

**République Algérienne Démocratique Et Populaire**  
**Ministère De l'Enseignement Supérieur Et De La Recherche Scientifique**  
**Université Abderrahmane MIRA**  
**Faculté De Technologie**  
**Département De Génie Electrique**

**Mémoire De Projet De Fin D'Études**

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

*Option : Automatique*

**Thème**

---

**Automatisation d'un système de remplissage de quatre  
trémies de sucre, avec supervision HMI,**

**CEVITAL**

---

Présenté par:

M<sup>lle</sup> NAMAOUI Nazha

Encadré par:

Mr. LEHOUCHE Hocine

Mr. RECHAM Hassen

Examiné par:

Mr. ARKOUB

Mr. CHARIKH

*Promotion 2014*

## *Remerciement*

*A Dieu Le Tout Miséricordieux, ton amour, ta miséricorde et Tes grâces à mon endroit m'ont fortifiée dans la persévérance et l'ardeur au travail.*

*Je tiens à remercier avant tout, les membres du jury qui me font l'honneur de participer à l'examen de ce travail.*

*Je souhaite tout d'abord remercier Mr RECHAM Hassen, responsable service méthode à l'entreprise CEVITAL, de m'avoir encadré, soutenu, durant mon stage, et fait partager sa passion pour la recherche et sa rigueur scientifique. Un grand merci pour sa patience et sa disponibilité, ainsi que ses qualités humaines et scientifiques, J'ai apprécié ses connaissances technologiques qui m'ont permis d'acquérir de nouvelles compétences que j'ai pu exploiter dans le développement de mon mémoire.*

*Je remercie également Mr LEHOUICHE Hocine, Docteur à l'Université A.MIRA, de m'avoir encadré, et encourager et guider à poursuivre mon projet, pour les nombreux échanges que nous avons eus et pour sa disponibilité et son aide dans des moments de doute.*

*Ce mémoire ayant été réalisée au sein de l'entreprise CEVITAL Bejdja, je tiens donc à remercier tout naturellement son directeur.*



*J'associe à ces remerciements pour Mr. Mustapha LOUIBA que je considère comme mon mentor, et sans oublier mes enseignants pour leurs enseignements et leurs efforts fournis durant toute la période d'étude.*

*Enfin, je remercie sincèrement tout le personnel de service méthode qui m'ont accepté et agréablement accueillis de puis le premier jour, et toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce travail.*

*Nazha*



## Dédicace

A l'âme de mon père, qui me conseillait et me poussait vers le but le plus sublime dans la vie, comme il a toujours agi tenacement afin de surmonter toutes les difficultés, ton amour et ta sollicitude à mon égard ne manqueront à jamais.

A la mémoire de mon grand père Mr. Mohand MAMMERRI, qui a disparu en martyr pour son honorable pays.

A celle qui m'a transmis la vie, l'amour, le courage, à toi chère maman toutes mes joies, mon amour et ma reconnaissance.

A mon grand frère, ZID pour son encouragements, son assistance et soutiens.

A mes sœurs, HAYATS, CHEZALA et LAHINA pour leurs encouragements, de m'avoir toujours supporté dans mes décisions, pour tout votre amour et votre confiance, et surtout de m'avoir aidé à ranger mon éternel désordre! Je vous aime beaucoup.

Sans oublier ma petite princesse AFNNANE, et mes petits chéris ANAS, WANIS et YAHIA.

A tous mes chère amis: M. NADIR, MOULOUUD, KAHIS, LIDYA, B. NADIR, TASSADIT, NADIRA, DJIDJA, SIHAM et son adorable mari CHABANE.

Et plus particulièrement à celui qui m'a donné une joyeuse vision de la vie, LYES.

Nazha

*Encadré par : Mr. Hassen RECHAME*

*Mr. Hocine LEHOUCHE*

---

## *Liste Des Figures*

---

Figure 1.1 : Organigramme du complexe CEVITAL.....	04
Figure 1.2: Situation géographique du complexe CEVITAL.....	05
Figure 1.3 : Structure d'un système automatisé.....	07
Figure 1.4 : Automate programmable industriel.....	07
Figure 1.5 : Architecture d'un API.....	12
Figure 1.6 : Les interfaces d'entrées/sorties.....	15
Figure 1.7 : Cycle d'un API .....	16
Figure 1.9: Exemple d'ouverture d'une vanne .....	17
Figure 1.10 : exemple d'un programme GRAFCET du fonctionnement d'un vérin.....	18
Figure 1.11 : Exemple d'allumage d'une LED sous langage a List.....	19
Figure 1.12 : Aperçu de la fenêtre Wincc flexible.....	23
Figure 2.1: Section ensachage .....	24
Figure 2.2 : Capteur LSH utilisé dans les trémies.....	25
Figure 2.3: Emplacement capteur LSH dans des silos contenant des solides.....	26
Figure 2.4 : Emplacement du détecteur de bourrage dans notre circuit.....	26
Figure 2.5 : Détecteur de fin de course montée sur la vanne guillotine.....	27
Figure 2.6 : Capteur de niveau.....	28
Figure 2.7 : Emplacement du capteur de niveau dans les trémies.....	29
Figure 2.8 : Une vis.....	30
Figure 2.9 : Moteur asynchrone .....	31
Figure 2.10 : Vérin simple effet avec son distributeur 3/2.....	31
Figure 2.11: Vérin double effet avec son distributeur 4/2 .....	32
Figure 2.12 : Distributeur.....	32
Figure 2.13 : Emplacement du distributeur sur la vanne guillotine .....	33
Figure 2.14 : Vanne guillotine.....	34
Figure 3.1 : Blocs de programme.....	38
Figure 3.2 : le remplissage de la trémie T1022A.....	41
Figure 3.3 : fenêtre SIMATIC pour accéder au matériel de la CPU.....	41
Figure 3.4 : fenêtre SIMATIC pour accéder à la propriété de la CPU.....	42
Figure 3.5: Fenêtre SIMATIC pour choisir l'octet de Memento de cadence.....	42
Figures 3.6 : Création d'un nouveau projet.....	43
Figures 3.7 : Appellation du nouveau projet.....	44
Figures 3.8 : Configuration du matériel.....	44
Figure 3.9 : choisir la CPU pour la configuration du matériel.....	45

Figure 3.10 : Hiérarchie du programme STEP7.....	46
Figure 3.11 : Une partie de la table des Mnémoniques du projet.....	46
Figure 3.12 : Création d'une fonction .....	47
Figure 3.13: Appellation d'une fonction .....	48
Figure 3.14: Mise à l'échelle de la trémie T1022A.....	48
Figure 3.15 : Insertion d'un nouveau réseau.....	49
Figure 3.16 : Insertion d'une table de variable.....	50
Figure 3.17 : Configuration du PLCSIM.....	51
Figure 3.18 : Système d'automatisation avec un pupitre.....	52
Figure 3.19 : Insertion d'un nouveau réseau.....	52
Figure 3.20 : Choix du pupitre.....	53
Figure 3.21 : Configuration du réseau .....	53
Figure 3.22 : Réseau MIP après configuration.....	54
Figure 3.23 : Configuration de la liaison pupitre vers l'automate.....	55
Figure 3.24: création du bargraphe.....	56
Figure 3.25 : configuration de l'affichage de la trémie T1022A.....	56
Figure 3.25 : configuration d'animation de la vanne EV1019A.....	57
Figure 4.1 : faire appeler les fonctions.....	59
Figure 4.2: Activation de la simulation.....	60
Figure 4.3: Activation de la simulation.....	60
Figure 4.4: Démarrer la simulation de la vanne EV1019B.....	61
Figure 4.5 : Visualisation l'ouverture d'EV1019B dans la table de variable.....	62
Figure 4.6: Visualisation l'ouverture d'EV1019B dans FC2.....	63
Figure 4.7: Visualisation le bit d'ouverture d'EV1019B.....	63
Figure 4.8: Visualisation du remplissage de la trémie T1022D.....	64
Figure 4.9: Visualisation du soutirage de sucre dans la trémie T1022D.....	64
Figure 4.10: Activé la visualisation avec Runtime.....	65
Figure 4.11: Visualisation l'ouverture d'EV1019B sur le pupitre.....	66

*Encadré par : Mr. Hassen RECHAME*

*Mr. Hocine LEHOUCHE*

---

## ***Liste Des Tableaux***

---



Tableau 3.1 : Les différents types de variable.....40

Tableau 3.2 : memento de cadence.....41

Tableau 4.1 : cas de l'ouverture de la vanne EV1019B.....60

*Encadré par : Mr. Hassen RECHAME*

*Mr. Hocine LEHOUCHE*

---

# ***Sommaire***

---

Introduction générale .....01

## Chapitre 01 : Généralités

I.1. Présentation de l'entreprise CEVITAL.....02

I.1.1.Introduction..... 02

    I.1.2. Présentation de l'entreprise.....02

    I.1.3. Mission et objective.....02

    I.1.4. Commercialisation.....03

    I.1.5. Structure du complexe CEVITAL.....03

    I.1.6. Situation géographique.....03

I.2. Les API et les logiciels associés.....05

    I.2.1. Le but de l'automatisme.....06

    I.2.2. L'automatisation.....06

    I.2.3. Les systèmes automatisés.....06

        I.2.3.1. La Partie Commande.....06

        I.2.3.2. La Partie Opérative.....06

        I.2.3.3. L'interface.....07

    I.2.4. Les automates programmables industriels (API) .....07

        I.2.4.1. Nature des informations traitées par l'automate.....07

        I.2.4.2. Architecture interne d'un API.....08

            a. Le CPU.....08

            b. Les bus .....10

            c. La mémoire.....10

            d. L'unité d'alimentation.....10

            e. L'unité d'entrées-sorties.....11

    I.2.5. Cycle d'un API.....11

    I.2.6. Critères de choix d'un automate.....12

    I.2.7. Programmer les API.....13

    I.2.8. Description du logiciel Step7.....14

    I.2.9. Langages programmation.....15

I.2.9.1. Les langages graphiques.....	15
a. CONT et LOG.....	15
b. Langage FBD.....	16
c. Langage SFC.....	17
I.2.9.2. Les langages texte.....	18
a. Langage IL.....	18
b. SCL.....	18
I.2.10. Description du logiciel Wincc.....	19
I.2.11. WinCC flexible Runtime.....	20
I.2.11.1 Principe.....	20
I.2.11.2. Variable.....	20
I.2.12. Domaines d'emploi des automates.....	20
I.3. Conclusion.....	20

*Chapitre 02 : Description processus et identification du matériel  
utilisé*

II.1. Introduction.....	21
II.2. Description de la section maturation.....	21
II.2.1. Circuit s'ensachage.....	22
II.2.1. Description des équipements du système existant.....	23
II.2.1.1.les capteurs.....	23
A. Capteur logique.....	24
A.1. Capteur LSH.....	24
A.2. Détecteur de bourrage.....	25
A.3. Détecteur de fin de course.....	26
B. Capteur analogique.....	27
B.1. Détecteur de niveau.....	27
B.1.1. Paramètre des capteurs.....	28
II.2.1.2.Convoyeur.....	29
A. Vis sans fin.....	29
B. Les moteurs asynchrones.....	30
II.2.1.3.Les actionneurs.....	30

A. Un servomoteur.....	30
B. Vérin.....	30
B.1. Vérin simple effet (VSE).....	30
B.2. Vérin double effet (VDE) .....	31
C. Distributeur pneumatique.....	32
D. Les vannes.....	32
C.1. Les vannes pneumatiques TOR.....	32
C.2. Catégories de vannes.....	32
C.2.1. Vanne guillotine.....	33
C.3. La commande manuelle.....	34
II.2.1.4.Les automates SIEMENS.....	34
II.2.1.5.Présentation de l'automate SIEMENS S7-300.....	35
II.3. Conclusion.....	35

## Chapitre 03 : Programmation et supervision du procédé

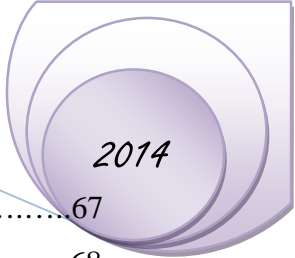
III.1. Introduction.....	36
III.2. Programmation.....	36
III.2.1. Le cahier de charge.....	36
III.2.2. Principe de la programmation sous STEP7.....	38
III.2.3. Structure du programme.....	38
III.2.4.Les opérands et types de données autorisés dans la table des mnémoniques...39	
III.2.5. L'adressage des modules E/S .....	40
III.2.6. Utilisation du front montant.....	41
III.2.7. Utilisation des memento de cadence.....	41
III.3. Création d'un projet STEP7.....	43
III.3.1. Configuration du matériel.....	44
III.3.2.Création de la table des Mnémonique.....	46
III.3.3.Elaboration du programme.....	47
III.3.3.1. Création du FC1.....	47
III.3.3.2. Création des autres fonctions.....	49
III.3.3.3. Éditer des Mnémoniques.....	50

III.3.3.4. Ouverture de l'application PLCSIM.....	50
A. Configuration du PLCSIM.....	51
III.3.3.5. Etat de fonctionnement de la CPU.....	51
III.4. Supervision.....	52
III.4.1. Interface homme/machine.....	52
III.4.2. Wincc flexible.....	52
III.4.2.1. Insertion d'une station HMI dans SIMATIC manager.....	52
III.4.2.2. Choix de la station.....	53
III.4.2.3. Configuration du réseau.....	53
III.4.2.4. Activer la liaison.....	55
III.5. Structure du projet.....	55
III.5.1. Création des trémies.....	56
III.5.2. Configurer l'affichage du bargraphe dans le pupitre.....	56
III.5.3. Créations des vannes.....	57
III.6. Conclusion.....	58

## Chapitre 04 : Tests et simulation du procédé

IV.1. Introduction.....	59
IV.2. Simulation sous step7.....	59
IV.2.1. Activation de la simulation.....	59
IV.2.2. Chargement du programme.....	60
IV.2.3. Le lancement de la simulation.....	61
IV.3. Résultats.....	61
IV.3.1. Visualiser les variables.....	62
IV.3.1.1. Visualisation de l'ouverture de la vanne EV1019B.....	62
IV.3.1.2. Visualisation du remplissage de la trémie T1022D.....	64
IV.3.1.3. Visualisation du soutirage de la trémie T1022D.....	64
IV.4. Simulation avec WINCC.....	65
IV.4.1. Activé la simulation.....	65
IV.4.2. Activé la simulation.....	66
IV.5. Conclusion.....	66

# SOMMAIRE



Conclusion générale .....	67
Bibliographies.....	68
Annexe .....	70

*Encadré par : Mr. Hassen RECHAME*

*Mr. Hocine LEHOUCHE*

---

# *Introduction **G**énérale*

---



## ***Introduction générale***

Depuis toujours l'homme est en quête de bien être, cette réflexion (qui rejoint la notion de besoin), peut paraître bien éloignée d'un cours de sciences industrielles, pourtant c'est la base de l'évolution des sciences en général, et de l'automatisation en particulier.

L'homme a commencé par penser, concevoir et réaliser, lorsqu'il a fallu multiplier le nombre d'objets fabriqués, produire en plus grand nombre, l'automatisation des tâches est alors apparue : remplacer l'homme dans des actions pénibles, délicates ou répétitives.

Notre travail consiste à réaliser un programme sous STEP7, avec un automate programmable SIEMENS S7-300, a fin de commander des vannes pour le remplissage des quatre trémies de sucre suivant leurs priorités établies au sein de l'entreprise CEVITAL, suivant d'une supervision sous WINCC flexible.

Pour cela, nous allons procéder à une étude de la station ensachage de CEVITAL, puis nous proposerons une solution d'automatisation du système que nous réaliserons conformément aux exigences de l'entreprise et nous achèverons ce projet par des tests de bon fonctionnement.

Au premier lieu, dans le premier chapitre nous allons présenter l'entreprise en passant par ses objective, sa mission et son statut géographique, et en second partie nous donnerons des généralités sur l'automatisme, on faisant apparaitre l'architecture matérielle interne d'un automate programmable industriel (API), et les langages de programmation qui lui associer.

Le second chapitre est consacré à l'identification du matériel utilisé, et la présentation de la section ensachage, en donnant des définirons sur composant du système, la structure de l'automate siemens S7-300, et la description du logiciel STEP7.

Le troisième chapitre, englobe la conception du programme et la méthodologie à suivre pour le réaliser, et aussi qui permet de gérer l'installation.

Enfin le dernier chapitre présente les différent procéder à suivre pour simuler le programme, avec des tests sous step7, et une supervision sous le WINCC flexible.

Pour terminer, une conclusion générale récapulative du travail effectué.

*Encadré par : Mr. Hassen RECHAME*

*Mr. Hocine LEHOUCHE*

---

*Chapitre 01:*

*Généralités*

---

## **I.1. Présentation de l'entreprise CEVITAL :**

### **I.1.1.Introduction :**

*Automatique*, c'est l'ensemble de théories et de techniques pour la prise de décision et la commande des systèmes.

L'automatique est ainsi la discipline scientifique permettant de caractériser les systèmes automatisés et de choisir/concevoir/réaliser la commande des systèmes, les systèmes de commande s'inspirent le plus souvent de l'homme.

D'autre part, un système automatisé est un ensemble d'éléments en interaction organisés dans un but précis, agir sur une matière d'œuvre afin de lui donner une valeur ajoutée.

CEVITAL continue à être le premier complexe agroalimentaire en matière de production en Algérie.

Dans ce premier chapitre nous allons entamer en premier lieu dans une première partie la présentation de l'entreprise CEVITAL, et en enchainera dans une deuxième partie des généralités sur les automates programmables industriels et leurs logiciels associés.

### **I.1.2. Présentation de l'entreprise :**

CEVITAL est une entreprise algérienne à une part dominante du marché sur plusieurs produits de base dont les huiles, margarines et le sucre.

La première étape créée est de lancer une industrie de raffinage de sucre roux de grande envergure, à base des dernières évolutions technologiques en la matière pour couvrir les besoins du marché national.

L'entreprise CEVITAL a en possession de deux raffineries de sucre, elles ont une capacité de production 2000 tonnes/jour et l'autre une capacité de 3000 tonnes/jour.

Elles sont conçues pour le traitement du sucre roux.

Enfin, les activités de CEVITAL sont regroupées en 5 pôles sectoriels : agroalimentaire, services, industrie lourde, grande distribution, construction.

### **I.1.3. Mission et objective :**

L'entreprise a pour mission principale de développer et d'assurer la qualité et le conditionnement des huiles, des margarines et du sucre à des prix nettement plus compétitifs et cela dans le but de satisfaire le client Les objectifs visés par CEVITAL peuvent se présenter comme suite :

- L'extension de ses produits sur tout le territoire national.
- L'importation des graines oléagineuses pour l'extraction directe des huiles brutes.
- Optimisation de ses offres d'emploi sur le marché de travail.
- L'encouragement des agriculteurs par des aides financières pour la production locale de graines oléagineuse.
- Le positionnement de ces produits sur le marché étranger par leurs exportations.

#### **I.1.4. Commercialisation :**

Un immense budget est consacré à la publicité pour amener un nombre considérant de commerçants dans les différentes wilayas du pays à se faire agréer par CEVITAL, et pour gagner la confiance du consommateur qui devient de plus en plus exigeant.

CEVITAL présente maintenant dans toutes les régions du pays tente de rapprocher au maximum ses produits aux consommateurs en offrant le meilleur rapport qualité prix, il faut savoir gérer l'excédent commerciale existant.

#### **I.1.5. Structure du complexe CEVITAL :**

Pour représenter le complexe CEVITAL d'une manière générale, la figure 1.1 relate la structure hiérarchique du complexe ainsi que les différentes directions des services :

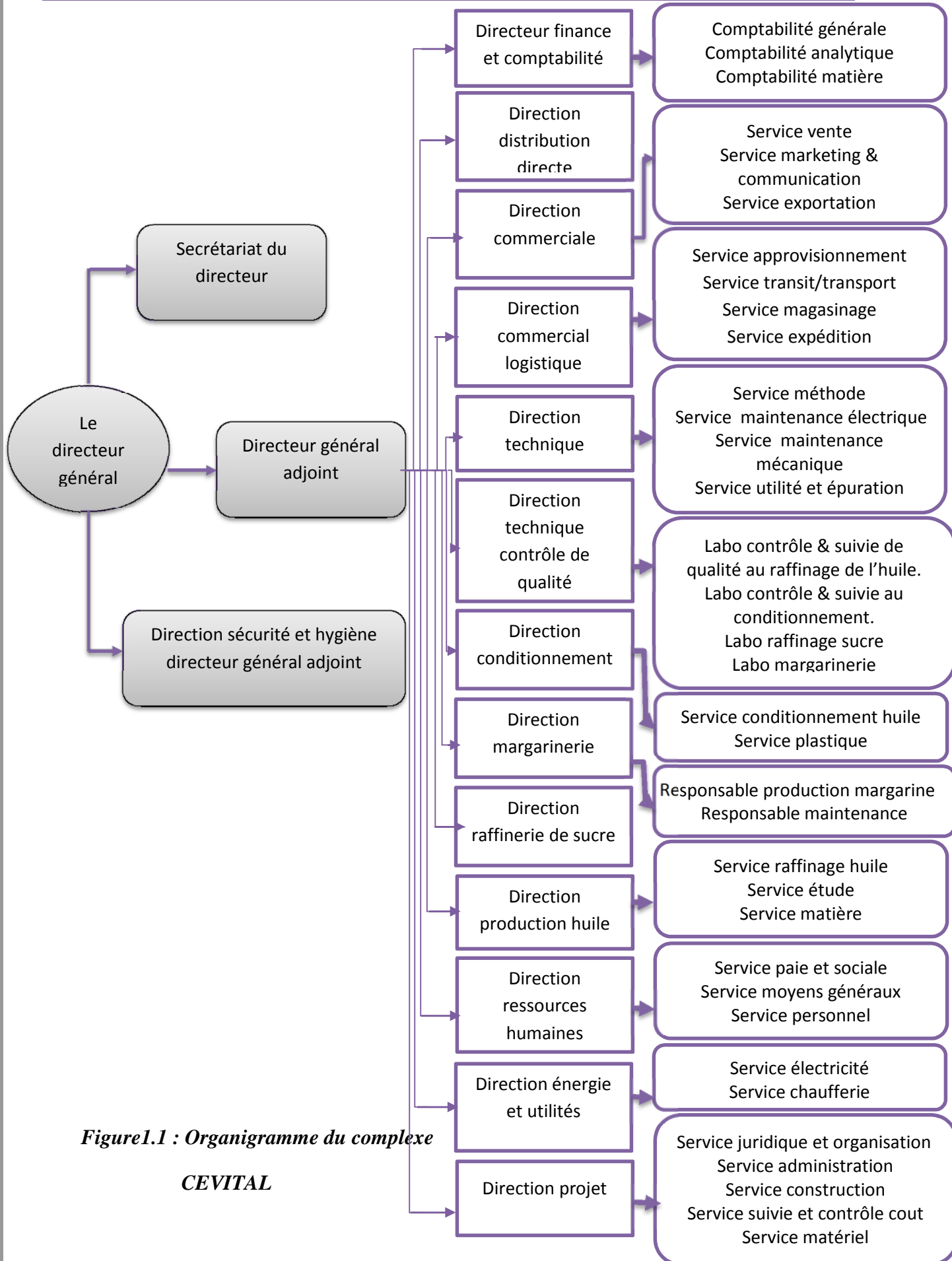


Figure 1.1 : Organigramme du complexe

CEVITAL

### ***1.1.6. Situation géographique :***

CEVITAL est implanté au niveau du nouveau quai du port de BEJAIA à 3 Km du sud-ouest de ce dernier, à proximité de la RN 26. Cette situation géographique de l'entreprise lui a beaucoup profité étant donné qu'elle lui donne l'avantage de proximité économique. En effet, elle se trouve proche du port et de l'aéroport.



***Figure1.2: Situation géographique du complexe CEVITAL***



## **I.2. Les API et les logiciels associés :**

### **I.2.1. Le but de l'automatisme :**

- Effectuer une production qualitative. (Pas d'erreur humaine : Zéro défaut.)
- Effectuer une production quantitative. (On peut solliciter un système automatisé 24h/24h)
- Supprimer les tâches ou actions physiques peu ou pas gratifiantes pour l'homme.
- Pouvoir accéder à des milieux de travail hostiles. (chimique, nucléaires ...) ou des sites inaccessibles à l'homme (mer, espace).

### **I.2.2. L'automatisation :**

L'automatisme est la discipline traitant d'une part, de la caractérisation des systèmes automatisés et d'autre part, du choix de la conception et de la réalisation de la partie commande.

Il s'agit donc d'étudier les systèmes :

- Réalisant leurs fonctions en relative autonomie.
- Assurant un contrôle des performances par la mise en place possible d'une chaîne de retour.

L'automatisation de la production consiste à transférer tout ou une partie des tâches de coordination, auparavant exécutées par des opérateurs humains.

### **I.2.3. Les systèmes automatisés :**

Un Système automatisé est toujours composé d'une partie commande (PC) et d'une partie opérative (PO), pour faire fonctionner ce système, l'opérateur (personne qui va faire fonctionner le système) va donner des consignes à la PC, celle-ci va traduire ces consignes en ordres qui vont être exécutés par la PO.

