redresseur combiné à un onduleur. Le redresseur va permettre d'obtenir un courant quasi continu. A partir de ce courant continu, l'onduleur (bien souvent à modulation à largeur d'impulsion ou MLI) va permettre de créer un système triphasé de tension alternative, dont on pourra faire varier la valeur efficace et la fréquence (Figure II.17).



**Figure II.17 :** constituants d'un variateur de vitesses.

Il permet :

- ✓ Une gamme de vitesses de 5% à 200% de la vitesse nominale.
- ✓ Une conservation du couple sur toute la gamme de vitesses.
- ✓ Des rampes d'accélération et de décélération.
- ✓ Deux sens de rotation.

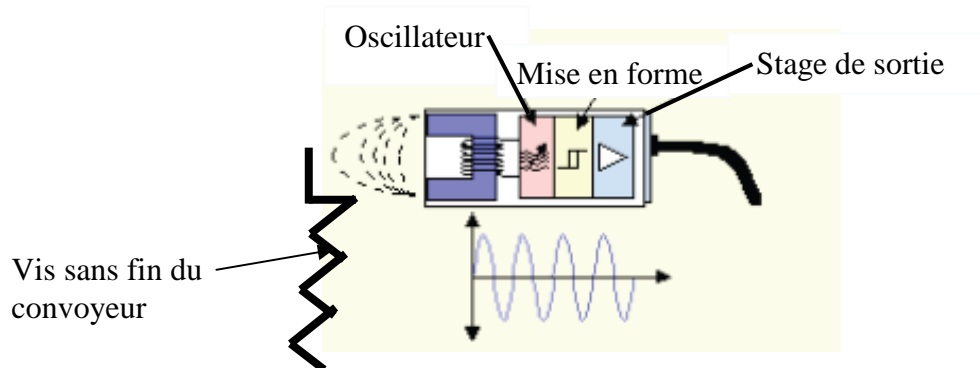
### 8.3. Détecteur de rotation

C'est un détecteur d'arrêt de mouvement indépendant. Il est composé d'un détecteur inductif et d'un contacteur d'arrêt, l'ensemble est dans un corps de 18mm de diamètre (Figure II.18). Aucun besoin de contact entre le détecteur et l'objet à contrôler (dans notre cas le convoyeur). Le détecteur de proximité capte les impulsions émises par un index fixé sur l'objet.



**Figure II.18 :** détecteur de rotation.

L'appareil est muni d'une temporisation réglable, afin de laisser le temps à l'objet à contrôler d'atteindre sa vitesse nominale. Ensuite, si aucune impulsion n'est reçue pendant ce temps, un signal d'arrêt de mouvement est déclenché. Dans la figure II.19 nous présenterons le schéma synoptique de fonctionnement



**Figure II.19** : schéma synoptique de fonctionnement de détecteur.

#### 8.4. Vanne

On utilise une vanne manuelle à la sortie du canal d'évacuation du silo, pour prévenir contre les pannes des dispositifs en amont.

### 9. Modélisation du nouveau système

#### 9.1. Grafct niveau 1

Dans la figure II.20 nous présenterons le grafct niveau 1 de notre station.

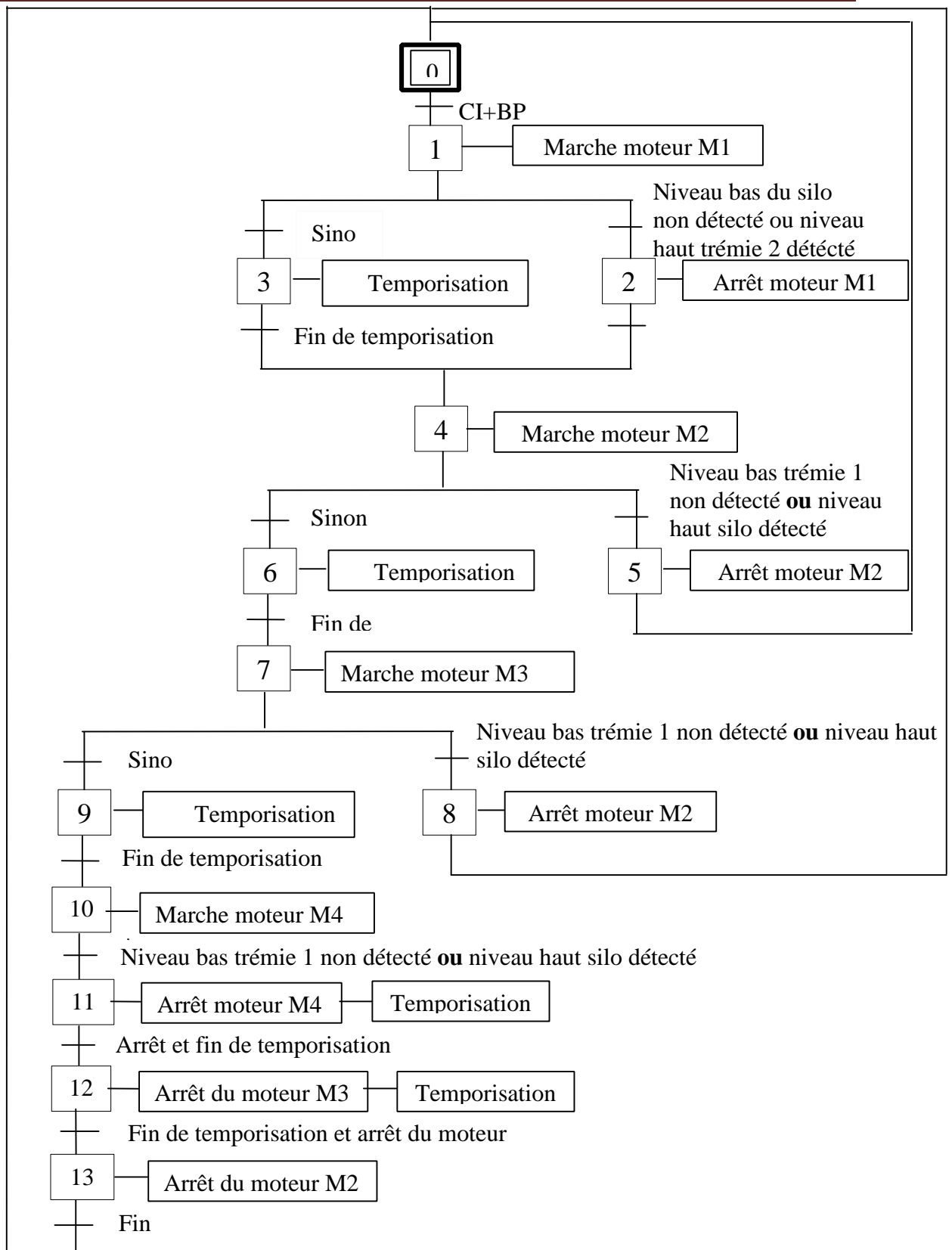


Figure II.20 : Grafcet niveau 1.

## **9.2 Listes des actionneurs et pré-actionneurs et capteurs**

Le GRAFCET niveau 2 nous impose les abréviations suivantes :

BP : Bouton poussoir de démarrage du cycle.

DM1 : moteur M1 démarre.

ARM1 : arrêt du moteur M1.

DM2 : moteur M2 démarre.

DM3 : moteur M3 démarre.

DM4 : moteur M4 démarre.

ARM4 : arrêt du moteur M4.

ARM3 : arrêt du moteur M3.

ARM2 : arrêt du moteur M2.

NBS : niveau bas du silo.

NHT2 : niveau haut de la trémie 2.

NHS : niveau haut du silo.

NBT1 : niveau bas de la trémie 1.

## **9.3 Grafcet Niveau 2**

Dans la figure II.21 nous présenterons le grafcet niveau 2 de notre station.

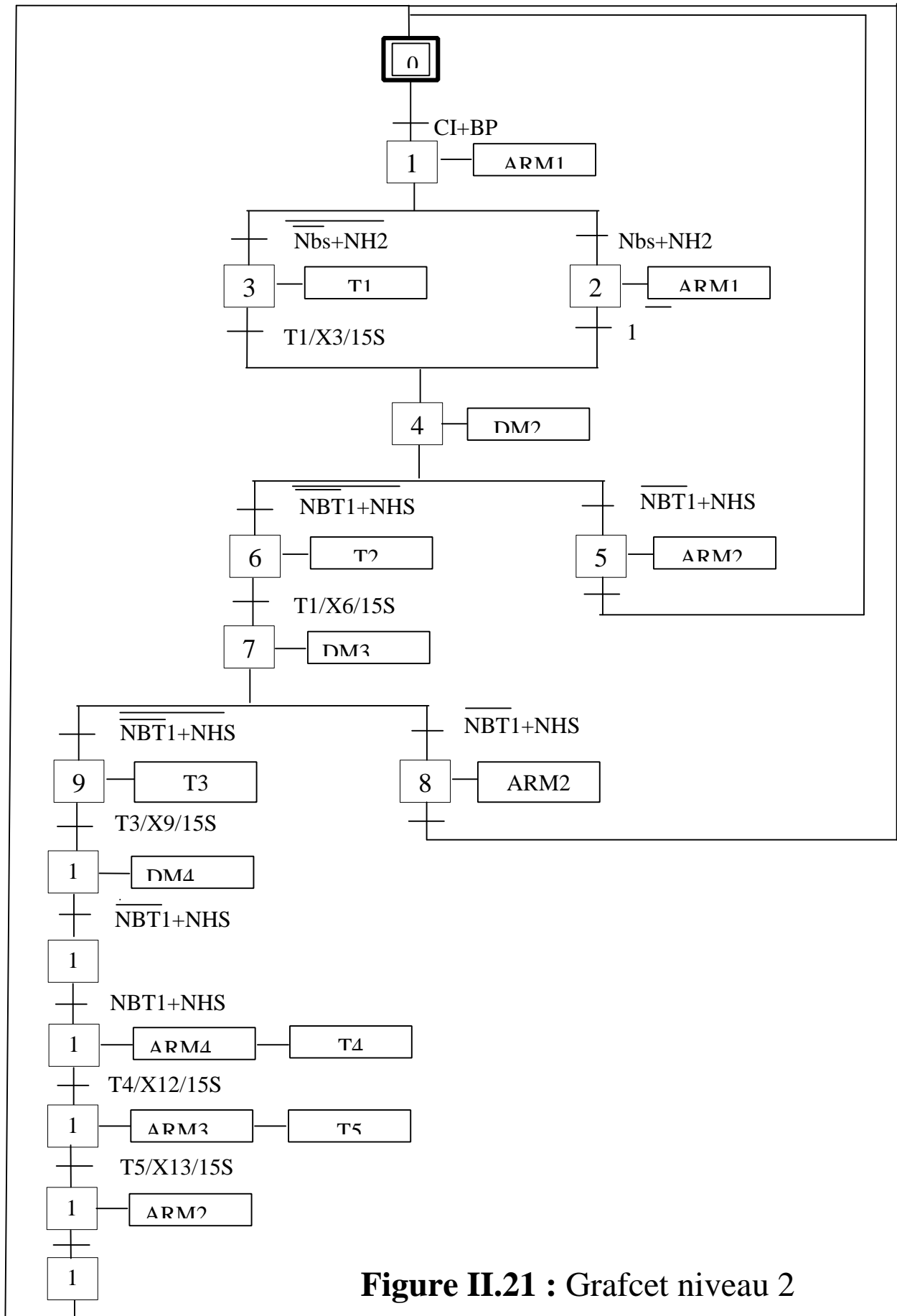


Figure II.21 : Grafcet niveau 2

## **12. Conclusion**

Au terme de ce chapitre, nous avons élaboré la nouvelle solution de commande de la station de transport du sucre, tout en introduisant les changements et les apports effectués à cette station. En tenant compte de la complexité et la difficulté rencontrée lors de l'étude du fonctionnement de cette station, nous avons élaboré le grafcet qui obéit aux caractéristiques de la station de transport du sucre.

