



République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique



Université Mohamed Khider de Biskra  
Faculté des Sciences et de la Technologie  
Département de génie électrique

## **Mémoire de Master**

En vue de l'obtention du Diplôme de Master en Automatique  
Option : Automatique et informatique industrielle

### **Thème :**

---

Programmation de la séquence de concassage et de transport  
de la matière d'ajout au ciment par l'automate S7-300

---

Présenté et soutenu par :

**TOUHAMI Abdelhakim**

#### **Devant le jury :**

**Dr : BENCHAAABANE Fateh**

**MCA**

**Président**

**Dr : MESSAOUDI Abdelhamid**

**MCB**

**Encadreur**

**Dr : ABDOU Latifa**

**PROF**

**Examineur**

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire

وزارة التعليم العالي و البحث العلمي

Ministère de l'enseignement Supérieur et de la recherche scientifique



Université Mohamed Khider Biskra

Faculté des Sciences et de la Technologie

Département de Génie Electrique

Filière : Automatique

Option : Automatique et informatique industrielle

Mémoire de Fin d'Etudes

En vue de l'obtention du diplôme :

**MASTER**

**Thème :**

Programmation de la séquence de concassage et de transport  
de la matière d'ajout au ciment par l'automate S7-300

**Présenté par :**

**TOUHAMI Abdelhakim**

**Avis favorable de l'encadreur :**

**MESSAOUDI Abdelhamid**

**Avis favorable du Président du Jury**

**Cachet et signature**

الجمهورية الجزائرية الديمقراطية الشعبية  
République Algérienne Démocratique et Populaire  
وزارة التعليم العالي و البحث العلمي  
Ministère de l'enseignement Supérieur et de la recherche scientifique



**Université Mohamed Khider Biskra**  
**Faculté des Sciences et de la Technologie**  
**Département de Génie Electrique**  
**Filière : Automatique**  
**Option : Automatique et informatique industrielle**

**Mémoire de Fin d'Etudes**  
**En vue de l'obtention du diplôme :**  
**MASTER**

**Thème :**

**Programmation de la séquence de concassage et de transport  
de la matière d'ajout au ciment par l'automate S7-300**

Présenté par :  
**TOUHAMI Abdelhakim**

Dirigé et encadré par :  
**MESSAOUDI Abdelhamid**

**Résumé :**

L'objectif de ce travail est de faire l'automatisation et la supervision d'un processus industriel (la séquence de concassage et de transport de la matière d'ajout au ciment), pour ça on a utilisé l'automate programmable industriel S7-300 programmé avec le logiciel STEP7 par langage à contact.

On a développé aussi une interface graphique HMI (Human Machine Interface) pour la supervision utilisant le logiciel WinCC flexible.

Ce travail nous a permet de familiariser avec l'automate programmable S7-300 et la programmation de processus industriels On a aussi développé quelques connaissances dans le domaine industriel.

## ملخص :

الهدف من هذا العمل هو أتمتة والإشراف على عملية صناعية (تسلسل تكسير ونقل المواد المضافة للأسمنت)، استخدمنا في هذا العمل المتحكم المنطقي القابل للبرمجة الصناعية S7-300، الذي قمنا ببرمجته باستخدام برنامج STEP7، كما تم تطوير واجهة رسومية HMI (واجهة الجهاز البشري) للمراقبة باستخدام برنامج WinCC. لقد سمح لنا هذا العمل بالتعرف على جهاز المتحكم المنطقي القابل للبرمجة S7-300 وبرمجة العمليات الصناعية ، كما سمح لنا بتطوير بعض المعارف في المجال الصناعي.

# Remerciements

*Je tiens tout d'abord à remercier Dieu de m'avoir donné la force et la patience pour réaliser ce modeste travail.*

*Je tiens à remercier vivement mon encadreur MESSAOUDI Abdelhamid, pour ses conseils précieux et pour l'assistance qu'il m'a apportée durant mon étude et ma réalisation de ce travail.*

*Mes remerciements les plus vifs s'adressent aussi aux Messieurs le président et les membres de jury d'avoir accepté d'examiner et d'évaluer mon travail.*

*J'exprime également ma gratitude à tous les enseignants qui ont collaboré à ma formation depuis mon premier cycle d'étude jusqu'à la fin de mon cycle universitaire.*

# Dédicaces

*Je dédie ce travail  
à mes chers parents  
à tous mes amis*

# Liste des figures

Figure 1.1: SPA BISKRIA CIMENT.....	3
Figure 1.2: Organigramme de l'usine SPA BISKRIA CIMENT.....	5
Figure 1.3: Les composants du ciment.....	6
Figure 1.4: Carrière.....	7
Figure 1.5: Concasseur à marteau.....	7
Figure 1.6: Concasseur à cylindre.....	7
Figure 1.7: Stockage circulaire.....	8
Figure 1.8: Stockage linéaire.....	8
Figure 1.9: Atelier de dosage.....	9
Figure 1.10: Broyeur cru.....	9
Figure 1.11: Zone cuisson.....	10
Figure 1.12: Préchauffeur ou cyclones.....	10
Figure 1.13: Four rotatif.....	11
Figure 1.14: Zone ciment.....	13
Figure 1.15: Zone expédition 1.....	13
Figure 1.16: Zone expédition 2.....	14
Figure 1.17: Les étapes de fabrication du ciment.....	14
Figure 2.1: Structure d'un système automatisé.....	16
Figure 2.2: Un automate programmable industriel.....	17
Figure 2.3: Fonctionnement cyclique d'un API.....	18
Figure 2.4: Automate de type compacte.....	19
Figure 2.5: Automate de type modulaire.....	19
Figure 2.6: Structure interne d'un API.....	22
Figure 2.7: Automate S7-300.....	23
Figure 2.8: Les modules du S7-300.....	24
Figure 3.1: Icône du STEP7.....	30
Figure 3.2: Création d'un projet STEP7.....	30
Figure 3.3: Choix de la station SIMATIC 300.....	31
Figure 3.4: Configuration hardware pour la station S7300.....	32
Figure 3.5: Sélection des modules.....	33
Figure 3.6: Table des mnémoniques.....	34
Figure 3.7: Edition des programmes.....	34

Figure 3.8: Exemple d'un programme dans l'OB1.....	35
Figure 3.9: Interface du simulateur S7-PLCSIM.....	37
Figure 3.10: Exemple de système supervision.....	38
Figure 3.11: La fenêtre principale du WinCC flexible.....	40
Figure 3.12: Paramétrage de la connexion.....	43
Figure 4.1: Organigramme de l'application.....	46
Figure 4.2: Le concasseur.....	48
Figure 4.3: Transporteur à tablier métallique.....	48
Figure 4.4: Séparateur magnétique.....	49
Figure 4.5: Les détecteurs de position.....	50
Figure 4.6: Des capteurs de vibrations.....	51
Figure 4.7: Moteur asynchrone triphasé.....	52
Figure 4.8: Contacteur.....	52
Figure 4.9: La séquence de concassage et de transport de la matière ajout.....	53
Figure 4.10: Grafcet du cahier de charge.....	54
Figure 4.11: Configuration de ces matériels.....	56
Figure 4.12: Table des mnémoniques.....	57
Figure 4.13: Simulation de l'application.....	58
Figure 4.14: Réseau de l'étape 0 dans le bloc OB1.....	59
Figure 4.15: Transformation d'une entrée analogique a un intervalle de température...59	59
Figure 4.16: La fenêtre « PIW256 ».....	60
Figure 4.17: Le comparateur avant d'atteindre la condition d'arrêt.....	60
Figure 4.18: La fenêtre « PIW256 » à la valeur maximale.....	60
Figure 4.19: L'arrêt du comparateur.....	61
Figure 4.20: Vue globale dans la séquence de notre application.....	62
Figure 4.21: Variables de notre application.....	63
Figure 4.22: La liaison entre le pupitre et la station.....	63
Figure 4.23: L'exécution du système (en état d'arrêté).....	64
Figure 4.24: Forçage des variables en S7-PLCSIM.....	65
Figure 4.25: L'exécution du système (en état de marche).....	66



# Liste des Tableaux

Tableau 2.1: Les modules prévus pour l'alimentation des CPU de S7-300.....	24
Tableau 4.1: Signification des symboles de GRAFCET.....	55

# Table des matières

<b>Introduction générale</b> .....	1
<b>Chapitre 1: Présentation de l'usine de ciment SPA BISKRIA CIMENT</b>	
1.1 Introduction .....	3
1.2 Présentation du l'usine de ciment SPA BISKRIA CIMENT.....	3
1.2.1 Description .....	3
1.2.2 Les produits du SPA BISKRIA .....	4
1.2.3 Organigramme de l'usine SPA BISKRIA CIMENT .....	5
1.2.4 C'est quoi le ciment? .....	5
1.2.5 Les zones du S.P.A BISKRIA CIMENT.....	6
1.2.5.1 Zone d'extraction.....	6
1.2.5.2 Zone cru.....	8
1.2.5.3 Zone cuisson.....	9
1.2.5.4 Zone ciment.....	12
1.2.5.5 Zone expédition.....	13
1.3 Conclusion.....	14
<b>Chapitre 2: Les automates programmables industriels</b>	
2.1 Introduction.....	15
2.2 Systèmes automatisés.....	15
2.2.1 Définition.....	15
2.2.2 Structure d'un système automatisé.....	15
2.3 Généralité sur l'automate programmable industriel.....	16
2.3.1. Historique.....	16
2.3.2. Définition d'un l'automate programmable industriel.....	16
2.3.3. Principe de fonctionnement.....	17
2.3.4. Type des automates programmables industrielles.....	18
2.3.5. Les éléments principales de l'automate programmable.....	20
2.3.6. Architecture interne de l'automate programmable.....	21
2.2.7. Critère de choix un API.....	22
2.4 L'automate S7-300.....	22
2.4.1. Caractéristique.....	23
2.4.2. Les modules du S7-300.....	23
2.4.3. Les langages de programmations.....	26

2.5 Conclusion.....	27
<b>Chapitre 3: Logiciels utilisés</b>	
3.1 Introduction.....	28
3.2 STEP7.....	28
3.2.1 Définition.....	28
3.2.2 Création d'un projet STEP7.....	30
3.2.3 Configuration matérielle.....	31
3.2.4 Définitions des mnémoniques.....	33
3.2.5 Edition des programmes.....	34
3.2.6 Simulation de module.....	36
3.3 WinCC flexible.....	37
3.3.1 Définition.....	37
3.3.2 Généralités sur la supervision.....	38
3.3.3 Les éléments de WinCC flexible.....	39
3.3.4 Intégration de WinCC dans STEP7.....	42
3.3.5 Exécution de la simulation du système de supervision.....	44
3.4 Conclusion.....	45
<b>Chapitre 4: Application</b>	
4.1 Introduction.....	46
4.2 Les étapes de développement de notre travail.....	46
4.3 Séquence de concassage et transport de matière ajout.....	47
4.3.1 Description.....	47
4.3.2 Les éléments principaux de la séquence.....	47
4.3.3 Les capteurs.....	49
4.3.4 Les actionneurs.....	51
4.3.5 Les pré-actionneurs.....	52
4.4 Principe de fonctionnement.....	53
4.4.1 Cahier de charge.....	53
4.4.2 Grafcet du cahier de charge.....	54
4.5 Programmation sur STEP7.....	55
4.5.1 Configuration du matériel dans STEP7.....	55
4.5.2 Table des mnémoniques.....	57
4.5.3 Simulation dans PLCSIM.....	58
4.5.4 Programme en STEP7.....	58
4.6 Création de la station HMI sur WinCC flexible.....	61

4.6.1 Création d'une vue d'interface principale de notre application.....	62
4.6.2 Création des variables de notre application.....	62
4.6.3 Configuration de la liaison.....	63
4.6.4 La simulation de projet à l'aide de WinCC flexible Runtime.....	64
4.7 Conclusion.....	66
<b>Conclusion générale.....</b>	<b>67</b>
<b>Bibliographie.....</b>	<b>68</b>
<b>Annexe.....</b>	<b>70</b>

# **Introduction générale**

## Introduction générale

Au cours des dernières décennies, les procédés industriels ont connu un essor considérable, grâce à l'automatisme. C'est ce qui incite les entreprises à automatiser leurs processus de production, afin d'assurer leur pérennité, tout en améliorant les conditions de travail de leur personnel, en éliminant les tâches pénibles et répétitives [8].

Depuis plus d'une décennie, les entreprises Algériennes ont commencé à renouveler et à automatiser leurs installations. La cimenterie SPA BISKRIA CIMENT a fait un grand pas dans ce domaine, Elle prend actuellement le défi pour améliorer ces performances, par la proposition des solutions correctives de l'état actuel, qui visent le doublement de la capacité de production de l'usine [19].

L'automate programmable industriel API (Programmable Logic Controller PLC) est l'un des appareils de commande des systèmes de production et d'automatisme les plus utilisés dans l'industrie, Ils sont devenus aujourd'hui les constituants les plus répandus des installations automatisées [8].

Notre travail est effectué au sein de l'usine SPA BISKRIA CIMENT. L'objectif est de faire l'automatisation et la supervision de processus industriel (la séquence de concassage et de transport de la matière ajout du ciment) par automate S7-300.

Pour atteindre notre objectif, nous avons utilisé le logiciel de programmation STEP7 fourni par SIEMENS, qui contient un simulateur des automates SIEMENS tel que le S7-300, ainsi que le logiciel de supervision WinCC flexible.

Pour une meilleure présentation de notre travail, le mémoire a été organisé en quatre chapitres. Le premier est consacré à la présentation de l'usine SPA BISKRIA CIMENT et ses zones. Le chapitre deux donne un bref aperçu sur la structure des systèmes automatisés ainsi que les automates programmables industriels, en particulier l'automate S7-300. Le troisième chapitre expose les logiciels de programmations utilisées dans notre travail (STEP7 et WinCC flexible).

En fin, le dernier chapitre sera consacré à la description de la séquence de concassage et transport de la matière ajout du ciment avec ses différents composants, ainsi que la simulation et la supervision du programme correspondant au cahier des charges de notre processus.

Le choix de cet exemple d'application vise à appliquer nos connaissances théoriques à la programmation des processus industriels tous dans logiciel STEP7 et WinCC, et ce qui nous permet d'expérimenter plus facilement et en toute sécurité des nouvelles solutions, que nous ne pouvons pas exécuter sur le terrain. Ainsi d'approfondir nos connaissances théoriques en automatisation industrielle.

# **Chapitre 1:**

Présentation de l'usine SPA  
BISKRIA CIMENT



## 1.1 Introduction

L'Algérie occupe une position économique importante dans le secteur de la construction et du ciment, grâce aux sociétés de ciment modernes.

Une de ces sociétés est la société SPA BISKRIA CIMENT. Dans le cadre de notre projet fin d'étude, nous présenterons dans ce chapitre la société SPA BISKRIA CIMENT et ses zones principales.

## 1.2 Présentation de l'usine de ciment SPA BISKRIA CIMENT

### 1.2.1 Description

La société SPA BISKRIA CIMENT est une entreprise de fabrication et ventes des ciments au Capital social : 870.000.000,00 DA.



Figure 1.1: SPA BISKRIA CIMENT

Le ciment est le plus important matériau de construction de notre temps. C'est un produit «high-tech» issu de la nature, qui a bénéficié d'un développement s'étendant sur des siècles. Le ciment est plus précisément un liant. Le matériau de construction proprement dit, c'est le béton, qui base sur sa composition / qualité sur le ciment ce dernier est le composants le plus chère du béton.

Le but de cette société est d'améliorer l'efficacité au niveau du développement socio-économique en Algérie, et conforter l'avantage concurrentiel au niveau national et international; Donc l'objectif en premier lieu consiste à satisfaire le marché national d'un produit local tout en permettant un accès stable aux matériaux de construction en termes de qualité, de choix, de services à des prix abordables. Et en deuxième lieu d'ouvrir des marchés internationaux.

La cimenterie possède de trois lignes de productions avec une capacité totale de quatre million Ton/an [1].

### **1.2.2 Les produits du SPA BISKRIA**

La SPA BISKRIA CIMENT produit des ciments de qualité qui sont systématiquement contrôlés par le laboratoire de la cimenterie et périodiquement par le Centre d'études et de Services Technologiques de l'Industrie des Matériaux de Construction CETIM (selon le référentiel ISO 17025). Ces produits sont: [1]

- Ciment Portland Composite CEM II/A 42,5 NA 442
- Ciment Portland Composite CEM II/B 32.5 NA 442
- Ciment Résistant aux Sulfates (CRS) CEM I 42.5 – ES NA 443
- Ciment Résistant aux Sulfates (CRS) CEM IV/A-SR 42.5 NA 443

### 1.2.3 Organigramme de l'usine SPA BISKRIA CIMENT

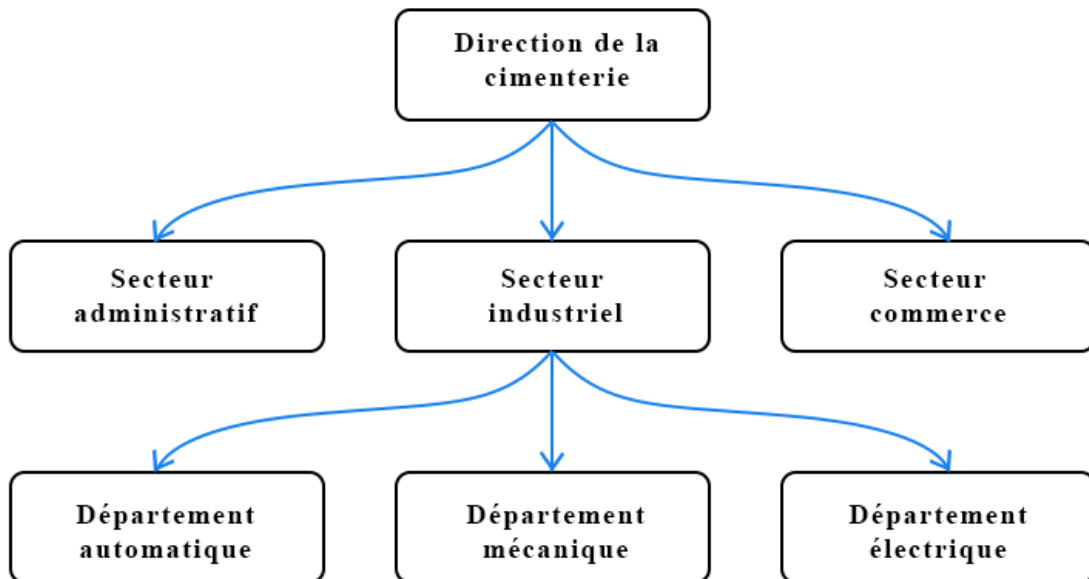


Figure 1.2: Organigramme de l'usine SPA BISKRIA CIMENT

La S.P.A BISKRIA CIMENT procède trois secteurs [1] :

- Secteur administratif : contient de services administratifs pour gérer la société
- Secteur industriel : contient les trois lignes de production d'une capacité de quatre million T/an.
- Secteur commerciale : contient le service de vente et d'expédition

### 1.2.4 C'est quoi le ciment?

Le ciment est un liant hydraulique, c'est-à-dire un matériau minéral finement moulu qui, gâché avec de l'eau, forme une pâte qui fait prise et durcit par suite de réactions et de processus d'hydratation et qui, après durcissement conserve sa résistance et sa stabilité, même sous l'eau. Le ciment est obtenu par broyage et cuisson à 1 450°C, d'un mélange de calcaire et d'argile. Appelé CLINKER, ce matériau granulaire est pour l'essentiel d'une combinaison de chaux, de silice, d'alumine et d'oxyde de fer. Le ciment est un mélange pulvérulent, de couleur grise, qui, gâché avec de l'eau, forme une pâte durcissant tant sous l'eau qu'à l'air [2].

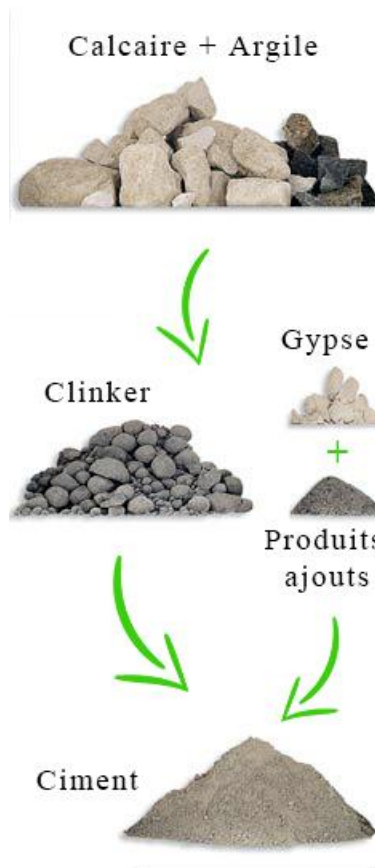


Figure 1.3: Les composants du ciment

## 1.2.5 Les zones du S.P.A BISKRIA CIMENT

Les zones d'usine du SPA BISKRIA CIMENT représentent respectivement les étapes de fabrication du ciment.