Une fois les ordres accomplis, la PO va le signaler à la PC (elle fait un compte-rendu) qui va à son tour le signaler à l'opérateur, ce dernier pourra donc dire que le travail a bien été réalisé.[1]

#### **I.2.3.1. La Partie Commande:**

C'est la partie qui gère le fonctionnement du système automatisé, elle est en général composée d'un ordinateur qui contient dans ses mémoires un programme. Elle transmet les ordres aux actionneurs de la partie opérative à partir :

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

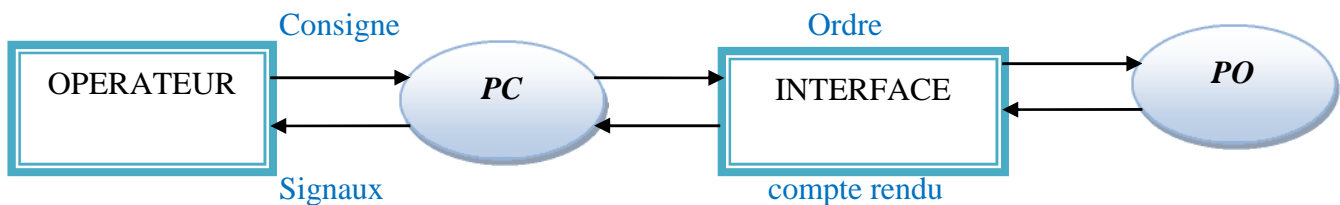
### **1.2.3.2. La Partie Opérative :**

Elle consomme de l'énergie électrique, pneumatique (air) ou hydraulique (huile), elle comporte en général un boîtier (appelé aussi bâti) contenant :

- Des actionneurs (transforment l'énergie reçue en énergie utile : moteur, vérin, lampe).
- Des capteurs (transforment la variation des grandeurs physiques liée au fonctionnement de l'automatisme en signaux électriques : capteur de position, de température, bouton poussoir).

### **1.2.3.3. L'interface :**

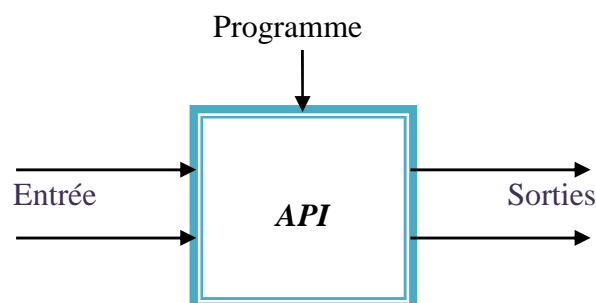
Elle relie la partie opérative (PO) et la partie commande (PC). C'est un système de traduction d'informations entre la PC et la PO.



**Figure 1.3 : Structure d'un système automatisé**

### **1.2.4. Les automates programmables industriels (API) :**

Un automate programmable industriel est une forme particulière de contrôleur à microprocesseur qui utilise une mémoire programmable pour stocker les instructions et qui implémente différentes fonctions, qu'elles soient logique de séquençage, de temporisation de comptage ou arithmétique pour commander les machines et les processus (voir figure 1.4). [2]



**Figure 1.4 : Automate programmable industriel**



Les API sont comme les ordinateurs, toutefois alors que les ordinateurs sont optimisés pour les tâches de calcul et d'affichage, les API le sont pour les tâches de commande et les environnements industriels, voici ce qui caractérise les API :

- Ils sont solides et conçus pour supporter les vibrations, les températures basses ou élevées, l'humidité et le bruit.
- Les interfaces des entrées et des sorties sont intégrées à l'automate.
- Ils sont faciles à programmer et leurs langage de programmation sont facile à comprendre est principalement orienté sur les opérations logique et de communication.

#### **I.2.4.1. Nature des informations traitées par l'automate :**

Les informations peuvent être de type :

- **Tout ou rien (TOR)** : l'information ne peut prendre que deux états (vrai/faux, 0 ou 1), c'est le type d'information délivrée par un détecteur, un bouton poussoir ...
- **Analogique** : l'information est continue et peut prendre une valeur comprise dans une plage bien déterminée, c'est le type d'information délivrée par un capteur (pression, température)
- **Numérique** : l'information est contenue dans des mots codés sous forme binaire ou bien hexadécimale, c'est le type d'information délivrée par un ordinateur ou un module intelligent.

#### **I.2.4.2. Architecture interne d'un API :**

De manière générale, un API est structuré autour de plusieurs élément de base, que sont l'unité de traitement, la mémoire, l'unité d'alimentation, interface d'entrées-sorties, l'interface de communication et le périphérique de programmation (voir figure 1.5).[2]

Il est constitué d'une unité centrale de traitement (CPU), qui comprend le microprocesseur, la mémoire, les entrées-sorties du système, elle contrôle et exécute toute les opérations de l'API, il est muni d'une horloge dont la fréquence détermine la rapidité de fonctionnement de l'API.

Au sein de l'API toutes les informations sont transmises au moyen de signaux numérique, les chemins par lesquels passent ces signaux sont appelé bus.

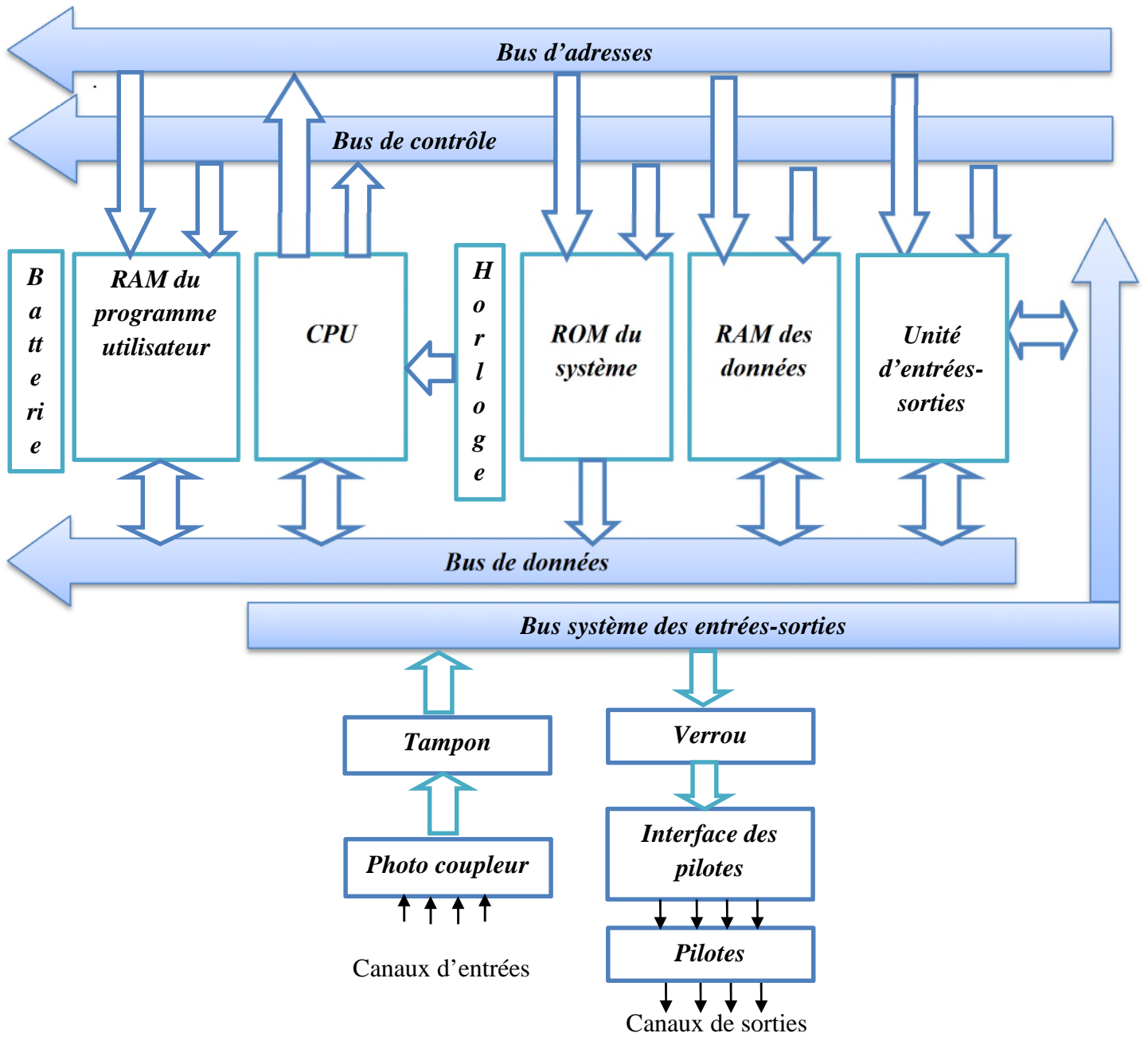


Figure 1.5 : Architecture d'un API

a. **CPU** : (CPU Central Processing Unit):

Elle contient le microprocesseur la CPU interprète les signaux d'entrées et effectue les actions de commande conformément au programme stocker en mémoire en communiquant aux sorties les décisions sous forme de signaux d'actions.

b. **Bus** :

Les bus représentent les chemins au sein de l'API, les informations sont transmises en binaire sous forme de groupe de bit, et un bit est un chiffre binaire qui vaut 1 ou 0, un mot est un groupe de bits qui constitue une information.

Le système comprend quatre bus :

- 1) ***Bus de données*** : transporte les données utilisées dans le traitement effectué par la CPU, il est bidirectionnel, le nombre de fils de ce bus varie suivant les microprocesseurs.
- 2) ***Bus d'adresse*** : transporte les adresses des emplacements mémoire pour que chaque mot puisse être localisé en mémoire chaque emplacement possède une adresse unique que la CPU utilise pour accéder aux données enregistrées à cet emplacement, que ce soit pour lire ou pour y écrire, c'est le bus d'adresse qui fournit les informations stipulant l'adresse à laquelle la CPU doit accéder. si le bus est constitué de 8 lignes le nombre de mots de 8 bits et par conséquent le nombre d'adresses distinctes est égale à  $2^8$  c'est-à-dire 256, il est possible d'accéder à 256 emplacements.
- 3) ***Bus de contrôle*** : transporte les signaux utilisés par la CPU pour le contrôle, il sert à informer les dispositifs mémoires s'ils vont recevoir des données à partir d'une entrée ou s'ils vont envoyer des données et à transmettre les signaux de minutage qui permettent de synchroniser les opérations.
- 4) ***Bus système*** : sert aux communications entre les ports d'entrées-sorties et l'unité d'entrées sorties.

c. **Mémoire** :

Elle est conçue pour recevoir, gérer, stocker des informations issues des différents secteurs du système que sont le terminal de programmation (PC ou console) et le processeur, qui lui gère et exécute le programme, elle reçoit également des informations en provenance des capteurs.

La mémoire centrale est l'élément fonctionnel qui peut recevoir conserver et restituer les données.

- 1) **La mémoire morte : (ROM : Read Only Memory):** Elle représente un espace de stockage permanent pour le système d'exploitation et les données figées, c'est à dire en lecture seulement, utilisées par le CPU.
- 2) **La mémoire vive : (RAM Random Access Memory) :**C'est une mémoire utilisable en lecture-écriture pendant le fonctionnement, elle s'efface automatiquement à l'arrêt de l'automate (nécessite une batterie de sauvegarde).
- 3) **La mémoire morte reprogrammable (EPROM :Erasable and Programmable Read Only Memory) :** Elle est parfois employée pour stocker de manière permanente les programmes.

Dans un API la mémoire est découpée en plusieurs zones :

- La zone mémoire réservé au système
- La zone mémoire programme (programme à exécuter)
- La zone mémoire des données (état des entrées et des sorties, valeurs des compteurs et des temporisateur)
- Une zone où sont stockés des résultats de calculs utilisés ultérieurement dans le programme.
- Une zone pour les variables internes

**d. L'unité d'alimentation :**

Elle est indispensable puisqu'elle convertit une tension alternative en une base de tension continue (5V) nécessaire au processeur et au module d'entrées sorties.

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

**e. L'unité d'entrées-sorties**

L'unité d'entrées-sorties apporte l'interface entre le système et le monde extérieur, elle permet d'établir des connexions avec des dispositifs d'entrée comme les capteurs, et des dispositifs de sorties comme les moteurs.

C'est également par l'intermédiaire de cette unité que se fait la saisie des programmes depuis un terminal.

Le nombre de ces entrées et sorties varie suivant le type d'automate. Les cartes d'E/S ont une modularité de 8, 16 ou 32 voies. Les tensions disponibles sont normalisées (24, 48, 110 ou 230V continu ou alternatif ...)

L'interface d'entrée comporte des adresses d'entrée, chaque capteur est relié à une de ces adresses.

L'interface de sortie comporte de la même façon des adresses de sortie, chaque pré-actionneur est relié à une de ces adresses.

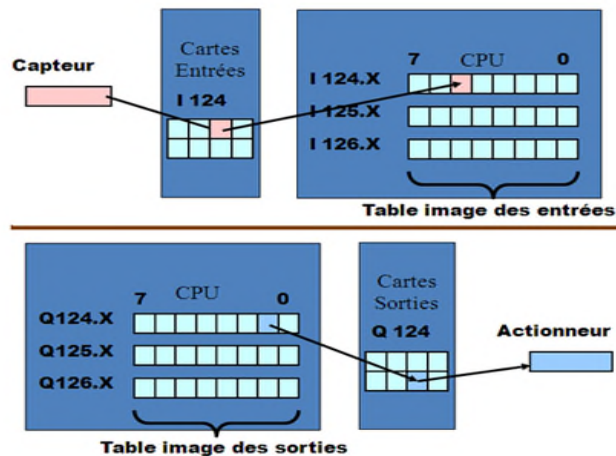


Figure 1.6: Les interfaces d'entrées/sorties

### 1.2.5. Cycle d'un API : [3]

Tous les automates fonctionnent selon le même mode opératoire :

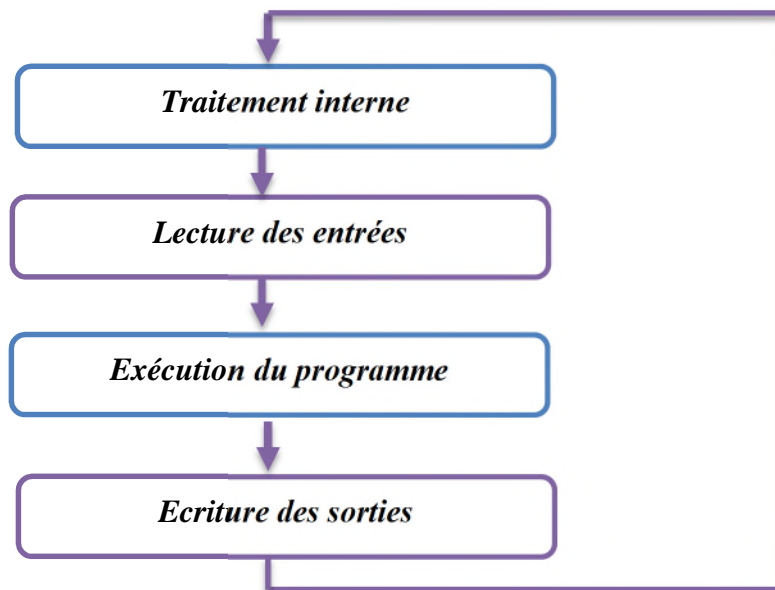


Figure 1.7 : Cycle d'un API

- **Traitement interne** : L'automate effectue des opérations de contrôle et met à jour certains paramètres systèmes (détection des passages en RUN / STOP,...).
- **Lecture des entrées** : L'automate lit les entrées (de façon synchrone) et les recopie dans la mémoire image des entrées.

- **Exécution du programme** : L'automate exécute le programme instruction par instruction et écrit les sorties dans la mémoire image des sorties.
- **Ecriture des sorties** : L'automate bascule les différentes sorties (de façon synchrone) aux positions définies dans la mémoire image des sorties.

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

### **I.2.6. Critères de choix d'un automate :**

Le choix d'un automate programmable est en premier lieu le choix d'une société ou d'un groupe et les contacts commerciaux et expériences vécues sont déjà un point de départ.

Le personnel de maintenance doit toutefois être formé sur ces matériels et une trop grande diversité des matériels peut avoir de graves répercussions. Les automates utilisant des langages de programmation de type GRAFCET est également préférable pour assurer les mises au point et dépannages dans des meilleures conditions.

La possession d'un logiciel de programmation est aussi source d'économies (achat du logiciel et formation du personnel). Des outils permettant une simulation des programmes sont également souhaitables.

Il faut ensuite quantifier les besoins :

- Nombre et type d'entrées / sorties : le nombre de cartes peut avoir une incidence sur le nombre de racks dès que le nombre d'entrées / sorties nécessaires devient élevé.
- Type de processeur : la taille mémoire, la vitesse de traitement et les fonctions spéciales offertes par le processeur permettront le choix dans la gamme souvent très étendue.
- Fonctions ou modules spéciaux : certaines cartes, permettront de "soulager" le processeur et devront offrir les caractéristiques souhaitées (résolution, ...).
- Fonctions de communication : l'automate doit pouvoir communiquer avec les autres systèmes de commande (API, supervision ...) et offrir des possibilités de communication avec des standards normalisés.

### **Remarque :**

Les systèmes automatisés sont par nature, source de nombreux dangers, (tensions utilisées, déplacements mécaniques, jets de matière sous pression ...).

Placé au cœur du système automatisé, l'automate se doit d'être un élément fiable car :

- un dysfonctionnement de celui-ci pourrait avoir de graves répercussions sur la sécurité des personnes,
- un arrêt de la production peut avoir de lourdes conséquences sur le plan financier.

### **I.2.7. Programmer les API :**

La programmation de l'API concerne principalement la mise en œuvre d'opérations logique et de communication, les dispositifs d'entrée c'est-à-dire des capteurs comme des interrupteur et les dispositifs de sortie c'est à dire moteurs, vannes...etc, du système sont connecter à l'API.

L'operateur saisit une séquence d'instructions du programme dans la mémoire de l'API, l'automate surveille ensuite les entrée et les sorties conformément aux instructions du programme et met en œuvre les règles de commande définies.

Une unité de programmation peut être un appareil portatif, un terminal de bureau ou un ordinateur, après que le programme a été conçu et tester sur l'unité de programmation il est prêt à être transférer dans la mémoire de l'API. [3]

- ❖ *Un appareil de programmation portatif* : dispose généralement d'une quantité de mémoire suffisante pour concevoir les programmes afin de les déplacer d'un endroit à un autre.
- ❖ *Les terminaux de bureau* sont généralement équipés d'un système d'affichage graphique, avec un clavier et un écran, comme c'est le cas à l'entreprise CEVITAL.
- ❖ *Les ordinateurs personnels* : sont souvent configurés comme des stations de développement des programmes, pour certain API, l'ordinateur doit simplement disposer du logiciel approprié, généralement à l'entreprise CEVITAL ils utilisent le langage Step7.

### **I.2.8. Description du logiciel Step7 :**

STEP7 est le logiciel de base pour la configuration et la programmation de systèmes d'automatisation SIMATIC S300, il fait partie de l'industrie logicielle SIMATIC. Le logicielle de base assiste dans toute les phases du processus de création de la solution d'automatisation.

SIMATIC Manager est une interface graphique assurant le traitement en ligne ou bien hors ligne d'objet STEP7, tels que les objets, fichiers de programmes utilisateur, bloc, station matérielles et outils.

STEP7 permet l'accès de base aux automates SIEMENS, il permet de programmer individuellement un automate en différents langages, il prend également en compte le réseau des automates ce qui permet d'accéder à tout automate du réseau pour programmer et éventuellement aux automates d'envoyer les messages entre eux. [4]

### 1.2.9. Langages programmation : [5]

SIMATIC STEP met à disposition de puissants éditeurs pour la programmation des automates SIMATIC S7.

- ✚ Texte structuré (SCL), schéma à contacts (CONT) et logigramme (LOG) sont disponibles pour tous les automates.
- ✚ Liste d'instructions (LIST) et programmation de chaînes séquentielles (GRAFSET) sont en outre disponibles pour les familles d'automates S7-300.

#### 1.2.9.1. Les langages graphiques :

##### a. *CONT et LOG* :

Avec STEP 7, les langages de programmation graphique sont supportés par de nouveaux compilateurs à hautes performances, les puissants outils et fonctionnalités intégrées, comme, par exemple, la programmation indirecte, accroissent l'efficacité du développement des programmes. Les éditeurs CONT et LOG, conviviaux et entièrement graphiques, offrent une très bonne visibilité et une navigation rapide dans l'éditeur de modules par :

- ouverture et fermeture de réseaux entiers.
- affichage et masquage de symboles et adresses
- zoom direct et enregistrement d'agencements
- multitude de raccourcis clavier.
- fonction lasso, copie et insertion d'instructions et de structures d'instructions
- commentaires.

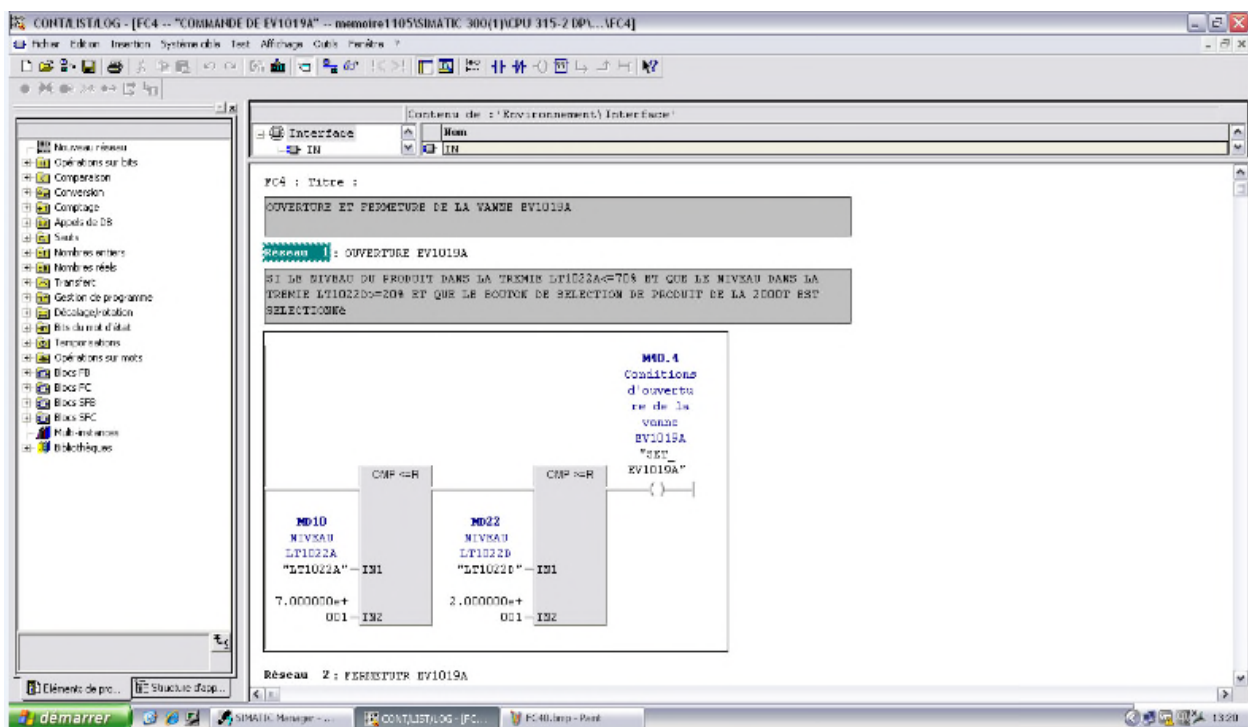


Figure 1.8 : Exemple d'une temporisation sous l'langage a contact



**b. Langage FBD (Function Block Diagram ou Logigramme):**

Langage FBD est un langage de programmation graphique qui utilise les blocs de l'algèbre de Boole pour représenter les opérations logiques. Les fonctions complexes, comme les fonctions mathématiques, peuvent être représentées directement combinées avec les blocs logiques. (voir annexe 01).