Le grafcet nous a servi à identifier les fonctions nécessaires au bon fonctionnement de la station, et nous a facilité le passage de la description à la modélisation, qui nous permettra au chapitre suivant d'aborder la programmation de la partie opérative, qui pilotera le procédé.

## 1. Introduction

Les automates programmables industriels, sont apparus à la fin des années soixante, à la demande de l'industrie automobile américaine (GM), qui réclamait plus d'adaptabilité de leurs systèmes de commande. Les coûts de l'électronique permettant alors de remplacer, avantageusement, les technologies actuelles. Le premier automate fut créé grâce à MODICON en 1968.

Depuis le début des années 80, l'intégration des automates programmables pour le contrôle des différents processus industriels, est plus qu'indispensable. A l'origine, l'automate programmable était considéré comme une machine séquentielle, capable de remplacer des automatismes réalisés en logique traditionnelle, en apportant toutefois de profonds bouleversements dans la manière de concevoir et d'organiser le contrôle d'un processus. L'intégration de l'automate programmable renforce le degré de fiabilité de l'équipement, et offre une très grande adaptabilité face aux évolutions de l'environnement.

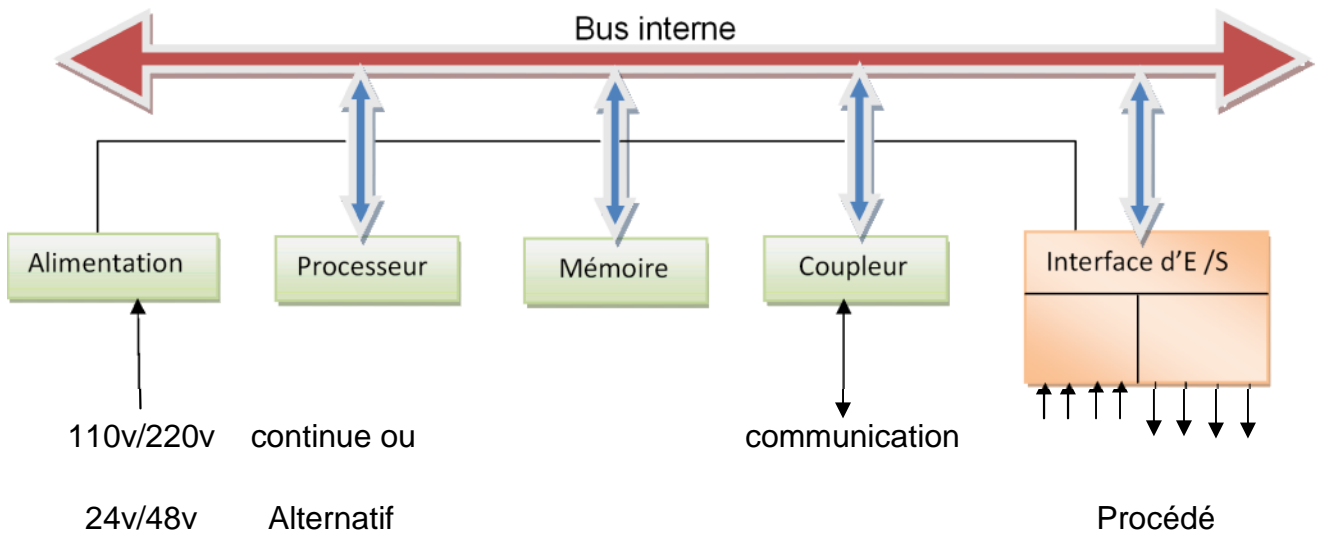
## 2. Définition d'un automate

L'automate programmable industriel(API) est un système de traitement logique d'informations, dont le programme de fonctionnement est effectué à partir d'instructions établies en fonction du processus à réaliser. Son fonctionnement est défini par un programme, donnant des ordres aux prés-actionneurs, de la partie opérative, à partir des données d'entrées (capteur, détecteur, etc.). Il rend des comptes en permanence de son état et dialogue avec l'opérateur et le processus. L'automate programmable peut traiter aussi :

- Des fonctions de calcul arithmétique, temporisation, comptage, comparaison.
- Des liaisons avec d'autres appareils (imprimantes, calculateur, etc.)

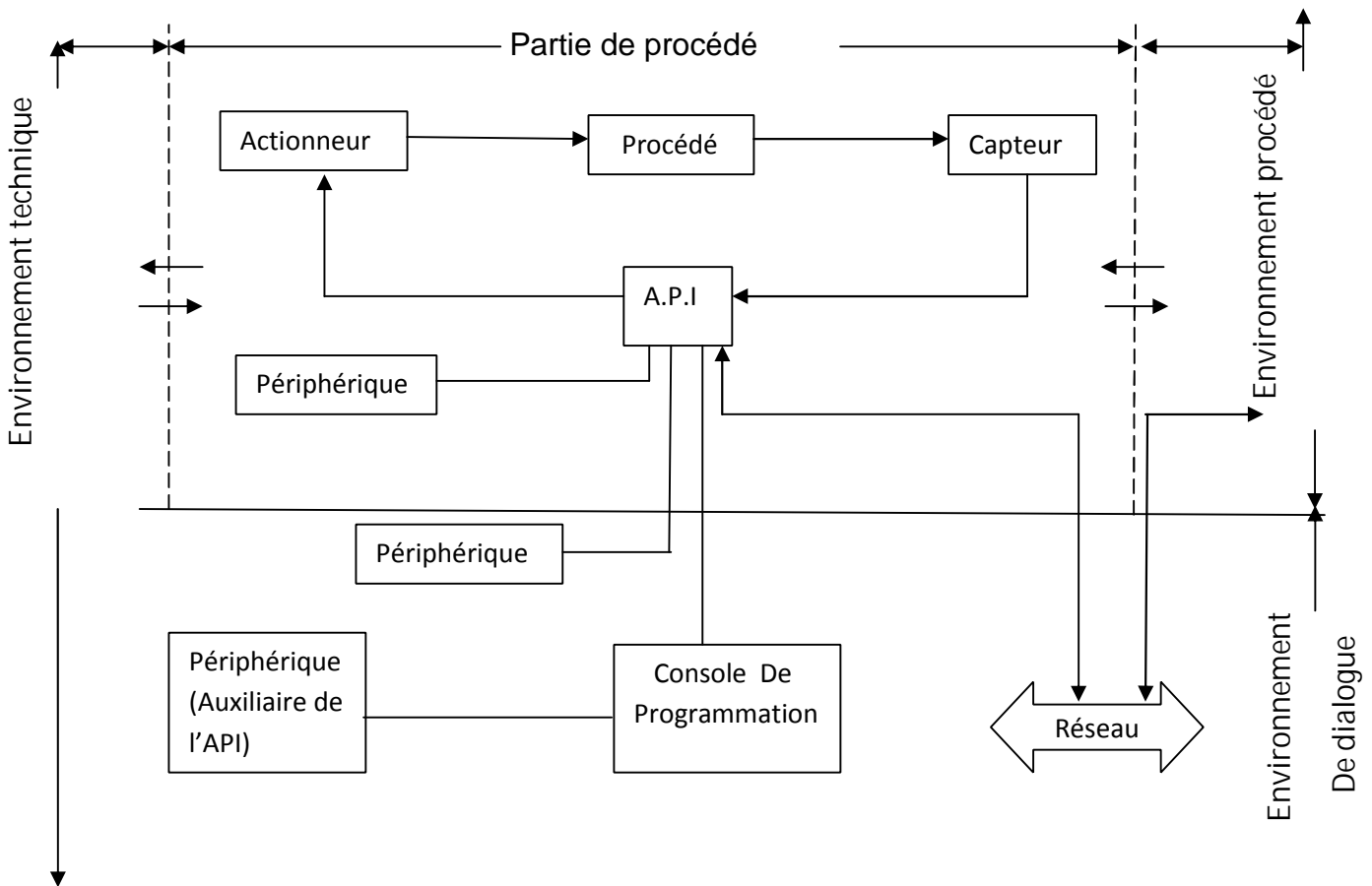
Comme il peut aussi réaliser des fonctions de régulation.

La structure de base d'un automate programmable comprend plusieurs ensembles fonctionnels (Figures III.1).



**Figure III.1 :** structure d'un automate programmable.

Pour conduire sa mission vis-à-vis d'un procédé, un API doit disposer des entrées/sorties(E/S) industrielles adaptées en nombre, en nature et en caractéristique physique (Figure III.2).



**Figure III.2 :** l'environnement de l'automate programmable industriel.

### 3. Choix d'un automate

Pour choisir un automate programmable, l'automaticien doit préciser :

- Le nombre et la nature des entrées et des sorties.
- Le type de programmation souhaitée et les besoins de traitement permettant le choix de l'unité centrale et la taille de la mémoire utilisateur.
- La nature de traitement (temporisation, couplage, etc.).
- Le dialogue (la console détermine le langage de programmation).
- La communication avec d'autres systèmes.
- La fiabilité et la robustesse.