### 1.2.5.1 Zone d'extraction

L'extraction des matières premières vierges (comme le calcaire « 75% à 80% » et l'argile « 20% à 25% ») à partir de la carrière. Ces matières premières sont extraites des parois rocheuses par abattage à l'explosif ou à la pelle mécanique. La roche est acheminée par des tombereaux, des dumpers ou des bandes transporteuses vers un atelier de concassage [3].



Figure 1.4: Carrière

### 1.2.5.1.1 Concassage

Le concassage est fait pour l'optimisation et la facilité de stockage et la manutention des matières premières [3].

Dans la société SPA BISKRIA on a trois types de concasseurs :

- concasseur à marteau pour le calcaire.
- concasseur a cylindre pour l'argile et le minerai de fer.
- concasseur a cylindre pour gypse.

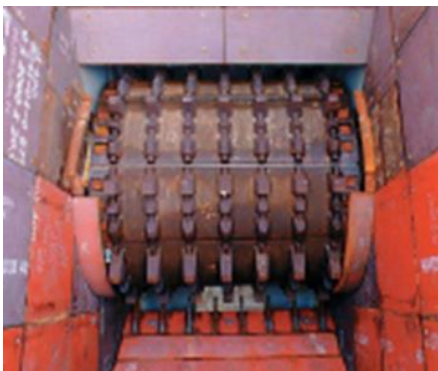


Figure 1.6: Concasseur à cylindre



Figure 1.5: Concasseur à marteau

### 1.2.5.1.2 Atelier de stockage

Après l'opération de concassage de ces trois constituants de base on obtient une granulométrie de 0 à 25 males constituant sont acheminés vers l'usine par des de tapis roulant couvert puis ce stocké dans deux halls de pré-homogénéisation [3]:

- Hall de pré-homogénéisation et stockage linéaire, Figure 1.8.
- Hall de pré-homogénéisation et stockage circulaire, Figure 1.7.



Figure 1.8: Stockage linéaire



Figure 1.7: Stockage circulaire

### 1.2.5.2 Zone cru

#### 1.2.5.2.1 Atelier de dosage

La matière cru est constitué d'un mélange des matière premières de correction dans des proportion qui sont définies suivant les valeurs des modules chimiques de cru, en général la matière cru est constituée de 70% de calcaire, 20% d'argile et 10% entre le sable et le minerai de fer, quatre doseurs sont installés sous les trémies, le calcaire et l'argile et le minerai de fer et le sable sont respectivement extraits par ces doseurs, chaque doseur est systématisé d'une manière automatique, les matières premières extraites sont chargé sur un seul convoyeur à bande puis transportées au broyeur cru[4].



Figure 1.9: Atelier de dosage

### 1.2.5.2.2 Le broyeur cru

Son composant principal est un cylindre de plus gros diamètre et de longueur plus petite qui peut tourner par l'intermédiaire du mécanisme de transmission. Les matériaux sont placés dans le cylindre à travers l'ouverture d'alimentation et ensuite broyés à fond, car au cours de la chute des billes de fer et le cru peut produire un impact énorme de chute broyage. Le cru est alors réduit en poudre (farine) et ce chargé sur un convoyeur à puis transportées au silo homogènes [5].



Figure 1.10: Broyeur cru

### 1.2.5.3 Zone cuisson

La zone cuisson se compose de trois étapes principales, ce sont :

- Réchauffement ou cyclones
- Four rotatif
- Le refroidissement

La figure 1.11 montre les sous-zones de la zone cuisson

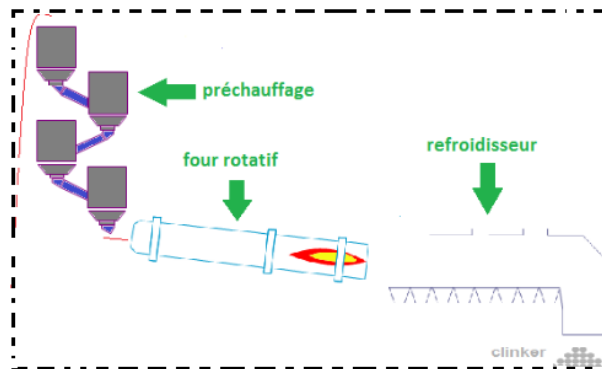


Figure 1.11: Zone cuisson

### 1.2.5.3.1 Préchauffeur ou cyclones

Les gazes réchauffent la poudre crue qui circule dans les cyclones en sens inverse, par gravité. Figure 9. La Farine cru s'échauffe ainsi jusqu'à 800 °C environ et perd donc son gaz carbonique (CO<sub>2</sub>) et son eau. C'est un échangeur à contre-courant destiné à préchauffer la farine avant son entrée dans le four, en récupérant la chaleur de gaz sortant du four "environ égale à 100 c», par le fait que l'argile et le calcaire ont la même densité (2,70 g/cm<sup>3</sup>), un exhausteur monté sur les cyclones aspire les gazes de combustions et le mélange carrière, faisant une tornade garantie que tous les grains du crue subit la chaleur des gazes. [4]



Figure 1.12: Préchauffeur ou cyclones



### 1.2.5.3.2 Four rotatif

La cuisson de la farine ce passe à une température de 1450 °C dans un four rotatif, Le four constitue par une virole cylindrique de 35m de long et de 5.6m de diamètre protégé par de brique réfractaire, incliné selon un angle de 1 à 4 degrés par rapport à l'horizontale, Voir la figure 1.13.

Le four tourne à des vitesses de rotation compris entre 1.8 et 3.5tr/min, la farine cru va suivre des différentes étapes de transformation lors de sa lente progression dans le four vers la partie basse, à la rencontre de la flamme.

Cette source de chaleur est alimentée au charbon broyé La température nécessaire au CLINKER en est de l'ordre de 1 450 °C. L'énergie consommée se situe entre 3 200 et 4 200 MJ par tonne de CLINKER, qui est le produit semi-fini obtenu à la fin du cycle de cuisson.

Il se présente sous forme de granules grises, et à la sortie du four le CLINKER doit être refroidi. [5]



Figure 1.13: Four rotatif

### 1.2.5.3.3 Le refroidissement

A la sortie de four rotatif le CLINKER doit refroidissement pour qu'il préserver sa formule chimique, par un refroidisseur a grilles il est équipé d'une batterie de puissants ventilateurs fournissant l'air de refroidissement.

Le refroidisseur a pour rôle d'abaisser la température du CLINKER tombant du four à une Température d'environ 1135 c jusqu'a 80-100 c, le CLINKER va progresser a l'intérieur grâce aux à coup répétés des grilles lesquelles il repose, au travers des grilles de puissants ventilateurs vont souffler sous le CLINKER afin de le refroidir et le transporté avec un convoyeur a silo de CLINKER. [4]

### 1.2.5.4 Zone ciment

Pour obtenir un ciment aux propriétés hydrauliques actives, le CLINKER doit être à son tour broyé très finement. Ce broyage s'effectue dans des broyeurs à boulets. Les corps broyants sont constitués de boulets d'acier qui, par choc, font éclater les grains de CLINKER et amènent progressivement le ciment à l'état de fine farine, ne comportant que très peu de grains supérieurs à 40 microns.

A la sortie du broyeur, un cyclone sépare les éléments suffisamment fins des autres qui sont envoyés à l'entrée du broyeur.

C'est également lors du broyage que l'on ajoute au CLINKER le gypse (3 à 5%) indispensable à la régulation de prise du ciment.

Les ciments « à ajouts » sont obtenus par l'addition au CLINKER, lors de son broyage, l'élément minéral supplémentaire contenu par exemple dans les laitiers de hauts fourneaux, les cendres de centrales thermiques, les fillers calcaires, les pouzzolanes naturelles ou artificielles.

Ainsi sont obtenues les différentes catégories, de ciments qui permettront la réalisation d'ouvrage allant du plus courant au plus exigeant. [5]

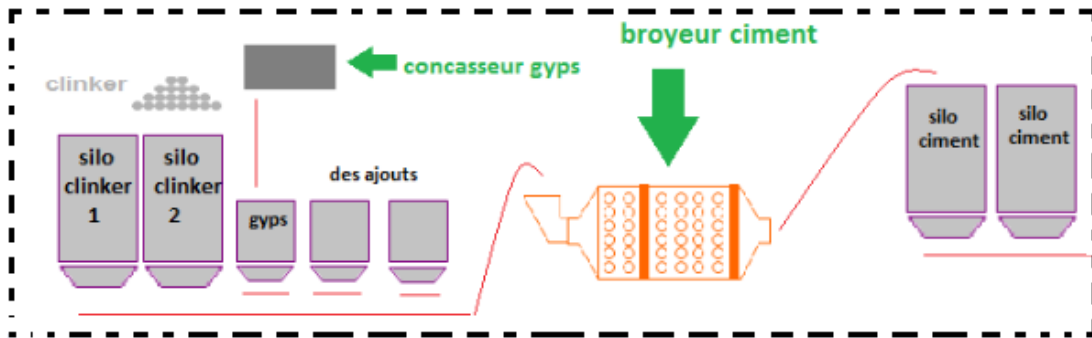


Figure 1.14: Zone ciment

### 1.2.5.5 Zone expédition

Acheminés vers les silos de stockage par transport pneumatique ou mécanique, les ciments quittent l'usine en sacs ou en vrac.

Les sacs contiennent généralement 25 ou 50 kg de ciment et l'ensachage atteint fréquemment 100 tonnes par heure.

Les sacs sont acheminés vers des palettiseurs qui constituent des palettes de 1500 kg transportées par camion.

Le ciment livré en vrac constitue 70% de la production. Il est transporté par camion-citerne, par train ou par bateau [5].



Ensachage



palettisation



Expedition

Figure 1.15: Zone expédition 1

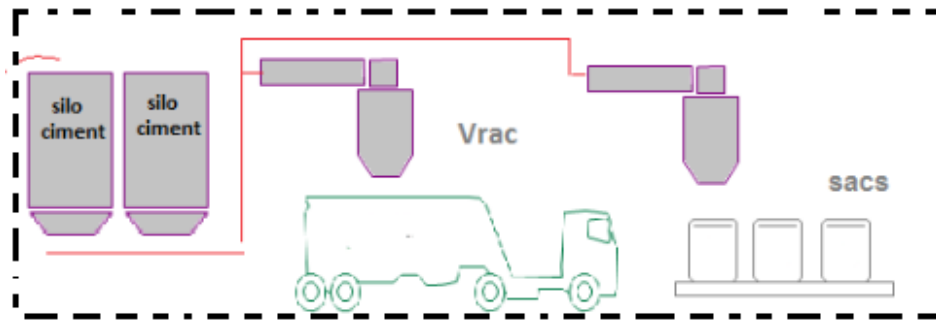


Figure 1.16: Zone expédition 2

La figure 7 illustre les étapes de fabrication du ciment

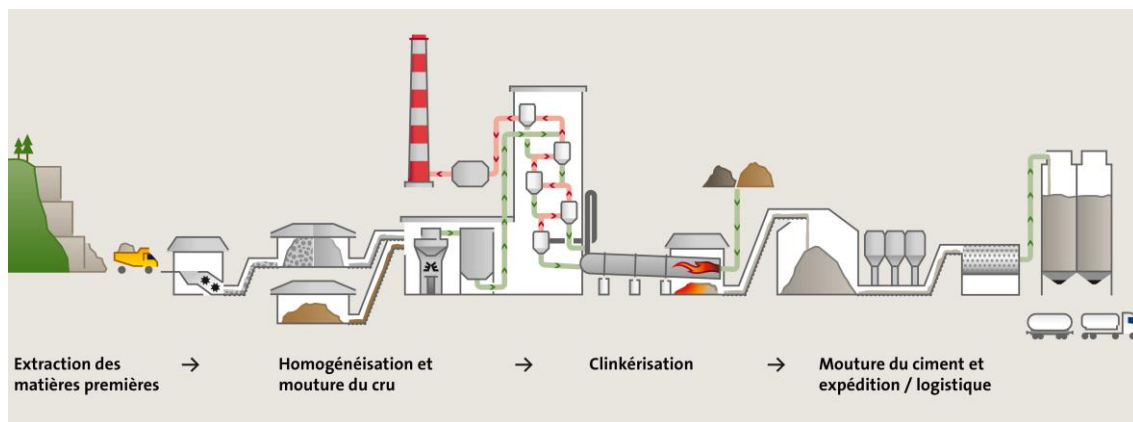


Figure 1.17: Les étapes de fabrication du ciment

### 1.3 Conclusion

On a présente dans ce chapitre l'usine du ciment et ses différentes zones et les étapes de production de ciment. Et l'explication du fonctionnement de ces zones principales.

Ensuite nous avons présenté la séquence de concassage et transport de matière ajout, ce qui sera notre thème étude appliquée.

## **Chapitre 2:**

# Les automates programmables industriels

## 2.1 Introduction

L'automate programmable industriel API(ou Programmable Logic Controller PLC) est l'un des appareils de commande des systèmes de production et d'automatisme les plus utilisés dans l'industrie.

Dans ce chapitre, nous donnons un bref aperçu sur la structure des systèmes automatisés ainsi que les automates programmables industrielles, à travers un bref aperçu historique, et quelques définitions principales, Ensuite nous présenterons l'automate S7-300 en détail et de ses composants internes et l'architecture interne.

## 2.2 Systèmes automatisés

### 2.2.1 Définition

L'automatisation d'un système consiste à transformer l'ensemble des tâches de commande et de surveillance, réalisées par des opérateurs humains, dans un ensemble d'objets techniques appelés partie commande. Cette dernière mémorise le savoir-faire des opérateurs, pour obtenir l'ensemble des actions à effectuer sur la matière d'œuvre, afin d'élaborer le produit final.

Parmi les objectifs de l'automatisation on peut citer : réalisation des tâches répétitives, sécurité, économie des matières premières et l'énergie, augmentation de la productivité [8].

### 2.2.2 Structure d'un système automatisé

Un système automatisé est toujours composé d'une partie commande (PC) et une partie opérative (PO) [9].

Pour faire fonctionner ce système, l'opérateur va donner des consignes à la PC. Celle-ci va traduire ces consignes en ordres qui vont être exécutées par la PO. Une fois les ordres accomplis, la PO va le signaler à la PC, par un retour d'information, qui va à son tour le signaler à l'opérateur, ce dernier pourra donc dire que le travail a bien été réalisé.

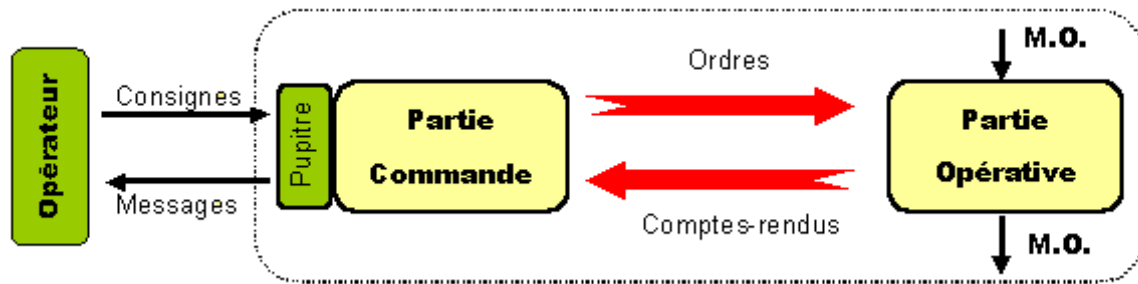


Figure 2.1: Structure d'un système automatisé

## 2.3 Généralité sur l'automate programmable industriel

### 2.3.1. Historique

Les automates programmables sont apparus aux USA vers les années 1969, à l'époque ils étaient destinés essentiellement à automatiser les chaînes de montages automobiles. Ils sont apparus en France en 1971, ils sont de plus en plus employés dans l'industrie.

Un Automate Programmable Industrielle (API) est distingué d'un ordinateur par le fait qu'il s'agit d'un système électrique programmable, spécialement adapté pour les non-informaticiens. Il est généralement destiné à être mis entre les mains d'un personnel dont la formation a été surtout orientée vers l'électromécanique.

C'était une révolution industrielle à cette époque, en raison de sa facilité d'utilisation et de programmation [7].

### 2.3.2. Définition d'un l'automate programmable industriel

Un Automate programmable est un appareil dédié au contrôle d'une machine ou d'un processus industriel, constitué de composants électroniques, comportant une mémoire programmable par un utilisateur non informaticien, à l'aide d'un langage adapté. En d'autres termes, un automate programmable est un ordinateur logique, ou ordinateur, au jeu d'instructions volontairement réduit, destiné à la conduite et la surveillance en temps réel de processus industriels [10].

Trois caractéristiques fondamentales distinguent totalement l'Automate Programmable Industriel (API) des outils informatiques tels que les ordinateurs (PC industriel ou autres) :

1. Peut être directement connecté aux capteurs et pré-actionneurs grâce à ses entrées/sorties industrielles,
2. Conçu pour fonctionner dans des ambiances industrielles sévères (température, vibrations, micro-coupure de la tension d'alimentation, parasites, etc.),
3. Sa programmation est faite à partir de langages spécialement développés pour le traitement de fonctions d'automatisme fait en sorte que sa mise en œuvre et son exploitation ne nécessitent pas une forte connaissance en informatique [10].

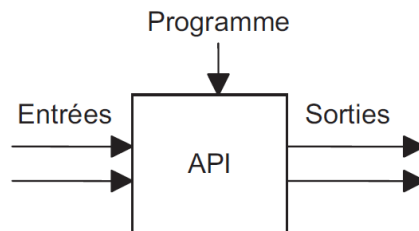


Figure 2.2: Un automate programmable industriel

### 2.3.3. Principe de fonctionnement

L'automate programmable reçoit des données par ses entrées, celles-ci sont ensuite traitées par un programme défini, le résultat obtenu étant délivré par ses sorties. Ce cycle de traitement est toujours le même, quel que soit le programme, néanmoins le temps d'un cycle d'API varie selon la taille du programme et la puissance de l'automate.

C'est l'unité centrale qui gère l'automate programmable : elle reçoit, mémorise et traite les données entrantes et détermine l'état des données sortantes en fonction du programme établi [11].



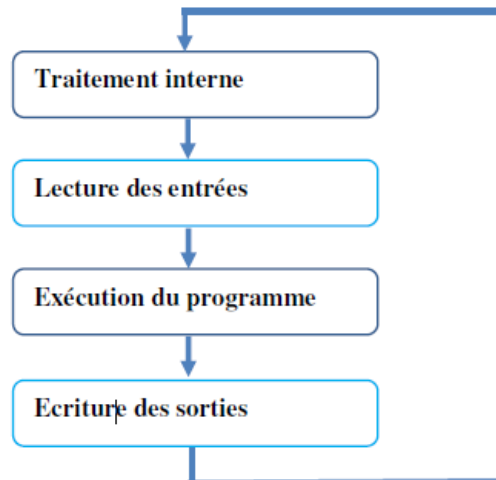


Figure 2.3: Fonctionnement cyclique d'un API

#### 2.3.4. Type des automates programmables industrielles

Les automates peuvent être de type compact ou modulaire [11]:

##### Type compact :

Un API compact est un API qui intègre le processeur, l'alimentation, les entrées, les sorties dans seule modules. Selon les modèles et les fabricants, il pourra aussi réaliser certaines fonctions supplémentaires comme comptage rapide, E/S analogique et recevoir des extensions en nombre limité.

Ces automates, de fonctionnement simple, sont généralement destinés à la commande de petite machine et des petites installations et pour l'automatisme décentralisés sur les annexes de grosses et moyennes machines. Quelque exemple de type compact.



Figure 2.4: Automate de type compacte

### Type modulaire :

Dans ce type d'API le processeur, l'alimentation et les interfaces d'entrées/sorties sont repartis dans unité séparées pour plusieurs raisons dont on peut citer la facilitation des opérations de la maintenance.

Ces différents modules sont fixés sur des Racks contenant le fond du panier, les bus et les connecteurs occurrence. Les Racks sont des lamelles métalliques qui une forme géométrique bien spécifique qui lui permet de recevoir les déférent modules des automates. Installé et enlever ces modules est particulièrement facile. Figure suivante illustre mieux ce dispositif.