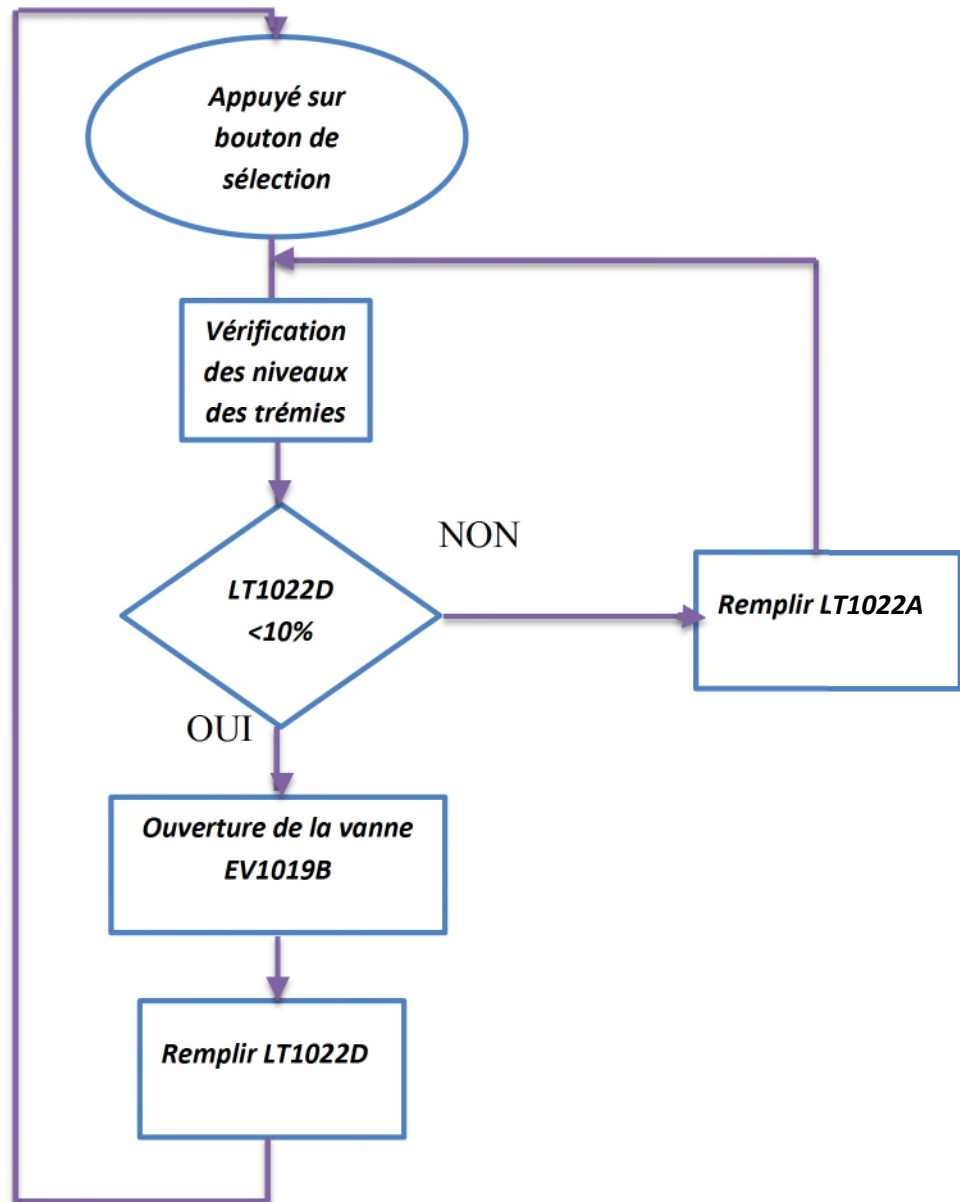


Figure 1.9: Exemple d'ouverture d'une vanne

### c. Langage SFC (Sequential function char) : Le Grafcet :

Le GRAFCET c'est un outil graphique de description du comportement déterministe de la Partie Commande, il décrit les interactions informationnelles à caractère déterministe à travers la frontière d'isolement entre la partie commande, et la partie opérative d'un système isolé, il est utilisé pour la description de procédures séquentielles à séquences alternatives ou parallèles,

Les procédures se configurent et se programment clairement et rapidement dans un mode de visualisation standardisé, le processus se décrit graphiquement en se décomposant en différentes étapes à fonctions élémentaires.

- ✚ Structure flexible des chaînes séquentielles: Branchements simultanés ou alternatifs, sauts à l'intérieur des chaînes séquentielles, activation et désactivation d'étapes.
- ✚ Exécution sélective des étapes : Le temps d'exécution d'une chaîne séquentielle est ainsi fonction du nombre d'étapes.
- ✚ Synchronisation des modes automatique et manuel : le processus n'est plus alors synchronisé quand il a été amené dans un autre état en mode manuel. GRAPH supporte la recherche de points de synchronisation en vue de la reprise du mode automatique. Les étapes pertinentes sont à cet effet repérées. Les critères définissables peuvent être des transitions ou des verrouillages.
- ✚ Visualisation étape par étape donnant un aperçu de tous les détails d'une étape
- ✚ Gain de temps important par rapport à la programmation en CONT/LOG/LIST.

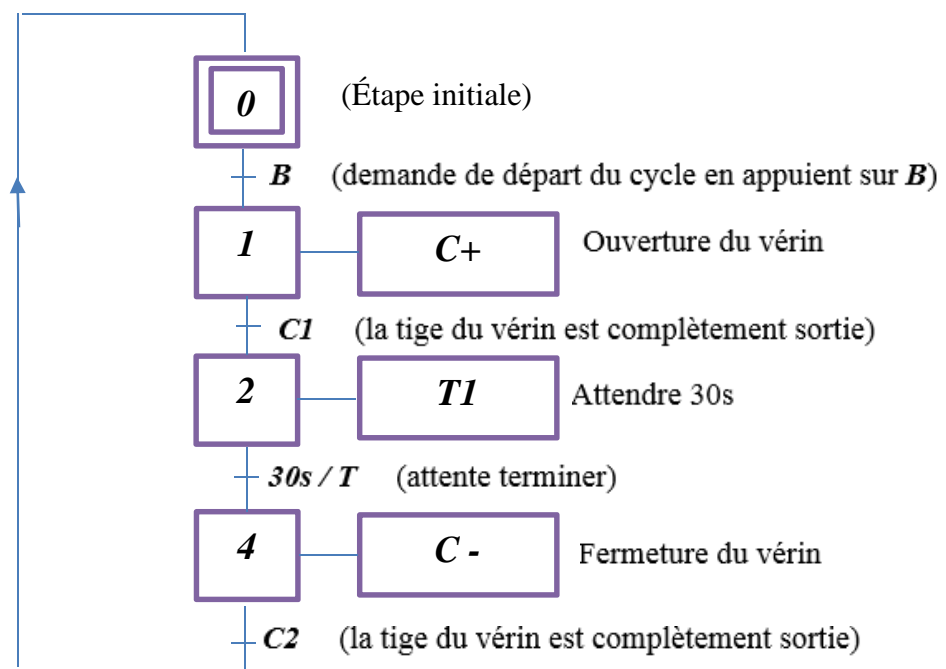


Figure 1.10 : exemple d'un programme GRAFCET du fonctionnement d'un vérin

### 1.2.9.2. Les langages textuels :

#### a. Langage IL (LIST):

Le langage de programmation textuelle LIST permet de créer des programmes d'applications à un niveau proche du matériel et en optimisant le temps d'exécution et la place en mémoire.

Il est l'abréviation de liste d'instructions. C'est un langage de liste d'instructions (*LIST*) est un langage de programmation textuel proche de la machine.

Dans un programme *LIST*, les différentes instructions correspondent, dans une large mesure, aux étapes par lesquelles la *CPU* traite le programme.

Pour faciliter la programmation, *LIST* a été complété par quelques structures de langage évolué (comme par exemple, des paramètres de blocs et accès structurés aux données).

Tout programme écrit en CONT ou en LOG peut être réécrit en LIST.

```
Réseau ?): Titre :
```

```
Commentaire :
```

```

A      M      10.5
A      M      10.0
O
A(
A      M      10.5
A      M      10.3
)
=      Q      124.6
```

Figure 1.11 : Exemple d'allumage d'une LED sous langage a List

#### b. SCL (Structured Control Language) : Programmation d'algorithmes complexes

Ce langage est un langage textuel de haut niveau ST (Structured Text), dédié aux applications d'automatisme.

SCL convient notamment à la programmation rapide d'algorithmes complexes et de fonctions mathématiques ou à des missions relevant du domaine du traitement des données difficilement modélisables avec les langages graphiques.

Le code SCL est plus simple, plus court et plus clair, ce qui en facilite et en accélère l'écriture et la manipulation.

### 1.2.10. Description du logiciel WinCC flexible :

WinCC flexible est l'interface homme-machine (IHM) idéale pour toutes les applications au pied de la machine et du processus dans la construction d'installations, de machines et de machines de série.

De part, sa conception généraliste, WinCC flexible permet de disposer d'un logiciel d'ingénierie pour tous les pupitres opérateur SIMATIC HMI, du plus petit Micro Panel jusqu'au Multi Panel ainsi que d'un logiciel de supervision *Runtime*.

Les projets peuvent être portés sans conversion et sont exécutables sur diverses plateformes IHM, Grâce à des logiciels et à des projets multilingues, WinCC flexible peut être utilisé dans le monde entier.[6]

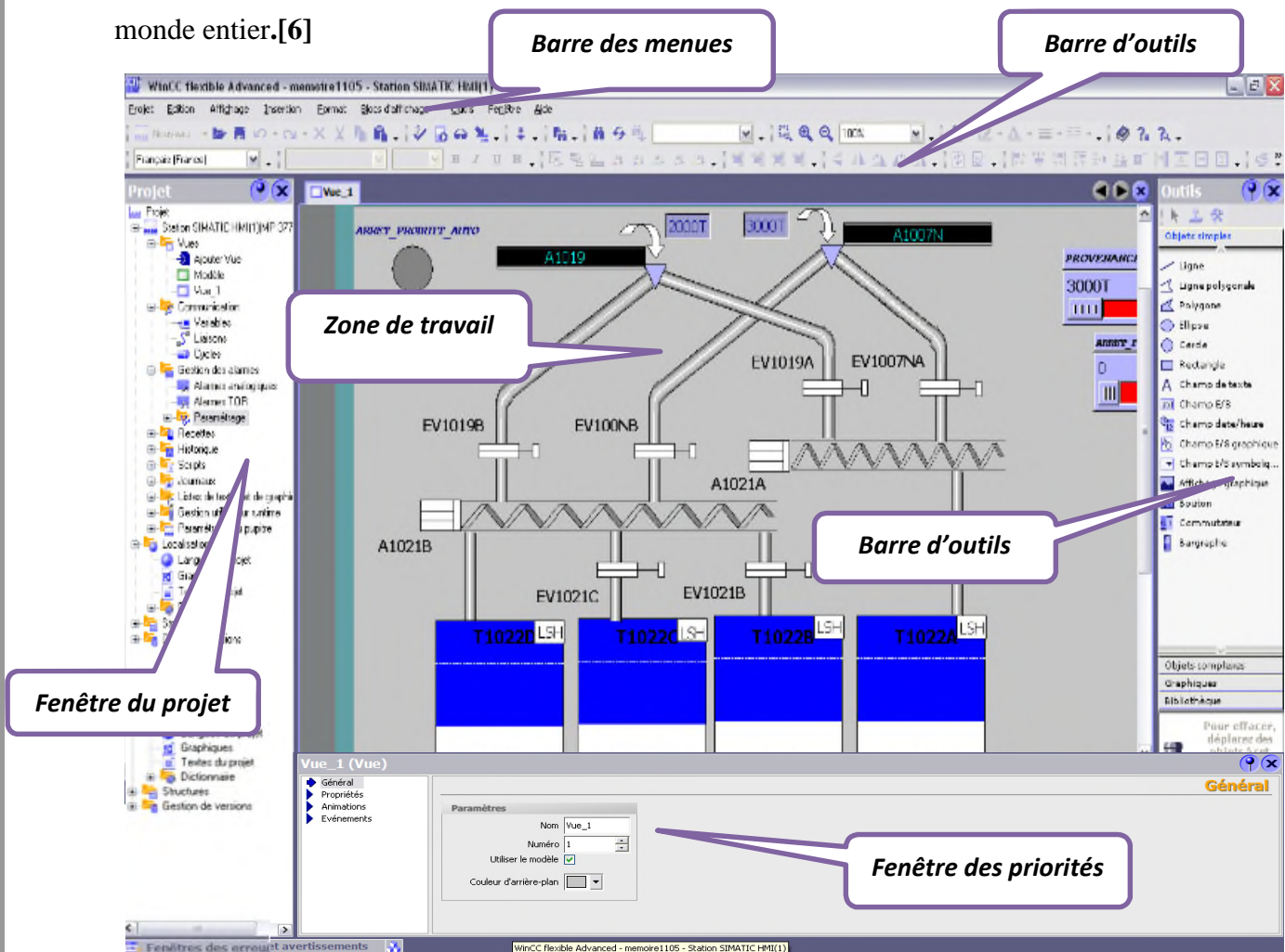


Figure 1.12 : Aperçu de la fenêtre WINCC flexible

L'environnement de travail de WinCC flexible se compose de plusieurs éléments, certains de ces éléments sont liés à des éditeurs particuliers et uniquement visibles lorsque cet éditeur est activé, il met à disposition un éditeur spécifique pour chaque tâche de configuration.

On peut configurer l'interface utilisateur graphique d'un pupitre opérateur avec l'éditeur «Vue».

### **I.2.11. WinCC flexible Runtime :**

WinCC flexible Runtime est le logiciel de visualisation de processus, dans Runtime, on exécute le projet en mode processus.

#### **I.2.11.1 Principe**

Au Runtime, l'opérateur peut réaliser le contrôle-commande du processus, les tâches suivantes sont alors exécutées:

- 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 événements d'alarme.

#### **I.2.11.2. Variable :**

On distingue deux types de variables, les variables externes et les variables internes :

- Les variables externes permettent de communiquer et d'échanger des données entre les composants d'un processus automatisé, entre un pupitre opérateur et un automate.
- Les variables internes ne possèdent aucun lien avec l'automate, elles sont enregistrées dans la mémoire du pupitre.

### **I.2.12. Domaines d'emploi des automates :**

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

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

### **I.3. Conclusion :**

Ce chapitre est séparé en deux parties, mais nous allons nous concentrer beaucoup plus sur la seconde partie, nous avons élaboré dans cette dernière la structure modulaire d'un automate programmable industrielle, ainsi que son architecture interne, et surtout comment choisir l'automate.

Cette étude est focalisée sur l'automate S7-300, en mettant en avant ses caractéristiques techniques pour une meilleure exploitation pendant sa programmation.

Et aussi, de cité les logiciels utilisés pour la programmation et la supervision qui seront l'objet du chapitre III.

*Encadré par : Mr. Hassen RECHAME*

*Mr. Hocine LEHOUCHE*

---

## *Chapitre 02:*

*Description **Du** **P**rocessus **Et***

*Identification **Du** **M**atériel **Utilisé***

---

### **II.1. Introduction :**

Les évolutions méthodologiques et technologiques des dispositifs constituant les systèmes automatisés ainsi que la complexité de plus en plus grande des applications, conduisent à multiplier les prises de mesures.

En effet, ces processus ont besoin pour contrôler, valider, commander leur actions, de connaître à tout instant un grand nombre d'informations internes nécessaires à leur fonctionnement, ses informations sont obtenues par mesure directes ou par déduction (technique d'observation), la chaîne d'instrumentation doit donc prendre en compte les spécificités du processus physique mesuré.

Dans ce chapitre nous allons décrire le processus élaboré, et l'appareillage de l'installation de la section ensachage.

### **II.2. Description de la section maturation:**

Ce que l'on nomme habituellement « le sucre » est une « substance de saveur douce », il peut se rapporter à une saveur sucrée que l'on a dit être une des quatre saveurs de base, d'où le plaisir qui lui est associé.

La section maturation se compose principalement de :

- a. ***Circuit de stockage*** : transport du sucre de la section séchage aux silos de maturation.
- b. ***Circuit de dépoussiérage*** : aspiration de la poussière de sucre.
- c. ***Installation de l'air conditionné*** : élimination de l'humidité dans le sucre pour maintenir sa fluidité.
- d. ***Circuit d'ensachage*** : transport du sucre vers les trémies d'ensachage.

### ***Remarque :***

La description suivante concerne les circuits de la raffinerie 2000T, les circuits de la raffinerie sont similaires avec la différence suivante : la raffinerie 3000T possède un seul silo de stockage d'une capacité de 80 000T et donc le nombre de vannes de soutirage et le nombre de transporteurs à bande plus important.

- Notre étude se concentre dans la quatrième section, nous allons donc nous intéresser à cette dernière.

**II.2.1. Circuit ensachage :**

Avant qu'il soit ensaché dans des sacs de 50Kgs et embarquer dans des semi-remorques, le sucre doit arriver dans deux raffineries différentes la 2000T ou 3000T ou il était stocker dans des trémies de 80T, il doit passer dans un circuit complexe qui contient des vannes, des vis, des élévateurs et des moteurs,...etc, pour qu'il soit mis dans quatre trémies qui sont à leurs tour remplis avec une priorité établie, et qui servent comme distributeur dans six quais différents.

Il se compose de :

**a.** Partie convoyage :

- Deux Convoyeurs à bande.
- Elévateur à godets.
- Deux transporteurs à chaines.
- Deux convoyeurs à vis.

**b.** Un aimant permanent : pour éliminer les débris ferromagnétiques.

**c.** Un crible (tamis) : élimine les particules dépassant le diamètre fixé par les normes de production.

**d.** Quatre trémies de stockage qui sont utilisées pour alimenter 6 lignes d'ensachage, leur remplissage est jusqu'ici se fait d'une manière manuelle, l'objectif de ce travaille est d'automatiser cette opération de remplissage.

**e.** Six doseuses : elles permettent de préparer des doses de 50 kg, elles sont composées principalement de pesants, unités de traitement et de casques (admission et vidange) pilotés par des vérins pneumatiques.

**f.** Couseuses : machines pour réaliser la fermeture des sacs par couture.

**g.** En camionneuses : permettent le transport des sacs cousus vers les camions.

La section ensachage est décrite dans la figure 2.1 qui suit :



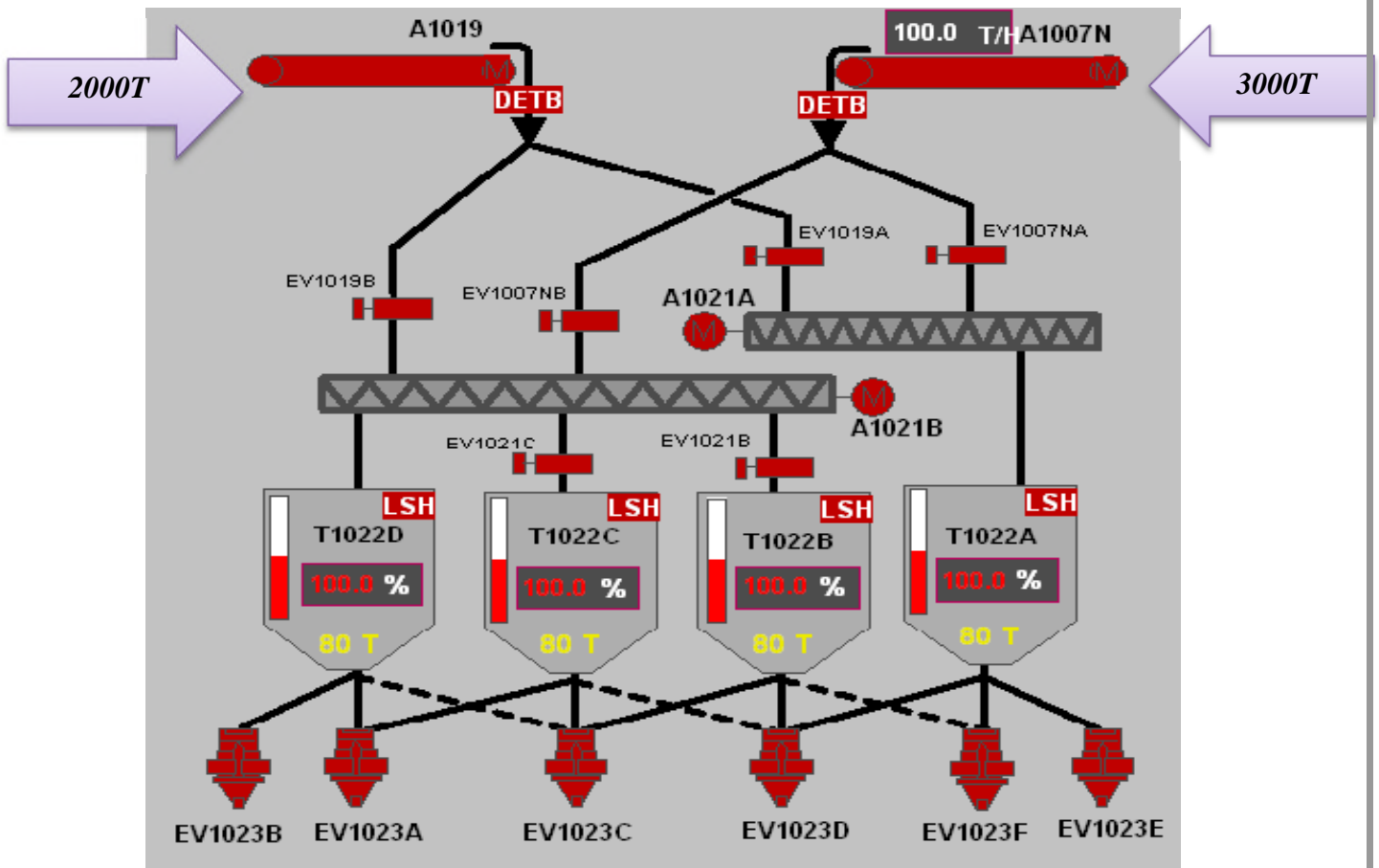


Figure 2.1: Section ensachage

- ❖ Pour ce qui suit, on va décrire les principaux éléments constituant le système.

### II.2.1. Description des équipements du système existant :

Notre système comporte principalement les éléments suivant : des capteurs, six (06) vannes, deux (02) moteurs asynchrones, deux (02) vis, quatre (04) trémies de capacité de chaque une d'elles 80Tonnes.

#### II.2.1.1. Capteurs :

Un capteur transforme une grandeur physique en une grandeur normée, généralement électrique, qui peut être interprétée par un dispositif de contrôle/commande, il produit une information qui permet au système automatisé de déclencher des actions.

Pour contrôler un système, il nous faut des consignes et quelque règle qu'il faut respecter et ce dernier consiste à satisfaire la consigne du programmeur, et pour arriver à cette satisfaction il faut intégrer dans notre programme des capteurs de tout genre afin d'arriver à cette satisfaction de chacun.

Les capteurs se divisent en deux groupes : les capteurs logiques (Le thermostat, Le pressostat, capteurs de position, Le capteur de rotation,...etc), et les capteurs analogiques (Le capteur de température, Capteurs de couple, Le débit mètre, Le capteur de pression,... etc). [7]

**A. Capteur logique :**

Ce type de capteur permet de détecter un événement ou un objet lié au fonctionnement du système technique.

Le signal électrique en sortie de ce capteur est de type logique, niveau logique (0) ou niveau logique (1).

**A.1. Capteur LSH (Locality Sensitive Hashing): [8]**

Le capteur LSH est un capteur logique(TOR), il détecte un seuil pouvant délivrer un contact ouvert au repos (normalement ouvert) ou un contact fermé au repos (normalement fermé).

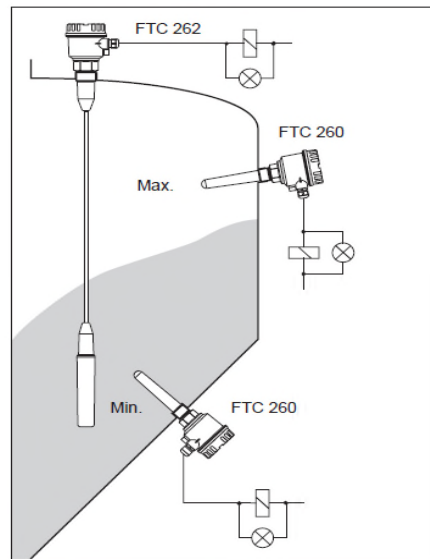
(Voir l'annexe 07).



**Figure 2.2 : Capteur LSH utilisé dans les trémies**

▪ **Principe de fonctionnement :**

Une plaque métallique à l'extrémité de la sonde, à l'intérieur de l'isolation, et l'environnement (par exemple : les parois du silo, comme dans notre système), forment les deux armatures du condensateur, si la sonde est découverte ou recouverte par le produit en vrac, la capacité change, et le capteur commute.



**Figure 2.3: Emplacement capteur LSH dans des silos contenant des solides**

**Remarque :**

Les détecteurs de niveau peuvent être utilisés comme détecteur de bourrage, comme c'est le cas dans notre système.

***A.2. Détecteur de bourrage :***

Les détecteurs de bourrage sont des instruments simples et économiques pour le contrôle des niveaux de produits en vrac dans des silos par mise en marche ou arrêt des systèmes d'alimentation.

Ils sont utilisés dans le contrôle des niveaux minimum et maximum de produit en vrac, poussiéreux, granuleux ou poudreux. [8]

▪ **Principe de fonctionnement :**

Le montage d'un seul détecteur peut suffire dans le cas où l'on ne désire obtenir qu'une seule indication (niveau minimum ou maximum).

Dans ce cas, le palpeur allume un voyant ou déclenche un système sonore ou tout autre dispositif électrique.

Ce capteur, on peut l'utiliser en haut du silo ou tout à fait en bas, ou bien comme dans notre système, à l'extrémité du convoyeur.