Ainsi, le choix de S7-300 est dû à la proposition faite par les responsables de Fruital Coca-Cola, car ils veulent automatiser d'autres stations, qui constituent la chaîne de fabrication. Le S7-300 s'avère un excellent choix d'automate qui permet une grande variété d'E/S, donc une simple et facile extension.

### 4. présentation générale de l'automate S7-300

Le système d'automatisation SIMATIC S7-300 est un automate modulaire de meilleure gamme. SIMATIC S7-300 désigne un produit de la société SIEMENS, il est synonyme de la nouvelle gamme des automates programmables (Figure III.3)



**Figure III.3** : l'automate S7-300.

Les automates programmables SIEMENS sont des appareils fabriqués en série. Tous les éléments logiques, fonctions de mémoire, temporisations, compteurs, etc., nécessaires à l'automatisation, sont prévus par le fabricant et sont intégrés à l'automate. Ils se distinguent principalement, par le nombre des :

- Entrées et sorties.
- Compteur.
- Temporisation
- Mémentos.
- La vitesse de travail.

### 4.1 Caractéristiques de l'automate S7-300

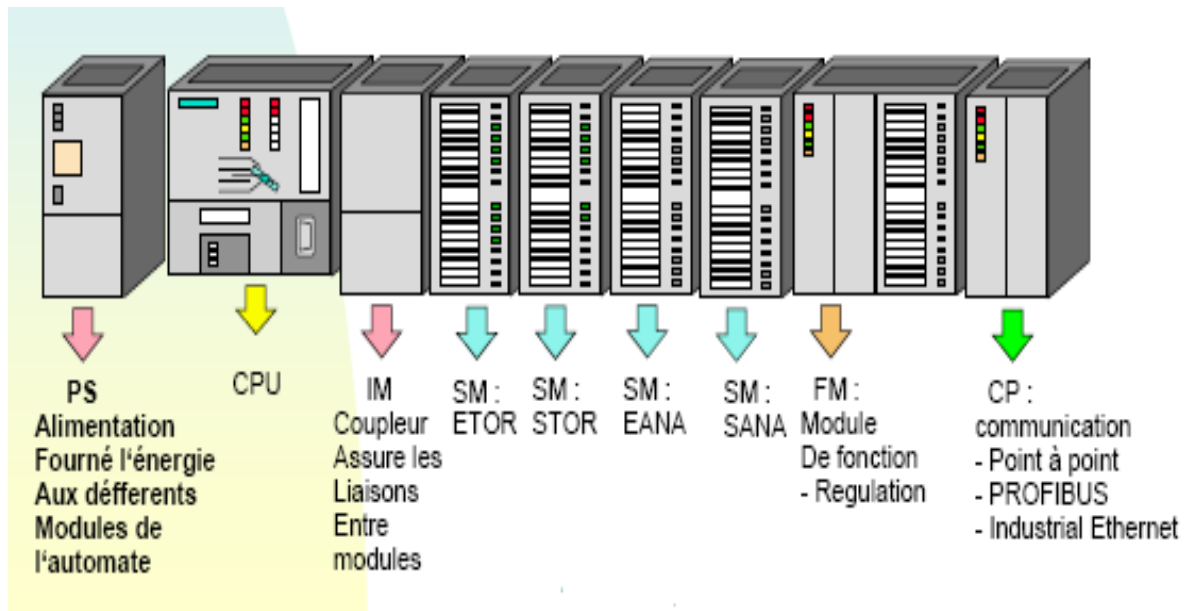
L'automate S7-300 offre les caractéristiques suivantes :

- Gamme diversifiée de CPU.
- Gamme complète de modules.
- Possibilité d'extension jusqu'à 32 modules.
- Bus de fond de panier, intégré au module.
- Possibilité de mise en réseau avec MPI PROFIBUS ou INDUSTRIAL ETHERNET.
- Raccordement central de la PG avec accès à tous les modules.
- Liberté de montage aux différents emplacements.
- Configuration et paramétrage à l'aide de l'outil configuration matérielle.

### 4.2 Constitutions de l'automate S7-300

L'automate programmable S7-300 (Figure III.4) est un système d'automatisation modulaire offrant la gamme des modules suivants :

- Module d'alimentation (PS) 2A, 5A, 10A.
- Unités centrales (CPU 312).
- Module de signaux (SM) pour entrées et sorties TOR et analogiques.
- Module d'extension (IM) pour configuration multi rangée du S7-300.
- Module de fonction (FM) pour les fonctions spéciales (par exemple activation d'un moteur asynchrone).
- Processus de communication (CP) pour la connexion au réseau.



**Figure III.4 :** présentation de l'automate S7-300.

#### 4. 2.1 Module d'alimentation

Le module d'alimentation transforme la tension secteur en tension d'alimentation, pour les modules de l'automate programmable. Cette tension s'élève à 24V. Les tensions pour les capteurs, actionneurs et voyants lumineux qui dépassent 24V, sont fournis par des blocs d'alimentation ou transformateurs supplémentaires.

#### 4.2.2 L'unité centrale (CPU)

La CPU est le cerveau de l'automate. Elle lit les états des signaux d'entrées, exécute le programme utilisateur et commande les sorties. Elle permet le réglage du comportement au démarrage et le diagnostic. Des défauts par les LEDs.

L'utilisateur a le choix parmi plusieurs CPU aux performances étagées, la CPU est logée dans un boîtier compact et comporte les éléments suivants :

- Les LEDs pour la signalisation d'état et de défauts.
- Un commutateur à clé pour les modules de fonctionnement :
  - Stop (arrêt).
  - Run / Run-P (marche).
- Un port pour interface MPI.
- Logement pour la carte mémoire (EPROM flash).

La CPU 312 est caractérisée par :

- Mémoire de travail : RAM 16KB.
- Carte mémoire micro Simatic : 64KB à 4MB.
- Langage de programmation : STEP7.
- Organisation de programme : linéaire, structurée.
- Temps d'exécution pour :
  - Opération sur bit : 0.3  $\mu$ s.
  - Opération sur mots : 4 $\mu$ s.
  - Opération de comptage/ temporisation : 10 $\mu$ s.

### **4.2.3 Module de coupleur**

Les coupleurs IM360 et IM365 permettent de configurer le S7-300 sur plusieurs rangées, et assurent la liaison entre les châssis et le couplage entre les différentes unités. Ainsi, la communication entre les entrées / sorties et d'autres périphérique et l'unité centrale, est assurée.

### **4.2.3 Module d'entrées /sortie (TOR)**

Les modules d'entrées /sorties sont des interfaces de communication entre l'unité centrale et les différents capteurs et actionneurs. Ils assurent le filtrage et l'adaptation des signaux électriques.

- **Les modules d'entrées :** Ils permettent à l'automate de recevoir des informations prévenantes, soit de la part des capteurs (entrées logique, analogiques ou numérique) ou bien du pupitre de commande.
- **Les modules de sorties :** Les modules de sorties permettent de raccorder l'automate avec les différents prés-actionneurs et actionneurs.
  - Les prés-actionneurs sont les (contacteurs, distributeurs, relais de puissance etc.).
  - Les actionneurs (moteur, éléments de signalisation, etc.)

Les sorties peuvent être logiques, analogiques ou bien numériques.

### **4.2.5 Module de fonction (FM)**

Ces modules réduisent la charge de traitement de la CPU, en assurant des tâches lourdes en calcul. Comme ils assurent, aussi, les fonctions spéciales telles que le comptage, la régulation et la commande numérique.

### **4.2.6 Module de communication (CP)**

Par des exigences très fortes en vitesse de transmission rapide, et de gros volumes de données, les modules de communication jouent un rôle clé dans le cadre de la communication industrielle. Ils permettent d'établir des liaisons homme-machine qui sont effectuées à l'aide des interfaces de communication.

### **4.2.7 Châssis d'extension**

Il est constitué d'un profilé support en aluminium et bus de fond de panier avec connecteur. Il permet le montage et le raccordement électrique de divers modules de l'automate.

### **4.2.8 Console de programmation**

Il existe deux types de consoles (PG ou PC), l'une permet le paramétrage et les relevés d'informations (modification des valeurs et visualisation), l'autre permet, en plus de la programmation, le réglage et l'exploitation.

## **5. Logiciel de programmation**

Le logiciel STEP-7 constitue l'outil standard pour les systèmes d'automatisation SIMATIC. Il permet à l'opérateur une utilisation simple et confortable de ces systèmes performants.

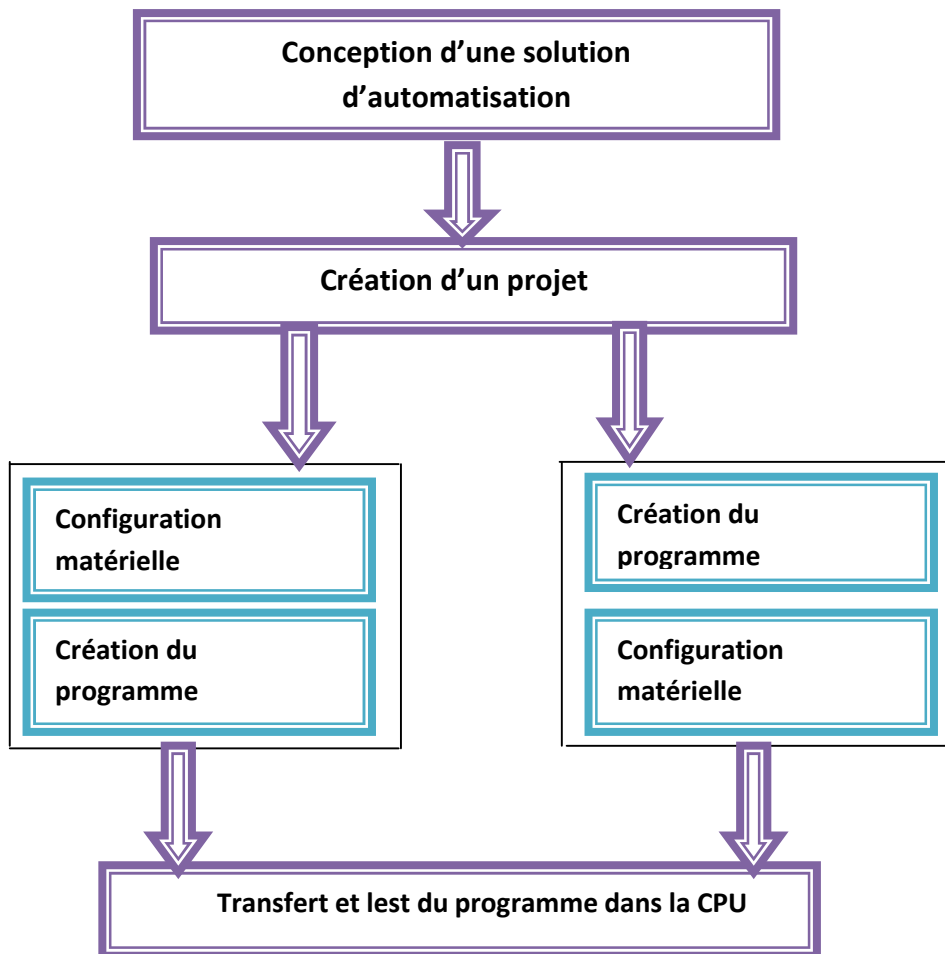
### **5.1 Création du projet**

Le logiciel STEP-7, dans ses différents langages de programmation (CONT, LIST, LOG), possède un nombre important de blocs, destinés à structurer le programme utilisateur, qui donne les avantages suivants :

- écriture des programmes importants mais clairs ;
- standardiser certaines parties du programme ;

- simplification de l'organisation du programme, car on peut l'exécuter section par section ;
- faciliter la mise en service.