Ces automates sont utilisés dans le cas où on a affaire à des structures moyennement et hautement complexes car ces derniers requièrent une puissance et une capacité de traitement assez élevées.



Figure 2.5: Automate de type modulaire

## 2.3.5. Les éléments principales de l'automate programmable

### 2.3.5.1. Unité central de traitement(CPU)

L'unité centrale de traitement est l'élément le plus important d'un API, les trois composants de CPU sont [11]:

- Le processeur
- La mémoire
- Les entrées-sorties

### 2.3.5.2. Processeur

Le processeur, ou unité traitement a pour rôle principal le traitement des instructions qui constituent le programme de fonctionnement de l'application (les fonctions logiques ET, OU, les fonctions de temporisation, de comptage, de calcul PID, etc.) [11].

Mais en dehors de cette tâche de base, il réalise également d'autres fonctions :

- Gestion des entrées/sorties,
- Surveillance et diagnostic de l'automate par une série de tests lancés à la mise sous tension ou cycliquement en cours de fonctionnement,
- Dialogue avec le terminal de programmation, aussi bien pour l'écriture et la mise au point du programme qu'en cours d'exploitation pour des réglages ou des vérifications.

### 2.3.5.3. La mémoire

Les mémoires RAM ou EEPPROM nécessaire pour stocker les programmes, les données, et les paramètres de configuration du système. La mémoire se compose de [11] :

- Mémoire système
- Mémoire principale (de travail)

#### **2.3.5.4. L'alimentation**

Il est composé de blocs, qui permettent de fournir à l'automate l'énergie nécessaire à son fonctionnement, à partir d'une alimentation en 220 volts alternatif. Ou bien de tension de 24V ,12V ou 5V en continu [11].

#### **2.3.5.5. Interface d'entrée /sortie**

Permettent au processeur de recevoir et d'envoyer des informations aux dispositifs extérieurs. Les entrées peuvent être des interrupteurs, ou d'autres capteurs, ... Les sorties peuvent être des bobines de moteur, des électrovannes, etc.

#### **2.3.5.6. Interface de communication**

Il est utilisée pour recevoir et transmettre des données sur des réseaux de communication qui relie l'API à d'autres API distants qui peut être [11]:

- Interface série utilisant dans la plupart des cas comme support de communication les liaisons RS232 ou RS422/RS485 pour la connexion à des terminaux (console, ou PC) pour assurer la communication Homme/Machine (programmation, supervision...)
- Interface pour assurer l'accès à un bus de terrain (Modbus, Profibus,...)
- Interface d'accès à un réseau Ethernet

#### **2.3.6. Architecture interne de l'automate programmable**

Quel que soit le type d'Automate programmable, compact ou modulaire, les automates sont organisés autour d'un même type d'architecture.

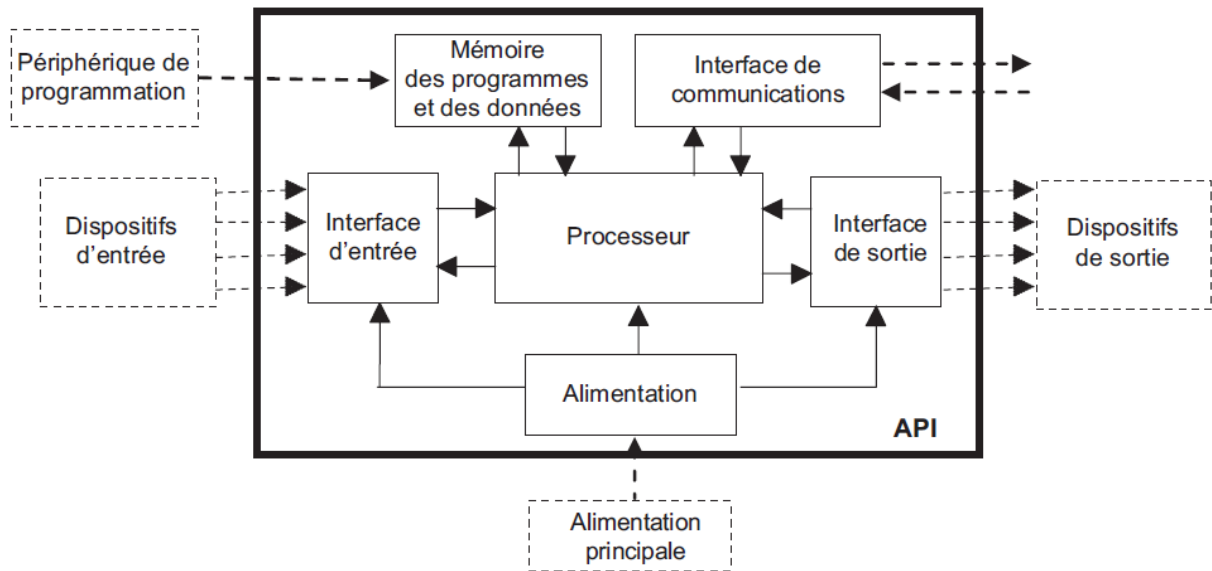


Figure 2.6: Structure interne d'un API

### 2.2.7. Critère de choix un API

Dans le marché actuel des API, il existe plusieurs marques et au sein de chaque marque on a une large gamme de modèles de différentes gammes de puissance.

Parmi les marques d'automates les plus utilisées, nous avons : Siemens, Allen Bradley, Schneider, Omron, Mitsubishi, Proface, Unitronics, General Electric etc...

Le choix d'un automate pour une application particulière dépend de plusieurs critères comme [8]:

- Nombres d'entrées/sorties intégrés.
- Temps de traitement (scrutation).
- Capacité de la mémoire.
- Nombre de compteurs.
- Nombre de temporisateurs

## 2.4 L'automate S7-300

La figure 2.7 représente l'automate programmable S7-300.



Figure 2.7: Automate S7-300

### 2.4.1. Caractéristiques :

- Mini-automate modulaire pour les applications d'entrée et de milieu gamme.
- Gamme diversité de CPU
- Possibilité d'extension jusqu'a 32 modules.
- Bus de fond de panier intègre aux modules.
- Liberté de montage aux différents emplacements.
- Possibilité de mise en réseau avec :
  - L'interface multipoint (MPI).
  - PROFIBUS.
  - Industrial Ethernet.

### 2.4.2. Les modules du S7-300

L'automate programmable S7-300 est subdivisé en plusieurs parties. La figure 8 montre les différents modules d'un Automate programmable S7-300.

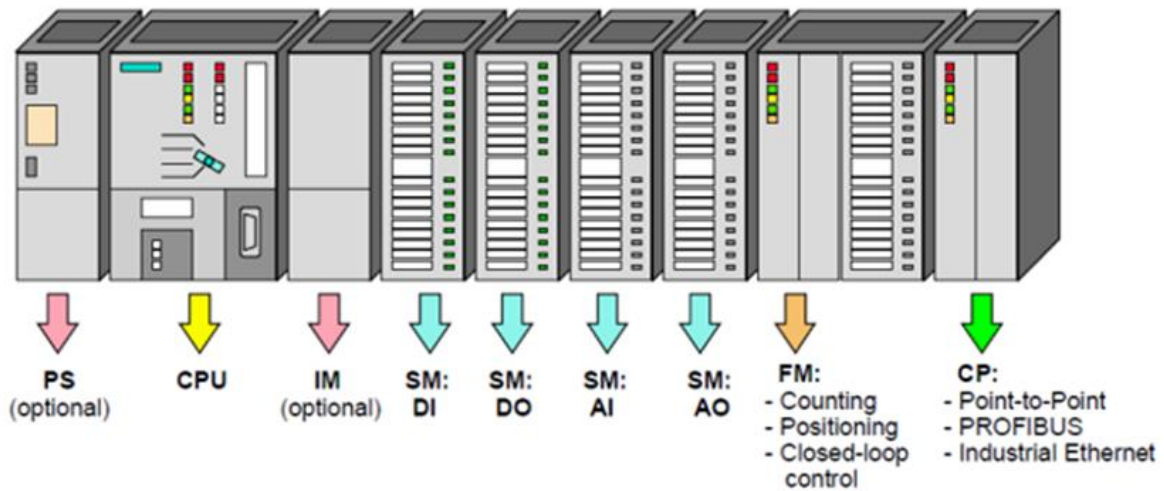


Figure 2.8: Les modules du S7-300

### 2.4.2.1. L'alimentation

Le module d'alimentation convertit la tension secteur 120V/230V en tension 24 VCC nécessaire pour l'alimentation de l'automate, Les modules prévus pour l'alimentation des CPU du S7-300 sont les suivants [10]:

Désignation	Courant de sortie	Tension à la sortie	Tension à l'entrée
PS 307-2A	2A	24VCC	120V/230V
PS 307-5A	5A	24VCC	120V/230V
PS 307-10A	10A	24VCC	120V/230V

Tableau 1: Les modules prévus pour l'alimentation des CPU de S7-300

### 2.4.2.2. L'unité centrale (CPU)

La gamme S7-300 offre une grande variété de CPU tels que la CPU312, 314M, 315, 315- P. Chacune d'elles possède certaines caractéristiques différentes des autres. Par

conséquent, le choix de la CPU, pour un problème d'automatisation donné, est conditionné par les caractéristiques offertes par la CPU choisie [11].

#### **2.4.2.3. Module des signaux (SM)**

Les modules de signaux (SM) servent d'interface entre le processus et l'automate. Il existe des modules d'entrées et des modules de sorties TOR, ainsi que des modules d'entrées et des modules de sorties analogiques [11].

#### **2.4.2.4. Module d'entrées /sorties TOR**

Les modules d'entrées/ sorties TOR constituent les interfaces d'entrée et de sortie pour les signaux tout ou rien de l'automate.

Ces modules permettent de raccorder à l'automate S7-300 des capteurs et des actionneurs tout ou rien les plus divers, en utilisant si, nécessaire des équipements d'adaptation (conditionnement, conversion, etc...) [11].

#### **2.4.2.5. Module d'entrée /sortie analogique**

Ces modules permettent de raccorder à l'automate des capteurs et actionneurs analogiques. Les modules d'entrées analogiques (SM 331) réalisent la conversion des signaux analogiques, issus de processus aux signaux numériques pour le traitement interne dans S7-300.

Les modules de sorties analogiques (SM 332) convertissent les signaux numériques interne (du S7-300) aux signaux analogiques destinés aux actionneurs ou pré-actionneurs analogiques.

Cependant, les modules d'entrées/ sorties analogiques (SM 334) réalisent les deux fonctions. Le choix des modules E/S dépend des critères suivants :

- Gamme de tension des capteurs utilisés.
- Tension et puissance d'alimentation des pré-actionneurs.
- Homogénéité des tensions entre les modules.



#### **2.4.2.6. Coupleur**

Ils ont pour rôle le raccordement du châssis d'extension au châssis de base. Pour la gamme S7-300, les coupleurs disponibles sont [11]:

- IM365 : pour les couplages entre les châssis distant d'un mètre au maximum,
- IM360/IM361 : pour les couplages allant jusqu'à 10 mètres de distance.

#### **2.4.2.7. Module de fonction (FM)**

Il a pour rôle, l'exécution de tâche de traitement des signaux du processus à temps critique et nécessitant une importante capacité de mémoire comme le comptage (FM 350), positionnement (FM 353/FM 357), et régulation (FM 355) [11].

#### **2.4.2.8. Module de communication (CP)**

Les processeurs de communication (CP) réalisent le couplage point-a-point que relie les partenaires de communication (automates programmable, scanner, PC, etc...)[11].

#### **2.4.2.9. Le châssis (RACK)**

Les châssis constituent des éléments mécaniques de base du SIMATIC S7-300, Ils remplissent les fonctions suivantes :

- Assemblage mécanique des modules
- Distribution de la tension d'alimentation des modules

### **2.4.3. Les langages de programmations**

La programmation peut s'effectuer de trois manières différentes :

- Sur l'API lui-même à l'aide de touches.
- Avec une console de programmation reliée par un câble spécifique à l'API.
- Avec un PC et un logiciel approprié.

Chaque automate dispose de son propre langage de programmation cependant il existe des langages universels à tous types d'automates. On distingue les langages graphiques et les langages textuels :

1. Les langages graphiques :

- ✓ LD : Ladder Diagram (Diagrammes échelle)
- ✓ FBD : Function Block Diagram (Logigrammes)
- ✓ SFC : Sequential Function Chart (Grafcet)

1. Les langages textuels :

- ✓ IL : Instruction List (Liste d'instructions).
- ✓ ST: Structured Text (Texte structuré).

Pour notre travail nous allons utiliser le langage LADDER qui utilise des symboles électriques qui assemblés forment le programme. Ce type de programmation à l'avantage de pouvoir être utilisé par du personnel électricien ou ayant une connaissance de la schématique électrique sans pour autant apprendre un langage spécifique.

## 2.5 Conclusion

Dans ce chapitre on a présenté d'une façon détaillée les caractéristiques de l'automate programmables industriel et son application dans l'industrie.

D'autres parts on a identifié les différents types d'automates programmables industriels qui permettent de commander un système automatisé.

Enfin, on a exposé en détail l'automate que nous allons utiliser (S7-300) et tous les modules accessoires et les langages de programmation du API.

# **Chapitre 3:**

## Logiciels utilisés

## 3.1 Introduction

L'automatisation des mécanismes industriels est réalisée avec des appareils (machines électroniques) spécialisés dans la conduite et la surveillance des processus industriels.

Ces appareils donnent des ordres de fonctionnement des procédés en exécutant une suite d'instructions appelée programme qu'on écrit par un langage de programmation pour une exécution cyclique.

Dans ce chapitre, nous donnons une présentation générale sur le logiciel de programmation des automates STEP7, ainsi que le logiciel WinCC flexible que nous allons utiliser pour la création d'une interface HMI pour la supervision.

## 3.2 STEP7

### 3.2.1 Définition

Le STEP7 est un logiciel de base pour la programmation et la configuration des systèmes d'automatisation SIMATIC. Il permet : la création et la gestion des projets, la configuration, le paramétrage du matériel et la communication, la gestion des mnémoniques et la création des programmes [12]. Il inclut six applications :

#### 3.2.1.1 Gestionnaire de projet SIMATIC Manager

Il gère toutes les données relatives à un projet d'automatisation. Il démarre automatiquement les applications requises pour le traitement des données sélectionnées.

#### 3.2.1.2 L'édition de mnémoniques

Les mnémoniques sont des noms symboliques qui vont être utilisés dans la programmation. L'utilisation de noms communs (mnémonique) est plus simple que la manipulation des adresses ou opérandes par exemple : on utilise la mnémonique «moteur» au lieu du bit de sortie A0.0.

L'éditeur mnémonique permet de gérer toutes les variables globales. C'est-à-dire la définition de désignations symboliques et les commentaires pour les signaux de processus (entrées/sorties).

### 3.2.1.3 La configuration du matériel

Il permet de configurer et paramétrer le matériel d'un projet. Il suffit juste de sélectionner le châssis (Rack), ensuite on peut ajouter tous les composants nécessaires come le CPU, les modules de fonctionnels (FM), le processeur de communication (CP)...etc.

### 3.2.1.4 Le diagnostic du matériel

Le diagnostic du matériel fournit un aperçu de l'état du système d'automatisation. Dans une représentation d'ensemble, un symbole permet de préciser pour chaque module, s'il est défaillant ou non.

Il permet aussi d'avoir des informations générales sur les modules, les causes des erreurs, comme on peut aussi détecter les causes des défaillances dans un programme.

### 3.2.1.5 La configuration de la communication NetPro

Le sous-logiciel NetPro permet de la création d'une vue graphique du réseau en question ainsi que les sous-réseaux qui le constituent tout en offrant la possibilité de choisir les participants à la communication et définir les liaisons de communication.

### 3.2.1.6 L'édition du programme

Il permet de l'Edition des programmes à travers trois langages inclus dans le logiciel de base :

**Le schéma à contact (CONT) :** langage graphique similaire aux schémas de circuit à relais ce qui permet de suivre facilement le trajet du courant.

**Liste d'instruction (LIST) :** langage textuel de bas niveau, à une instruction par ligne, similaire au langage assembleur.

**Le logigramme (LOG) :** langage de programmation graphique qui utilise les boîtes de l'algèbre de Boole pour représenter les opérations logiques.

L'éditeur de programme permet aussi la visualisation et forçage de variables.

### 3.2.2 Création d'un projet STEP 7

Un projet comprend deux types de données essentielles, les programmes et la configuration du matériel. On peut commencer par définir l'un ou l'autre. En premier lieu, il faut tout d'abord démarrer le programme SIMATIC Manager. Ce dernier est l'interface graphique qui permet la manipulation du projet et l'accès aux autres programmes de STEP 7 [13]. La figure 3.1 représente l'icône du STEP 7.



Figure 3.1: Icône du STEP7

Pour créer un nouveau projet, il suffit de cliquer sur le bouton 'Nouveau projet', définir un nom et le valider (Figure 3.2). Ensuite, on choisit une station de travail.



Figure 3.2: Création d'un projet STEP 7

Une station SIMATIC représente une configuration matérielle comportant un ou plusieurs modules programmables. Il en existe plusieurs types tels que SIMATIC 400, SIMATIC 300, SIMATIC S7...etc. La figure 3 montre comment choisir une station SIMATIC par exemple SIMATIC 300.

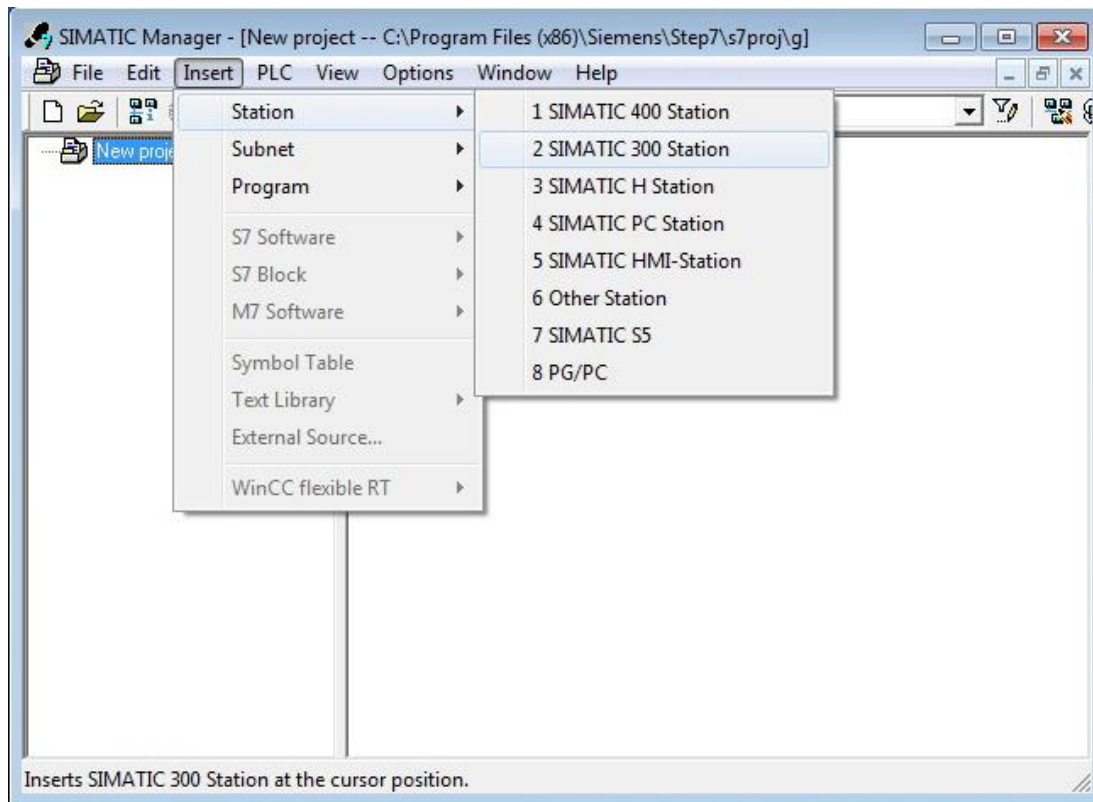


Figure 3.3: Choix de la station SIMATIC 300

Avant de commencer la programmation, nous devons tout d'abord configurer le matériel. Cela est fait en double cliquant sur «*Hardware*», ce qui ouvre l'application *HW Config*.

### 3.2.3 Configuration matérielle

Pour la configuration du matériel, il suffit de glisser des éléments du catalogue dans l'emplacement approprié. On choisit premièrement le châssis (Rack), l'alimentation, le CPU et les Entrées/Sorties.