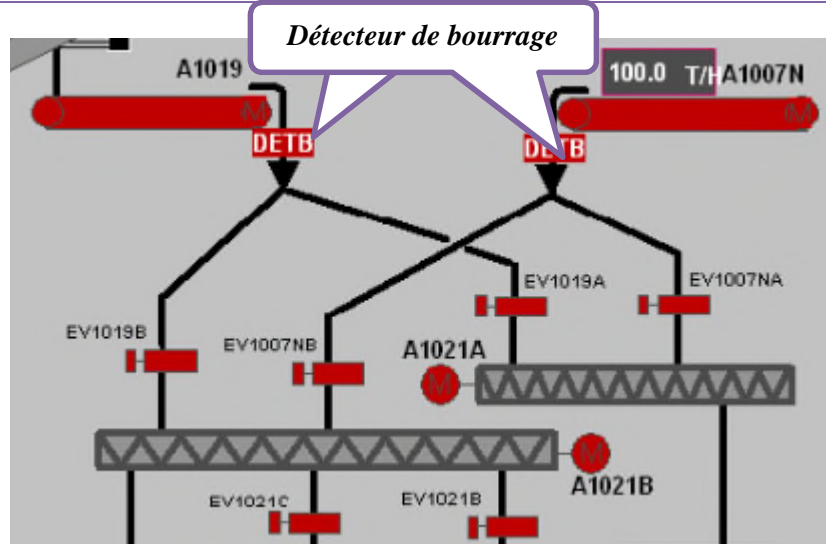


Figure 2.4 : Emplacement du détecteur de bourrage dans notre circuit

### A.3. Détecteur de fin de course:

Un contact de fin de course est un dispositif électromécanique utilisé pour déterminer le positionnement et les applications du contrôle en mécanique et en ingénierie des systèmes.

Il est utilisé pour détecter aisément la présence ou l'absence d'objets dans les zones où le contact physique est autorisé, il peut être monté sur un actionneur pneumatique muni ou non d'un positionneur pneumatique ou électropneumatique, il peut aussi être utilisé pour des valves où l'extension de la tige de robinet déclenche le contact de fin de course, indiquant une position ouverte ou fermée comme dans notre cas, il est monté sur les vannes guillotine.

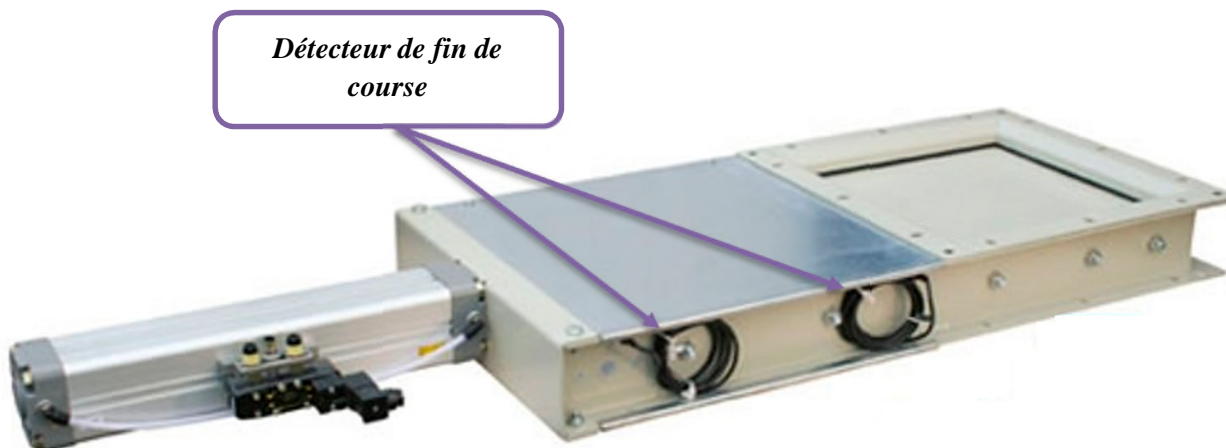


Figure 2.5 : Détecteur de fin de course montée sur la vanne guillotine

**B. Capteur analogique :**

La sortie du capteur analogique peut prendre n'importe quelle valeur entre deux extrêmes. A chaque valeur de la grandeur physique doit correspondre une valeur en sortie.

Une variation de la grandeur physique d'entrée du capteur produit une variation de la caractéristique électrique du capteur (courant, tension, fréquence, valeur moyenne, rapport cyclique,...).

**B.1. Détecteur de niveau :**

Le capteur utilisé pour ce système est un capteur analogique, détecteur de niveau compact, mise en service sans étalonnage également pour zones à poussières inflammables, il est conçu pour la détection de niveau dans les solides en vrac légers. (Voir l'annexe 06)



**Figure 2.6 : Capteur de niveau**

**Principe de fonctionnement :**

C'est un transmetteur utilisant le principe de la mesure du temps de parcours, la distance du point de référence à la surface du produit est mesurée.

Des impulsions haute fréquence sont émises et guidées le long d'une sonde, elles sont réfléchies par la surface du produit, captées par l'unité d'exploitation et converties en information de niveau.

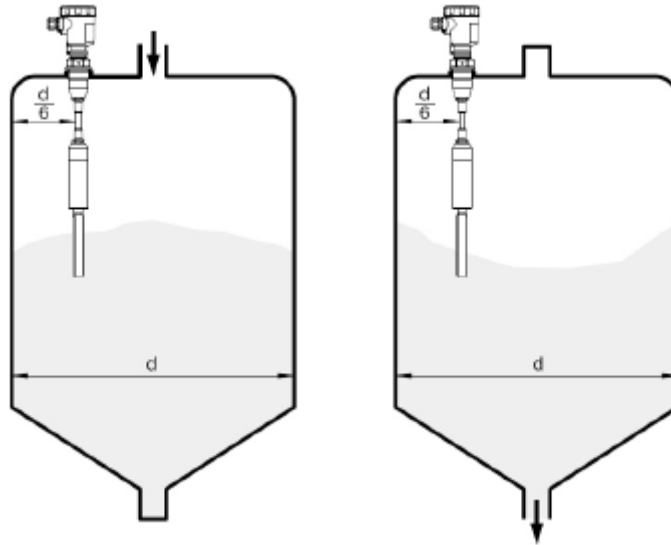


Figure 2.7 : Emplacement du capteur de niveau dans les trémies

### B1.1. Paramètre des capteurs : [9]

Comme beaucoup de composants ou ensembles électroniques, il existe un certain nombre de paramètres électriques communs à tous les capteurs, ils caractérisent les performances et donc leurs utilisations dans un environnement donné.

a) **Précision** : La précision d'un capteur est caractérisée par l'incertitude absolue obtenue sur la grandeur électrique obtenue en sortie du capteur. Elle s'exprime en fraction de la grandeur physique mesurée.

b) **Sensibilité** : Ce paramètre caractérise l'aptitude du capteur à détecter la plus petite variation de la grandeur à mesurer.

c) **Étendue de mesure** : cette caractéristique donne la plage de fonctionnement du capteur pour la grandeur à mesurer. Elle est souvent notée E.M.

d) **Linéarité** : Un capteur est dit linéaire s'il présente la même sensibilité sur toute l'étendue de sa plage d'emploi.

e) **Fidélité** : Un capteur est dit fidèle si le signal qu'il délivre ne varie pas pour une série de mesures concernant la même valeur de la grandeur d'entrée.

### **II.2.1.2. Convoyeur :**

Un convoyeur est un mécanisme ou machine qui permet le transport d'une charge isolée (cartons, bacs, sacs, ...) ou de produit en vrac (terre, poudre, aliments, sucre...) d'un point A à un point B.

Dans notre cas, le convoyeur est une vis sans fin, alimentée par un moteur asynchrone, qui sert à transporter le sucre pour qu'il soit mis dans les trémies.

#### **A. Vis sans fin :**

Un convoyeur à vis est un système de convoyage qui fonctionne à l'aide d'une vis hélicoïdale insérée dans un tube pour le transport des matériaux granulés, elle est spécialement conçue pour le transport et le déchargement des produits solides (sucre), du circuit jusqu'au silo de stockage, elles sont judicieusement dimensionnées en fonction des exigences d'utilisation.

Basé sur le système d'Archimède, ce système est utilisé dans différentes industries modernes pour le transport efficace des matériaux en vrac/semi-solides à l'horizontal ou sur un angle incliné. (Voir l'annexe 08)



**Figure 2.8 : Une vis**

#### ▪ **Principe :**

Un convoyeur à vis est souvent utilisé pour les matériaux en vrac ou les matériaux semi-solides tels que les copeaux, les agrégats, les graines des céréales, les aliments pour animaux, etc.

#### **Remarque :**

La vis est actionnée par un moteur asynchrone 220/380V.

**B. Les moteurs asynchrones :**

Ces moteurs sont robustes, ils sont intéressants, lorsque la vitesse du dispositif à entraîner n'a pas à être rigoureusement constante

La vitesse de la rotation du moteurs n'est pas la même que la vitesse du champ électrique crée (champ statorique) les deux vitesses sont asynchrone.



*Figure 2.9 : Moteur asynchrone*

**II.2.1.3. Les actionneurs :**

L'appellation "actionneur" désigne tous les systèmes capables d'agir physiquement sur un objet.

Placés dans la partie opérative d'un système automatisé, l'actionneur est un élément qui reçoit de l'énergie (électricité, vapeur d'eau...) et la transforme en une autre énergie capable de produire un mouvement, de la chaleur, de la lumière, un bruit.

**A. Un servomoteur :**

Un servomoteur électrique permet de piloter et de contrôler l'ouverture et la fermeture des vannes et industrielles. Selon les applications il peut gérer l'ouverture ou la fermeture totale d'une vanne, ou réguler un flux de façon plus ou moins intense et précise.

**B. Vérin :**

Un vérin pneumatique ou hydraulique sert à créer un mouvement mécanique, et consiste en un tube cylindrique (le cylindre) dans lequel une pièce mobile (le piston) sépare le volume du cylindre en deux chambres isolées l'une de l'autre, un ou plusieurs orifices permettent d'introduire ou d'évacuer un fluide dans l'une ou l'autre des chambres et ainsi déplacer le piston.

**B.1. Vérin simple effet (VSE) :**

Un vérin simple effet ne travaille que dans un sens (souvent, le sens de sortie de la tige), l'arrivée de la pression ne se fait que sur un seul orifice d'alimentation ce qui entraîne le piston dans un seul sens, son retour s'effectuant sous l'action d'un ressort ou d'une force extérieure (fréquent en hydraulique).



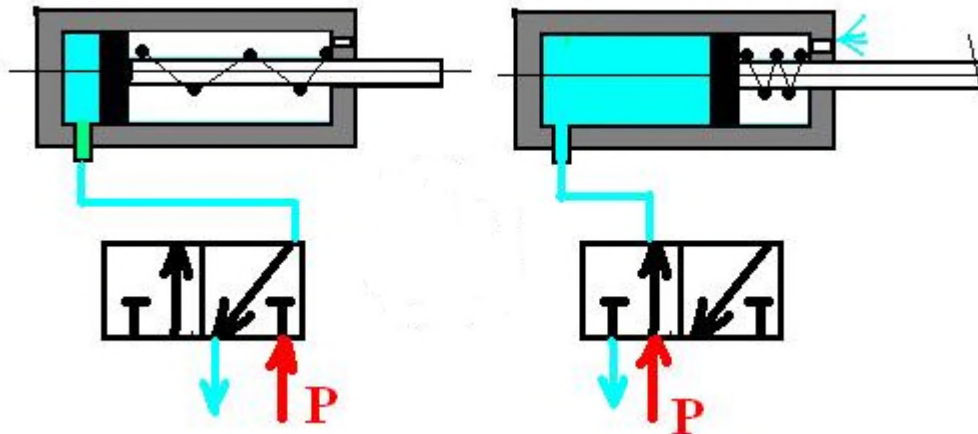


Figure 2.10 : Vérin simple effet avec son distributeur 3/2

### B.2. Vérin double effet (VDE) :

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

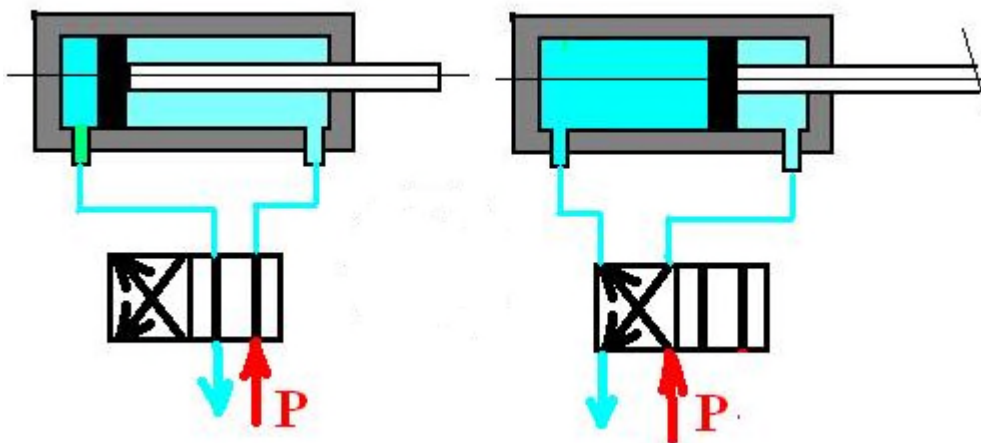


Figure 2.11: Vérin double effet avec son distributeur 4/2

#### Remarque :

Les vérins sont souvent équipés d'amortisseurs de fin-de-course qui évitent les chocs du piston.

### C. Distributeur pneumatique : [1]

Les distributeurs pneumatiques sont des éléments de la chaîne d'énergie, ils distribuent de l'air comprimé dans des canalisations qui aboutit aux chambres des vérins, aux actionneurs pneumatiques (vérins, générateurs de vide, moteurs à palettes...) à partir d'un signal de commande.

#### *Emplacement du distributeur*



Figure 2.12 : Distributeur

Figure 2.13 : Emplacement du distributeur sur la vanne guillotine

### D. Les vannes :

Elles sont des dispositifs permettant de régler le débit d'un fluide dans un ouvrage à écoulement libre, dans une conduite ou une canalisation.

#### C.1. Les vannes pneumatiques TOR:

La vanne tout ou rien que l'on trouve dans notre installation est une vanne pneumatique, c'est donc une pression d'air comprimé qui lui permet de bloquer ou de libérer la canalisation.

L'air comprimé est admis ou non dans le servomoteur de la vanne procédé grâce à une électrovanne dite vanne de commande ou pilote.

Cette vanne ne peut prendre que deux positions, ouverte ou fermée. On traduit en termes de course du clapet 0% ou 100% d'ouverture. Son rôle principal est d'assurer des fonctions de sécurités et d'utilités, on peut aussi dans certains cas les utiliser pour réguler une grandeur physique (Régulation discontinue). Lorsqu'elles sont fermées, ces vannes sont étanches.[10]

#### ▪ Principe :

L'énergie pneumatique est distribuée à l'actionneur sur ordre de l'unité de traitement, cette énergie pneumatique est transformée en énergie mécanique afin de mouvoir les effecteurs.

#### C.2. Catégories de vannes :

On peut ranger les vannes dans différentes catégories, selon leurs caractéristiques :

- *par la taille* : depuis le simple petit robinet jusqu'à la très grosse vanne de régulation du débit d'arrivée d'eau dans une centrale hydroélectrique.
- *la fonction* : vanne régulatrice de débit, vanne de purge, vanne d'arrêt, ...etc.

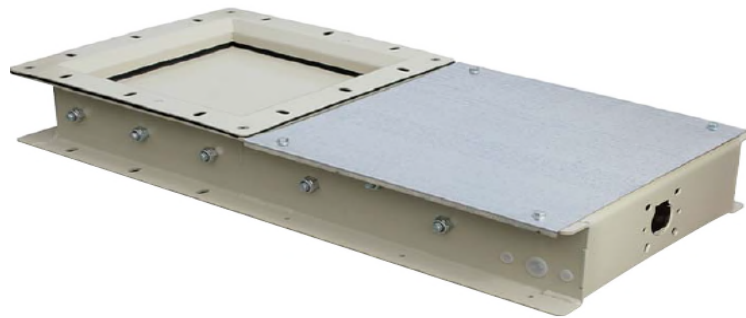
- **par le principe d'obstacle au débit employé** : vanne papillon, vanne à opercule, vanne à boisseau conique, vanne à boisseau sphérique, vanne trois-voies, vanne guillotine, ... etc.
- **par le système de commande** : manuelle, électrique.

### **C.3.1. Vanne guillotine : [10]**

La vanne guillotine est une vanne à montage entre brides étanche d'un côté, utilisée généralement pour des applications industrielles.

Elle est surtout utilisée dans l'industrie du papier et de la cellulose, l'industrie chimique, minière, les centrales électriques et l'industrie agro-alimentaire.

La vanne guillotine trouve cependant toujours plus d'emploi dans les méthaniseurs, les stations de traitement des eaux usées, l'industrie sucrière, la sidérurgie et l'industrie de l'alimentation animale. (Voir l'annexe 05).



**Figure 2.14 : Vanne guillotine**

#### ▪ **Principe : [10]**

Fonctionnement en double effet : un vérin pneumatique de type piston-crémaillère entraîne l'axe de la vanne dont le mouvement est un  $\frac{1}{4}$  de tour. Un distributeur envoie un signal pneumatique alternativement dans l'une des deux chambres du vérin permettant l'ouverture ou la fermeture de la vanne.

Fonctionnement en simple effet : dans ce cas, le vérin pneumatique est équipé de ressorts de compression du piston. Le signal pneumatique n'exécute qu'un seul des deux mouvements d'ouverture ou de fermeture, le second étant opéré par la pression des ressorts. Ce type de fonctionnement permet de prévoir une position de sécurité par manque de courant.

#### **Remarque :**

La vanne de notre installation est une vanne guillotine, fonctionnement en double effet.

### **C.3. La commande manuelle :**

La plus part des vannes de régulations peuvent être équipées d'un dispositif de commande manuelle, permettant de régler le débit en l'absence du signale de commande ou comme dans notre système limiter la course d'ouverture ou de fermeture, ce diapositive n'est à prévoir qu'en cas de nécessité absolue, l'utilisation d'une commande manuelle.

#### **II.2.1.4. Les automates SIEMENS :**

L'automate SIEMENS constitue une plate-forme d'automatisation universelle pour toutes les applications, Il offre une gamme échelonnée de CPU.

Parmi les avantages de cet automate, nous relevons, entre autres, sa construction compacte et modulaire ainsi que la large gamme de CPU adaptée à toutes les demandes de performances, il est répandu beaucoup plus dans les industries automobiles.

Il a un câblage pratique et fiable lui permet d'être installé en toute circonstance avec la solution.

La figure suivante nous monte un automate programmable SIEMENS :



***Figure 2.15: Automate programmable SIEMENS***

#### ***Remarque :***

L'automate utilisé dans la partie ensachage à l'entreprise CEVITAL est un automate ABB, mais pour notre projet on prendra un automate SIEMENS S7-300 l'étudier et le programmer à causes des possibilités de simulation et supervision qu'il offre.

### **II.2.1.5. Présentation de l'automate SIEMENS S7-300 : [11]**

L'API S7-300 de gamme SIEMENS, est un Mini-automate modulaire pour les applications de milieu de gamme, avec possibilité d'extension jusqu'à 32 modules d'entrées/sorties, et une mise en réseau par l'interface multipoint (MPI), PROFIBUS et industriel Ethernet. (Voir l'annexe 04)

Les caractéristiques principales des automates programmables industriels (API) Siemens S7- 300 sont les suivantes :

Il s'agit d'un matériel multiprocesseur :

- un processeur logique (bit processor).
- un processeur pour les opérations arithmétiques (word processor).
- un processeur dédié à la régulation de type PID.
- un processeur dédié à la gestion des communications.

Le logiciel Siemens S7 permet une programmation multi langage, c'est-à-dire qu'il peut être programmé dans plusieurs langages différents, qui peuvent être même mélangés dans un même programme (mais pas dans une même sous-routine) :

- Liste d'instructions ou Instruction List (IL).
- Langage à contacts ou Ladder diagramm (CONT).
- Logigramme ou Functional Block (LOG).

Le mode séquentiel est accessible :

- soit en utilisation une programmation en GRAFCET directement
- soit en créant une séquence d'exécution. Blocs d'organisation

### **II.3. Conclusion :**

Tous ce qui est décrit dans ce chapitre fait appel à des notions de la physique dans un plan très large et nous ne monterons que la partie utilisation technologique de ces concepts.

Ce chapitre a pour vocation de faire un état des lieux de la connaissance du matériel qui consiste la station ensachage.

Nous avons montré dans ce chapitre certain capteurs, l'automate siemens, vérin , vanne et moteurs qui sont utilisé dans notre application, et en générale dans l'industrie moderne.

*Encadré par : Mr. Hassen RECHAME*

*Mr. Hocine LEHOUCHE*

---

## *Chapitre 03:*

# *Programmation **E**t **S**upervision*

---

**III.1. Introduction :**

Comme tout système à microprocesseur les API fonctionnent sur la base d'un programme qui lui définit les tâches à exécuter.

Pour piloter le circuit, nous allons réaliser un programme que nous allons intégrer l'automate, pour se faire, nous utiliserons le logiciel de conception de programme pour des systèmes d'automatisations SIMATIC STEP7.

Dans ce chapitre, nous allons décrire l'implantation du programme d'automatisation élaboré à partir de l'analyse fonctionnelle, ainsi que la supervision.[13]

**III.2. Programmation :**

Pour pouvoir commencer toute programmation il nous faut un cahier de charges, qui résume toutes les tâches à accomplir tout au long du programme.[14]

**III.2.1. Le cahier de charges :**

Afin de mieux contrôler un système il nous faut son cahier de charges, qui contient des consignes et quelque règle qu'il faut respecter et ce dernier consiste à satisfaire la consigne du programmeur :

- Le sucre peut arriver des deux stations la 2000T ou la 3000T, mais pour le remplissage on doit choisir la raffinerie par un bouton de sélection.
- Le remplissage des trémies se fait par l'ordre suivant : TL1022D, TL1022A, TL1022B, TL1022C
- Les vannes EV1019B et EV1007NB ne doivent pas s'ouvrir au même temps.
- Les vannes EV1019A et EV1007NA ne doivent pas s'ouvrir au même temps.
- S'il y'a arrêt des moteur A1021A les vannes EV1019A & EV1007NA ne s'ouvrent pas, et il y'aura arrêt soutirage (arrêt moteur A1007N & A1019).
- S'il y'a arrêt des moteur A1021B les vannes EV1019B & EV1007NB ne s'ouvrent pas, et il y'aura arrêt soutirage (arrêt moteur A1007N & A1019).
- La vanne EV1019B :

Ouverture: si le sucre arrive de la 2000T &

$LT1022D \leq 20\%$ ,  $\forall$  le niveau des autres trémies OU

$LT1022D \geq 20\%$ , &  $LT1022A \geq 90\%$ , &  $LT1022B \leq 70\%$ , OU

$LT1022D \geq 20\%$ , &  $LT1022A \geq 90\%$ , &  $LT1022B \leq 90\%$ , &  $LT1022C \leq 70\%$ .

OU il y 'a une demande de produit dans T1022D OU T1022B OU bien LT1022

Fermeture :

Le sucre arrive de la 3000T, OU

$LT1022D \geq 20\%$ , &  $LT1022B \geq 90\%$ , &  $LT1022C \geq 90\%$  &  $LT1022A \leq 70\%$ .

- La vanne EV1019A :

Ouverture: le sucre arrive de la 2000T &  
 $LT1022D \geq 20\%$ , &  $LT1022A \leq 70\%$  OU  
Il y'a une demande de produit T1022A.

Fermeture :

Le sucre arrive de la 3000T, OU  
 $LT1022A \geq 90\%$ .

- La vanne EV1021B :

Ouverture : le sucre arrive de la 2000T OU de la 3000T &  
 $LT1022B \leq 70\%$ .

Fermeture:

$LT1022B \geq 90\%$ .

- La vanne EV1021C :

Ouverture: le sucre arrive de la 2000T OU de la 3000T &  
 $LT1022BC \leq 70\%$

Fermeture:

$LT1022C \geq 90\%$ .

- La vanne EV1007NB :

Ouverture: le sucre arrive de la 3000T &  
 $LT1022D \leq 20\%$ ,  $\forall$  le niveau des autres trémies OU  
 $LT1022D \geq 20\%$ , &  $LT1022A \geq 90\%$ , &  $LT1022B \leq 70\%$ , OU  
 $LT1022D \geq 20\%$ , &  $LT1022A \geq 90\%$ , &  $LT1022B \leq 90\%$ , &  $LT1022C \leq 70\%$ .

Fermeture :

Le sucre arrive de la 2000T, OU  
 $LT1022D \geq 20\%$ , &  $LT1022B \geq 90\%$ , &  $LT1022C \geq 90\%$  &  $LT1022A \leq 70\%$ .

- La vanne EV1007NA :

Ouverture: le sucre arrive de la 3000T &  
 $LT1022D \geq 20\%$ , &  $LT1022A \leq 70\%$

Fermeture :

Le sucre arrive de la 2000T, OU  
 $LT1022A \geq 90\%$ .