La stratégie pour programmer dans STEP-7 peut se résumer dans la Figure III.5.



**Figure III.5 :** programmation avec le logiciel STEP7.

Avant de commencer la programmation, il est nécessaire de créer un projet, dans lequel les données et le programme utilisateur, à créer, seront structurés. La manière, la plus simple, pour créer un nouveau projet, est d'avoir recours à l'assistant « **Nouveau projet** ». Pour l'appeler, nous avons choisi la commande **Fichier > Assistant « Nouveau projet »**. Il nous posera les questions nécessaires dans des boîtes de dialogue et créera le projet.

L'étape suivante est le choix de la CPU (Figure III.6), pour notre cas nous avons choisi la CPU 312 car notre programme n'est pas compliqué, en d'autres termes notre station ne possède pas beaucoup d'entrées sorties.

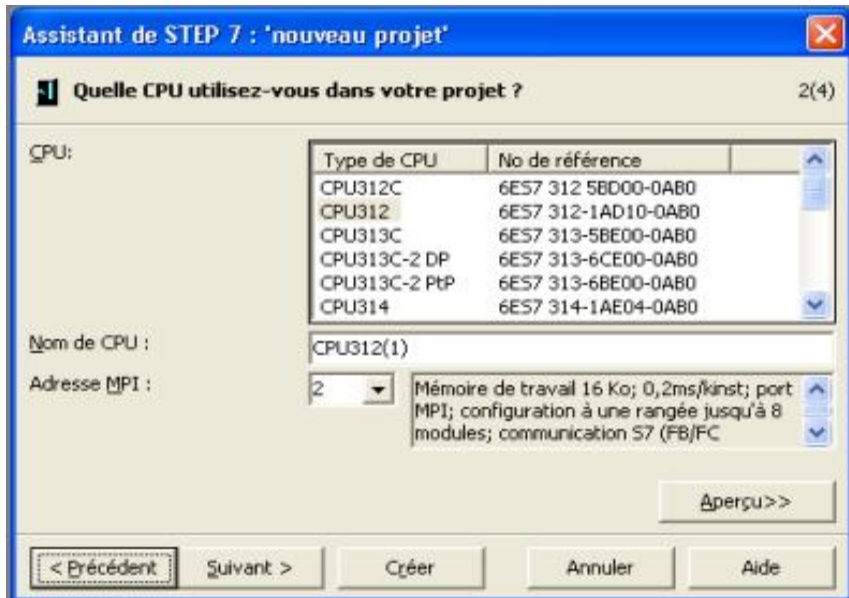


Figure III.6 : fenêtre du choix de la CPU.

Après, une autre fenêtre apparaîtra et permettant de choisir le langage de programmation (CONT) et L'OB1 (cycle d'exécution) (Figure III.7).

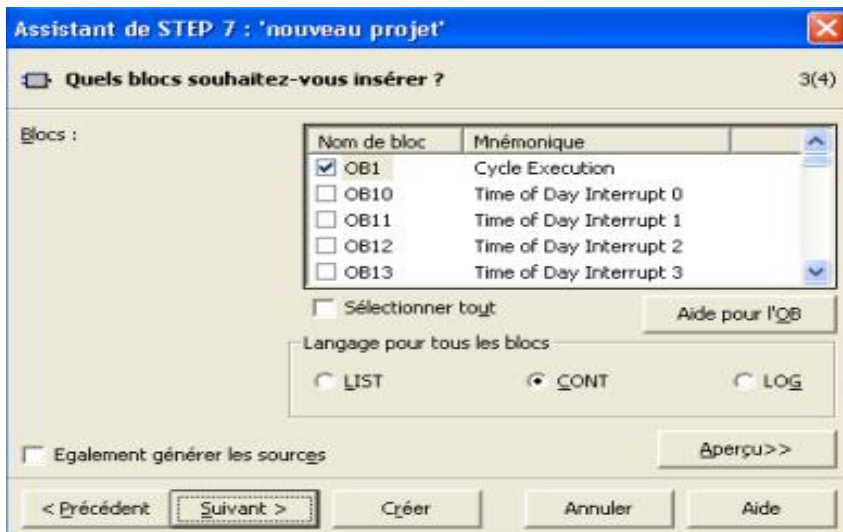


Figure III.7: sélection des blocs et mode de programme.

Et comme étape suivante la nomination du projet « transporteur du sucre »

## 5-2 Configuration et paramétrage du matériel

Nous entendons par « configuration », ce qui suit la disposition de profilés support ou châssis, de modules, d'appareils de la périphérie décentralisée et de cartouches interface dans une fenêtre de station. Les profilés support ou châssis sont représentés par une table

de configuration, dans la quelle l'on peut définir un nombre défini de modules, tout comme dans les profilés support ou châssis « réel ».

STEP-7 affecte automatiquement une adresse à chaque module dans la table de configuration. Nous pouvons modifier les adresses des modules d'une station, à condition que la CPU permette l'adressage libre. Aussi les erreurs éventuelles sont immédiatement détectées et signalées.

Pour la configuration matérielle de la station de transport du sucre, nous avons utilisé un seul rack (Figure III.8). Cela est justifié par le nombre d'entrée/sorties que possède la station ainsi que leur nature, par exemple E/S logique pour ce qui est boutons poussoirs, interrupteurs, capteurs, moteurs, etc.

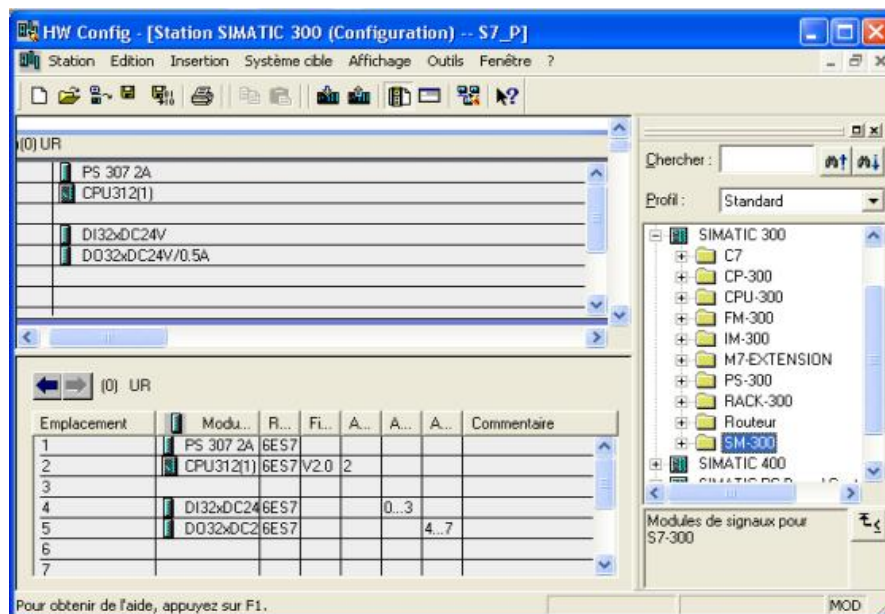


Figure III-8 : fenêtre de configuration matérielle de notre automate.

### ❖ Caractéristiques de la configuration matérielle

Selon les caractéristiques et la nature de nos entrées/sorties, le choix s'est porté sur les modules suivant :

- **unité centrale** : CPU312/312-1AD10-0AB0, Mémoire de travail : 16 KB.
- **Alimentation** : PS 307/307-1BA00-0AA0 (2A), Alimentation externe 120/230v continue / alternatif.
- **Module d'entrées** : DI32\*DC24v.
- **Module de sorties** : DO32\*DC24v/0.5A.

### 5.3 Créations du programme utilisateur

Le programme utilisé comprend toutes les constructions et déclaration ainsi que les données nécessaires au traitement de signaux de commande d'une installation ou d'un processus.

Un programme utilisateur devra être exécuté dans la CPU S7. Il est essentiellement constitué de blocs. Le programme utilisateur contient aussi des informations supplémentaires, telles que les données destinées à la configuration ou à la mise en réseau du système. En fonction de l'application, on peut donc créer, dans le programme utilisateur, les blocs de types suivants :

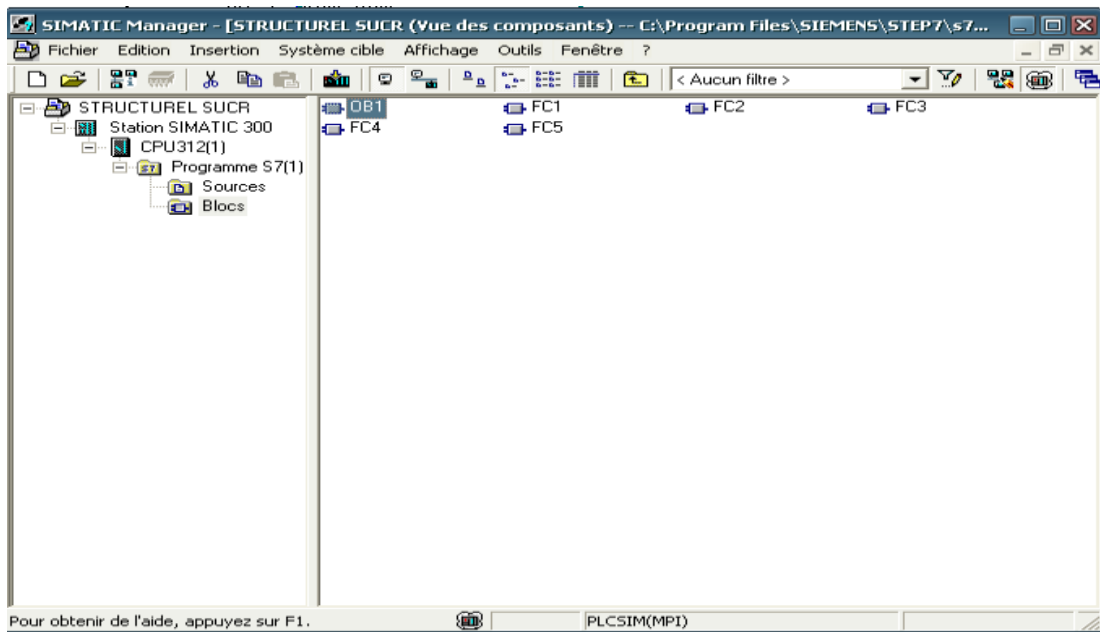
- Blocs d'organisation(OB).
- Blocs fonctionnels(FB).
- Blocs de données(DB).
- Fonctions(FC).