Dans le catalogue, on trouve les modules qu'on peut ajouter ou affecter à chaque type de station, on distingue principalement [13] :

- C7 : système intégré pour établir la liaison entre l'API et l'interface HMI,
- Module de communication (CP),
- Module de fonction (FM),
- Coupleur (IM),
- Module d'alimentation (PS),
- Module des signaux (SM),
- CPU,
- Modules d'entrées/sorties,

Les figures 3.4 et 3.5 représentent l'étape de configuration et de la sélection des modules.



Figure 3.4: Configuration hardware pour la station S300



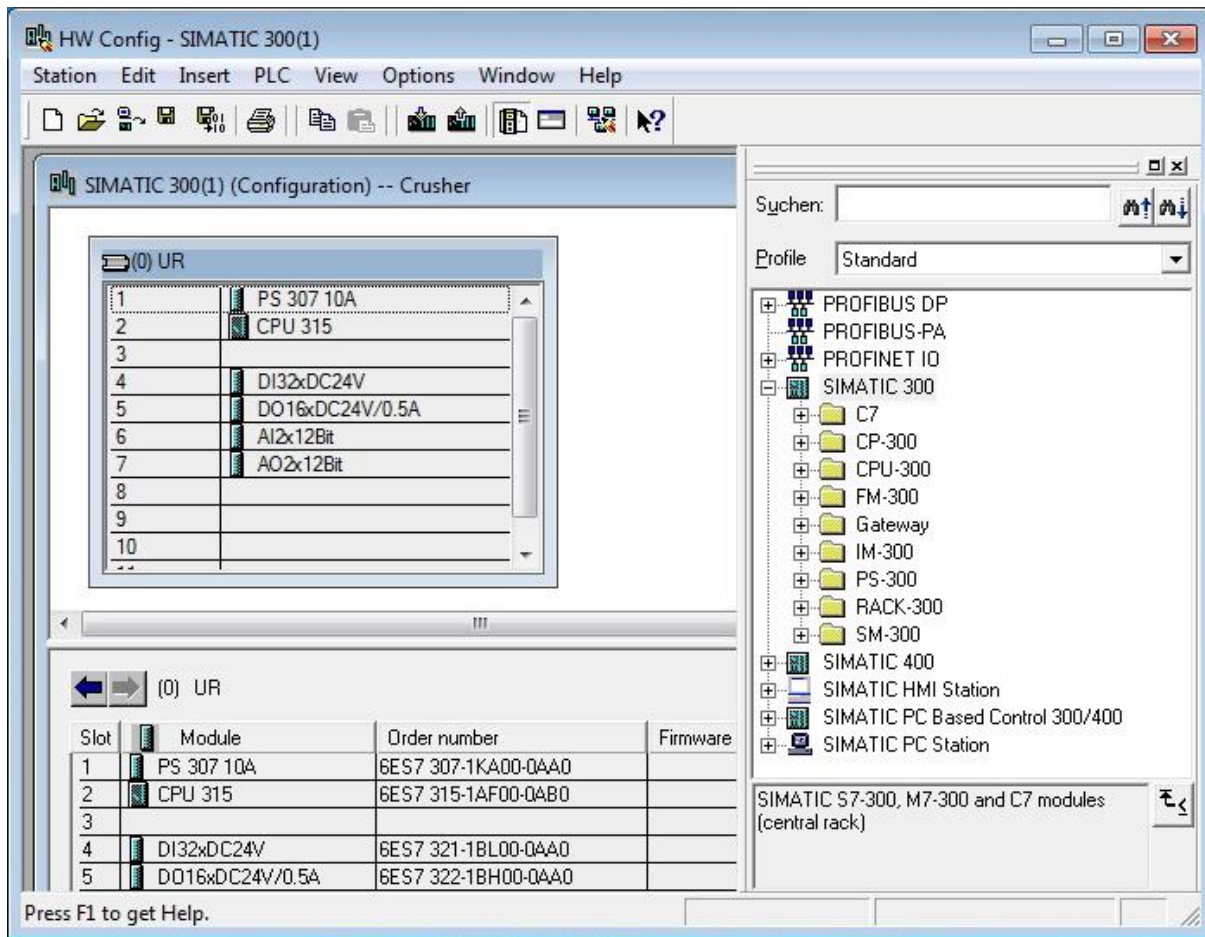
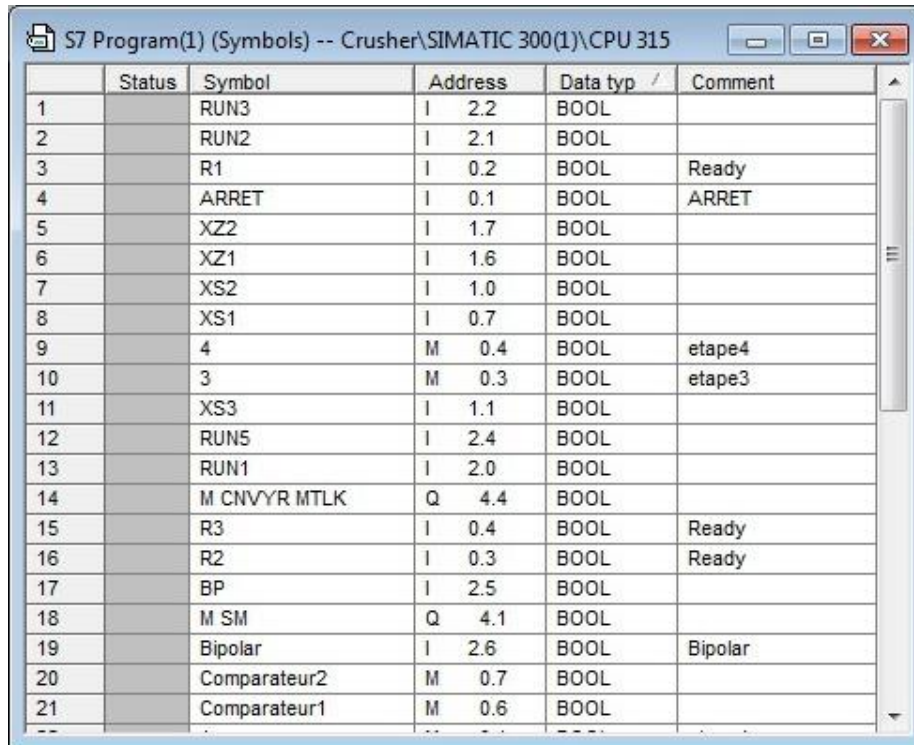


Figure 3.5: Sélection des modules

### 3.2.4 Définitions des mnémoniques

De retour dans le SIMATIC Manager, on trouve de nouveau éléments. On commence par créer les mnémoniques dans la section programmes. Pour créer ce tableau, on clique sur «Insert new object» et ensuite sur «Symbol table». La figure 6 illustre la table des mnémoniques.



	Status	Symbol	Address	Data typ /	Comment
1		RUN3	I 2.2	BOOL	
2		RUN2	I 2.1	BOOL	
3		R1	I 0.2	BOOL	Ready
4		ARRET	I 0.1	BOOL	ARRET
5		XZ2	I 1.7	BOOL	
6		XZ1	I 1.6	BOOL	
7		XS2	I 1.0	BOOL	
8		XS1	I 0.7	BOOL	
9		4	M 0.4	BOOL	etape4
10		3	M 0.3	BOOL	etape3
11		XS3	I 1.1	BOOL	
12		RUN5	I 2.4	BOOL	
13		RUN1	I 2.0	BOOL	
14		M CNVYR MTLK	Q 4.4	BOOL	
15		R3	I 0.4	BOOL	Ready
16		R2	I 0.3	BOOL	Ready
17		BP	I 2.5	BOOL	
18		M SM	Q 4.1	BOOL	
19		Bipolar	I 2.6	BOOL	Bipolar
20		Compareur2	M 0.7	BOOL	
21		Compareur1	M 0.6	BOOL	

Figure 3.6: Table des mnémoniques

### 3.2.5 Edition des programmes

Dans la section «*Bloc*» du SIMATIC Manager, on trouve par défaut le bloc d'organisation «OB1» qui représente le programme cyclique. On peut rajouter d'autres blocs à tout moment par une clique du bouton droit dans la section *Bloc* (Figure 3.7).

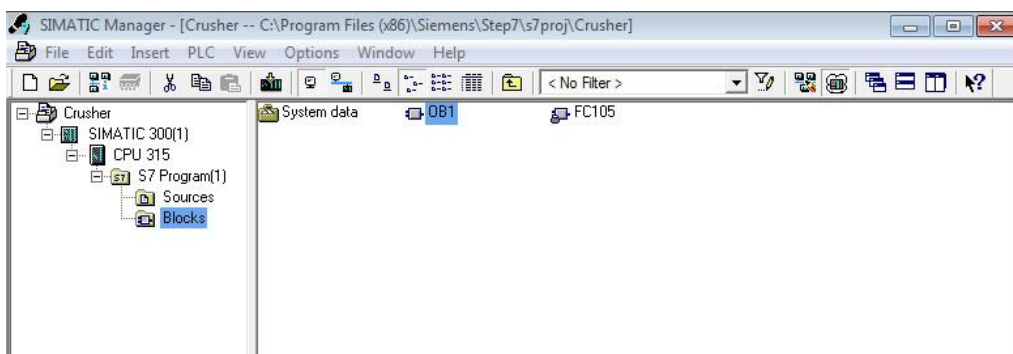


Figure 3.7: Edition des programmes

Le logiciel STEP7 permet de structurer le programme utilisateur en le subdivisant en différentes parties autonomes ou dépendantes. Cela permet d'écrire des programmes bien organisés, simples à tester et à modifier.

### 3.2.5.1 Blocs d'organisation

Les modules d'organisation (OB) sont, comme leur nom l'indique, utilisés pour l'organisation interne du programme et forment ainsi un moyen puissant et essentiel pour la programmation structurée. L'OB1 est examiné à chaque cycle d'automate. C'est donc à partir de ce bloc que l'on fera les appels aux différents blocs de programmes [12].

L'OB100 et l'OB101 sont uniquement appelés au démarrage (respectivement à chaud et à froid). On y appellera donc les blocs traitant les initialisations. La figure 3.8 montre un exemple d'un programme dans l'OB1.

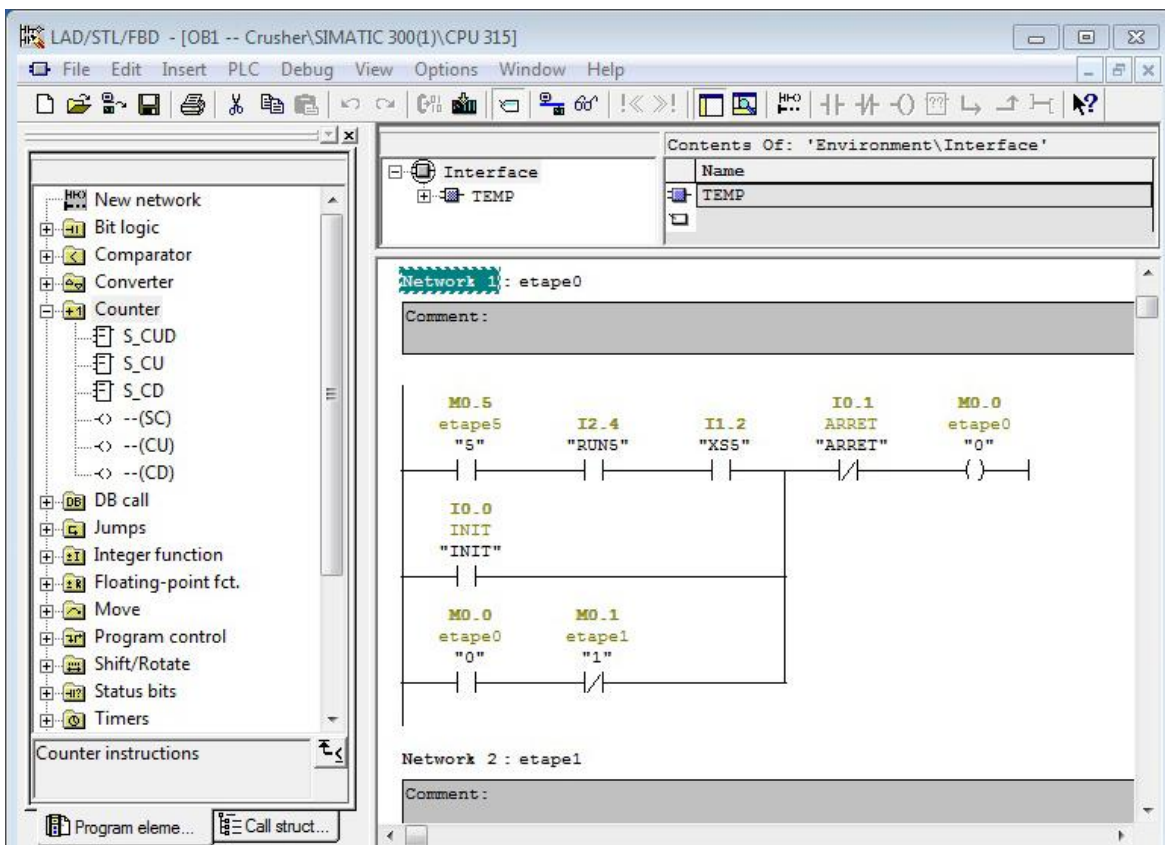


Figure 3.8: Exemple d'un programme dans l'OB1

### 3.2.5.2 Blocs fonctionnels

Les modules fonctionnels (*FB*) sont librement paramétrables et spécialement conçus pour la standardisation de fonctions complexes. C'est dans ces blocs que l'on met les instructions à exécuter. La numérotation est libre (de 0 à 255). Ces blocs n'ont pas de mémoire.

### 3.2.5.3 Bloc de données

Les blocs de données (*DB*) servent à l'enregistrement de données utilisateur. Les blocs de données globaux servent à l'enregistrement des données qui peuvent être utilisées par tous les autres blocs. Les blocs de données d'instances sont affectés à des blocs fonctionnels.

### 3.2.6 Simulation de module

Pour la simulation des programmes, on utilise le simulateur S7-PLCSIM ce qui permet d'exécuter et de tester le programme utilisateur destinés à la CPU l'automate. Le S7-PLCSIM dispose d'une interface simple permettant de visualiser et de forcer les différents paramètres utilisés par le programme comme par exemple : activer ou de désactiver des entrées [14]. La figure 3.9 représente l'interface du simulateur S7-PLCSIM.

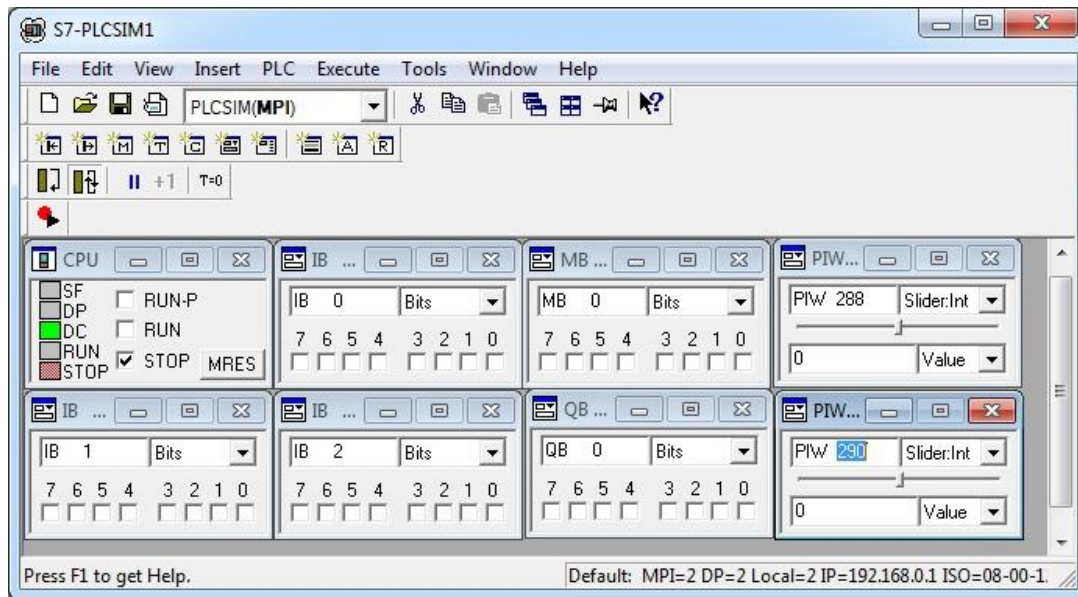


Figure 3.10: Interface du simulateur S7-PLCSIM

Le S7-PLCSIM possède les fonctions suivantes [14] :

- On peut créer des "fenêtres" dans lesquelles on a la possibilité d'accéder aux zones de mémoire d'entrée et de sortie.
- On peut sélectionner l'exécution automatique des temporisations ou encore les définir et les réinitialiser manuellement.
- On a la possibilité de changer l'état de fonctionnement de la CPU (STOP, RUN et RUNP) comme pour un CPU réel.

### 3.3 WinCC flexible

#### 3.3.1 Définition

Le WinCC (Windows Control Center) de SIEMENS, est un logiciel d'ingénierie pour la configuration de pupitres SIMATIC, de PC industriels SIMATIC et de PC standard avec le logiciel de visualisation WinCC Runtime.

Par la programmation, on peut effectuer les tâches de supervision concernant la surveillance de l'opérateur, de fabrication et le contrôle de l'automate de la production. Il

fournit des unités fonctionnelles appropriées à l'industrie pour la représentation graphique, les alarmes, l'archivage et le protocole [15].

### 3.3.2 Généralités sur la supervision

#### 3.3.2.1 Définition de la supervision

La supervision est une technique industrielle de suivi et de pilotage informatique des procédés de fabrication à système automatisés. La supervision est une forme évoluée de dialogue *Homme-Machine*. Elle consiste à surveiller l'état de fonctionnement d'un procédé et des paramètres de commande des processus généralement communiqués à des automates programmables. En informatique, la supervision est la surveillance du bon fonctionnement d'un système ou d'une activité [16]. La figure 3.10 illustre un exemple de système supervision.

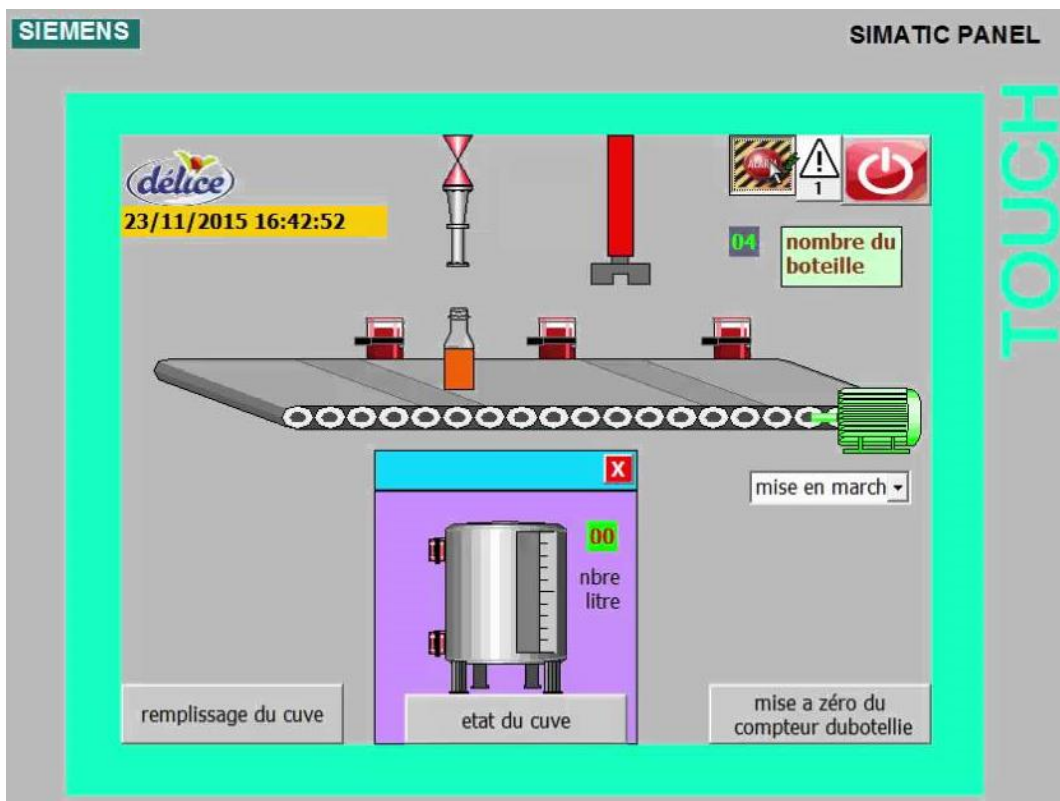


Figure 3.11: Exemple de système supervision

### 3.3.2.2 Système HMI

Un système *HMI* constitue l'interface entre l'homme (opérateur) et le processus (machine/installation). L'HMI est généralement une interface graphique qui affiche l'état actuel d'un processus, tel que l'état d'un moteur, l'état d'une vanne, les températures, les pressions et d'autres variables système. Le rôle de l'interface *Homme-Machine* est de fournir aux opérateurs, aux personnels de maintenance et à la direction les outils nécessaires pour accéder, contrôler et afficher les paramètres d'un processus [17].