### III.2.2. Principe de la programmation sous STEP7 :

Les fonctions suivantes peuvent être utilisées avec STEP7 pour l'automatisation d'un dispositif :

- Création et la gestion du projet.
- La configuration et le paramétrage du matériel et de la communication.
- La gestion des mnémoniques.
- La création du programme.
- Le chargement du programme dans les systèmes cible.
- Le teste et l'installation d'automatisation.
- Le diagnostic lors de perturbations de l'installation.

### III.2.3. Structure du programme : [11]

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

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

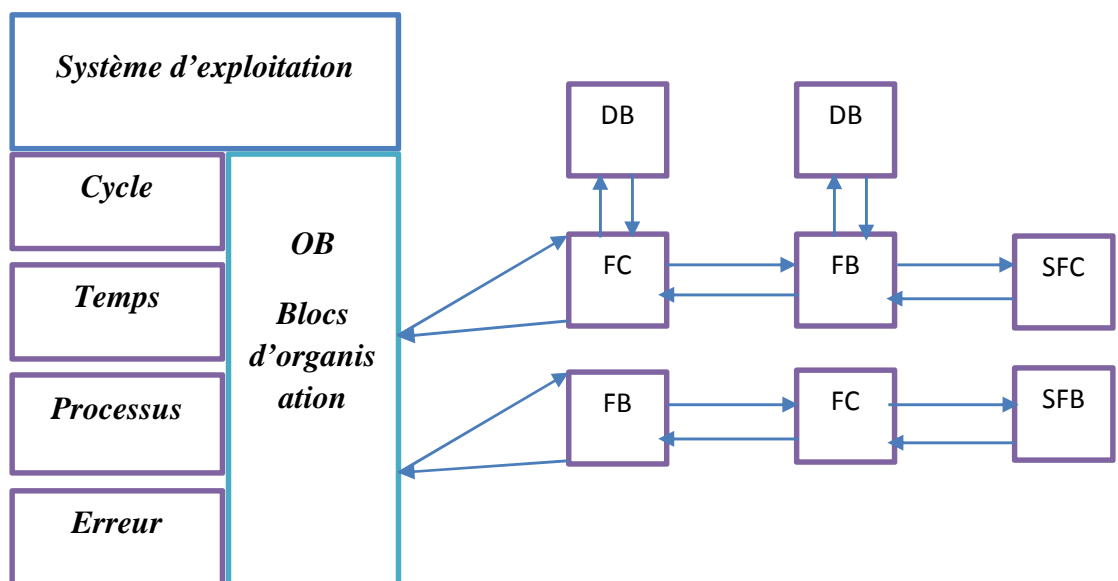


Figure 3.1 : Blocs de programme

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

✚ **Bloc d'organisation :** Les blocs d'organisation (**OB**) constituent l'interface entre le système **OB** d'exploitation et le programme utilisateur. L'ensemble du programme peut être concaténé dans un seul bloc OB1 (programme linéaire) appelé de manière cyclique par le système d'exploitation ou être structuré dans plusieurs blocs (programme structuré).

✚ **Fonction :** Une fonction (**FC**) assure une fonctionnalité spécifique du programme.

**FC, SFC** Les fonctions peuvent être paramétrables. Dans ce cas, des paramètres sont transmis à la fonction lorsqu'elle est appelée. Les fonctions conviennent donc pour la programmation de fonctionnalités récurrentes et complexes, par exemple pour effectuer des calculs.

Les fonctions système (**SFC**) sont des fonctions paramétrables, intégrées au système d'exploitation de la CPU, dont le numéro et la fonctionnalité sont définis de manière fixe. Pour de plus amples informations, se reporter à l'aide en ligne.

✚ **Bloc fonctionnel :** Du point de vue du programme, les blocs fonctionnels s'apparentent aux **FB, SFB** fonctions, mais ils disposent en plus de zones mémoire spécifiques, sous forme de blocs de données d'instance. Les blocs fonctionnels conviennent donc pour la programmation de fonctionnalités récurrentes encore plus complexes, par exemple pour assurer des tâches de régulation.

✚ **Blocs fonctionnels système (SFB) :** sont des blocs fonctionnels paramétrables, intégrés au système d'exploitation de la CPU, dont le numéro et la fonctionnalité sont définis de manière fixe. Pour de plus amples informations, se reporter à l'aide en ligne.

✚ **Blocs de données :** Les blocs de données (**DB**) sont des zones de données du programme **DB** utilisateur.

#### **III.2.4. Les opérandes et types de données autorisés dans la table des mnémoniques :** [12]

Saisir les mnémoniques est très utile, il vaut mieux saisir un programme entièrement en symbole qu'en adressage absolu, c'est beaucoup plus lisible et compréhensible.

Il suffit d'aller dans la table des mnémoniques et y entrer les différents éléments, le nom du symbole, son adresse réel, son type et son commentaire.

Les paramètres peuvent être des types simples tel que : **BOOL, WORD, INT, REAL, ...ect**, ou de types complexes tel que : **TIMER, COUNTER, ... ect**

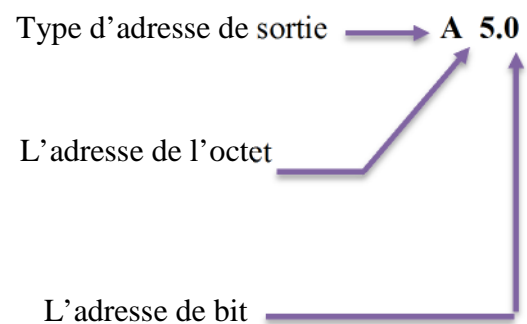
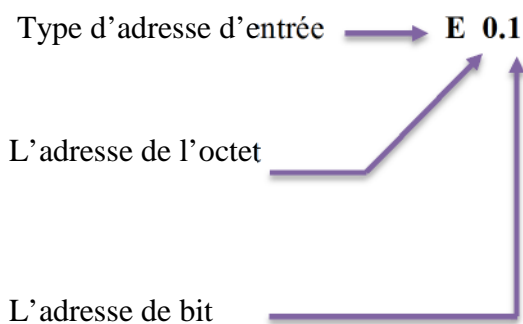
Les différents types de variables sont donnés dans le tableau qui suit, dans le cadre de cette première approche, on fera plus particulièrement attention aux variables suivantes :

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

**Tableau 3.1 : Les différents types de variable**

### III.2.5. L'adressage des modules E/S :

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



**III.2.6. Utilisation du front montant:**

Pour cela, il faut utiliser le bloc POS, il utilise un memento de front et l'entrée sur laquelle on veut détecter le front. [3]

Dans notre programme on l'a utilisé pour le remplissage et le soutirage du sucre dans les trémies, la détection du front montant pour déclencher l'additionneur à incrémenter une valeur désirer.

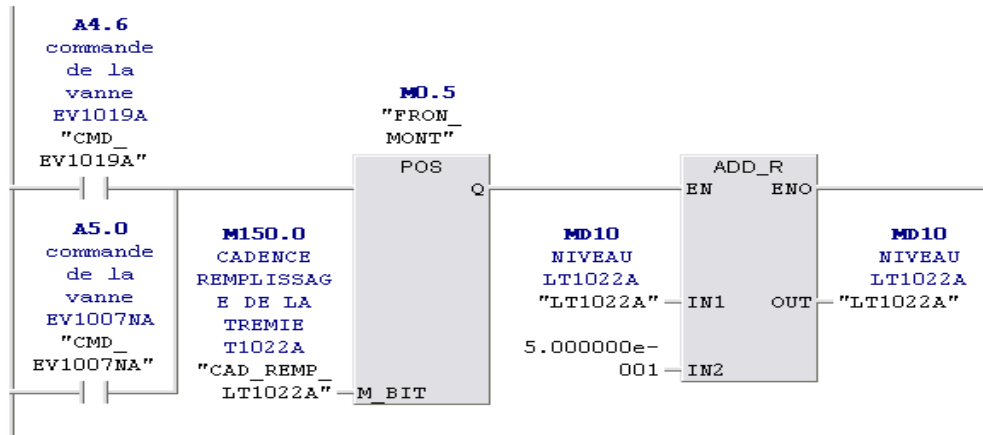


Figure 3.2 : le remplissage de la trémie T1022A

**III.2.7. Utilisation des memento de cadence : [12]**

Dans cette fonction, nous allons insérer des memento de cadence pour la simulation du remplissage et du soutirage de sucre dans les trémies et pour cela, nous incluons des additionneurs pour le remplissage et les soustracteurs pour le sous-titrage.

Le memento de cadence est un octet, chacun des bits de cet octet change d'état suivant une horloge interne.

Bit	7	6	5	4	3	2	1	0
Durée de période (S)	2	1.6	1	0.8	0.5	0.4	0.2	0.1
Fréquence(Hz)	0.5	0.625	1	1.25	2	2.5	5	10

Tableau 3.2 : memento de cadence

Sa configuration se fait comme suite :

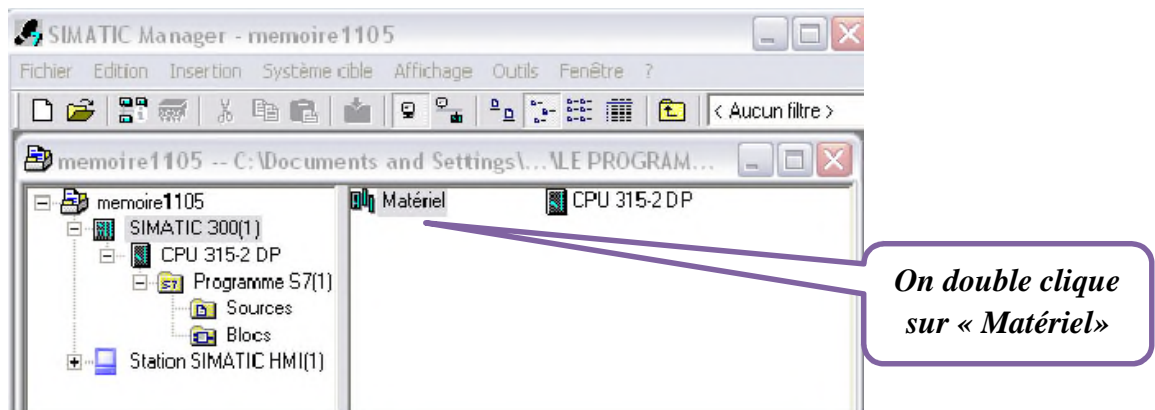


Figure 3.3 : fenêtre SIMATIC pour accéder au matériel de la CPU

- ❖ Ensuite double clic sur l'unité centrale (CPU 315-2 DP), et dans l'ongle « cycle et Memento de cadence», on choisit alors l'adresse du memento :

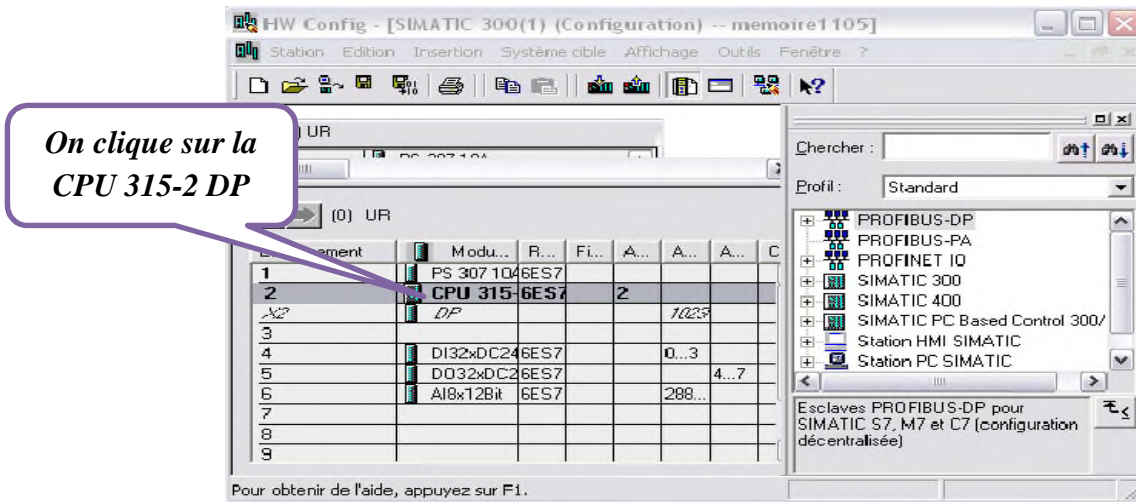


Figure 3.4 : fenêtre SIMATIC pour accéder à la propriété de la CPU

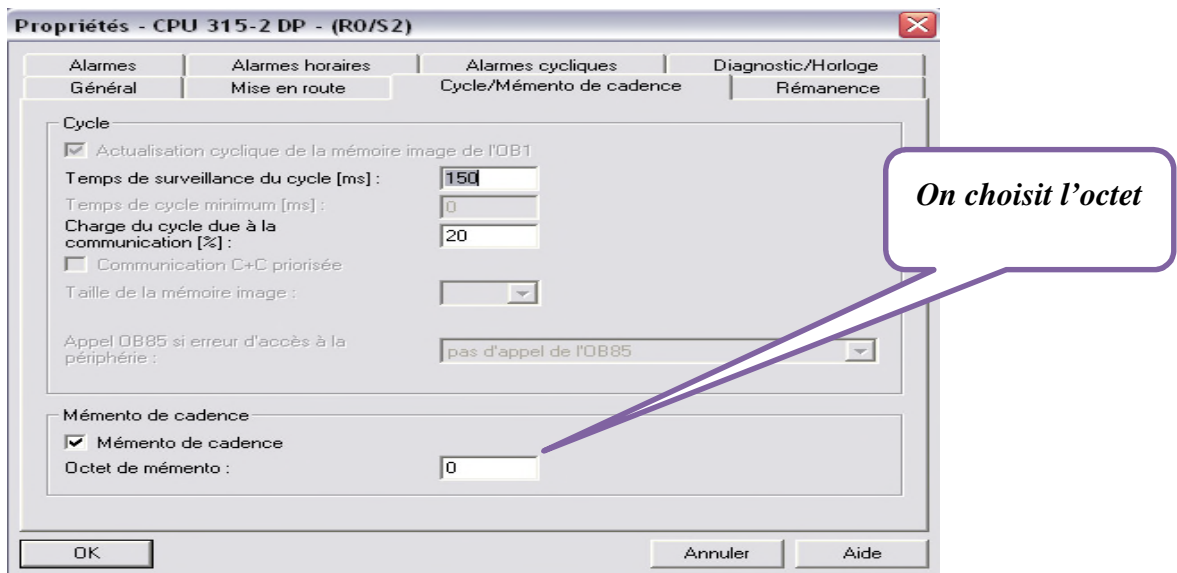


Figure 3.5: Fenêtre SIMATIC pour choisir l'octet de Mémento de cadence

**Remarque :**

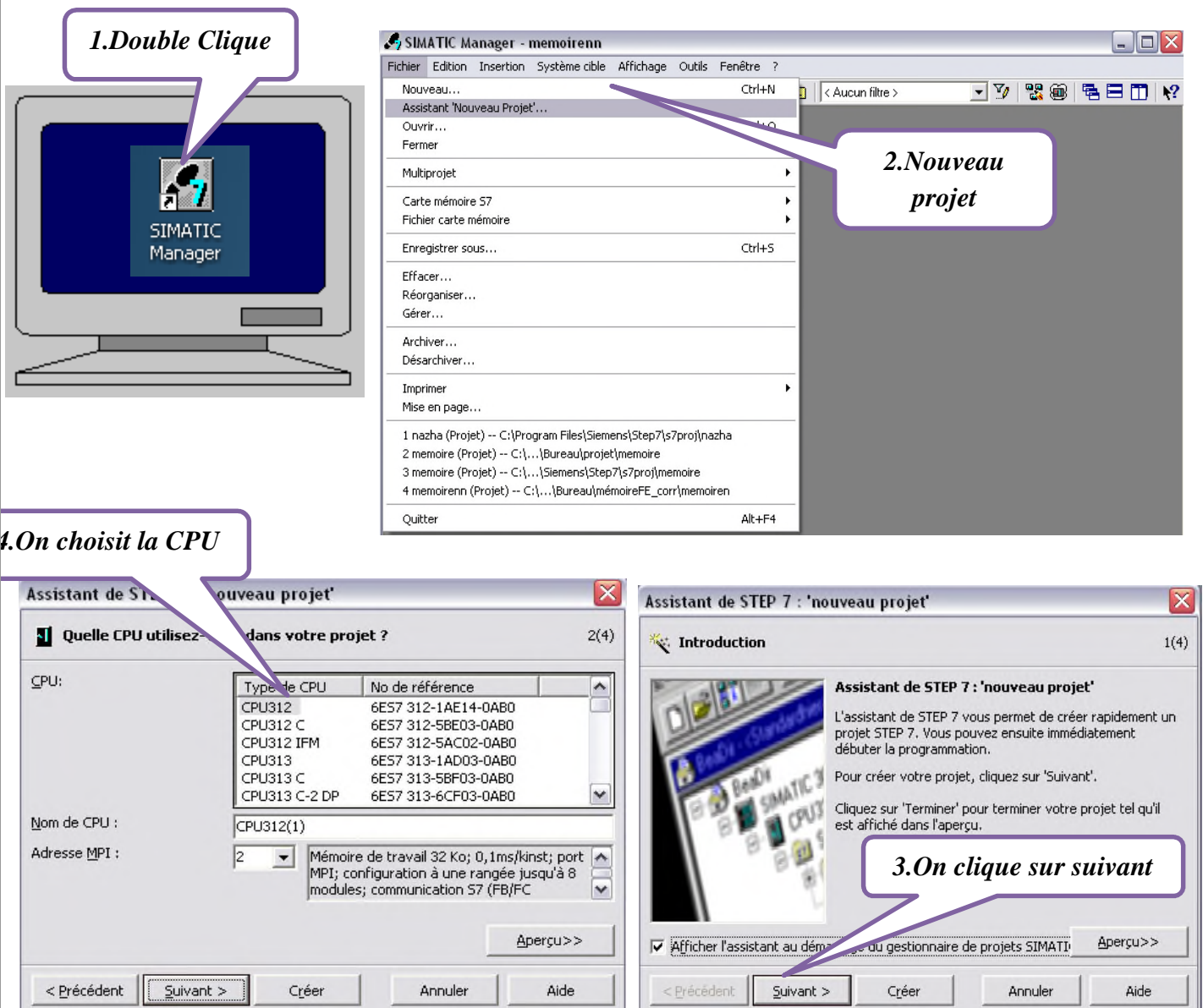
Le bit M 0.5 change toutes les secondes.

**III.3. Création d'un projet STEP7 : [12]**

Une fois Windows démarré, on trouve dans l'interface Windows une icône pour SIMATIC Manager qui permet d'accéder au logiciel STEP 7.

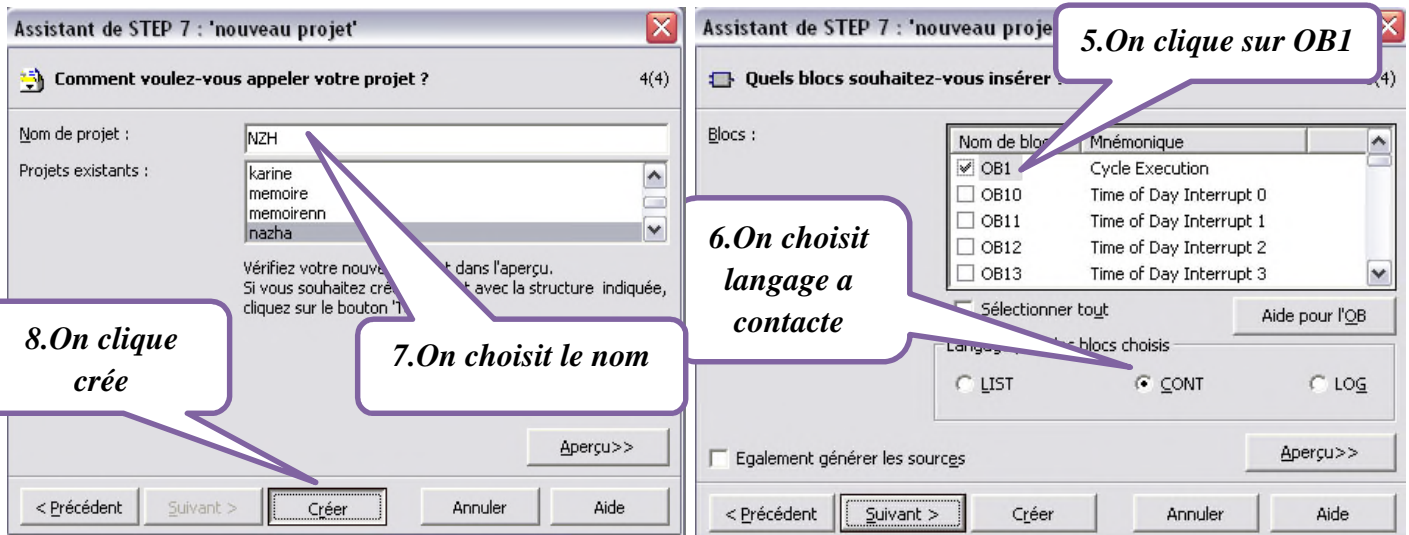
Afin de créer un nouveau projet STEP7 il nous est possible d'utiliser « l'assistant de création de projet » ou bien créer le projet soi-même et le configurer directement, cette dernière est un peu plus complexe mais nous permet aisément de gérer notre projet.

En sélectionnant l'icône SIMATIC Manager on affiche la fenêtre principale pour sélectionner un nouveau projet et le valider comme nous montre les figures suivantes :



Figures 3.6 : Création d'un nouveau projet





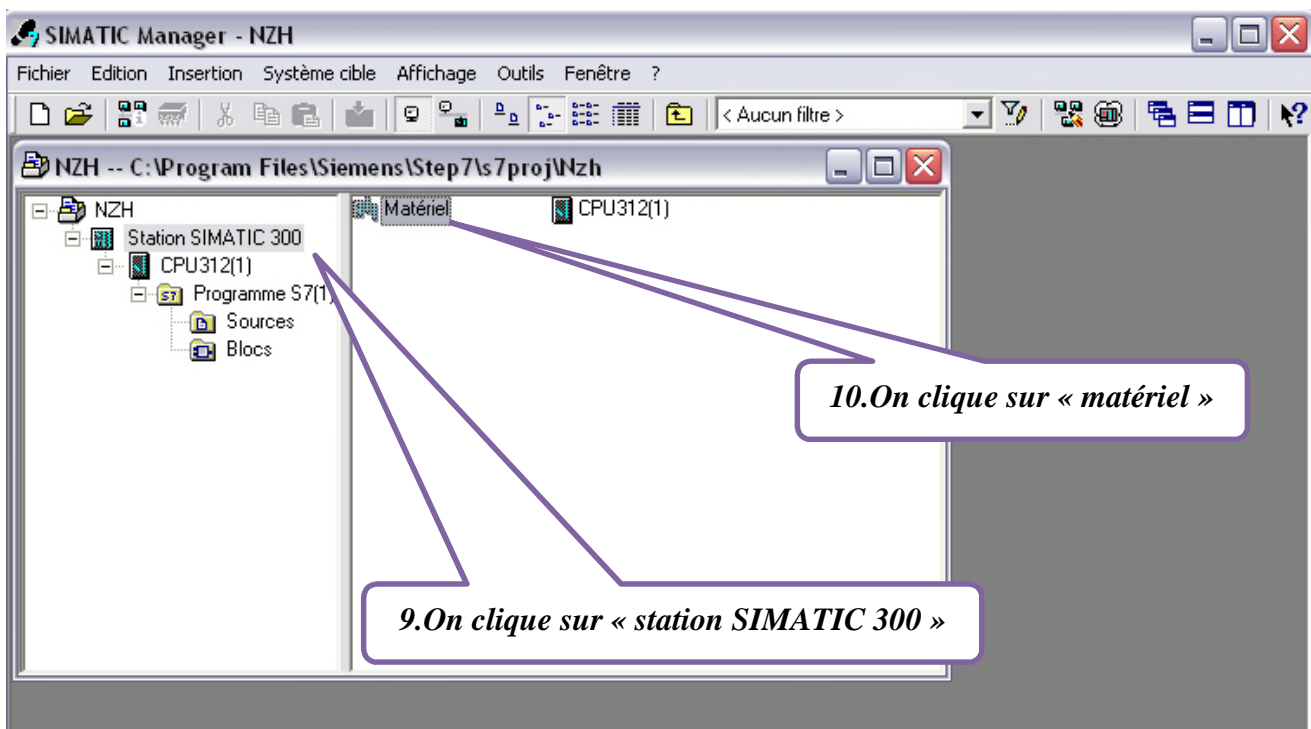
Figures 3.7 : Appellation du nouveau projet

### III.3.1. Configuration du matériel :

Après avoir créé le projet, on doit configurer le matériel, c'est une étape très importante, qui consiste à l'agencement des châssis, des modules et de la périphérie décentralisée.

Les modules sont fournis avec des paramètres définis par défaut en usine, une configuration matérielle est nécessaire pour :

- Les paramètres ou les adresses pré-régler d'un module.
- configurer les liaisons de communication.