Des blocs souvent utilisés, relevant du système tel que les blocs fonctionnels système (SFB) et les fonctions système(SFC), sont intégrés au logiciel.

### 5.4 Structure du programme utilisateur

La programmation peut être linéaire ou structurée en fonction de la nature de la tâche d'automatisation.

Le programme que nous avons utilisé est de type structuré (Figure III.9), car il simplifie l'organisation et la gestion du programme. Le test de ce dernier peut être fait section par section et facilite la mise en service.



**Figure III.9** : structure du programme de la station de transport du sucre.

L'OB1 est la structure importante du programme, il constitue l'interface entre le programme utilisateur et le système d'exploitation, c'est-à-dire que la CPU exécute uniquement l'instruction se trouvant sur ce bloc.

Nous avons choisi d'utiliser les fonctions FC qui, contrairement, au bloc fonctionnel ne possède pas de zone mémoire (Figure III-10). Une fois tout les FC sont programmés, nous les insérons dans le bloc d'organisation OB1 pour la phase de simulation.

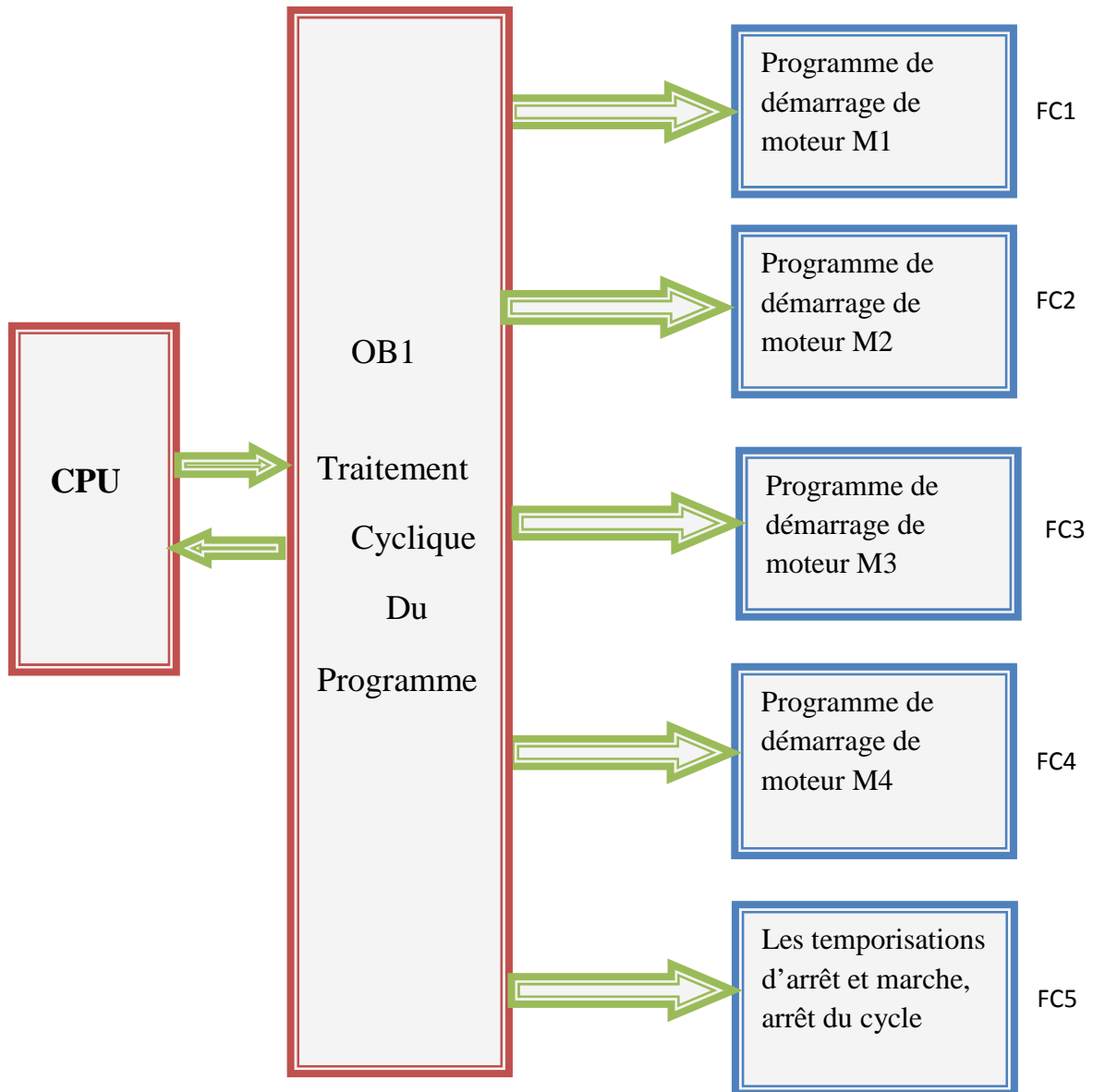


Figure III.10 : arborescence du programme.

## 6. Exemple d'une partie de notre programme

### 6.1. La fonction FC2

Dans ce bloc, nous avons programmé le démarrage de moteur M2 qui entraîne la vis sans fin de haut de silo pour évacuer le reste des blocs du sucre qui sont restés sur le tamis. Pour cela nous avons utilisé le réseau suivant :

Nous avons défini les conditions d'activation et de désactivation du moteur à l'aide d'une bascule SR.

- ❖ **Les entrées :** la figure III.11 suivante montre les entrées du système pour le bloc FC2.

Contenu de : 'Environnement\I'		
Nom	Type de données	
LOCAL_M1	Bool	
OM1	Bool	
LOCAL_M2	Bool	
HLS	Bool	
DEF_TH_2	Bool	
DET_RO_2	Bool	
ARR_SEQ	Bool	

**Figure III.11 :** fenêtre de représentation des entrées de FC2.

- ❖ **Les sorties :** la figure III.12 suivante montre les sorties du système pour le bloc FC1.

Contenu de : 'Environnement\Interface\OUT'			
Nom	Type de données	Commentaire	
OM2	Bool		

**Figure III.12 :** fenêtre de représentation des sorties de FC2

La figure III.13 suivante illustre Le programme de démarrage du moteur M2.

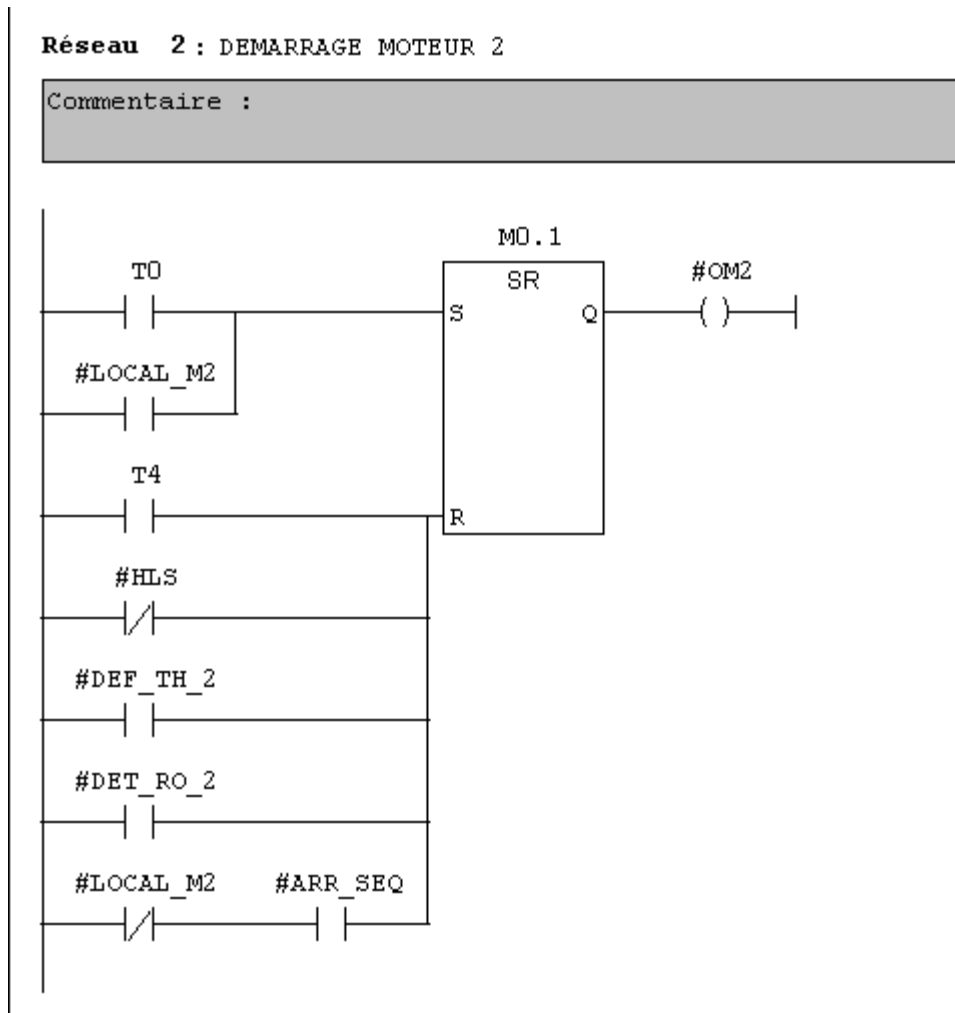


Figure III.13 : fenêtre de programme du moteur M2.