### 3.3.2.3 Les avantages de la supervision

Un système de supervision facilite à l'opérateur la conduite du processus. Son but est de présenter à l'opérateur des résultats expliqués et interprétés. Parmi les avantages principaux d'un système de supervision, on peut citer [16]:

- Surveiller le processus à distance.
- Interaction en temps réel.
- La détection des défauts.
- Le diagnostic et le traitement des alarmes.
- Traitement des données.

Dans notre application, nous avons utilisé le pupitre et le logiciel WinCC flexible de SIEMENS pour la supervision de notre séquence.

### 3.3.3 Les éléments de WinCC flexible

Le WinCC flexible comprend les éléments suivants :

- **WinCC flexible Engineering System** : est le logiciel avec lequel on réalise toutes les tâches de configuration requises. L'édition WinCC flexible détermine les pupitres opérateurs de la gamme SIMATIC HMI pouvant être configurés.
- **WinCC flexible Runtime** : est le logiciel de visualisation de *Process*. Dans le *Runtime*, on exécute le projet en mode *Process*.

- **Options WinCC flexible** : Les options WinCC flexible permettent d'étendre les fonctionnalités de base du WinCC flexible. Chaque option nécessite une licence particulière.

### 3.3.3.1 WinCC flexible Engineering System

Le WinCC flexible est le système d'ingénierie pour toutes les tâches de configuration. Il est un logiciel modulaire. Chaque incrément d'édition élargit l'éventail des appareils cibles et fonctionnalités pris en charge. Lorsqu'on crée ou on ouvre un projet sous WinCC flexible, l'écran de l'ordinateur de configuration affiche WinCC flexible *Workspace*. La fenêtre de projet affiche alors la structure du projet et permet ainsi de le gérer. La figure 3.12 montre la fenêtre principale du WinCC flexible [15].

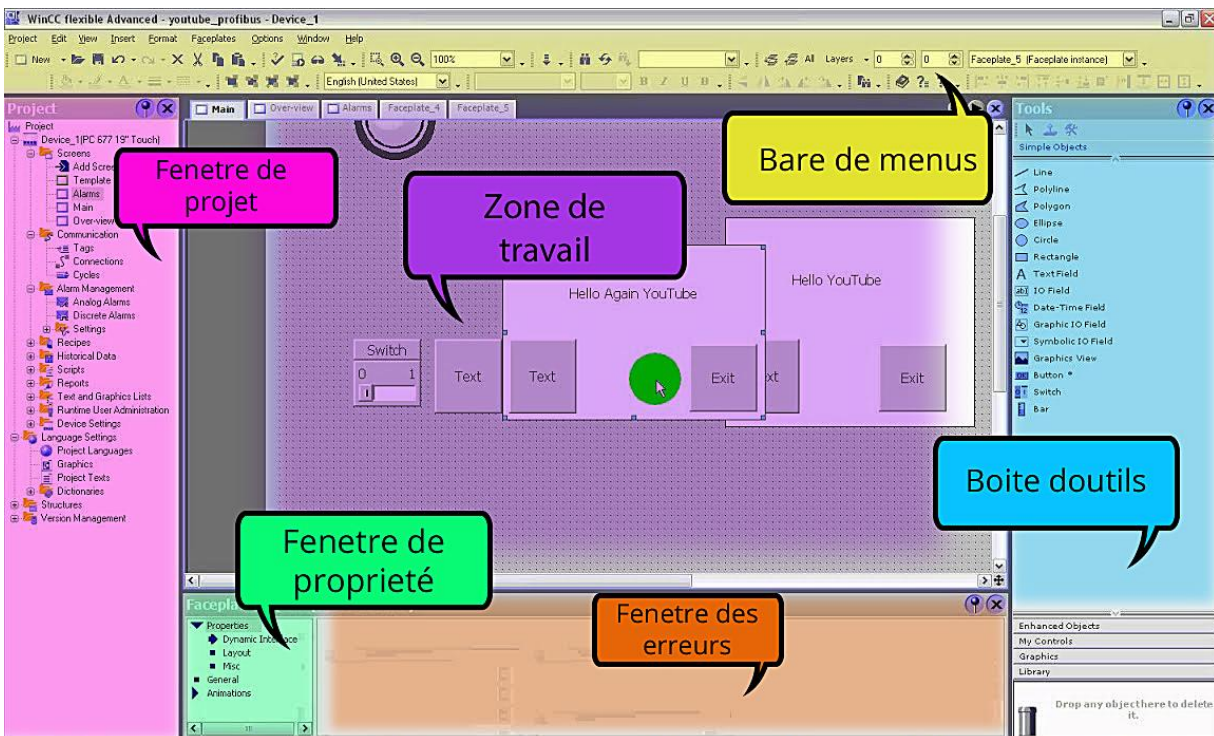


Figure 3.13: La fenêtre principale du WinCC flexible

Le WinCC flexible se compose des éléments suivants [15]:

- **Fenêtre de projet** : Tous les éléments et tous les éditeurs disponibles d'un projet sont affichés sous forme d'arborescence dans la fenêtre du projet et peuvent être ouverts à partir de cette fenêtre.



- **Barre des menus** : La barre des menus contient toutes les commandes nécessaires à l'utilisation de WinCC flexible. Les raccourcis disponibles sont indiqués en regard de la commande du menu.
- **Boîte d'outils** : La boîte d'outils propose une liste d'objets qu'on peut insérer dans les vues. Par exemple : les objets graphiques et les éléments de commande.
- **Zone de travail** : La zone de travail sert à éditer les objets du projet. Tous les éléments de WinCC flexible sont disposés autour de la zone de travail.
- **Fenêtre des propriétés** : Elle permet d'éditer les propriétés des objets, telle que la couleur des objets de vue. Elle n'est disponible que dans certains éditeurs.
- **Fenêtre des erreurs et avertissements** : La fenêtre des erreurs et avertissements affiche les alarmes système générées telle que l'alarme du test d'un projet.

### 3.3.3.2 WinCC flexible Runtime

Dans *Runtime*, l'opérateur peut réaliser le contrôle-commande du processus. Les tâches suivantes sont alors exécutées [15]:

- Communication avec les automates.
- Affichage des vues à l'écran.
- Commande du processus, par exemple spécification de consignes ou ouverture et fermeture de vannes.
- Archivage des données de *Runtime* actuelles, des valeurs processus et des événements d'alarme par exemple.

### 3.3.3.3 Options WinCC flexible

Des options sont disponibles pour les composants suivants [15] :

- WinCC flexible Engineering Système.
- WinCC flexible Runtime sur des pupitres opérateur basés sur PC.
- Pupitres opérateurs non basés sur PC.

### 3.3.3.4 Le projet de WinCC

Le projet est à la base de la configuration d'interface graphique. On crée puis on configure dans le projet tous les objets indispensables à la commande et au contrôle de la station. Dans notre cas, les objets nécessaires sont :

- Les vues pour représenter et commander la station.
- Les variables qui transmettent les données entre la station et le pupitre opérateur.
- Les alarmes qui affichent les états du fonctionnement de la station.
- Les boutons et les outils de commande ou d'entrée pour manipuler le système.

### 3.3.4 Intégration de WinCC dans STEP7

Lors de la configuration intégrée, on peut accéder aux données de configuration qu'on crée lors de la configuration de l'automate avec STEP 7. Ce qui offre les avantages suivants [15]:

- On peut utiliser le gestionnaire SIMATIC Manager comme un poste central de création, d'édition et de gestion des automates SIMATIC et des projets WinCC flexible.
- Les paramètres de communication de l'automate sont choisis par défaut lors de la création du projet WinCC flexible. Toute modification sous STEP 7 se traduit par une mise à jour des paramètres de communication sous WinCC flexible (figure 3.12).
- Lors de la configuration des variables et des pointeurs de zones, on peut accéder sous WinCC flexible directement aux mnémoniques de STEP 7. La simple sélection du mnémonique STEP 7 permet de lui affecter une variable. Les modifications des mnémoniques sous STEP 7 sont mises à jour sous le WinCC flexible.
- La définition des mnémoniques une seule fois sous STEP 7 permet de les utiliser sous STEP 7 et sous WinCC flexible.
- Les alarmes ALARM\_S et ALARM\_D configurées sous STEP 7 sont prises en charge sous WinCC flexible et peuvent être affichées sur le pupitre opérateur.

- On peut créer un projet WinCC flexible sans intégration dans STEP 7 et de l'intégrer ultérieurement dans STEP 7.
- Inversement, un projet intégré peut être désolidarisé de STEP 7 et être utilisé de façon autonome.
- Dans un multi projet STEP 7, on peut configurer des liaisons de communication sur plusieurs projets.



Figure 3.14: Paramétrage de la connexion

### 3.3.4.1 Conditions requises pour l'installation

Pour pouvoir intégrer WinCC flexible dans STEP 7, il y a lieu de tenir en compte de l'ordre d'installation. On installe tous d'abord le logiciel STEP 7 et ensuite WinCC flexible. La routine d'installation de WinCC flexible détecte la présence de STEP 7 et installe automatiquement le support pour l'intégration dans STEP 7 [15].

### 3.3.4.2 Utilisation du SIMATIC Manager pour manipuler un projet WinCC

Dans les projets intégrés, le SIMATIC Manager offre les possibilités suivantes [15]:

- Créer une station HMI ou PC avec WinCC flexible Runtime.
- Insertion d'objets WinCC flexible.
- Création de dossiers WinCC flexible.
- Ouverture de projets WinCC flexible.

- Génération et transfert de projets WinCC flexible.
- Exportation et importation de textes pour la traduction.
- Sélection de la langue.
- Duplication et déplacement de projets WinCC flexible.
- Archivage et désarchivage de projets WinCC flexible dans le cadre de projets STEP 7.

#### **3.3.4.3 Configuration de liaisons**

Des liaisons de communication sont nécessaires pour que WinCC flexible puisse échanger des données avec le niveau automatisation. Dans les projets intégrés, les liaisons sont créées avec :

- WinCC flexible,
- NetPro.

#### **3.3.5 Exécution de la simulation du système de supervision**

Le logiciel de configuration WinCC est fourni avec un simulateur qui est le WinCC Runtime Professional. Le WinCC Runtime est le logiciel qui permet de visualiser les processus. Ce logiciel est optionnel et permet de tester le projet et de contrôler le bon fonctionnement des vues et les messages d'alarmes configurés [12].

##### **3.3.5.1 Compilation et Simulation**

Après avoir créé le projet et effectuée la configuration, et s'il n'y a pas d'erreurs dans la cohérence du programme, la compilation permet de créer un fichier du projet complet. L'implantation sur l'automate virtuel et de le simuler avec le simulateur «*Runtime* » permet de détecter les erreurs logiques de la configuration et afficher le résultat à l'interface graphique [12].

### **3.4 Conclusion**

La description précédente de l'outil de programmation STEP7 et l'outil de supervision WinCC flexible, et les étapes de création et configuration des projets nous prépare le terrain pour la mise en œuvre de notre application sur la séquence de concassage et du transport de la matière ajout. C'est ce que nous allons voir dans le chapitre suivant.

# **Chapitre 4:**

## Application

## 4.1 Introduction

L'application de notre étude consiste de programmer la séquence de concassage et le transport de la matière ajout. On commence par une description générale de la séquence, ensuite on présente le cahier de charges qui décrit le processus de fonctionnement. Le cahier des charges sera traduit en un programme en langage LADDER en utilisant le logiciel STEP7. La supervision du processus est effectuée par une interface graphique HMI qu'on a réalisée spécialement avec le logiciel WinCC.

## 4.2 Les étapes de développement de notre travail

Pour le développement du système étudié, on a suivi le plan décrit par la figure 4.1.

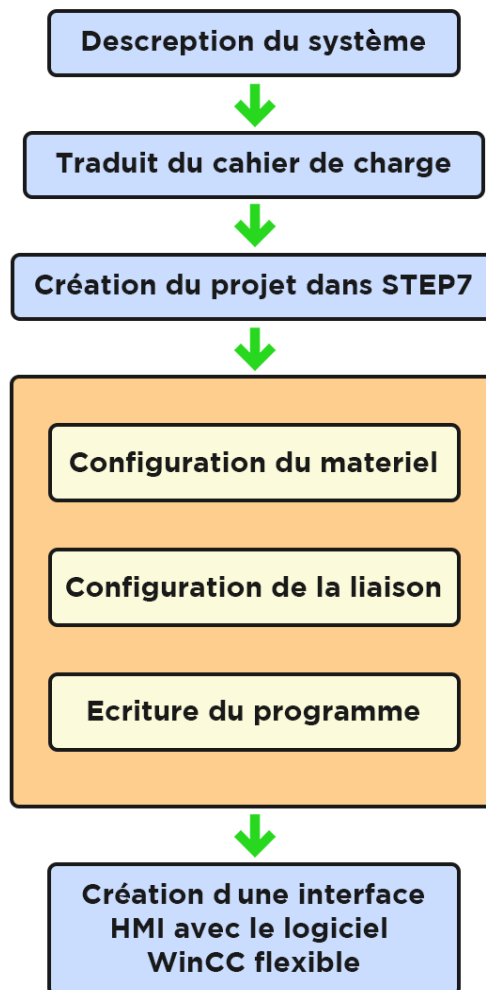


Figure 4.1: Organigramme de l'application

## **4.3 Séquence de concassage et transport de matière ajout**

### **4.3.1 Description**

La séquence de concassage et transport de matières ajoutés du ciment (calcaire principalement) représente la première étape dans le processus de fabrication du ciment.

Ce processus se produit dans la première zone de la cimenterie (zone d'extraction) la séquence de concassage et transport de matières ajoutés se compose de cinq parties principales :

- Le concasseur
- Transporteur tablier métallique
- Transporteur à bande
- Convoyeur du stacker
- Séparateur magnétique

### **4.3.2 Les éléments principaux de la séquence**

#### **4.3.2.1 Le concasseur**

Un concasseur est une machine conçue pour réduire les grosses roches en petites pierres, gravier, ou poussière de roche. Les concasseurs peuvent être utilisés pour réduire la taille ou changer la forme des déchets, afin ; qu'ils puissent être plus facilement éliminés ou recyclés [18].





Figure 4.2: Le concasseur

#### 4.3.2.2 Les convoyeurs

**Transporteur à tablier métallique:** Ces convoyeurs servent à l'alimentation des concasseurs en matière tout venant. Ils sont généralement inclinés de  $20^\circ$  ou  $23^\circ$  par rapport à l'horizontale, alors que les parois latérales de la trémie ont un angle d'inclinaison de  $20^\circ$  en général. La figure 3 un modèle de transporteur [18].



Figure 4.3: Transporteur à tablier métallique

#### 4.3.2.3 Le Séparateur magnétique

Le séparateur magnétique spécial sert d'une part à capturer les particules de métaux, d'autre part à protéger les équipements en aval (broyeuses, presses, convoyeurs à vis sans fin, etc.) contre l'endommagement par les objets ferromagnétiques (pièces de machines

arrachées, vis, écrous, fils, etc.). Les impuretés ferromagnétiques séparées du matériau sur le convoyeur sont transférées, par la bande de séparation, dans le but de les éliminées.



Figure 4.4: Séparateur magnétique

### 4.3.3 Les capteurs

#### 4.3.3.1 Capteur de rotation

Les capteurs de vitesse de rotation inductifs MSZ214 sont des transducteurs magnétiques électriques, ce capteur à un diamètre externe de 14 mm Les options de ce capteur sont [19] :

- Le circuit électronique de traitement des signaux est intégré dans le boîtier du capteur.
- Le circuit électronique est protégé contre la polarisation erronée de la tension d'alimentation.
- La sortie fourni la fréquence digitale correspondante aux nombres de dents.
- Livrables avec câble de raccordement (longueur 2 m) ou avec connecteur.

### 4.3.3.2 Les détecteurs de position

Ils sont des capteurs de position et sont surtout employés dans les systèmes automatisés pour assurer la fonction détecter les positions. Les capteurs de position sont des capteurs de contact. Ils peuvent être équipés d'un galet, d'une tige souple, d'une bille [20]. Ses applications principales:

- La détection de pièces machines (cames, butées, pignons)
- La détection de balancelles, chariots, wagons
- La détection directe d'objets

Dans la séquence de concassage et transport de la matière ajout, les détecteurs de position sont utilisés pour la détection de déviation des convoyeurs.

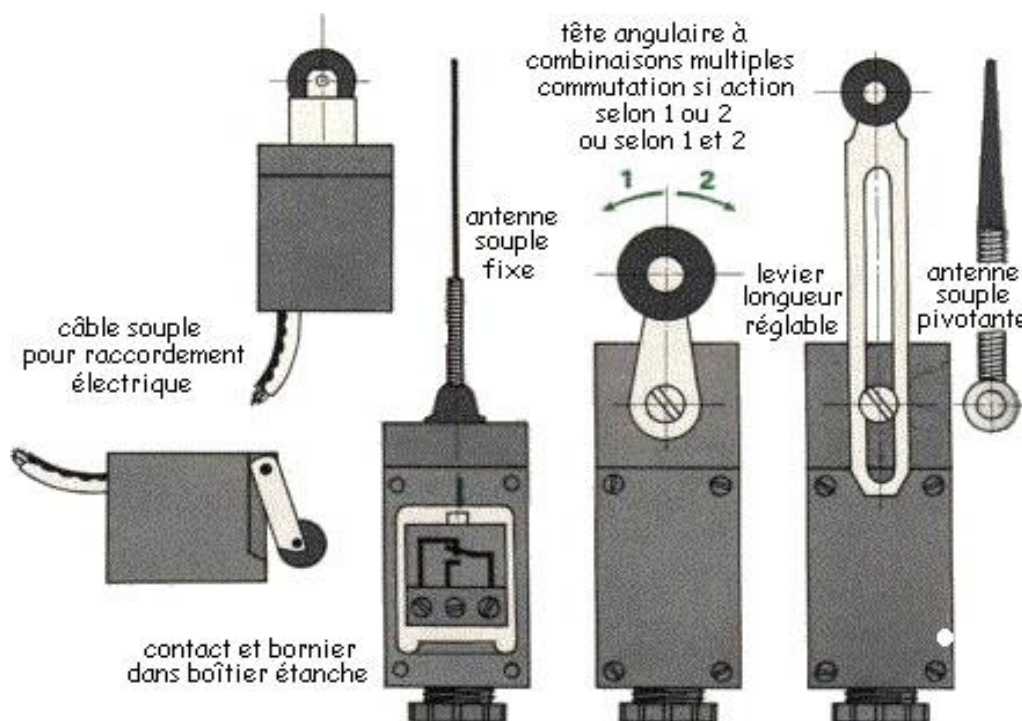


Figure 4.5: Les détecteurs de position

### 4.3.3.3 Capteur de vibration

Les capteurs de vibrations sont utilisés dans l'industrie pour la détection et la mesure des vibrations des machines. Avec un capteur de vibration, on mesure l'accélération des

oscillations, les impulsions de chocs. Dans la séquence de concassage et transport de la matière ajout, ils sont utilisés pour la détection de vibration de concasseur.



Figure 4.6: Des capteurs de vibrations

## 4.3.4 Les actionneurs

### 4.3.4.1 Moteur asynchrone

Le moteur asynchrone triphasé est largement utilisé dans l'industrie grâce à sa simplicité de construction (figure 4.7). En fait, c'est un moteur 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 bobinée, à cage d'écureuil. Les circuits magnétiques du rotor et du stator sont constitués d'un empilage de fines tôles métalliques pour éviter la circulation de courants de Foucault [5].



Figure 4.7: Moteur asynchrone triphasé

### 4.3.5 Les pré-actionneurs

Les pré-actionneurs font partie de la chaîne d'action d'un système automatisé. Un pré-actionneur permet de distribuer de l'énergie à un actionneur. Ce dernier convertit l'énergie reçue en une énergie utile [5].