Figures 3.8 : Configuration du matériel





- La configuration matérielle étant terminée, un dossier « programme STEP7 » est automatiquement inséré dans le projet, comme indique la figure suivante :

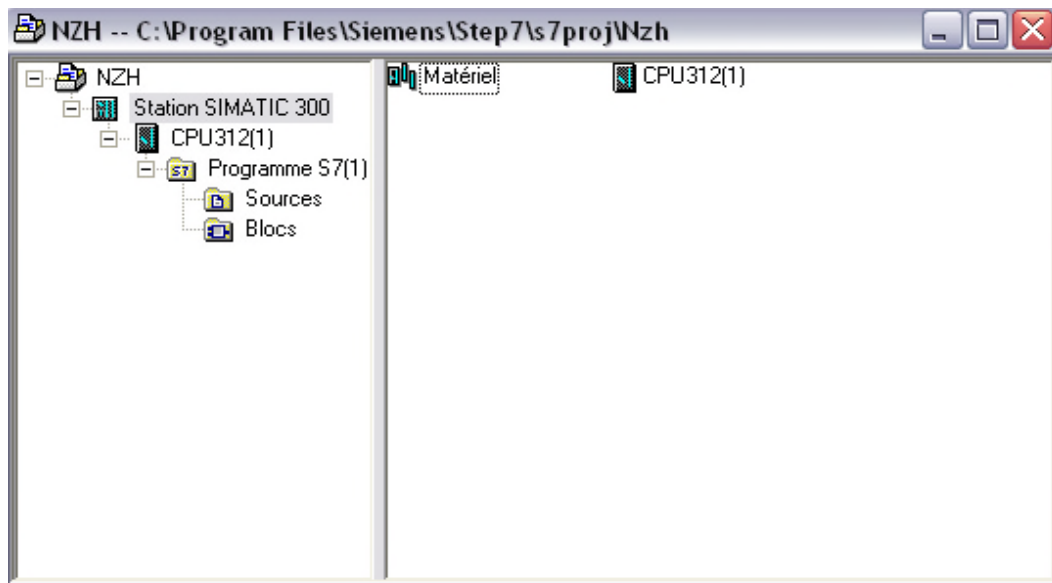


Figure 3.10 : Hiérarchie du programme STEP7

**III.3.2. Création de la table des Mnémonique :**

Une table des mnémoniques vide est automatiquement générée lors de la création d'un programme STEP7, elle se trouve dans le menu : <programme><table des mnémonique.

Dans tout programme, il faut définir la liste des variables qui vont être utilisés lors de la programmation, les mnémoniques sont des noms que l'on peut donner aux variables afin de faciliter la programmation en affectant des noms parlant plus faciles à retenir, ils améliorent la lisibilité du programme et servent également de documentation.

On édite la table des mnémonique en respectant le cahier de charges, pour les entrées et les sorties, la figure suivante présente une partie de la table des mnémonique : (voir annexe 02)

	Etat	Mnémonique	Opérande	Type de d	Commentaire
1		A	VAT 1		
2		ARRET_A1019 & A1007N	A 1.0	BOOL	ARRET DES TAPIS A1007N & A1019A
3		ARRET_PROD	A 7.0	BOOL	ARRET DU PRODUIT
4		ARRT_A1021A	E 5.1	BOOL	ARRET DU MOTEUR A1021A
5		ARRT_A1021B	E 5.3	BOOL	ARRET DU MOTEUR A1021B
6		CAD_LT1022A	M 150.0	BOOL	CADENCE DE LA TREMIE T1022A
7		CAD_LT1022B	M 150.1	BOOL	CADENCE DE LA TREMIE T1022B
8		CAD_LT1022C	M 150.2	BOOL	CADENCE DE LA TREMIE T1022C
9		CAD_LT1022D	M 150.3	BOOL	CADENCE DE LA TREMIE T1022D
10		CAD_VANNES	FC 9	FC 9	CADENCE DES VANNES

Figure 3.11 : Une partie de la table des Mnémoniques du projet

### **III.3.3. Elaboration du programme :**

Après avoir créé notre projet et configuré le matériel, pour commencer de programmer on doit créer des blocs de fonction ou on doit insérer notre langage à contact.

Dans notre programme, on a 10 fonctions (FC1 → FC10), elles sont réparties comme suite:

FC 1 : la mise à l'échelle des niveaux des trémies

FC 2 : la commande de la vanne EV1019B.

FC 3 : la commande de la vanne EV1007NB.

FC 4 : la commande de la vanne EV1019A.

FC 5 : la commande de la vanne EV1021B.

FC 6 : la commande de la vanne EV1021C.

FC 7 : la commande de la vanne EV1007NA.

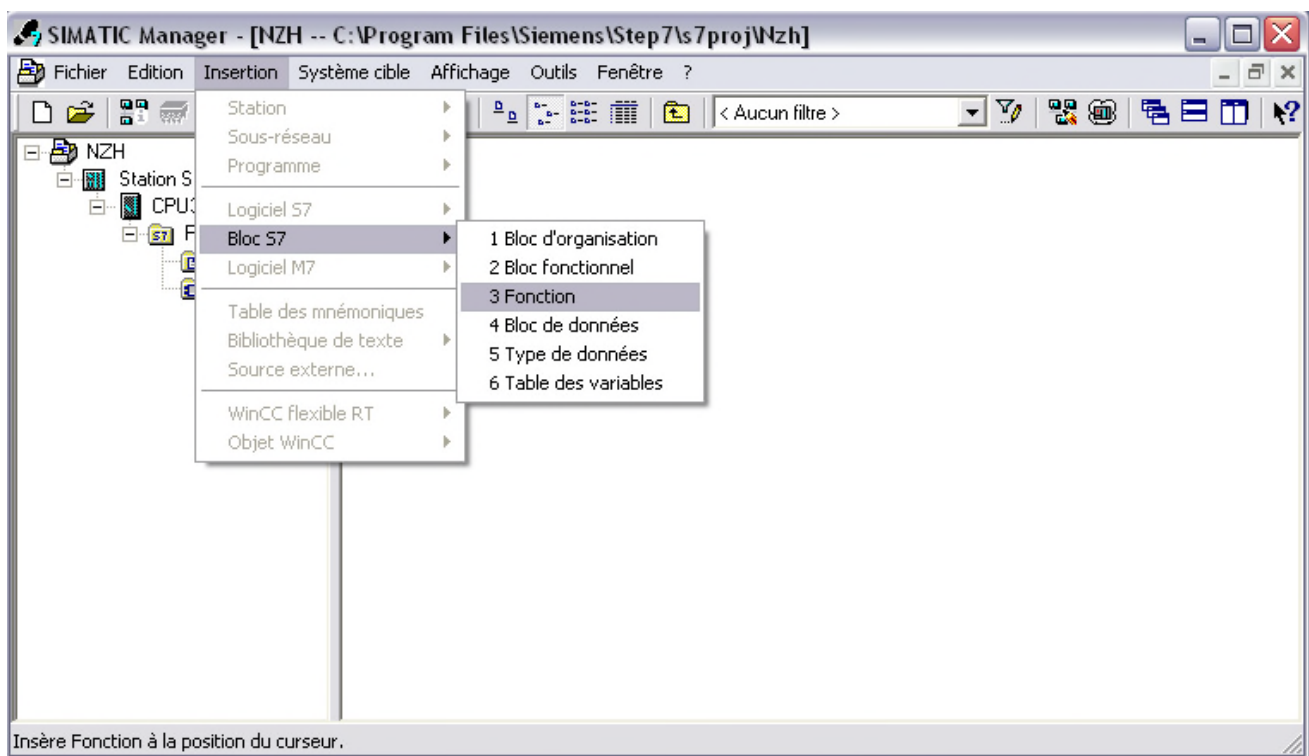
FC 8 : la temporisation pour les demandes manuelle, et arrêt soutirage.

FC 9 : le remplissage et le soutirage des trémies.

FC 10 : arrêt produit, et arrêt moteur.

#### **III.3.3.1. Création du FC1 :**

Dans cette fonction il n'y aura rien d'autre qu'appart la mise en échelle des trémies, et elle se fera de la manière suivante :



*Figure 3.12 : Création d'une fonction*

Une nouvelle fenêtre apparaît, et on peut donner un aperçu de ce qu'il contient comme programme, le bloc de fonction :

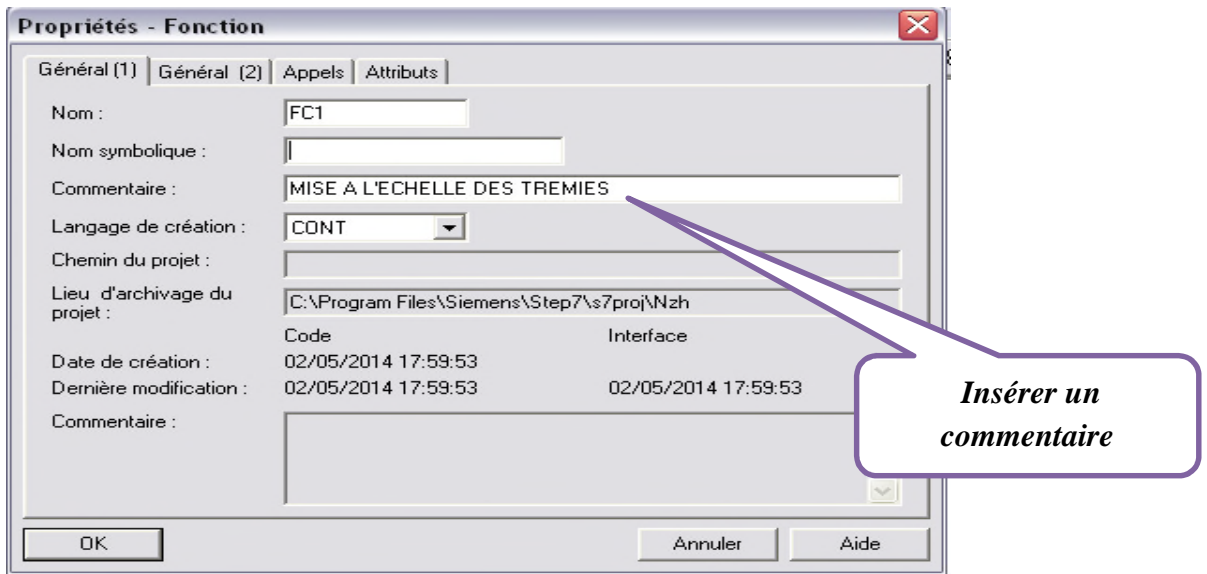


Figure 3.13: Appellation d'une fonction

Une fois que la fonction FC1 est créée, on doit insérer une bibliothèque, pour mettre la mise à l'échelle des trémies, comme la figure ci-dessus le montre : (voir annexe 03)

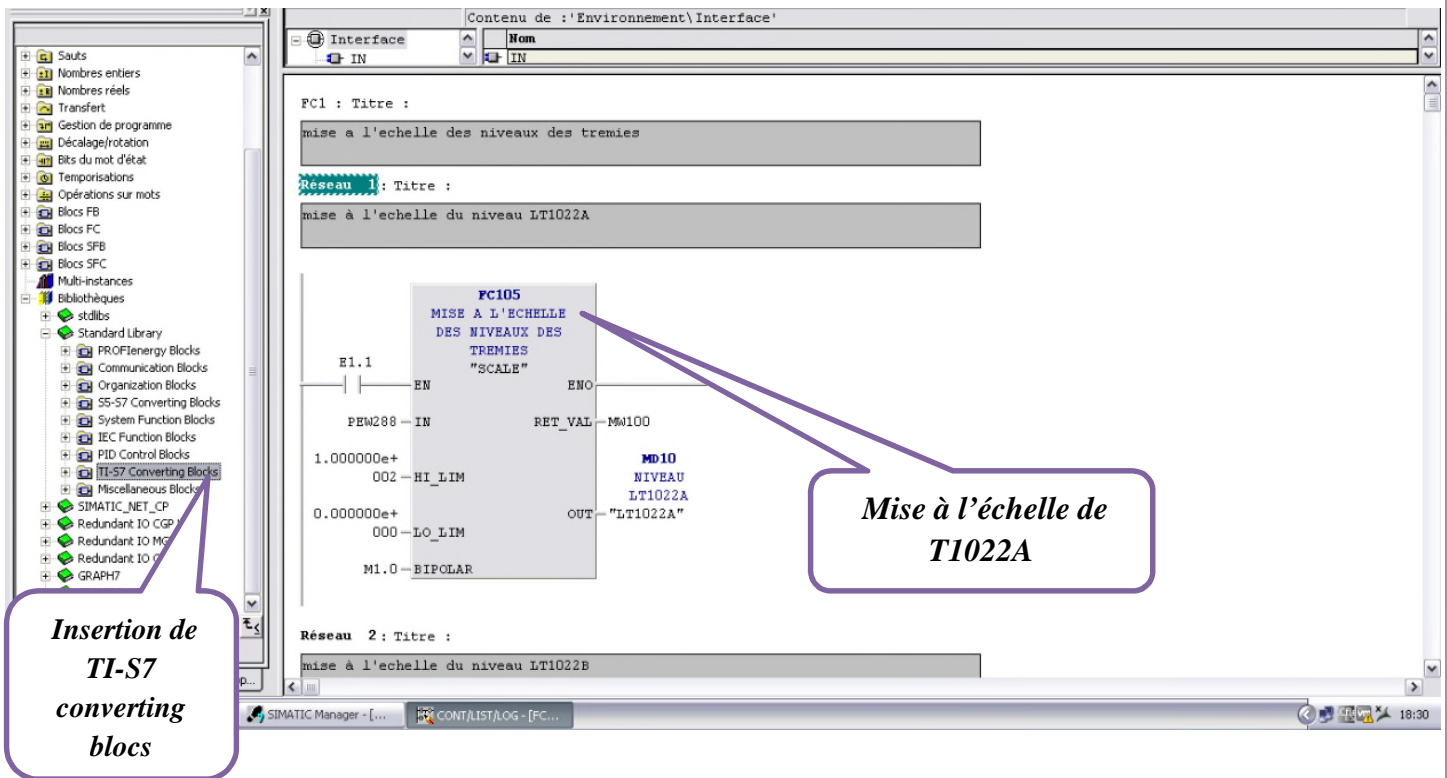


Figure 3.14: Mise à l'échelle de la trémie T1022A

- ❖ Cette action s'applique pour toutes les trémies restantes, on donnant l'adresse, périphérique Entrée Word (PEW) à chacune des trémies qui seront séparé de deux bit comme suite :

LT1022A → PEW 288  
 LT1022B → PEW 290  
 LT1022C → PEW 292  
 LT1022D → PEW 294

- ❖ Et aussi on incrémentant la sortie qui nous donne les niveaux des trémies de 4 bits, car c une adresse de double mot (Double Mot) :

LT1022A → MD 10  
 LT1022B → MD 14  
 LT1022C → MD 18  
 LT1022D → MD 22

- ❖ Ensuite des adresse Memento Word (MW), qu'elles seront séparé de 2 bit :

LT1022A → MW 100  
 LT1022B → M W 102  
 LT1022C → M W 104  
 LT1022D → M W 106

### III.3.3.2. Création des autres fonctions :

Pour crée les autres fonctions, elles se font de la même manière que la création du FC1, seulement bien sur le contenu qui change (le programme), comme par exemple le FC2, le programme de la vanne EV1019B, l'lorsqu'il est créé, un nouveau réseau est affiché directement, et pour crée un deuxième réseau, il suffit juste d'y aller à la fenêtre SIMATIC>>insertion>> réseau :

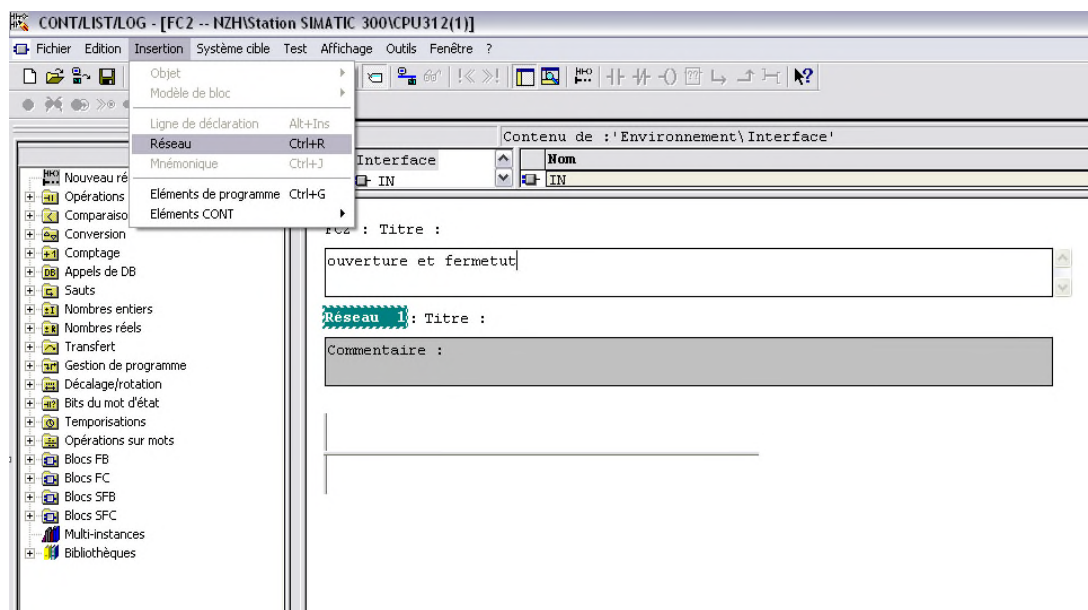


Figure 3.15 : Insertion d'un nouveau réseau

### III.3.3.3. Éditer des Mnémoniques :

Dans l'icône « insertion » >>blocS7>>table des variable, on peut accéder directement à la table des mnémoniques de manière à pouvoir la créer, la compléter ou la modifier lors même de la simulation :

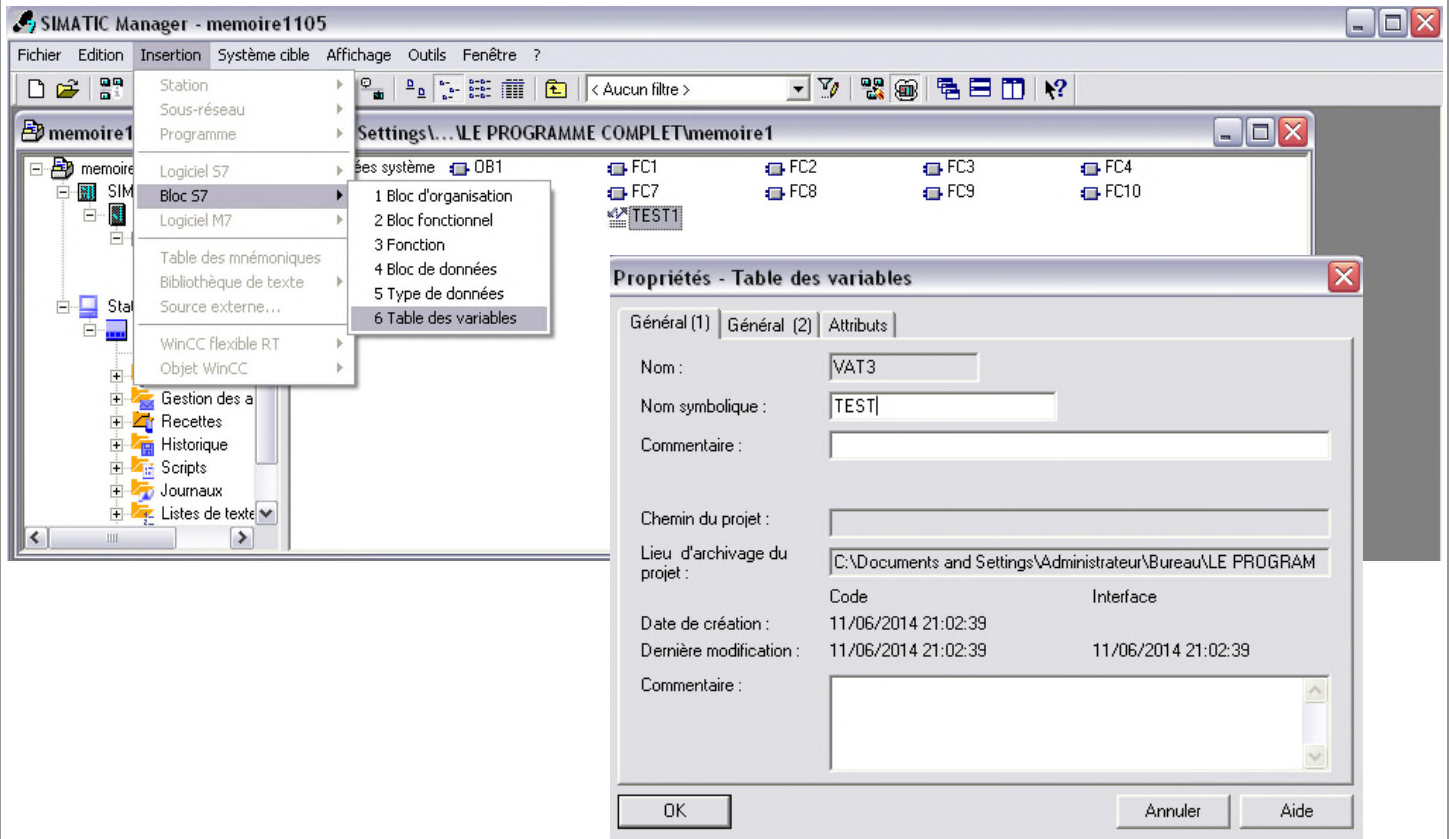


Figure 3.16 : Insertion d'une table de variable

#### Remarque :

Les tables de variables mémorisées dans le dossier blocs d'un programme S7 peuvent être ouvertes ultérieurement à des fins de test par un double-clic sur « TEST » qu'on a créé.

### III.3.3.4. Ouverture de l'application PLCSIM :

Pour l'utilisation du simulateur S7-PLCSIM, on suit les étapes suivantes pour sa mise en route.

1- Ouvrir le gestionnaire de projet SIMATIC.

2- Sélectionner la commande «outils» et puis «simulation de modules» ou en cliquant sur son icône qui se trouve dans la boîte d'outils du gestionnaire de projet SIMATIC. Cela lance l'application S7-PLCSIM et ouvre une fenêtre CPU ayant l'adresse MPI par défaut.



### A. Configuration du PLCSIM :

Dans l'application S7-PLCSIM, on crée de nouvelles fenêtres pour visualiser les informations provenant de l'automate programmable de simulation et pour créer les diverses fenêtres, on procède comme suite :

1- Créer une fenêtre permettant de modifier l'état des entrées intervenant dans le programme.

- Choisir la commande « insertion entrée » où à partir de la barre d'outils.

2- Créer une fenêtre permettant de modifier l'état des sorties intervenant dans le programme.

- Choisir la commande « insertion sortie » ou à partir de la barre d'outils.

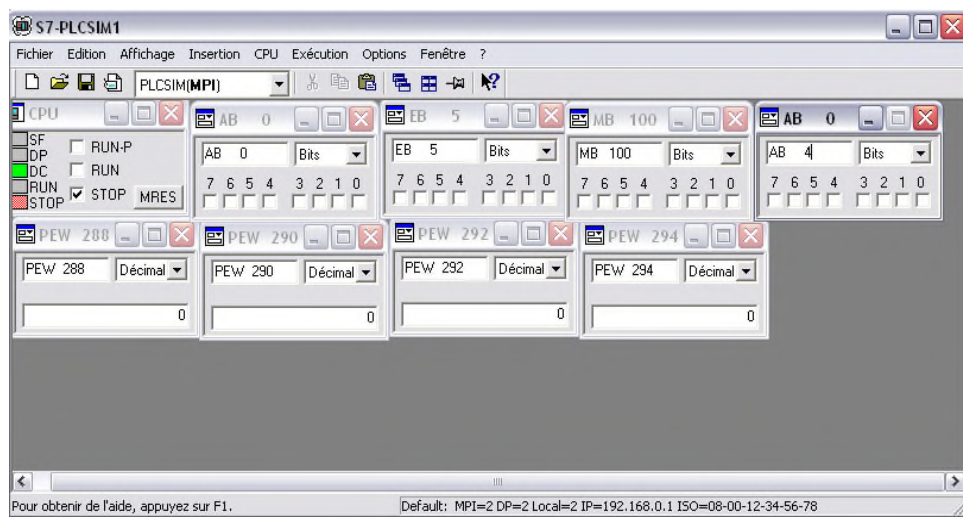


Figure 3.17 : Configuration du PLCSIM

#### III.3.3.5. Etat de fonctionnement de la CPU :

##### ❖ *Etat de marche (RUN-P) :*

La CPU exécute le programme et nous permet de le modifier ainsi que ses paramètres.

Afin de pouvoir utiliser les applications de STEP7 pour forcer un paramètre quelconque du programme durant son exécution, on doit mettre la CPU à l'état RUN-P.

##### ❖ *Etat de marche (RUN) :*

La CPU exécute le programme en lisant les entrées, exécutant le programme, puis en actualisant les sorties. Lorsque la CPU se trouve à l'état de marche (RUN), on ne peut ni charger un programme, ni utiliser les applications de STEP7 pour forcer un paramètre quelconque (comme les valeurs d'entrées).