## 7. Simulation avec le logiciel S7-PLCSIM

L'application de simulation de modules S7-PLCSM nous permet d'exécuter et de tester notre programme. La simulation étant complètement réalisée au sein du logiciel STEP-7.

L'API S7 de simulation nous permet de tester des programmes destinés aux CPU S7-300 et aux CPU S7-4000, puis de remédier à d'éventuelles erreurs.

S7-PLCSM dispose d'une interface simple nous permettant de visualiser et de forcer les différents paramètres utilisés par le programme (comme, par exemple, d'activer ou désactiver des entrées) (Figure III.14).



Figure III.14 : fenêtre du S7-PLCSIM.

- **Exemple de simulation de notre programme**

Dans cet exemple, nous allons simulé la marche du moteur M2 avec les conditions de démarrage, comme représenté sur les deux Figures III.15 et III.16.

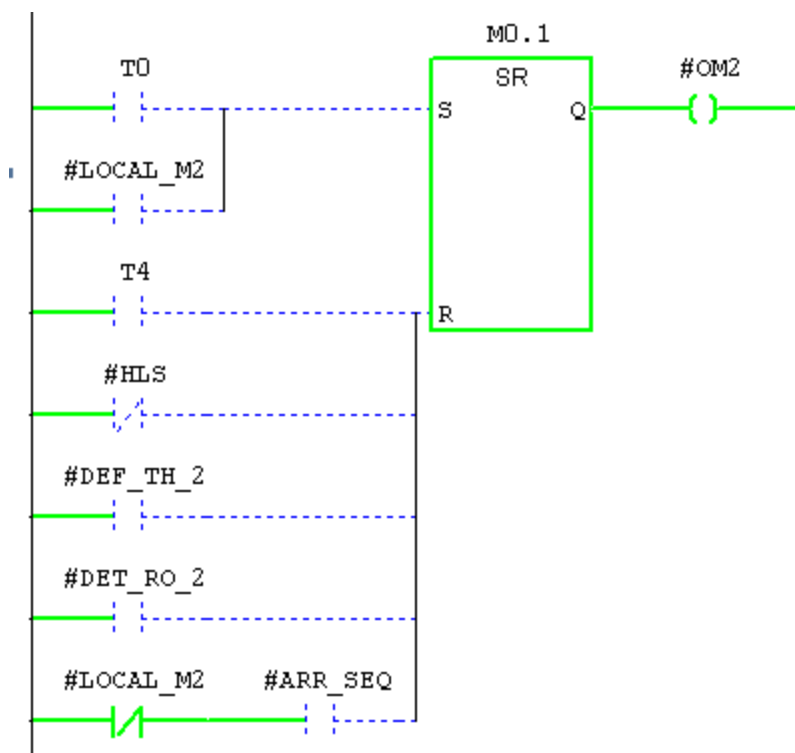


Figure III.15 : fenêtre de simulation de FC2.

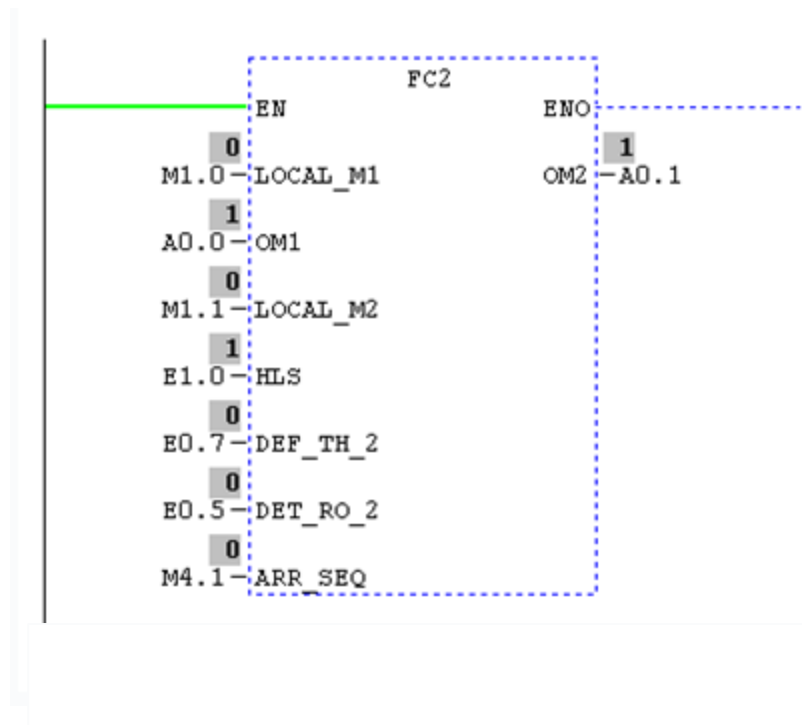


Figure III.16 : fenêtre de simulation d'OB1.

## 8. Conclusion

Aujourd'hui, l'automate programmable n'est plus seulement une machine séquentielle mais il est beaucoup plus considéré comme un calculateur de processus grâce aux énormes progrès, quant à la structure de base, la qualité et la diversité des outils proposés, et ses langages de programmation. Son insertion dans le procédé à automatiser constitue un passage pour augmenter la performance des processus.

## **1. Introduction**

Avec le développement de l'informatique, il est devenu possible de traiter des données dans le domaine industriel, grâce à des vues préalablement créées et configurées, et à l'aide d'un logiciel adéquat.

Ce logiciel est une entité capable de présenter à l'opérateur des informations utiles, afin qu'il prenne à temps les bonnes décisions pour la conduite du procédé. Il a, essentiellement, pour mission de collecter les données (acquisition et stockage) et les mettre en forme (traitement), afin de les présenter à l'opérateur (supervision).

## **2. La supervision**

La supervision se situe au plus haut niveau dans la hiérarchie des fonctions de production. Il est, donc, essentiel de présenter à l'opérateur, sous forme adéquate, les informations sur le procédé indispensable pour une éventuelle prise de décision. Cette présentation passe par les images synoptiques qui représentent un ensemble de vue. Le processus est représenté par une synoptique comprenant des images et des objets animés par l'état des organes de commande et les valeurs transmises par les capteurs.

## **3. Constitution d'un système de supervision**

La majorité des systèmes de supervision se compose, généralement, d'un moteur central (logique) auquel se rattachent des données provenant des équipements (automate). Ce moteur central assure l'affichage, le traitement et l'archivage des données, ainsi que la communication avec d'autres périphériques (Figure IV.1).

### **3.1. Module de visualisation**

Il permet d'obtenir et de mettre à la disposition des opérateurs des éléments d'évaluation du procédé par ses volumes de données instantanées

### **3.2. Module d'archivage**

Il mémorise des données (alarme et événement) pendant une longue période, et permet l'exploitation des données pour des applications spécifiques à des fins de maintenance ou de gestion de production.

### 3.3. Module de traitement

Il permet de mettre en forme les données, afin de les présenter, via le module de visualisation, aux opérateurs sous une forme prédéfinie.

### 3.4. Module de communication

Le module de communication assure l'acquisition et le transfert des données et gère la communication avec les automates programmable industriels et autres périphériques.

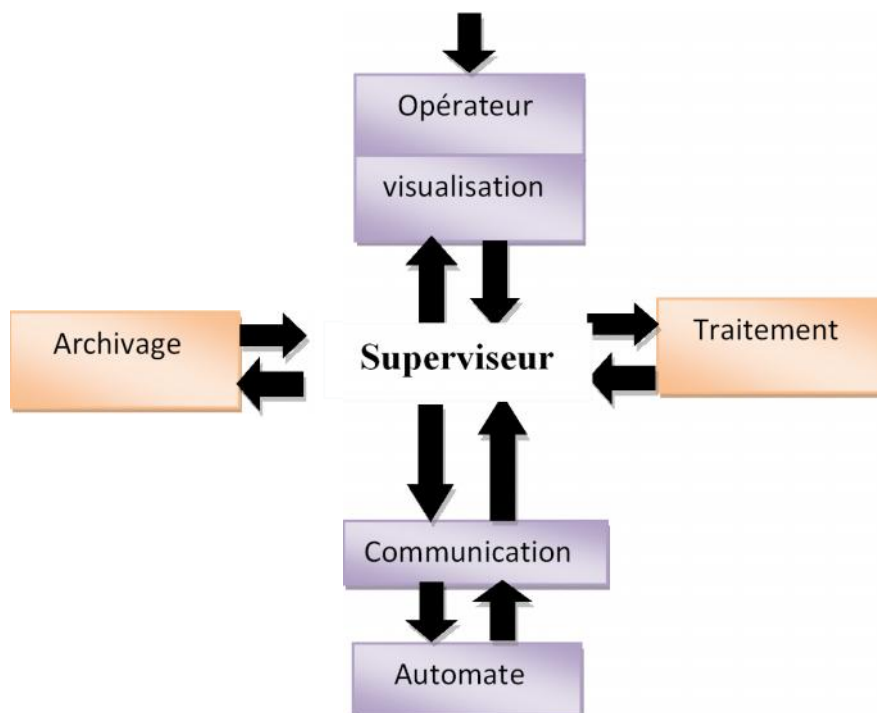


Figure IV.1 : schéma synoptique d'un système de supervision.

## 4. Logiciel de supervision WinCC

WinCC (Windows Control Centre) est un logiciel de supervision développé par SIEMENS. Il est caractérisé par sa flexibilité, c'est-à-dire qu'il peut être utilisé par un composant hors SIEMENS.

Ce logiciel est une Interface Homme Machine (IHM) graphique, qui assure la visualisation et le diagnostic du procédé. Il permet la saisie, l'affichage et l'archivage des données, tout en facilitant les tâches de conduite et de surveillance aux exploitants. Il offre une bonne solution de supervision, car il met à la disposition de l'opérateur des fonctionnalités adaptées aux exigences d'une installation industrielle.

## **4.1. Applications disponibles sous WinCC**

WinCC se compose de plusieurs applications pour accomplir la fonction de supervision. Il dispose des modules suivants :

### ***4.1.1 Graphic designer***

Il offre la possibilité de créer des vues de procédés, et de les configurer en leur affectant les variables correspondantes. A cet effet, il dispose d'une bibliothèque d'objets et permet de créer des objets selon le besoin. Il assure la fonction de visualisation grâce au **graphic runtime**.

### ***4.1.2 Tag logging***

On y définit les archives, les valeurs du processus à archiver et les temps de cycle de saisie et d'archivage. En outre on y configure la mémoire tampon sur le disque.