#### 4.3.5.1 Contacteurs

Le contacteur est un relais électromagnétique qui permet grâce à des contacts (pôles) de puissance d'assurer le fonctionnement de moteurs, de résistances ou d'autres récepteurs de fortes puissances [5].



Figure 4.8: Contacteur

## 4.4 Principe de fonctionnement

La figure 4.9 représente la séquence de concassage et de transport de la matière ajout.

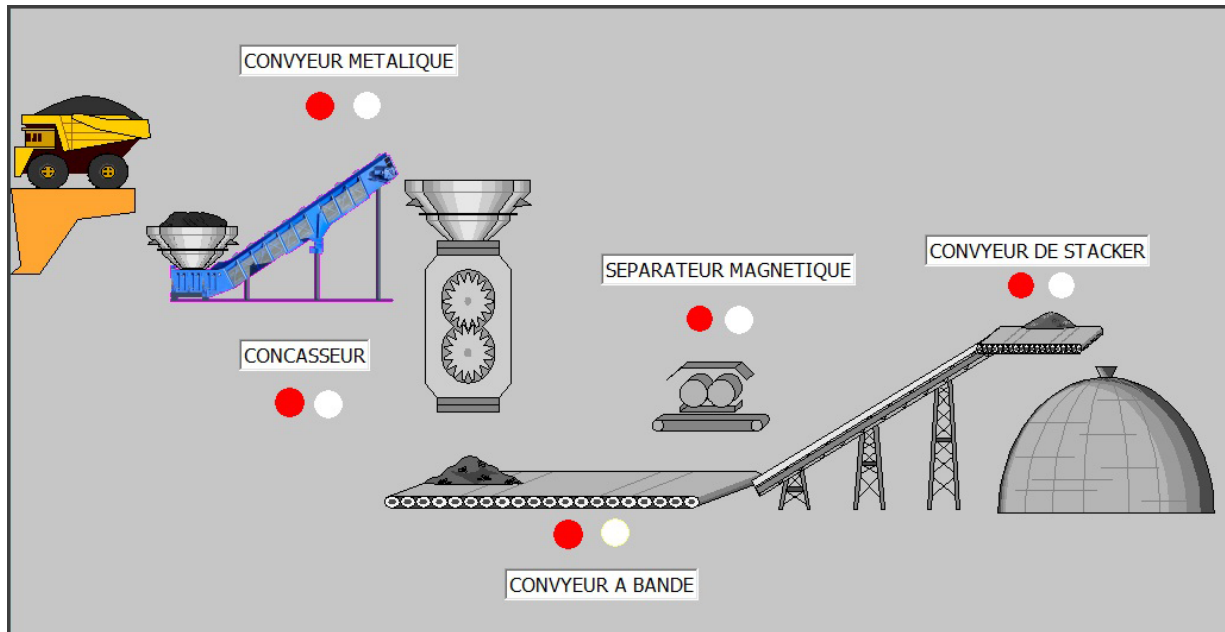


Figure 4.9: La séquence de concassage et de transport de la matière ajout

### 4.4.1 Cahier de charge

Le principe de fonctionnement de la séquence de concassage et de transport de la matière ajout est simple. Le principe de base est : lorsque la matière première est nécessaire, la chaîne est exécutée. La séquence de concassage et de transport se compose de cinq phases de base :

1. Transporteur à tablier métallique (convoyeur métallique),
2. Le concasseur,
3. Transporteur à bande,
4. Séparateur magnétique,
5. Convoyeur du Stacker.

Ces phases doivent être exécutées en ordre inverse c à d à partir de la fin de séquence (convoyeur du stacker) jusqu'au début (convoyeur métallique), et non pas en ordre direct.

Notre travail peut se résumer dans les étapes suivantes :

- Démarrage de moteur de convoyeur du Stacker,
- Démarrage de Séparateur magnétique,
- Démarrage de moteur de convoyeur à bande,
- Démarrage de moteur de concasseur,
- Démarrage de moteur de convoyeur métallique.

#### 4.4.2 Grafctet du cahier de charge

La figure 4.10 représente le GRAFCET du cahier de charges.

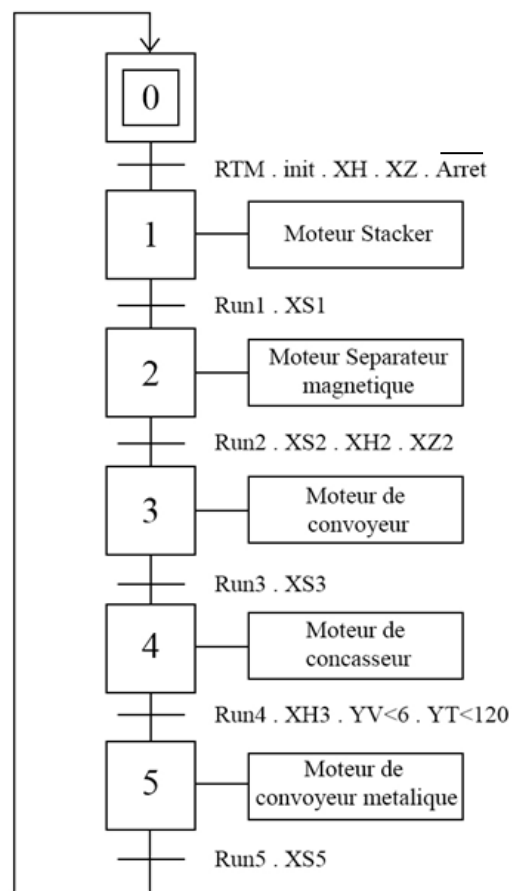


Figure 4.10: Grafctet du cahier de charge

Tableau 4.1: Signification des symboles de GRAFCET

Etape	Description
RTM	Ready tous les moteurs
Init	Initialisation
Arrêt	Bouton d'arrêt
XH 1, 2, 3	Les arrêts à câbles
XZ 1, 2, 3	Les capteurs de déviation de convoyeur
XS 1, 2, 3, 5	Les capteurs de rotation
Run 1, 2, 3, 4, 5	Feedback de chaque Moteur
YV<6 mm/s	Condition de vibration
YT<120 °C	Condition de température

## 4.5 Programmation sur STEP7

Un projet STEP7 est créé selon la procédure vue dans le chapitre 3. La création d'un projet sur STEP7 commence tout d'abord par définir une configuration matérielle. L'intérêt de cette dernière est de spécifier les modules d'alimentation, d'entrées et de sorties du projet. Cela est nécessaire pour pouvoir le relier avec la CPU correspondante au niveau de l'automate.

### 4.5.1 Configuration du matériel dans STEP7

On a choisit la configuration suivante pour notre travail:

- Station SIMATIC S7-300
- Module d'alimentation PS 307 10A
- Module de communication MPI
- CPU 315
- Entrée analogique AI4x/4...20mA
- Sortie analogique AO4x/4...20mA
- Entrée numérique DI16xDC24V
- Sortie numérique DO16xDC24V



La figure 4.11 représente la configuration de ce matériel.

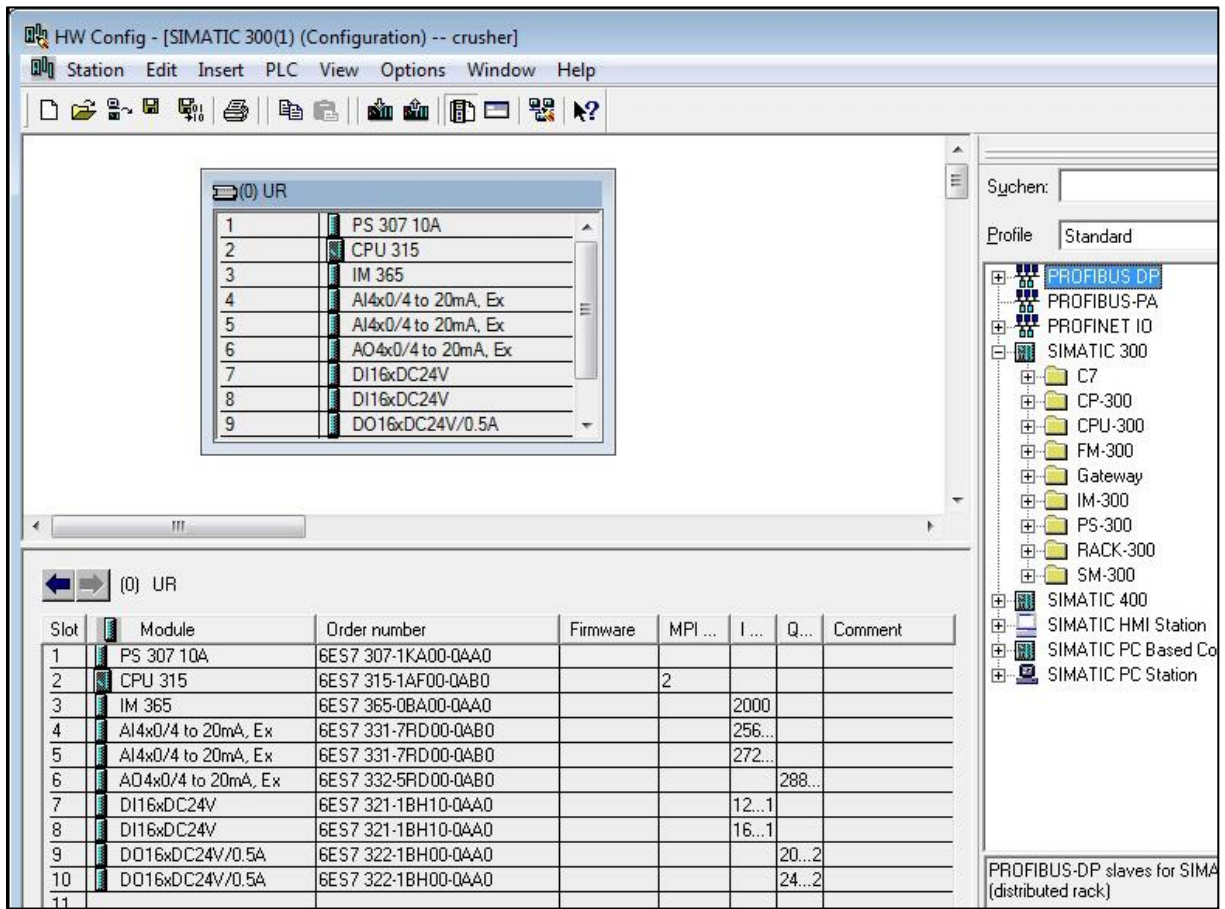
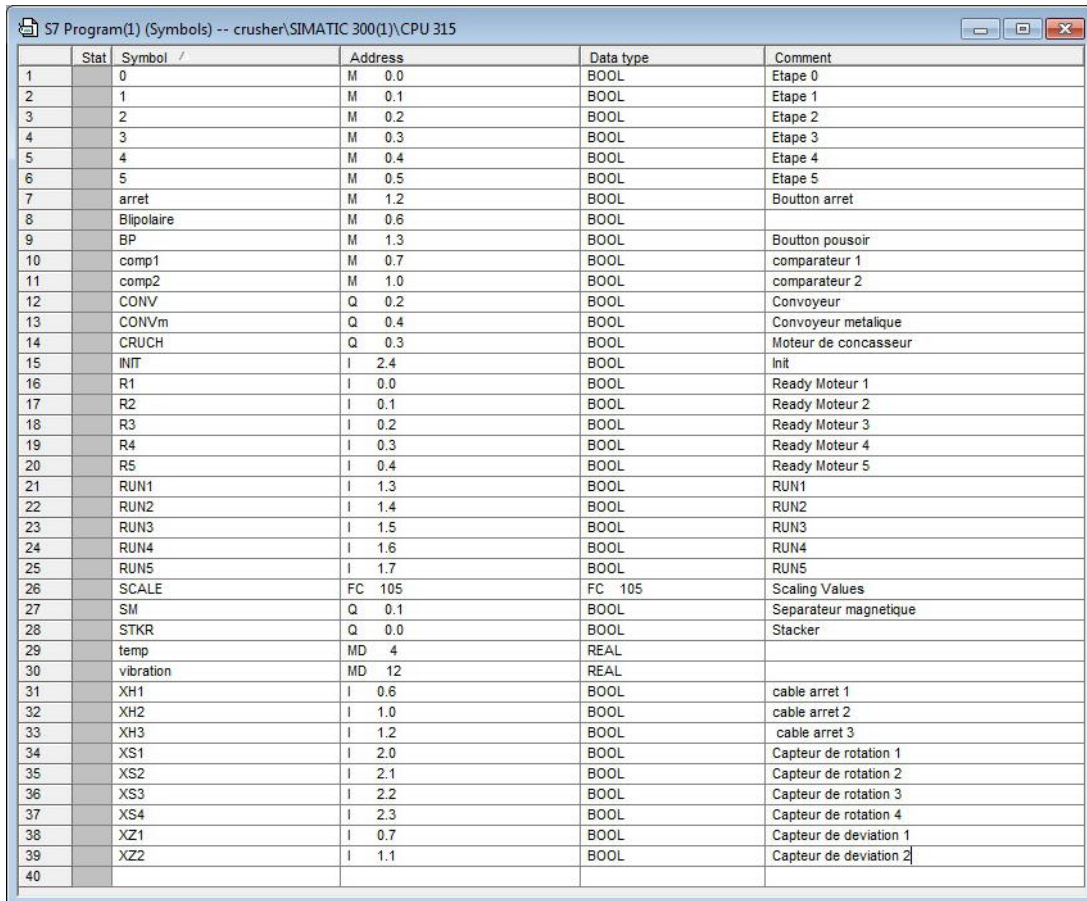


Figure 4.11: Configuration de ces matériels

## 4.5.2 Table des mnémoniques

La définition des variables du programme nécessite la création de la table des mnémoniques. La figure 4.12 représente notre table des mnémoniques.



	Stat	Symbol /	Address	Data type	Comment
1		0	M 0.0	BOOL	Etape 0
2		1	M 0.1	BOOL	Etape 1
3		2	M 0.2	BOOL	Etape 2
4		3	M 0.3	BOOL	Etape 3
5		4	M 0.4	BOOL	Etape 4
6		5	M 0.5	BOOL	Etape 5
7		arret	M 1.2	BOOL	Boutton arret
8		Blipolaire	M 0.6	BOOL	
9		BP	M 1.3	BOOL	Boutton pousoir
10		comp1	M 0.7	BOOL	comparateur 1
11		comp2	M 1.0	BOOL	comparateur 2
12		CONV	Q 0.2	BOOL	Convoyeur
13		CONVm	Q 0.4	BOOL	Convoyeur metalique
14		CRUCH	Q 0.3	BOOL	Moteur de concasseur
15		INIT	I 2.4	BOOL	Init
16		R1	I 0.0	BOOL	Ready Moteur 1
17		R2	I 0.1	BOOL	Ready Moteur 2
18		R3	I 0.2	BOOL	Ready Moteur 3
19		R4	I 0.3	BOOL	Ready Moteur 4
20		R5	I 0.4	BOOL	Ready Moteur 5
21		RUN1	I 1.3	BOOL	RUN1
22		RUN2	I 1.4	BOOL	RUN2
23		RUN3	I 1.5	BOOL	RUN3
24		RUN4	I 1.6	BOOL	RUN4
25		RUN5	I 1.7	BOOL	RUN5
26		SCALE	FC 105	FC 105	Scaling Values
27		SM	Q 0.1	BOOL	Separateur magnetique
28		STKR	Q 0.0	BOOL	Stacker
29		temp	MD 4	REAL	
30		vibration	MD 12	REAL	
31		XH1	I 0.6	BOOL	cable arret 1
32		XH2	I 1.0	BOOL	cable arret 2
33		XH3	I 1.2	BOOL	cable arret 3
34		XS1	I 2.0	BOOL	Capteur de rotation 1
35		XS2	I 2.1	BOOL	Capteur de rotation 2
36		XS3	I 2.2	BOOL	Capteur de rotation 3
37		XS4	I 2.3	BOOL	Capteur de rotation 4
38		XZ1	I 0.7	BOOL	Capteur de deviation 1
39		XZ2	I 1.1	BOOL	Capteur de deviation 2
40					

Figure 4.12: Table des mnémoniques

### 4.5.3 Simulation dans PLCSIM

L'application S7-PLCSIM nous permet de simuler notre programme sans besoin de faire la liaison au matériel S7-300. La S7-PLCSIM fournit une interface utilisateur graphique permettant de visualiser les variables du programme de commande et de l'exécuter sur la CPU, ainsi que de modifier l'état de fonctionnement de l'API de simulation [9].

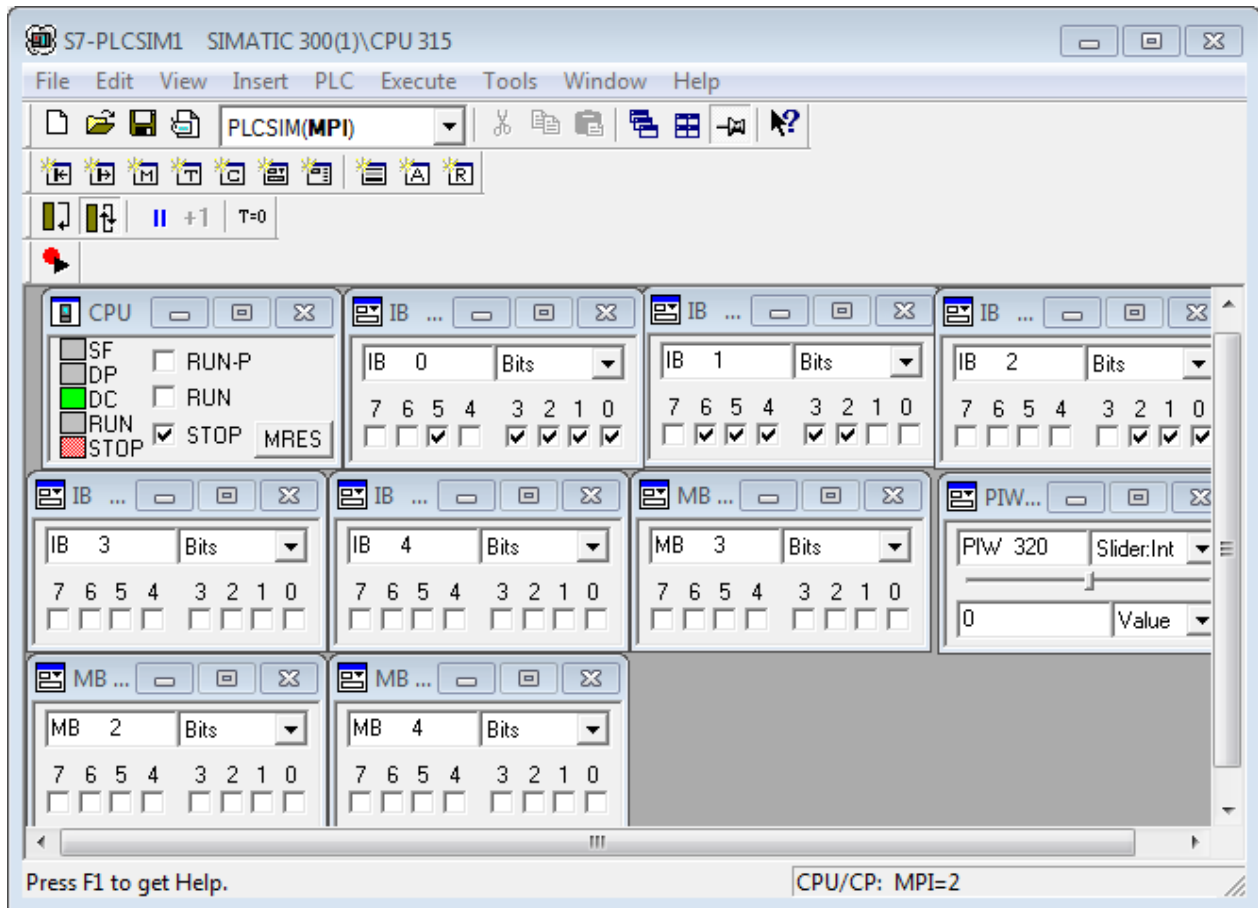


Figure 4.13: Simulation de l'application

### 4.5.4 Programme en STEP7

Les figures suivantes représentent quelques parties du programme principal en langage LADDER (langage à contacts). Le programme complet est donné dans l'annexe.