##### ❖ *Etat d'arrêt (STOP) :*

La CPU n'exécute pas le programme. Contrairement à l'état d'arrêt (STOP) des CPU réelles, les sorties ne prennent pas de valeurs (de sécurité) prédéfinies, mais conservent l'état auquel elles étaient lorsque la CPU est passée à l'état d'arrêt (STOP). Nous pouvons charger des programmes dans la CPU lorsqu'elle est à l'arrêt.

### III.4. Supervision :

La supervision est une forme évoluée de dialogue Homme-Machine, elle sert à représenter et surveiller l'état de fonctionnement d'un procédé.

#### III.4.1. Interface homme/machine : [5]

Les interfaces homme/machine sont des logiciels spécifiques au contrôle-commande dans SIMATIC, les systèmes de visualisation du processus SIMATIC WinCC et SIMATIC WinCC flexible sont des systèmes de base indépendants des branches et technologies d'utilisation qui comportent toutes les fonctions indispensables au contrôle-commande.

#### III.4.2. WinCC flexible:

WinCC flexible est l'interface homme-machine (IHM) idéale pour toutes les applications au pied de la machine et du processus dans la construction d'installations.

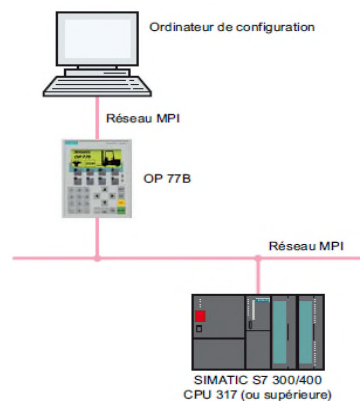


Figure 3.18 : Système d'automatisation avec un pupitre

#### III.4.2.1. Insertion d'une station HMI dans SIMATIC manager :

Pour insérer une station HMI, il faut d'abord aller dans: SIMATIC manager >>insertion>>station>>station HMI, les étapes à suivre nous les montre les figures suivante :

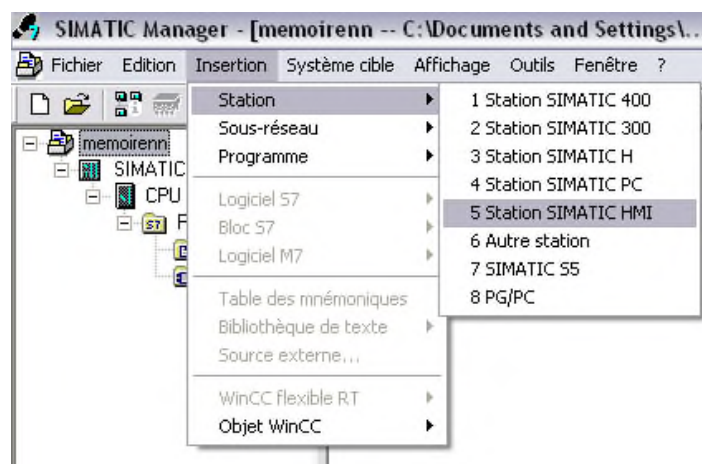
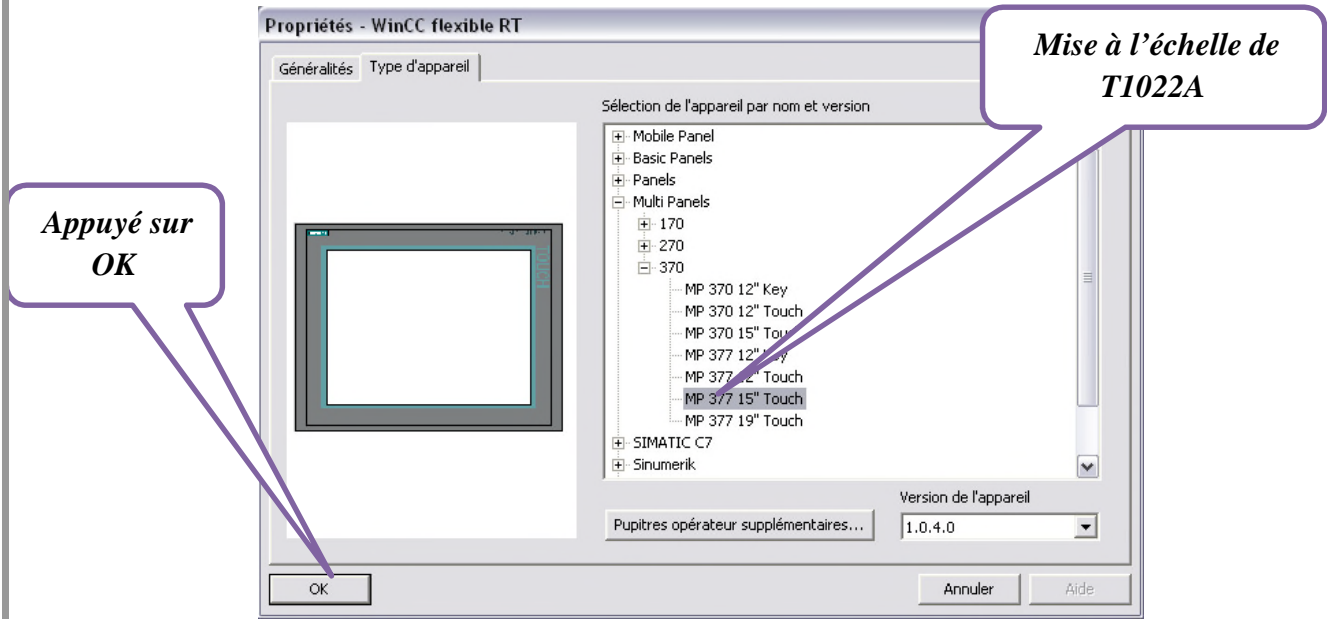


Figure 3.19 : Insertion d'un nouveau réseau


**III.4.2.2. Choix de la station (pupitre) :**

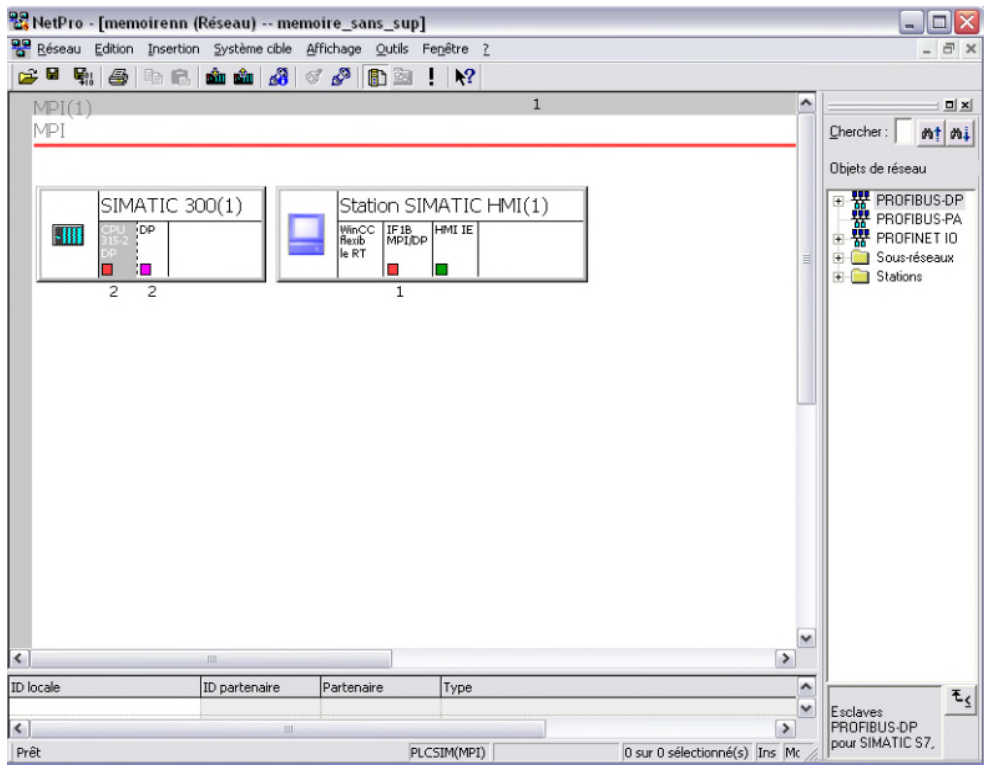
Dans notre cas MP377 15" :



**Figure 3.20 : Choix du pupitre**

**III.4.2.3. Configuration du réseau:**

Aller dans NETPRO (icone  ), on clique dessus, et la fenêtre suivante apparait, tout ce qu'on doit faire c'est de relier MIP (la ligne rouge) avec SIMATIC 300 et station SIMATIC HMI :



**Figure 3.21 : Configuration du réseau**



Une solution d'automatisation complète est composée d'une IHM telle que le WinCC flexible, d'un système d'automatisation, d'un bus système et d'une périphérie. Dans l'assistant du projet WinCC flexible, où on crée étape par étape notre projet IHM, on sélectionne notre projet STEP7, dans lequel on veut intégrer notre projet IHM.

- Double cliquer sur la partie CPU 315-2DP pour paramétrer l'interface de communication automate :

Type : MPI

Adresse : 3

Connecté : oui

- Après double cliquer sur la partie IF 1B MPI DP pour paramétrer l'interface de communication pupitre :

Type : MPI

Adresse : 2

Connecté : oui

- Enfin enregistrer et compiler en appuyant sur l'icone 

Une nouvelle fenêtre apparaît, en montrant clairement que le réseau MIP est configuré :

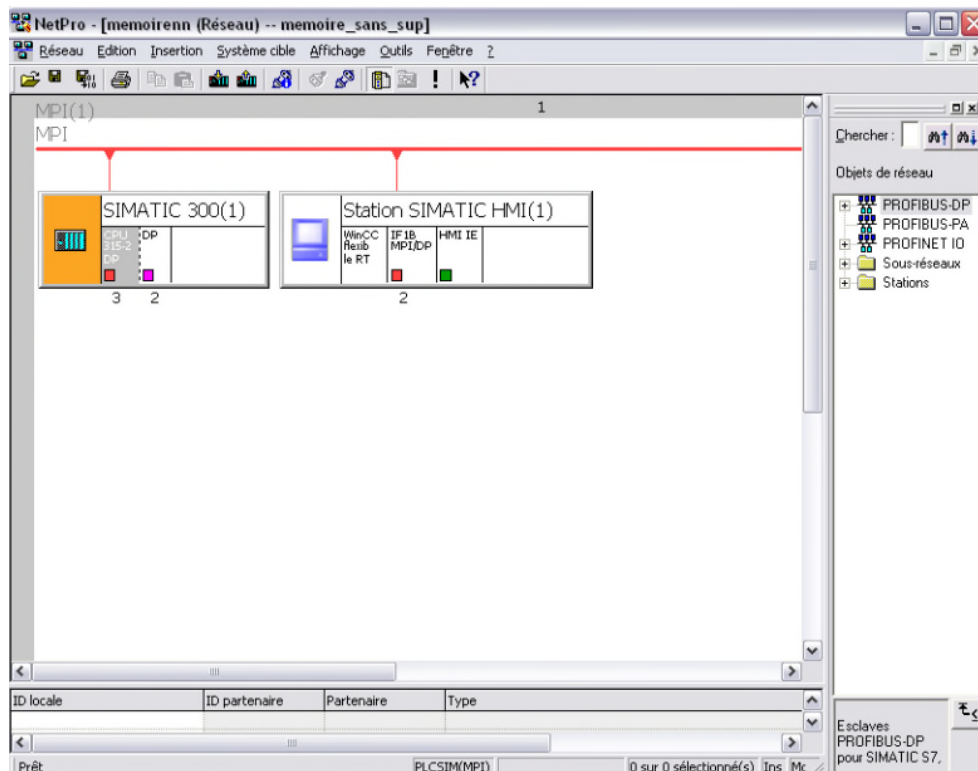


Figure 3.22 : Réseau MIP après configuration

### III.4.2.4. Activer la liaison:

Aller dans SIMATIC manager >> station SIMATIC HMI >> WinCC flexible RT >> communication >> liaison (choisir l'automate, activer la liaison).

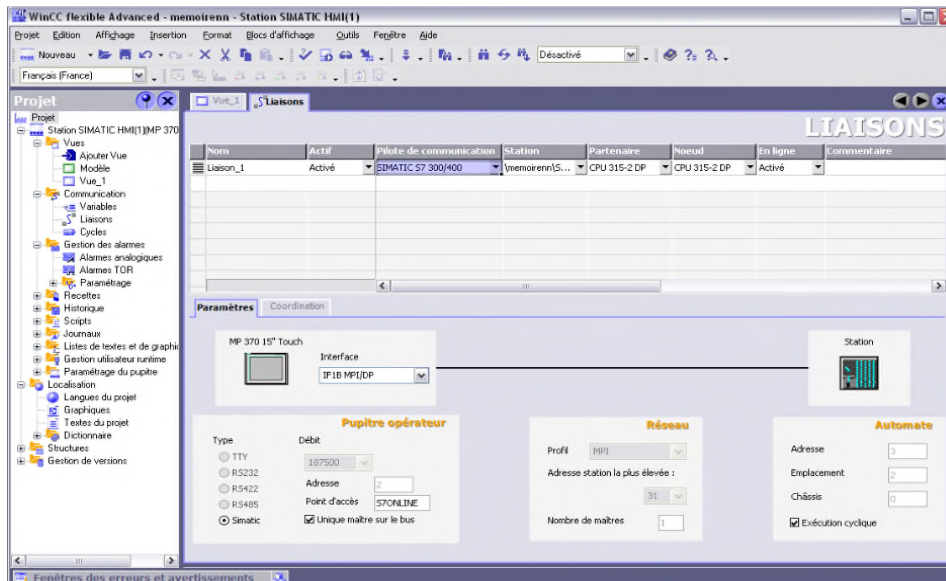


Figure 3.23 : Configuration de la liaison pupitre vers l'automate

### III.5. Structure du projet :

Le projet est à la base de la configuration de l'interface graphique, on va créer et configurez dans le projet tous les objets indispensables à la commande et au contrôle de l'installation du remplissage des quatre trémies, et sa touchera:

- Contrôle du niveau de remplissage des trémies.
- Surveillance des états des vannes.

Pour permettre l'exécution de ces tâches, le pupitre opérateur doit être «configuré ». Le manuel de mise en route pour débutants décrit les étapes de configuration nécessaires :

- Création du projet
- Création des trémies
- Configuration de switch (2000T/3000T)
- Création des vannes et les configurer
- Création des deux moteurs et les configurer
- Ajout de changements de vue
- Test et simulation du projet

**III.5.1. Création des trémies :**

Pour l’affichage des niveaux de remplissage, on doit insérer un paragraphe dans la vue :

Objets simple>>bargraphe

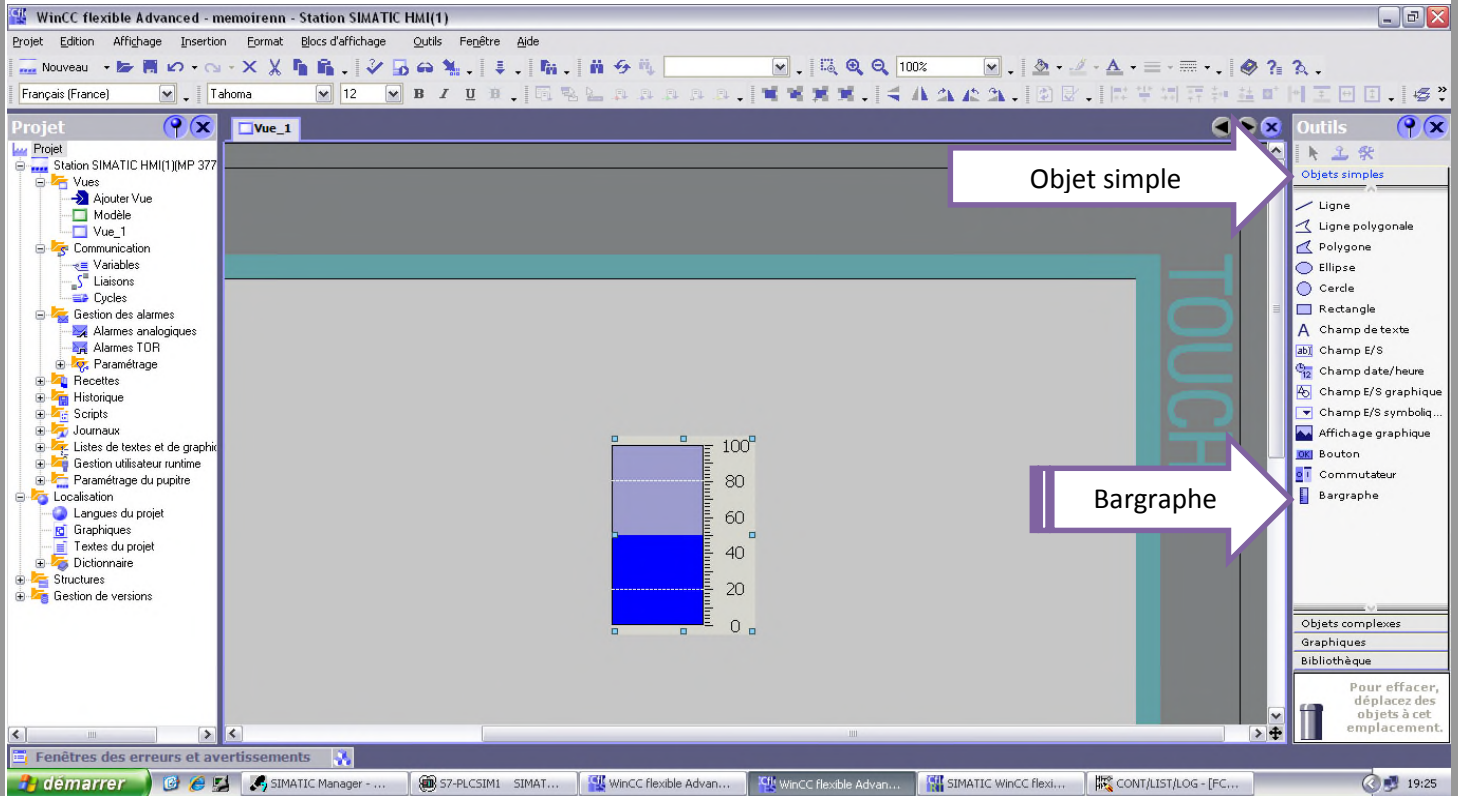


Figure 3.24: création du bargraphe

**III.5.2. Configurer l’affichage du bargraphe dans le pupitre :**

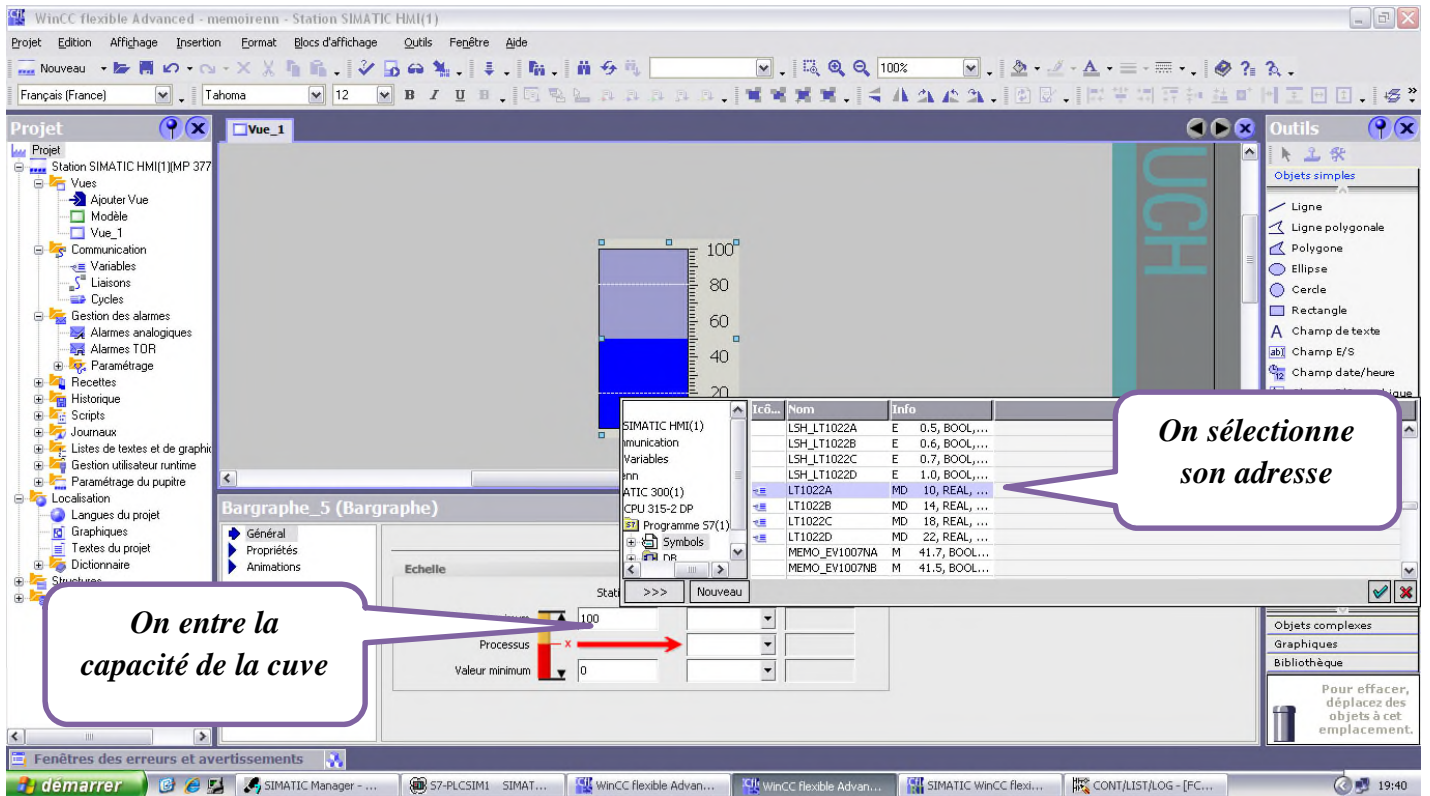


Figure 3.25 : configuration de l’affichage de la trémie T1022A

**Remarque :**

Nous avons choisi pour la capacité de la trémie 100 mais c'est juste un pourcentage, mais en réalité notre trémie est de 80T.

- ❖ On referra cette opération pour les trois autres trémies qui reste T1022B, T1022C et T1022D.

**III.5.3. Créations des vannes :**

Pour les vannes, on a dessiné juste un rectangle, et on l'a configuré à l'état zéro et à son état un par un déplacement horizontal, puis on configurera son adresse comme suite :

Nom	Info
CAD_SOUT_LT10...	M 151.0, BOO...
CADENCE DES T...	M 10.5, BOOL...
CMD_EV1007NA	A 5.0, BOOL...
CMD_EV1007NB	A 4.2, BOOL...
CMD_EV1019A	A 4.6, BOOL...
CMD_EV1019B	A 4.0, BOOL...
CMD_EV1019C	A 4.4, BOOL...
CMD_EV1021B	A 5.4, BOOL...
CMD_EV1021C	A 5.2, BOOL...
DEF_FER_EV100...	M 30.7, BOOL...

Objet	0	1	Non	Non
CMD_EV1019B				

Figure 3.25 : configuration d'animation de la vanne EV1019A

- ❖ On referra cette opération pour les cinq autres vannes qui reste EV1019B, EV1007NB, EV1007NA, EV1021B et EV1021C.

**III.6. Conclusion :**

Après avoir opté pour une configuration adéquate de l'automate, le programme qui a été élaboré et qui va être chargé dans l'API a été établi d'après le cahier des charges et l'analyse fonctionnelle effectuée.

Dans ce chapitre, nous avons créé le programme sous STEP7 puis l'interface graphique (HMI), ensuite nous les avons mis en liaison.

Pour cela, nous avons procédé comme suit :

- Création du programme.
- Mettre en liaison l'automate et le pupitre.
- Transfert du projet sur le pupitre.



*Encadré par : Mr. Hassen RECHAME*

*Mr. Hocine LEHOUCHE*

---

## *Chapitre 04:*

# *Testes **E**t **S**imulations*

---

**IV.1. Introduction :**

Dans ce chapitre nous allons simuler notre programme, on introduisant quelque valeur des niveaux aux trémies pour pouvoir voir le mouvement des vannes commander et aussi le remplissage et le soutirage du sucre, et nous allons présenter les résultats obtenus dans la partie réalisation et dans la partie programmation.

**IV.2. Simulation sous step7 :**

La simulation permet de détecter des erreurs logiques de configuration, par ex. des valeurs limites incorrectes.

Avant tout, on doit faire appeler toutes les fonctions dans FC1 par la commande « CALL ».

OB1 : "Main Program Sweep (Cycle)"

Commentaire :

Réseau 1: Titre :

Commentaire :


```
CALL "MISE A L'ECHELLE"
CALL "COMMANDE DE EV1019B"
CALL "COMMANDE DE EV1007NB"

CALL "COMMANDE DE EV1019A"
CALL "COMMANDE DE EV1021B"
CALL "COMMANDE DE EV1021C"
CALL "COMMANDE DE EV1007NA"
CALL FC      8
CALL "CAD_VANNES"
|
CALL FC     10
```

*Figure 4.1 : faire appeler les fonctions*

**IV.2.1. Activation de la simulation :**

La figure ci-après explique comment simuler l'affichage du niveau de remplissage des trémies et des états des vannes.