### ***4.1.3 Alarm logging***

Il se charge de l'acquisition et de l'archivage des alarmes, en mettant à la disposition des utilisateurs. Les fonctions nécessaires à la prise des alarmes issues du procédé, à leur traitement, leur visualisation, leur acquittement et leur archivage.

### ***4.1.4 Global script runtime***

Il dispose de deux éditeurs : l'éditeur C et l'éditeur Visuel Basic (VBS), à l'aide desquels on crée des actions et des fonctions qui ne sont pas prévues dans le WinCC.

### ***4.1.5 Report designer***

Il contient des informations avec lesquelles on peut lancer la visualisation d'une impression ou d'un ordre d'impression. On y trouve aussi des modules de mise en page du journal, qu'on peut adapter en fonction du besoin.

### ***4.1.6 User administrator***


C'est là que s'effectue la gestion des utilisateurs et des automatisations. On y crée des nouveaux utilisateurs, on leur attribue des mots de passe et on leur affecte la liste des autorisations.

## 5. Applications développée sous WinCC

La supervision de la station de transport du sucre a été élaborée à l'aide du logiciel WinCC, vu ses particularités. Les différentes étapes à suivre, pour créer notre application sont :

- Créer un projet.
- Sélectionner et installer L'API.
- Définir les variables dans l'éditeur de variables.
- Créer et éditer les vues (vue d'accueil, vue de tous les ouvrages) dans l'éditeur graphic designer.
- Paramétrer les propriétés de WinCC runtime.
- Activer les vues dans le WinCC runtime.
- Utiliser le simulateur pour tester les vues du processus.

### a) Création d'un projet WinCC

On démarre WinCC en double cliquant sur l'icône (  ), la fenêtre du choix de projet s'affiche. On sélectionne un projet monoposte (Figure IV-2).

La communication entre l'API S7-300 et WinCC consiste à configurer le système de supervision en sélectionnant, à partir de l'éditeur gestion des variables, le pilote « SIMATIC S7 Protocol suite » et choisir le MPI comme réseau de communication.

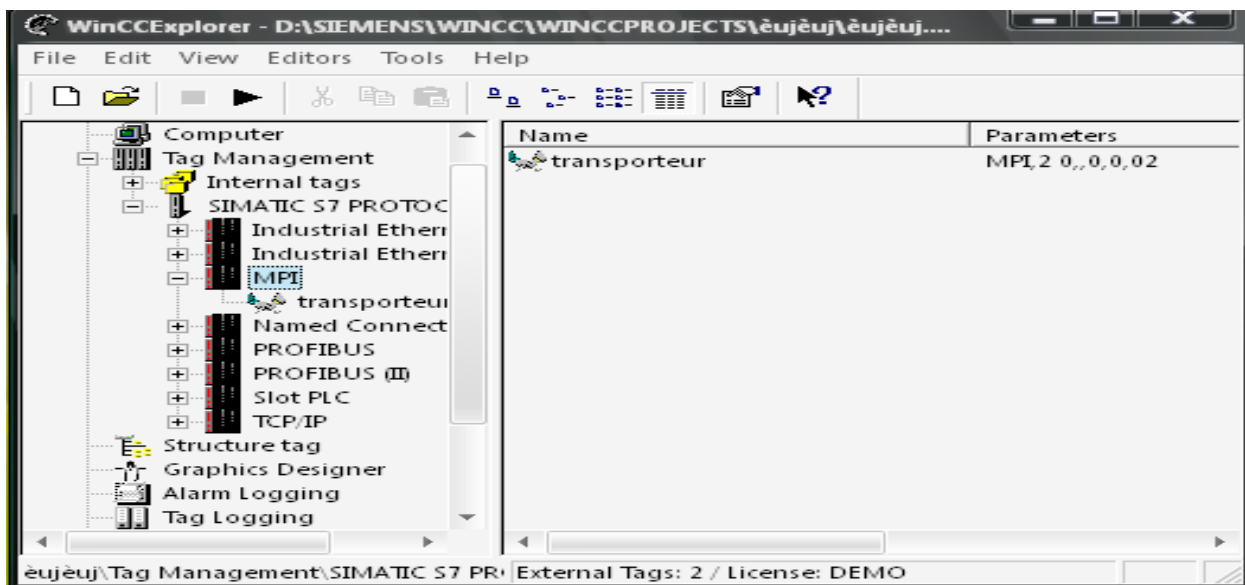


Figure IV.2 : configuration du système de supervision.

## b) Création des paramètres du processus

L'étape suivante est l'introduction des variables de processus. Ces variables correspondent aux variables manipulées par le programme de l'API S7-300(Figure IV.3).

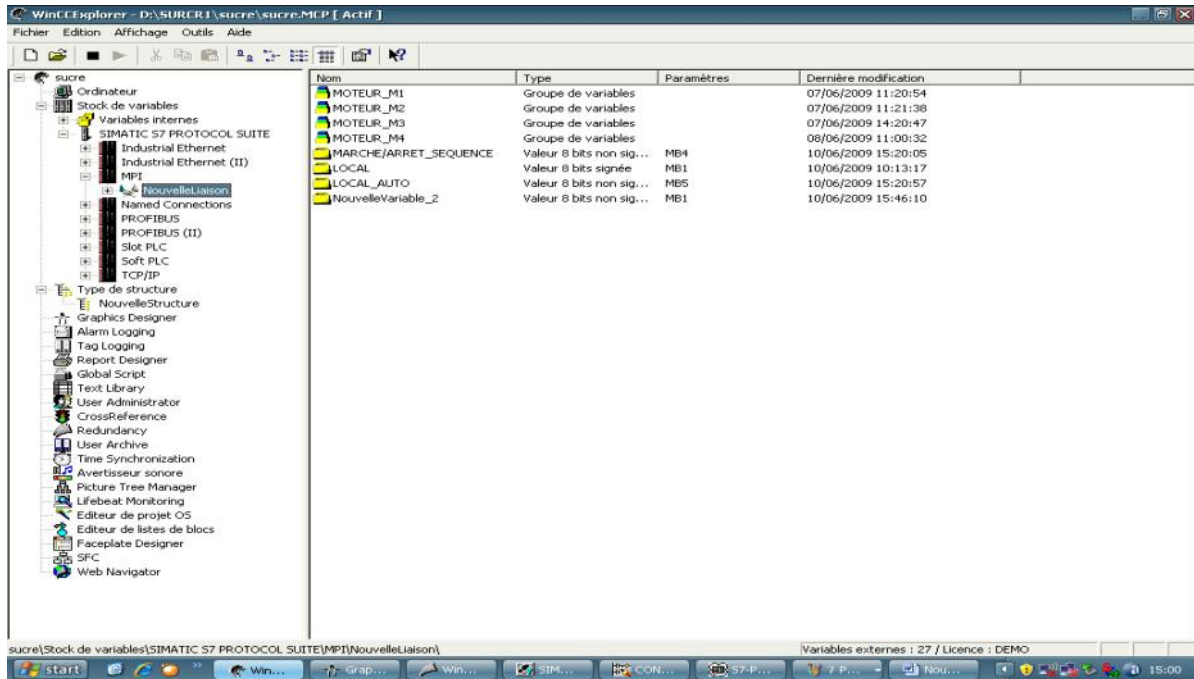


Figure IV.3 : création des paramètres au processus.

## c) Création des vues

Cette étape consiste à créer des vues dans l'éditeur « Graphiques designer », qui nous permettent d'insérer les différents objets dont nous avons besoin, grâce à la bibliothèque interne de WinCC. Les figures suivantes nous montrent le nombre de vues créées dans notre projet.

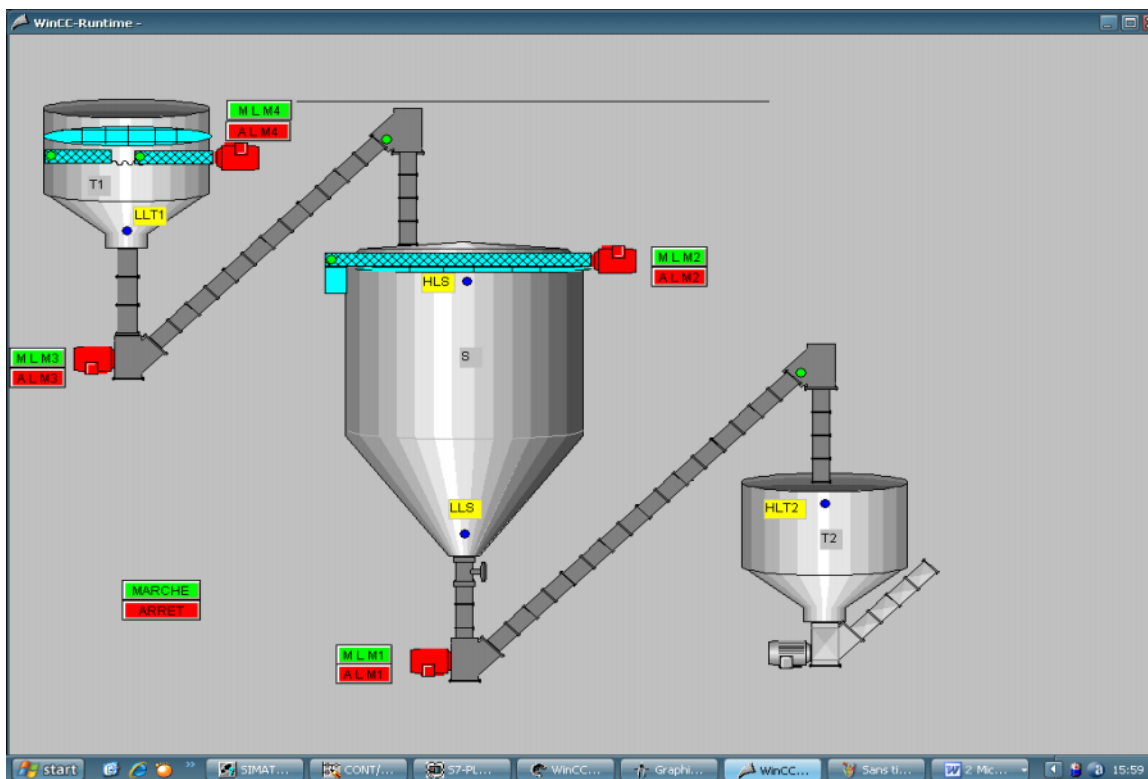
A travers ces vues, la station de transport du sucre est constituée, essentiellement, des moteurs, des capteurs et détecteurs de rotation :

- Vue de la station (arrêt).
- Vue de la station (démarrage des moteur M1 et M2).
- Vue de la station (démarrage complet de la station).