- Réseau de l'étape initiale

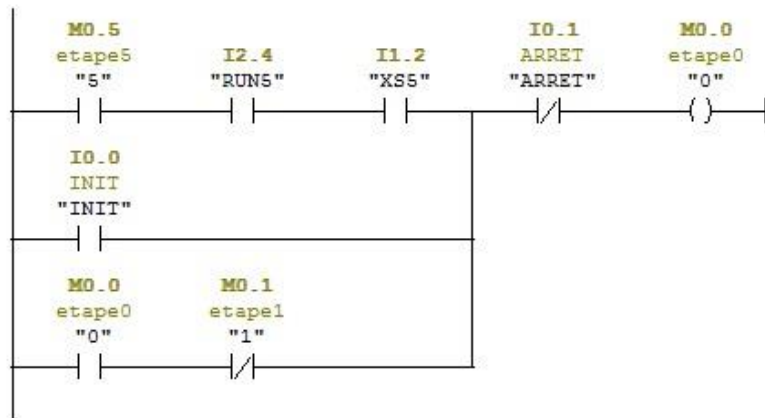


Figure 4.14: Réseau de l'étape 0 dans le bloc OB1

- Le réseau suivant représente la fonction « FC105 Scaling values » suivi par un comparateur qui compare la valeur analogique de la température avec la valeur donnée au comparateur.

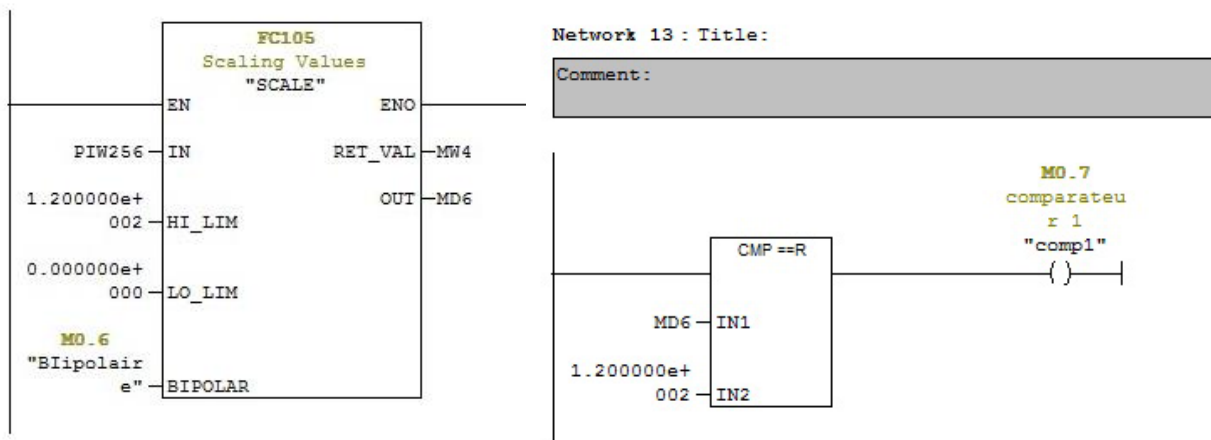


Figure 4.15: Transformation d'une entrée analogique a un intervalle de température

On utilise la fenêtre « PIW256 » dans S7-PLCSIM pour contrôler la valeur de température.

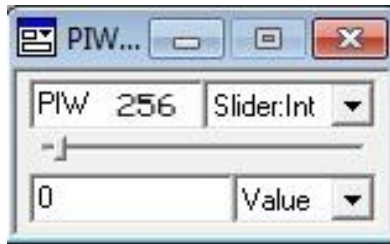


Figure 4.16: La fenêtre « PIW256 »

Le résultat de simulation avant d'atteindre la condition est donné à la figure 17.

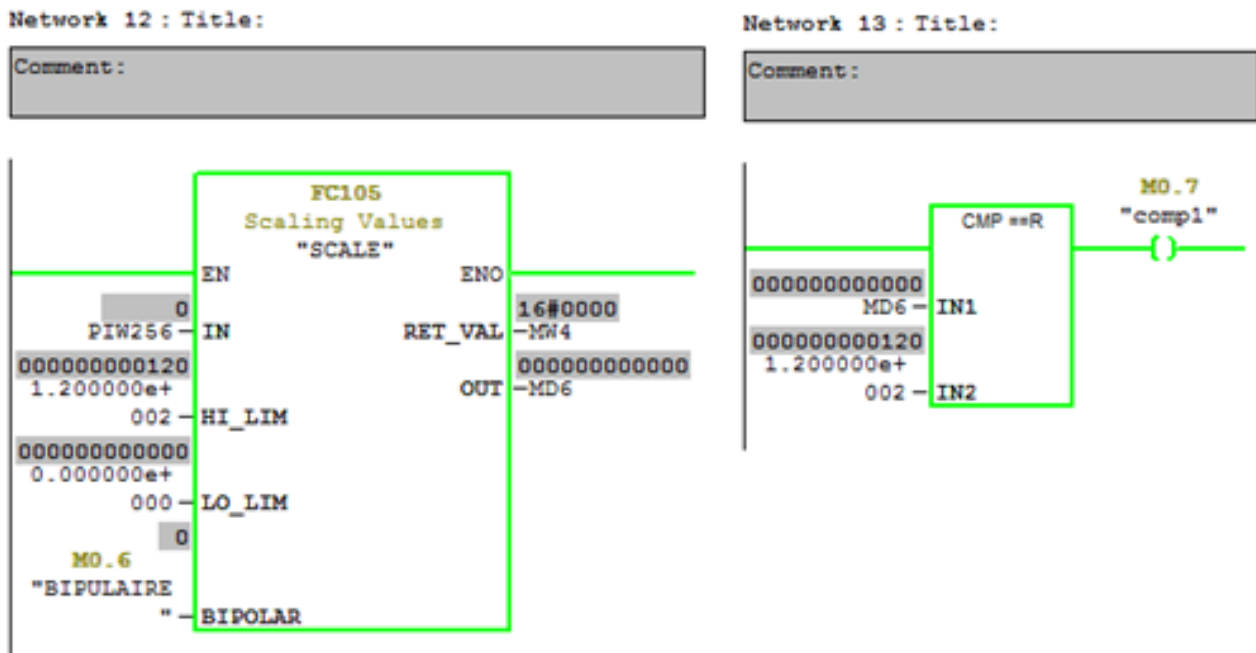


Figure 4.17: Le comparateur avant d'atteindre la condition d'arrêt

Si nous donnons au contrôleur « PIW256 » la valeur maximale qui vérifie la condition d'arrêt le comparateur s'arrête.

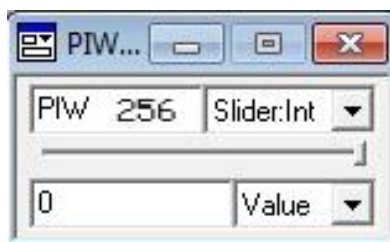


Figure 4.18: La fenêtre « PIW256 » à la valeur maximale

Dans ce cas le comparateur s'arrête comme illustré à la figure 4.19.

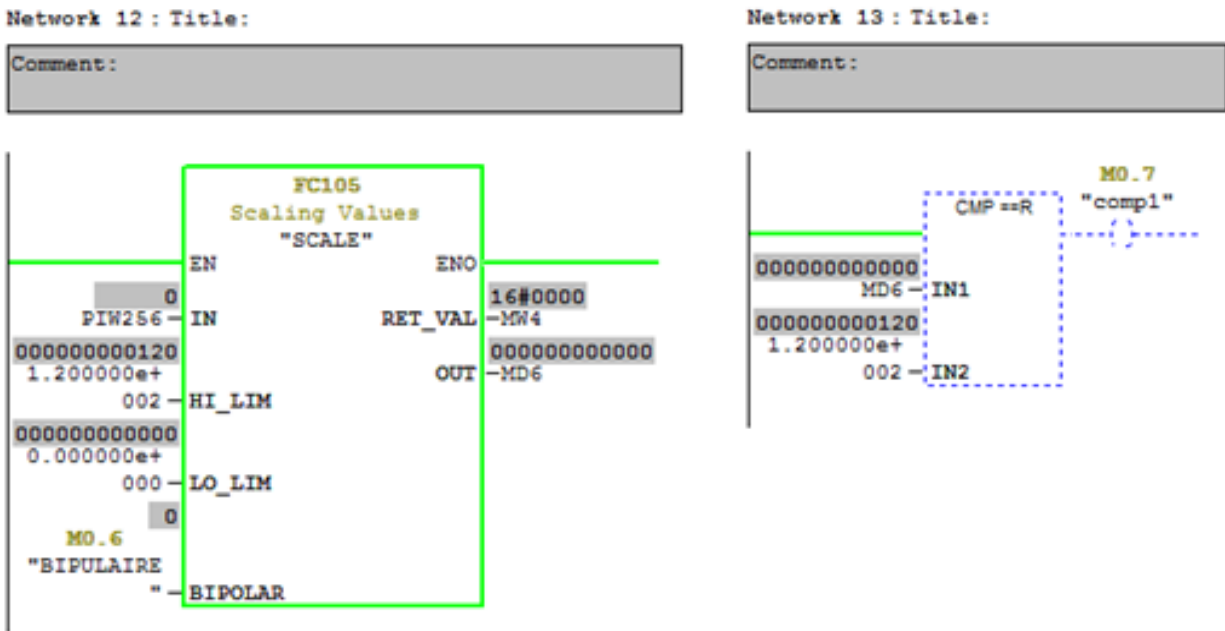


Figure 4.19: L'arrêt du comparateur

## 4.6 Création de la station HMI sur WinCC flexible

En suivant les étapes décrites dans le chapitre 3 « Logiciels utilisés », nous avons créé un nouveau projet.

Dans notre projet on a introduit une station HMI, en choisissant le type de pupitre SIMATIC MP 370 15" Touch. Sur lequel, les informations seront transmises.

### 4.6.1 Création d'une vue d'interface principale de notre application

On utilise les éléments graphiques de la bibliothèque du logiciel pour dessiner une vue de l'interface. Ensuite on doit programmer les boutons de contrôle et les alarmes qui affichent les états du fonctionnement de la station. La figure 4.20 montre une vue de la séquence de concassage et de transport de la matière ajout.

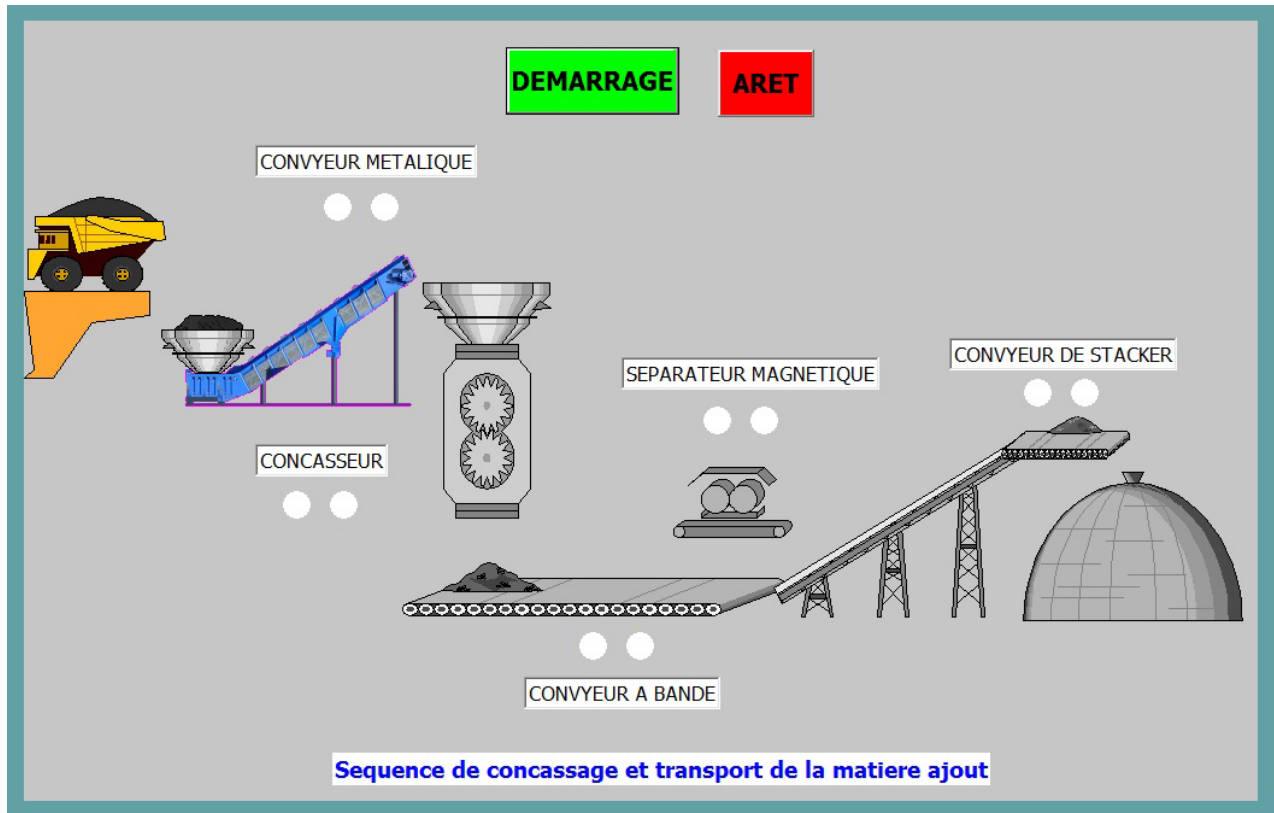


Figure 4.20: Vue globale dans la séquence de notre application

### 4.6.2 Création des variables de notre application.

La figure 4.21 montre le tableau de variables défini pour la transmission des données entre la station et le pupitre opérateur.

Name	Connection	Data type	Symbol	Address	Array elements	Acquisition cycle
R4	CPU 315	Bool	R4	I 0.3	1	1 s
CONVm	CPU 315	Bool	CONVm	Q 0.4	1	1 s
arret	CPU 315	Bool	arret	M 1.2	1	1 s
CONV	CPU 315	Bool	CONV	Q 0.2	1	1 s
SM	CPU 315	Bool	SM	Q 0.1	1	1 s
R5	CPU 315	Bool	R5	I 0.4	1	1 s
R2	CPU 315	Bool	R2	I 0.1	1	1 s
CRUCH	CPU 315	Bool	CRUCH	Q 0.3	1	1 s
STKR	CPU 315	Bool	STKR	Q 0.0	1	1 s
BP	CPU 315	Bool	BP	M 1.3	1	1 s
R1	CPU 315	Bool	R1	I 0.0	1	1 s
R3	CPU 315	Bool	R3	I 0.2	1	1 s

Figure 4.21: Variables de notre application

### 4.6.3 Configuration de la liaison

La figure 4.22 illustre l'étape de la configuration de la liaison indispensable. Cette dernière est établie en choisissant le protocole de communication MPI.

The screenshot shows the 'Connections' window with a table at the top listing the connection between 'CPU 315' and 'CPU 315' using the 'SIMATIC 57 300/400' driver. Below the table, the 'Parameters' section is expanded to show the configuration for the 'MP 377 15" Touch' HMI device. The 'Interface' is set to 'IF1B MPI/DP'. The 'HMI device' section shows 'Type' set to 'Simatic', 'Baud rate' at '187500', 'Address' at '1', and 'Access point' at '57ONLINE'. The 'Network' section shows 'Profile' set to 'MPI', 'Highest station address (HSA)' at '31', and 'Number of masters' at '1'. The 'PLC device' section shows 'Address' at '2', 'Expansion slot' at '2', 'Rack' at '0', and 'Cyclic operation' checked.

Figure 4.22: La liaison entre le pupitre et la station



#### 4.6.4 La simulation de projet à l'aide de WinCC flexible Runtime

La simulation du système est faite dans l'environnement WinCC flexible Runtime. La figure 4.23 illustre l'exécution de la séquence de de concassage et de transport de la matière ajout. La séquence est en état d'arrêt et les alarmes sont en rouge signifiant l'état d'arrêt de chaque unité. Les alarmes en blanc indiquent l'absence de signal ready.

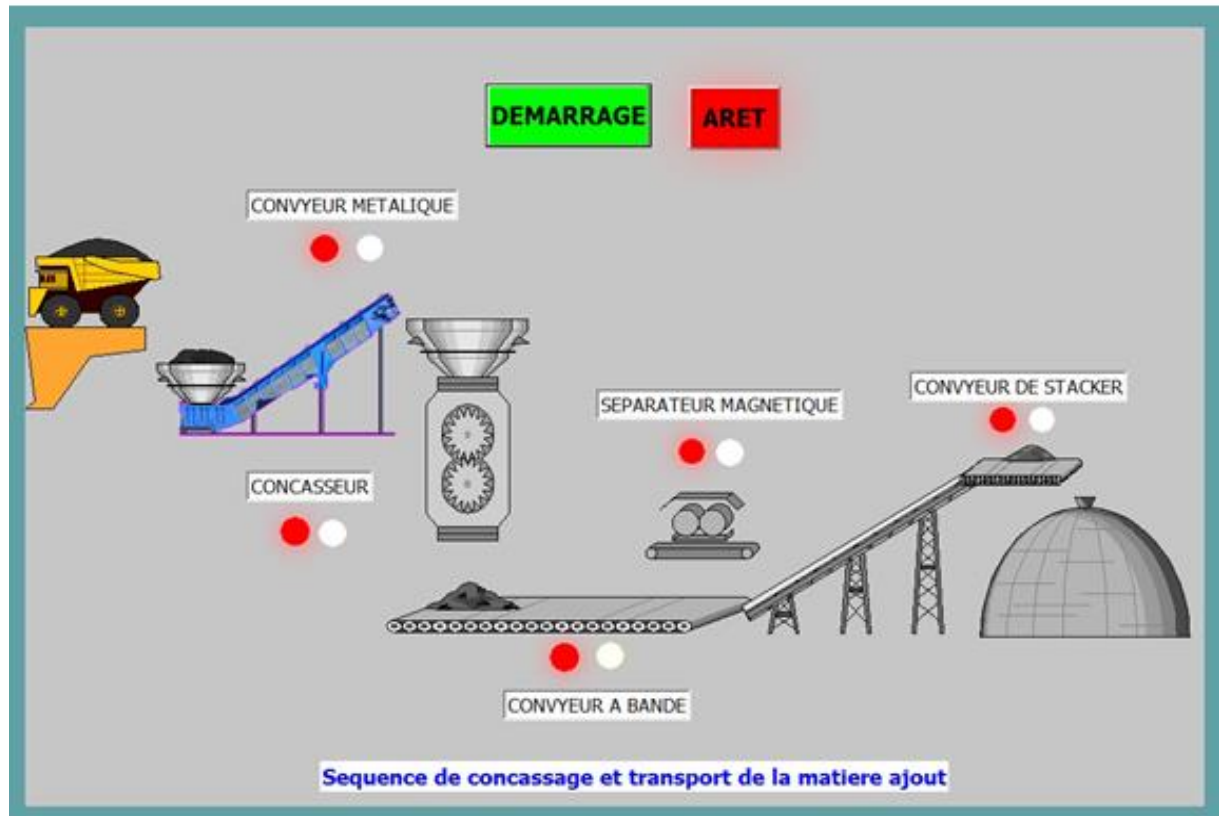


Figure 4.23: L'exécution du système (en état d'arrêt)

On utilise S7-PLCSIM pour forcer les variables, on coche les cases de signal ready pour chaque unité.

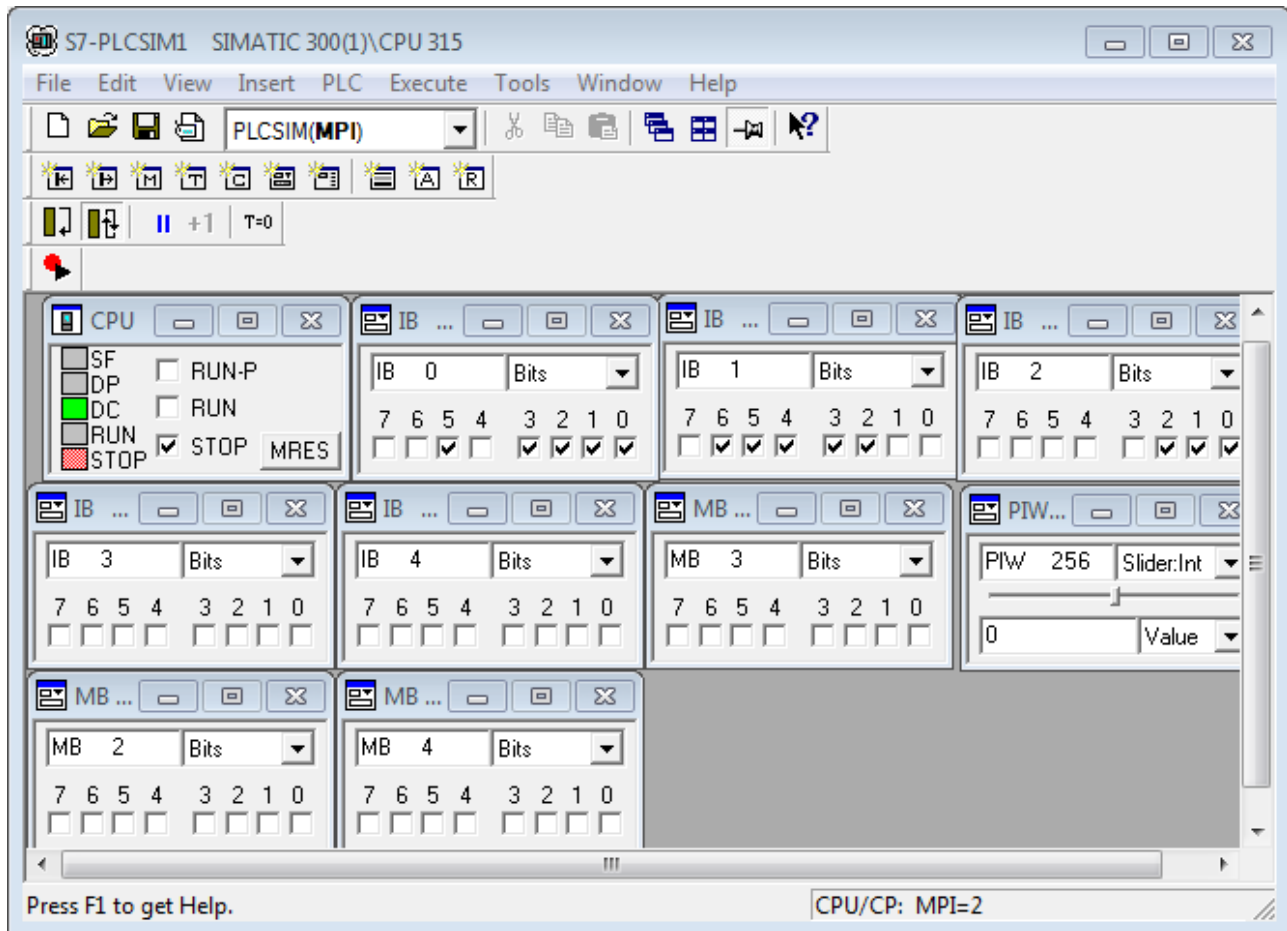


Figure 4.24: Forçage des variables en S7-PLCSIM

Les alarmes en blanc se transforment en jaune (ce signifie que la machine est en état ready pour démarrer). Si on appuie sur le bouton de démarrage on remarque que tous les alarmes en rouges se transforment en vert, et le système commence à fonctionner.

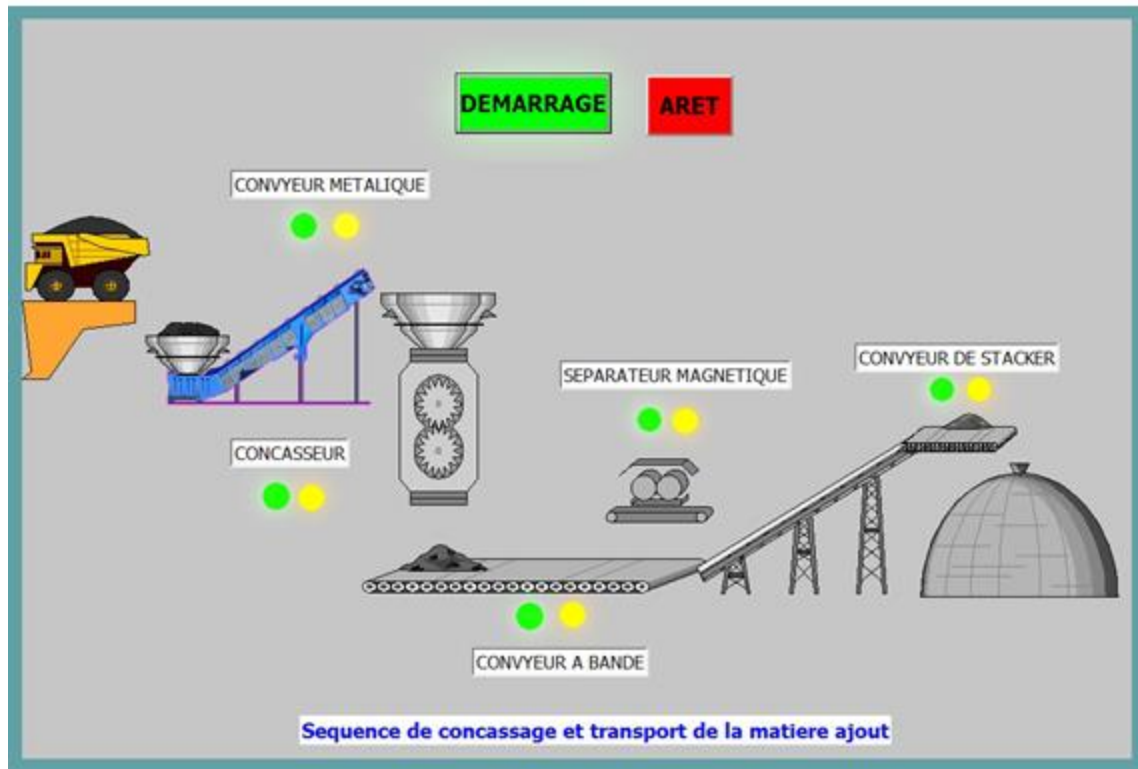


Figure 4.25: L'exécution du système (en état de marche)

## 4.7 Conclusion

Nous avons développé dans ce chapitre l'automatisation d'un processus industriel qui est la séquence de concassage et transport de matière ajout. On a traduit notre système à un cahier des charges qui est converti en un programme utilisant le logiciel de programmation STEP7. En dernière partie, on a créé une interface HMI de supervision qui correspond à notre système à l'aide du logiciel WinCC flexible. Ceci nous a facilité le suivi et le contrôle de notre séquence en temps réel.

# **Conclusion générale**

## **Conclusion générale**

Notre travail consistait à faire l'automatisation d'un processus industriel au sein de l'usine SPA BISKRIA CIMENT (la séquence de concassage et de transport de la matière ajout du ciment), Pour cela, il nous fallait étudier le fonctionnement de ce processus pour pouvoir programmer et simuler et superviser le programme avec les logiciels STEP7 et WinCC.

Notre connaissance du logiciel STEP7 nous a permis d'implémenter quelques tâches d'automatisation sur l'automate S7-300, Ce que nous ne pouvions pas réellement appliquer, Sinon cela aurait causé de graves problèmes.

Après la programmation de la séquence, on a récupéré l'état des variables qui nous intéressaient pour créer l'interface homme-machine. Pour la conception du projet de supervision du système, nous avons exploité les performances de WinCC flexible qui est un logiciel permettant de gérer les interfaces graphique avec des visualisations et des animations actualisées.

Ce projet était une occasion d'approfondir nos connaissances acquises durant notre formation et de les confronter en étude de simulation à un problème d'industrie réel. Cela nous a permis d'acquérir une expérience dans le domaine de la pratique.

# **Bibliographie**

## Bibliographie

- [1] : Site Web [<http://biskriaciment-dz.com>] Révisé Le 05/06/2018
- [2] : Site Web [<http://www.ecocem.fr/fabrication-du-ciment/>] Révisé Le 01/07/2018
- [3] : Mémoire Master Présentation et supervision d'un système de graissage à l'aide d'un automate programmable S7-300, par Abdelaziz Abdelali 2018
- [4] : Mémoire Master Automatisation et supervision d'un système de dosage de matière clinker de ciment par automate S7.300 et WinCC flexible par Ben Hassin Lotfi 2018
- [5] : Documentations de l'usine SPA BISKRIA CIMENT
- [6] : Automates Programmables industriels, William Bolton, 2e édition, DUNOD, 2015
- [7] : Historique de l'API Article disponible sous : [<https://www.institut-numerique.org/iii41-historique-de-lapi-4e1b375ea28df>] Révisé Le 03/04/2019
- [8] : Mémoire Master commande et supervision d'un processus de fabrication de margarine via un automate programmable CHERCHOUR Hamza 2015
- [9] : Support de cours Automatisme industriels, Dr. SAADOUNE Achour, 2013
- [10] : Support de cours, Automate Programmable Industriel disponible sous: [<https://www.automation-sense.com/blog/automatisme/>]
- [11] : Manuel Système d'automatisation S7-300 Caractéristiques des modules 2013.
- [12] : Mémoire Master Automatisation avec supervision à l'aide d'un automate Siemens S7-300 et logiciel WinCC, Nahali Oussama et Ouacifi Hamza 2011
- [13] : Manuel SIEMENS, «Programmation avec STEP7», SIMATIC, 2008.
- [14] : Manuel SIEMENS, « S7PLCSIM, Testez vos programmes », SIMATIC, 2008.
- [15] : SIMATIC HMI WinCC flexible 2008 Compact/ Standard/ Advanced Manuel d'utilisation.
- [16] : Mémoire Master Automatisation et supervision d'une station de Thermolaquage par un automate S7-1200, Chikh Baelhadj Brahim 2017
- [17] : Site Web <http://www.seid.it/> révisé 15/05/2018
- [18] : MERABTI Soufiane, Les étapes de la fabrication du ciment, Article disponible sous: [<https://www.slideshare.net/soufiane0806/les-tapes-de-la-fabrication-du-ciment-soufiane-merabti>] Révisé Le 12/06/2019

[19] : Mémoire Master Etude et automatisation de système du dosage des composants de ciment par un automate s7-300, OUAFI Mohamed 2018

[20] : Détecteurs de position et de proximité Article disponible sous:  
[<https://sitelec.org/cours/hu/detecteurs.htm#position>] Révisé Le 17/06/2019



# **Annexe**

# OB1 - <offline>

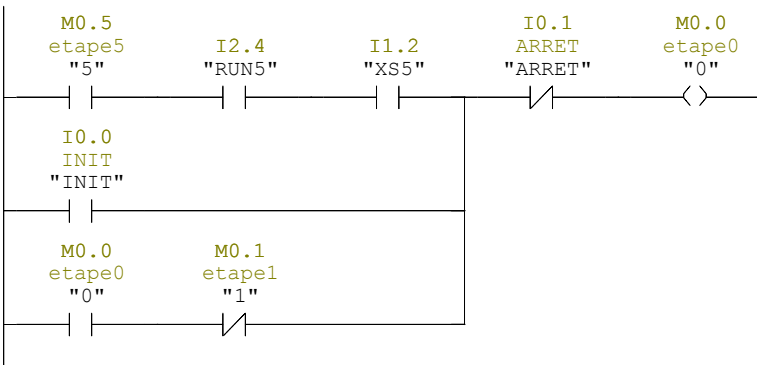
""

**Name:** **Family:**  
**Author:** **Version:** 0.1  
**Block version:** 2  
**Time stamp Code:** 07/01/2019 12:19:12 PM  
**Interface:** 02/15/1996 04:51:12 PM  
**Lengths (block/logic/data):** 00480 00338 00030

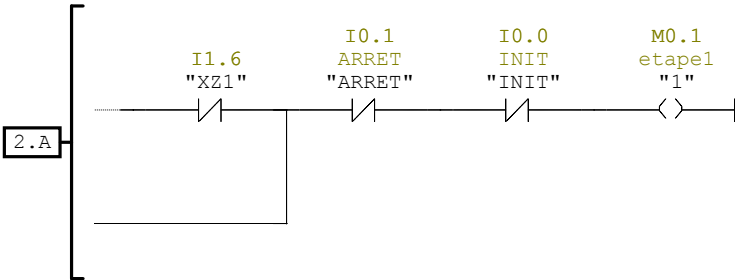
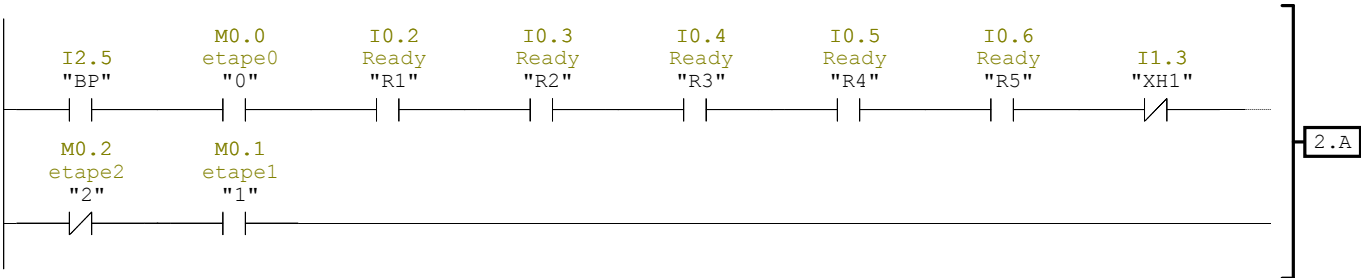
Name	Data Type	Address	Comment
TEMP		0.0	
OB1_EV_CLASS	Byte	0.0	Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
OB1_SCAN_1	Byte	1.0	1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1)
OB1_PRIORITY	Byte	2.0	Priority of OB Execution
OB1_OB_NUMBR	Byte	3.0	1 (Organization block 1, OB1)
OB1_RESERVED_1	Byte	4.0	Reserved for system
OB1_RESERVED_2	Byte	5.0	Reserved for system
OB1_PREV_CYCLE	Int	6.0	Cycle time of previous OB1 scan (milliseconds)
OB1_MIN_CYCLE	Int	8.0	Minimum cycle time of OB1 (milliseconds)
OB1_MAX_CYCLE	Int	10.0	Maximum cycle time of OB1 (milliseconds)
OB1_DATE_TIME	Date_And_Time	12.0	Date and time OB1 started

**Block: OB1 "Main Program Sweep (Cycle)"**

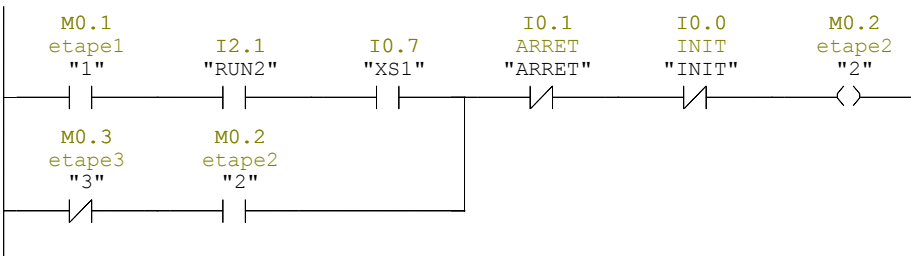
Network: 1 etape0



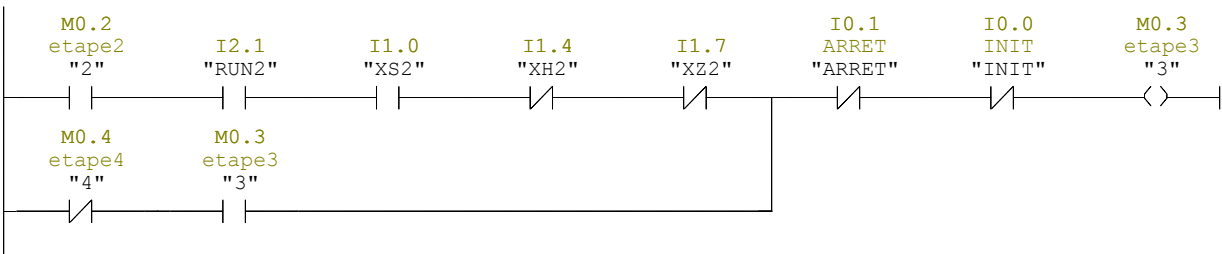
Network: 2      etape1



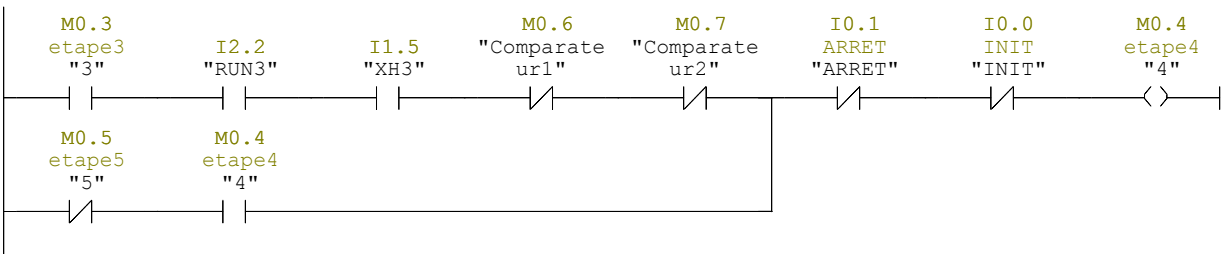
Network: 3      etape2



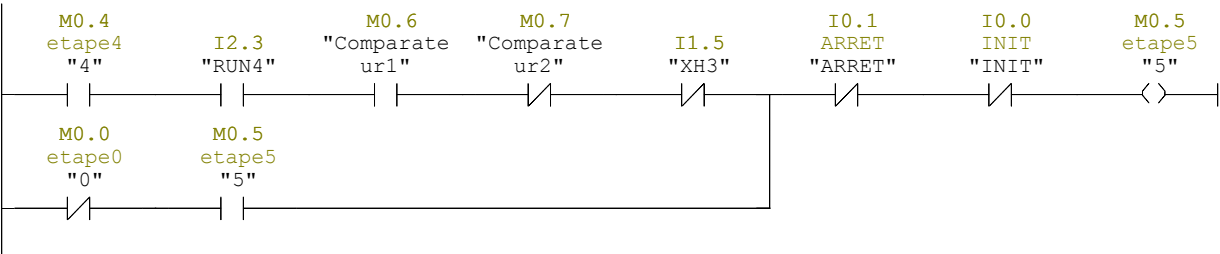
Network: 4      etape3



Network: 5      etape4



Network: 6      etape5



Network: 7



Network: 8



Network: 9



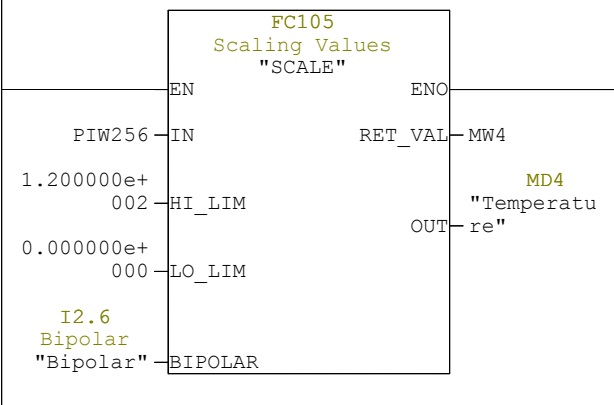
Network: 10



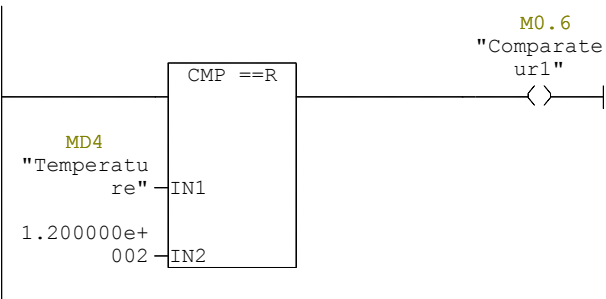
Network: 11



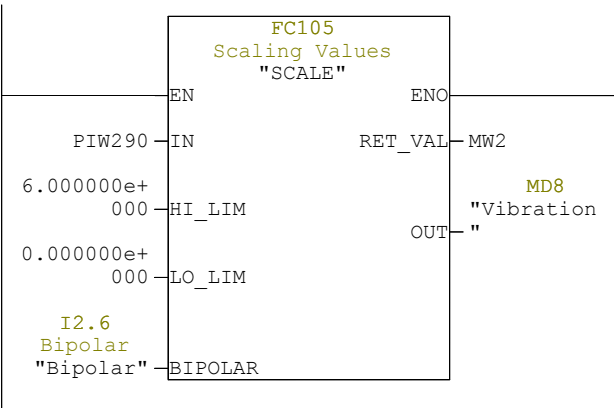
Network: 12



Network: 13



Network: 14



Network: 15