## d) Activation du projet

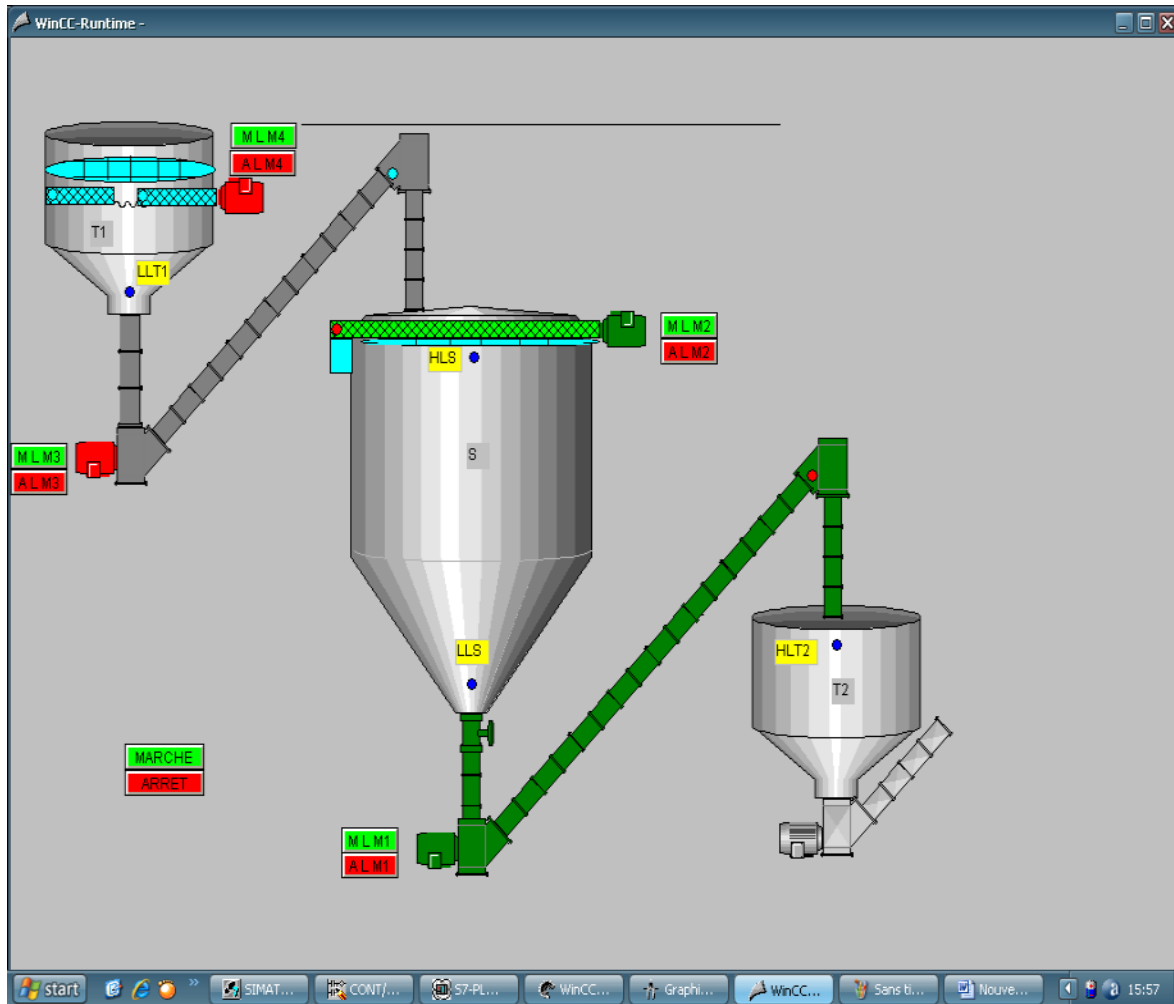
Après le paramétrage des propriétés WinCC runtime, nous accédons à l'activation projet en cliquant sur **Fichier-Activer**

La figure IV.4, nous présente la vue complète de la station de transport du sucre à l'arrêt, qui comprend des boutons marche et arrêt local pour chaque moteur et un bouton poussoir de démarrage et arrêt de la station complète (automatique).



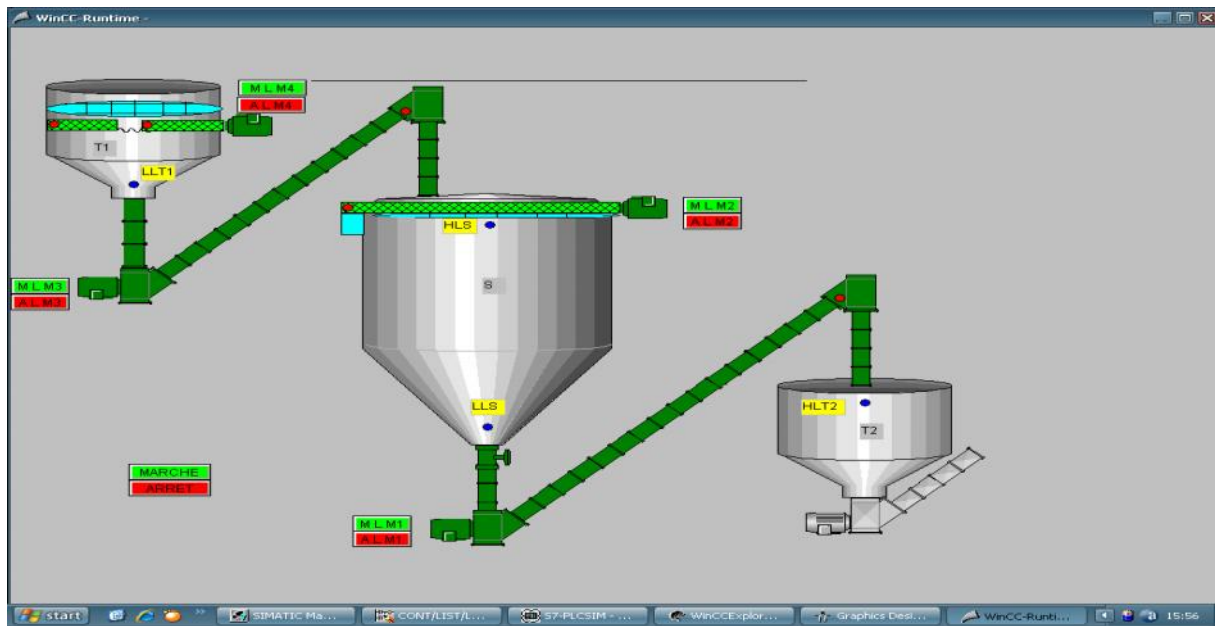
**Figure IV.4 :** vue de la station en arrêt

La vue suivante (Figure IV.5), présente la station de transport du sucre à l'état de marche (début), où les moteurs M1 et M2 sont en marche et les moteurs M3 et M4 attendent l'écoulement de la temporisation.



**Figure IV.5:**vue de démarrage des moteurs M1 et M2.

A la figure IV.6 nous avons présenté la vue complète de la station en marche.



**Figure IV-6** : vue de démarrage de toute la station.

## 6. Conclusion

Dans ce dernier chapitre, qui est consacré à la supervision de la station de transport du sucre, nous avons décrit la supervision en précisant sa place dans l'industrie. Nous avons élaboré, sous le logiciel WinCC, les vues qui permettent de suivre l'évolution de la station en fonction du temps. Cela se fait par activation des variables à partir du simulateur de modules physiques, le S7-PLCSIM.

### Conclusion générale

Dans ce travail, il nous a été demandé d'automatiser, à l'aide d'un automate programmable industriel, la station de transport du sucre en sein de Fruital Coca-Cola Alger. Ce transporteur étant manuellement commandé, d'où la nécessité de la substituer avec un moyen d'automatisation plus évolué qui est l'API.

Nous avons, en premier lieu, étudié le fonctionnement de la station de transport du sucre. Ensuite, nous avons élaboré une modélisation cohérente de notre procédé à l'aide du GRAFCET. Le modèle grafcet développé nous a beaucoup aidés au passage vers la programmation en langage STEP-7 et l'élaboration d'une solution programmable dans l'automate programmable industriel. Nous avons effectué une simulation avec le logiciel S7-PLCSIM, qui nous permet de visualiser et de valider nos résultats obtenus.

Dans notre travail, nous avons élaboré une solution de supervision dont le but est de contrôler le déroulement du processus par l'intermédiaire de graphismes et de schémas en temps réel. Il est donc facile de cibler, en cas de panne, un élément défectueux parmi les capteurs et les actionneurs. Ainsi, l'opérateur peut intervenir et prendre les décisions appropriées pour remédier aux défauts survenus en un temps minime.

Cependant, le remplissage de la trémie 1, via un Clark, reste une tâche toujours manuelle et son automatisation ne fera qu'accélérer, davantage, la production. Espérant que ça fera un sujet pour les promotions à venir.

Enfin, ce projet nous a permis de nous familiariser avec : les automates programmables industriels (S7-300), le logiciel de supervision WinCC, de mettre en œuvre nos connaissances théoriques et de faire face à une situation purement industrielle. Il nous a permis aussi de forger nos connaissances dans le domaine d'automatisation des procédés industriels, et d'apprendre à dépasser toutes les difficultés que présente une telle situation d'automatisation.

# Bibliographie

- [1] : Documentation technique de l'entreprise Fruitful Coca-Cola.
- [2] : Documentation technique SIEMENS 2006 sur les API S7-300.
- [3] : Documents fournis avec le logiciel de programmation STEP7.
- [4] : Documents fournis avec le logiciel de supervision WinCC.
- [5] : Sites internet.
- [6]: Data sheet de nivocompact FTC 731
- [7] : **Mahmoudi. N, Ahmed Ouamer. D** « développement d'une station programmable et supervision à base d'un API S7-300 de la station de conditionnement du lait ». Mémoire de fin d'étude promotion 2008.
- [8] : **Bouya. S, Bouacem. M** « conceptions d'une commande programmable pour une presse à élaborer à base d'un API SIMATIC S7-300». Mémoire de fin d'étude promotion 2007.
- [9] : **Benmammam. S, Benbdi. B**« automatisations d'une chaîne de production de lait à base de l'API S7-300 ». Mémoire de fin d'étude promotion 2007
- [10] : **Khelil. Y, Djematene. F** « automatisations d'une soudeuse par point ». Mémoire de fin d'étude promotion 2005