Pour activé la simulation, on aura qu'a appuyé sur l'icone 

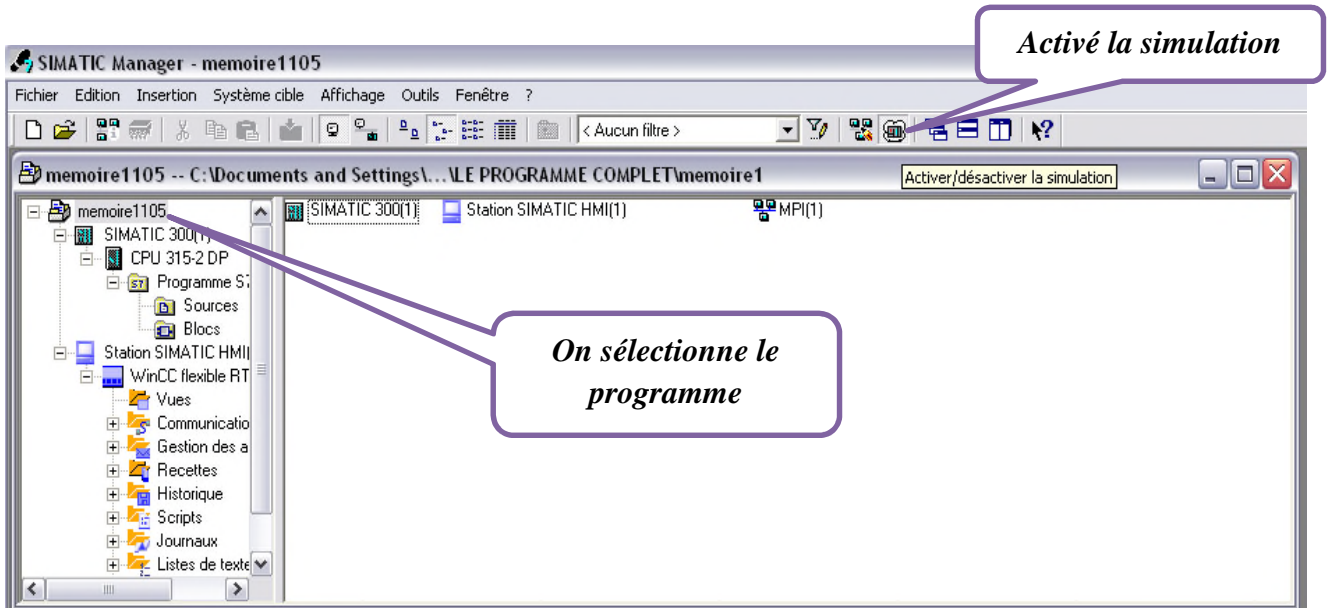



Figure 4.2 : Activation de la simulation

**IV.2.2. Chargement du programme :**

Le chargement du programme se fait on appuyant sur  , cette action permet de charger le programme dans la CPU pour son exécution.

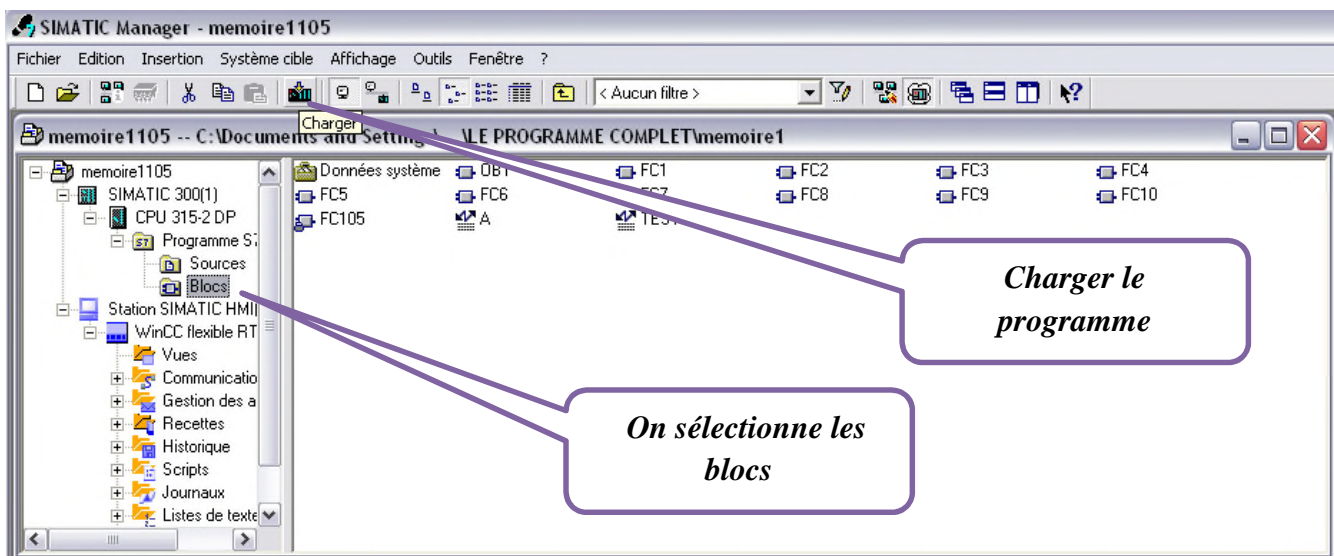


Figure 4.3: Activation de la simulation

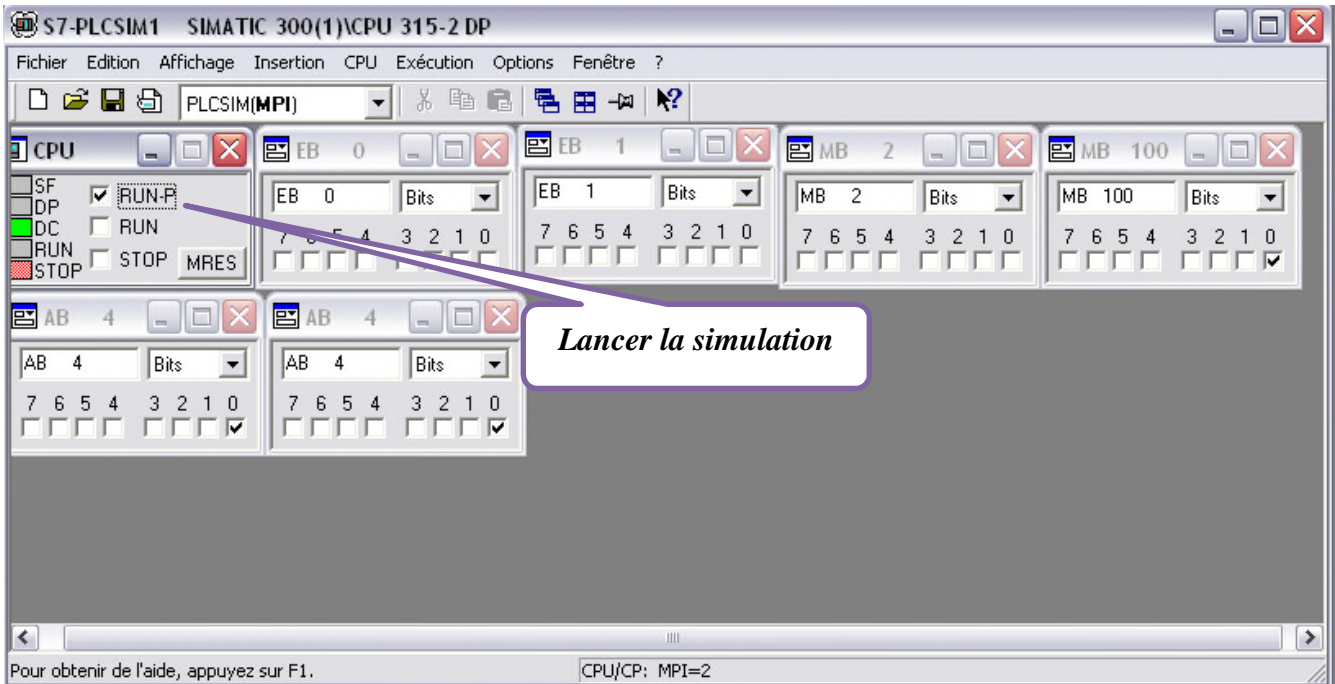
**Remarque :**

Il faut d’abord sélectionner « BLOCS » avant de charger le programme.



**IV.2.3. Le lancement de la simulation :**

Cela se fait en appuyant sur « RUN-P », comme le montre la figure ci-dessous :



**Figure 4.4 : Démarrer la simulation**

**IV.3. Résultats :**

Pour cette partie, nous allons juste tester pour une seule vanne (EV1019B), le tableau suivant nous résume quelques cas où cette dernière s’ouvre:



T1022A		T1022B		T1022C		T1022C		raffinerie
Valeur	Tendance	Valeur	Tendance	Valeur	Tendance	Valeur	Tendance	
X	X	X	X	X	X	LT1022D	DECENDANT	2000T
						<=20%		
LT1022A	MONTANT	LT1022B	DECENDANT	X	X	LT1022D	MONTANT	2000T
>=70%		<=70%				>=20%		
LT1022A	MONTANT	LT1022B	MONTANT	LT1022C	DECENDANT	LT1022D	MONTANT	2000T
>=70%		>=70%		<=70%		<=20%		

**Tableau 4.1 : cas de l’ouverture de la vanne EV1019B**

### IV.3.1. Visualiser les variables :

Dans l'outil configuration matérielle, on peut visualiser ou forcer directement les opérands des modules configurés, la fonction visualiser des variables permet de contrôler le câblage des modules d'entrée et la fonction Forcer des variables, celui des modules de sortie.

Il existe deux possibilités pour visualiser les variables :

- actualisation unique des valeurs d'état à l'aide de la commande **Variable -> Actualiser valeurs d'état** ou en cliquant sur 
- actualisation cyclique des valeurs d'état à l'aide de la commande **Variable -> Visualiser** ou en cliquant sur 

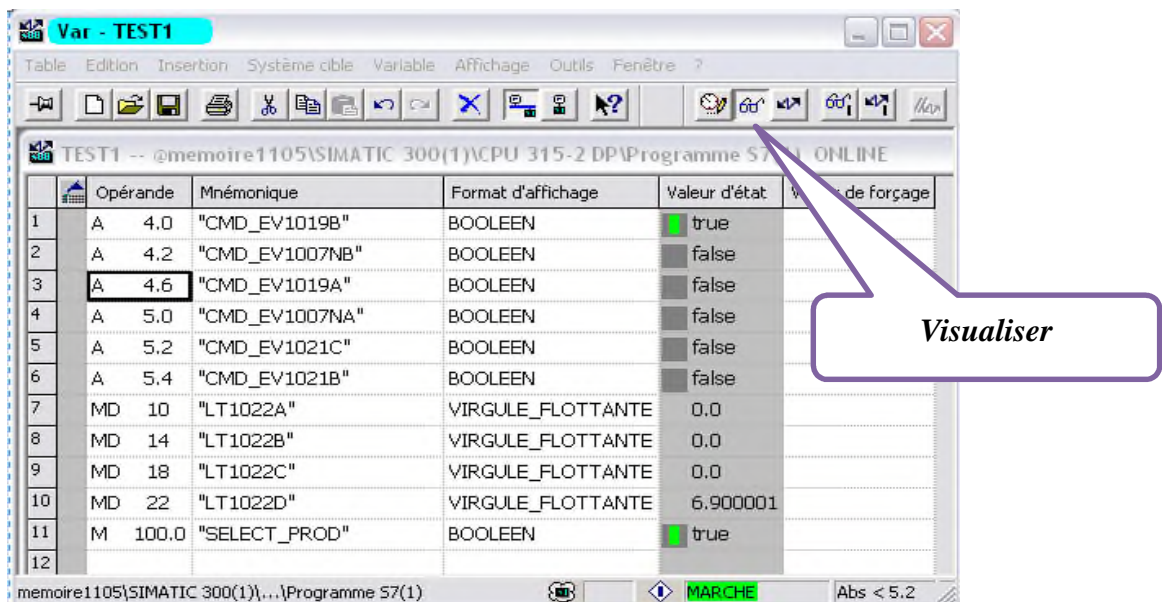
#### Remarque :

Avant de pouvoir activer les fonctions "Visualiser" et "Forcer", il faut établir une liaison en ligne avec la CPU.

#### IV.3.1.1. Visualisation de l'ouverture de la vanne EV1019B :


Après avoir chargé le programme dans notre CPU, on double clic sur la fonction « TEST » qu'on a créé au pare avant.

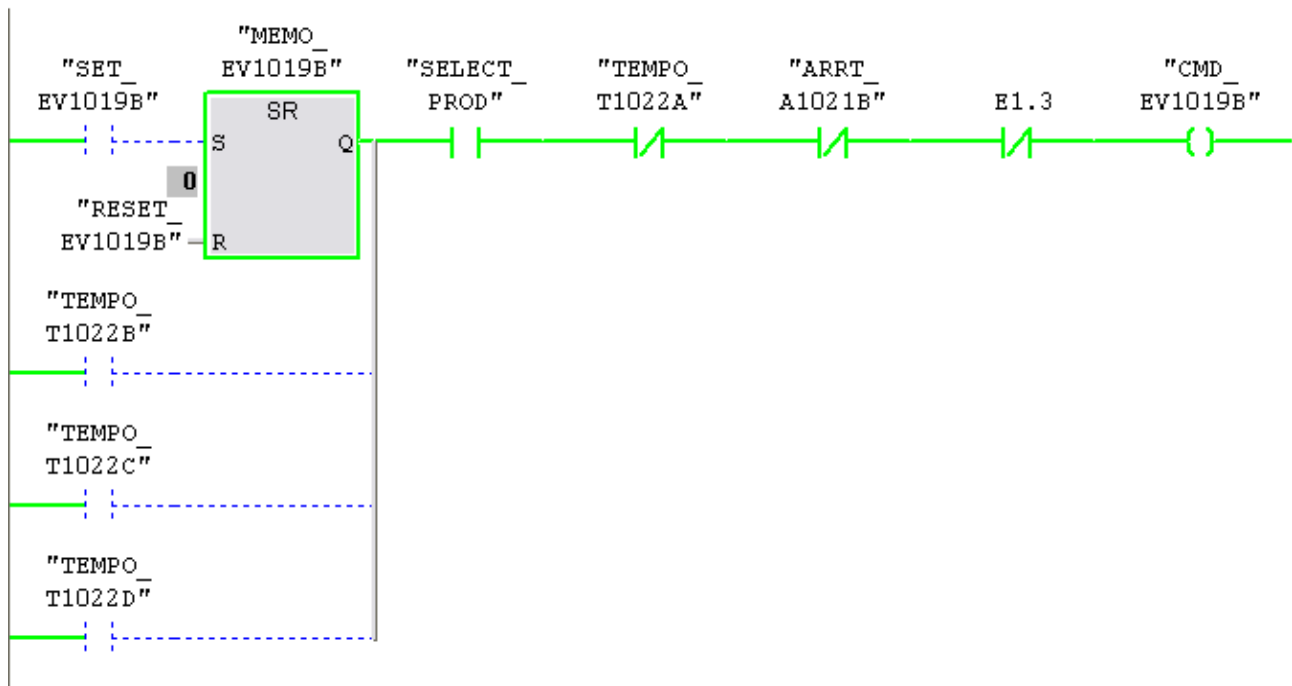
On aura la figure suivante :



	Opérande	Mnémonique	Format d'affichage	Valeur d'état	Valeur de forçage
1	A 4.0	"CMD_EV1019B"	BOOLEEN	true	
2	A 4.2	"CMD_EV1007NB"	BOOLEEN	false	
3	A 4.6	"CMD_EV1019A"	BOOLEEN	false	
4	A 5.0	"CMD_EV1007NA"	BOOLEEN	false	
5	A 5.2	"CMD_EV1021C"	BOOLEEN	false	
6	A 5.4	"CMD_EV1021B"	BOOLEEN	false	
7	MD 10	"LT1022A"	VIRGULE_FLOTTANTE	0.0	
8	MD 14	"LT1022B"	VIRGULE_FLOTTANTE	0.0	
9	MD 18	"LT1022C"	VIRGULE_FLOTTANTE	0.0	
10	MD 22	"LT1022D"	VIRGULE_FLOTTANTE	6.900001	
11	M 100.0	"SELECT_PROD"	BOOLEEN	true	
12					

Figure 4.5 : Visualisation l'ouverture d'EV1019B dans la table de variable

❖ On peut aussi voir la simulation de la vanne EV1019B dans le schéma bloc qui se trouve dans FC2, pour cela on appuiera seulement sur l'icone , comme le montre la figure



suivante :


Figure 4.6: Visualisation l'ouverture d'EV1019B dans FC2

❖ On peut voir que la vanne EV1019B est ouverte on vérifiant son bit (A 4.0) qu'il est à un (1)



Figure 4.7: Visualisation le bit d'ouverture d'EV1019B

**IV.3.1.2. Visualisation du remplissage de la trémie T1022D :**

On peut voir le remplissage de la trémie T1022D, avec une cadence de un demi (0.5), dans FC9 on appuyant sur  .

Réseau 4 : Titre :

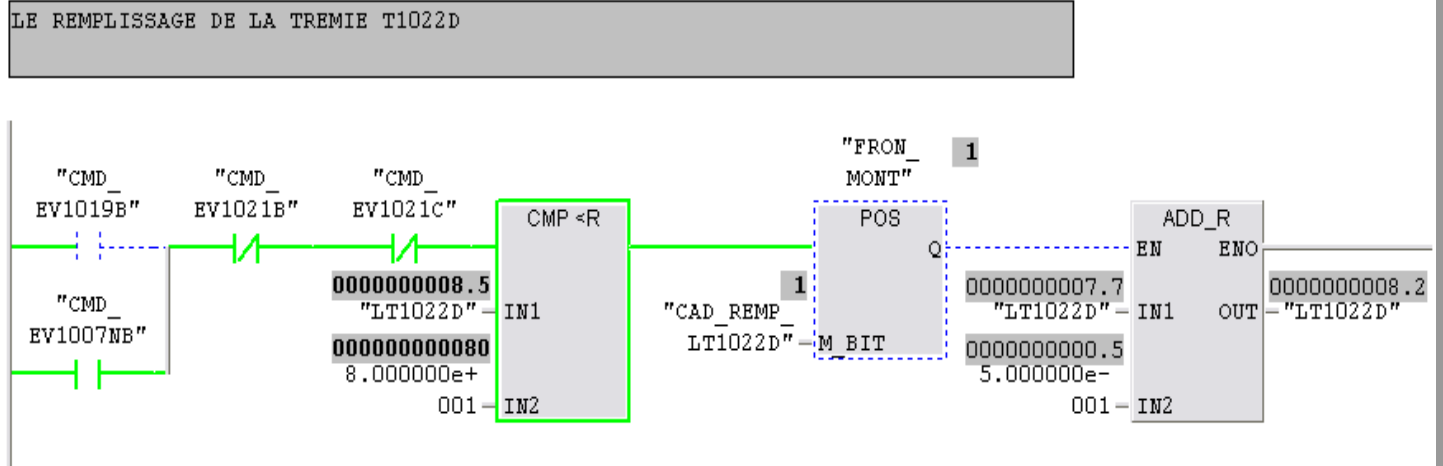



Figure 4.8: Visualisation du remplissage de la trémie T1022D

**IV.3.1.3. Visualisation du soutirage de la trémie T1022D :**

On peut voir soutirage du sucre avec une cadence moins élevé que le remplissage de la trémie T1022D, dans FC9 on appuyant sur  .

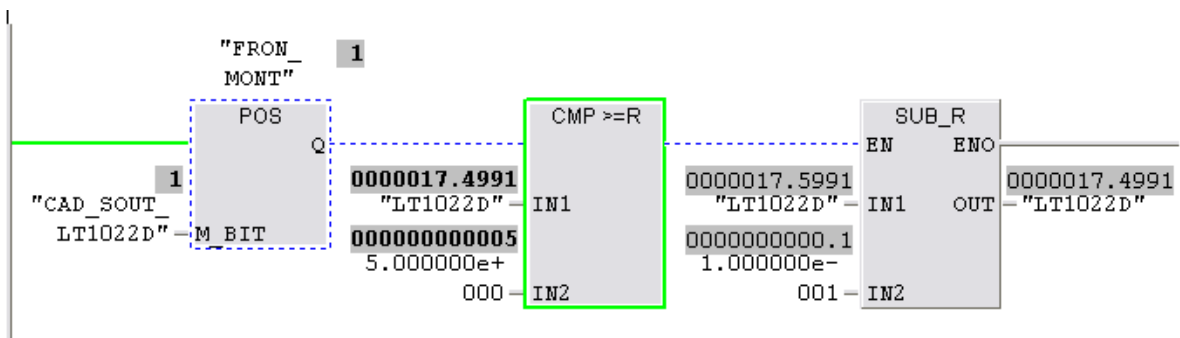


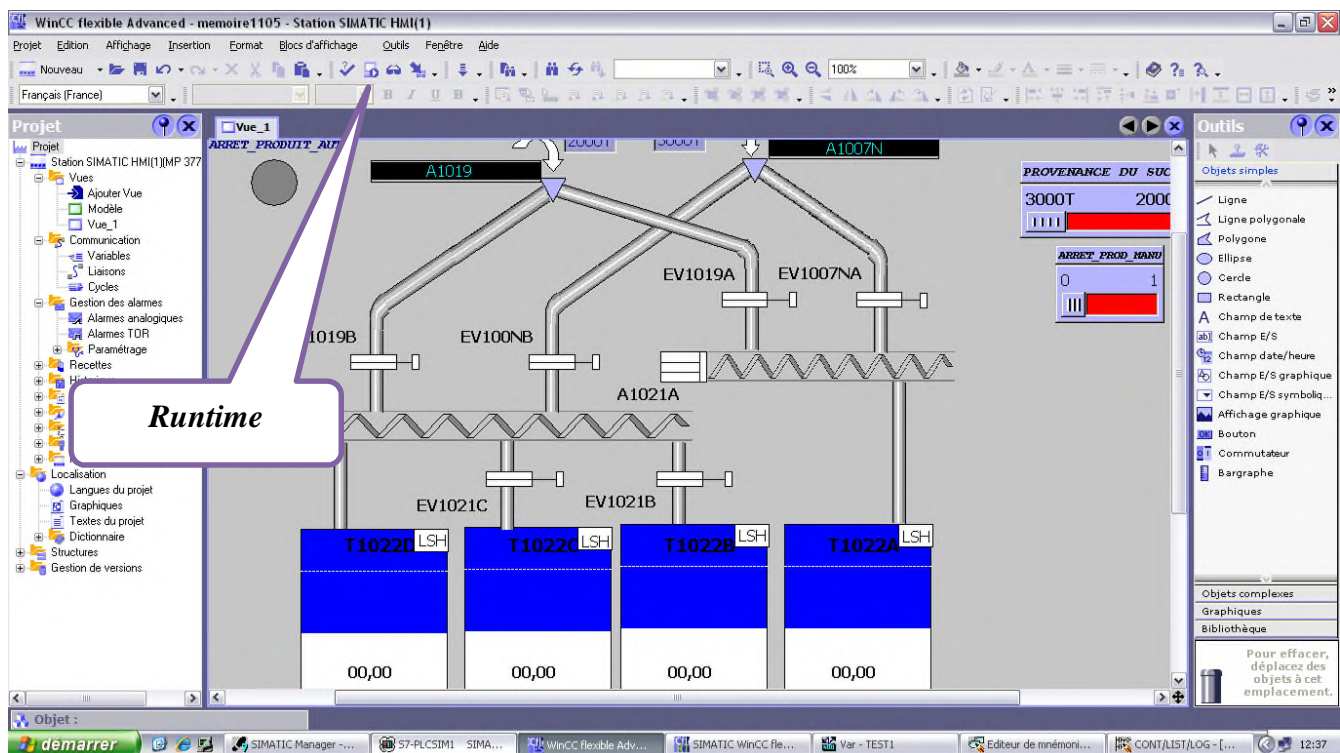
Figure 4.9: Visualisation du soutirage de sucre dans la trémie T1022D

#### IV.4. Simulation avec WINCC:

Une fois que la liaison est activée, tout ce qui nous reste à faire c'est de visualiser du processus sur le pupitre.

##### IV.4.1. Activé la simulation:

Cela se fait avec la commande « RUNTIME » , qui se trouve dans la barre d'état de la fenêtre WINCC flexible.



**Figure 4.10: Activé la visualisation avec Runtime**

#### **Remarque :**

On doit appuyer sur Runtime, une fois que le chargement des blocs a été fait, et l'automate a été démarré (le lancement de la simulation sous step7) également.

#### IV.4.2. Activé la simulation:

La fenêtre ci-dessous apparait, juste après qu'on a appuyé sur l'icône Runtime, on peut donc voir le remplissage des trémies et l'ouverture des vannes.

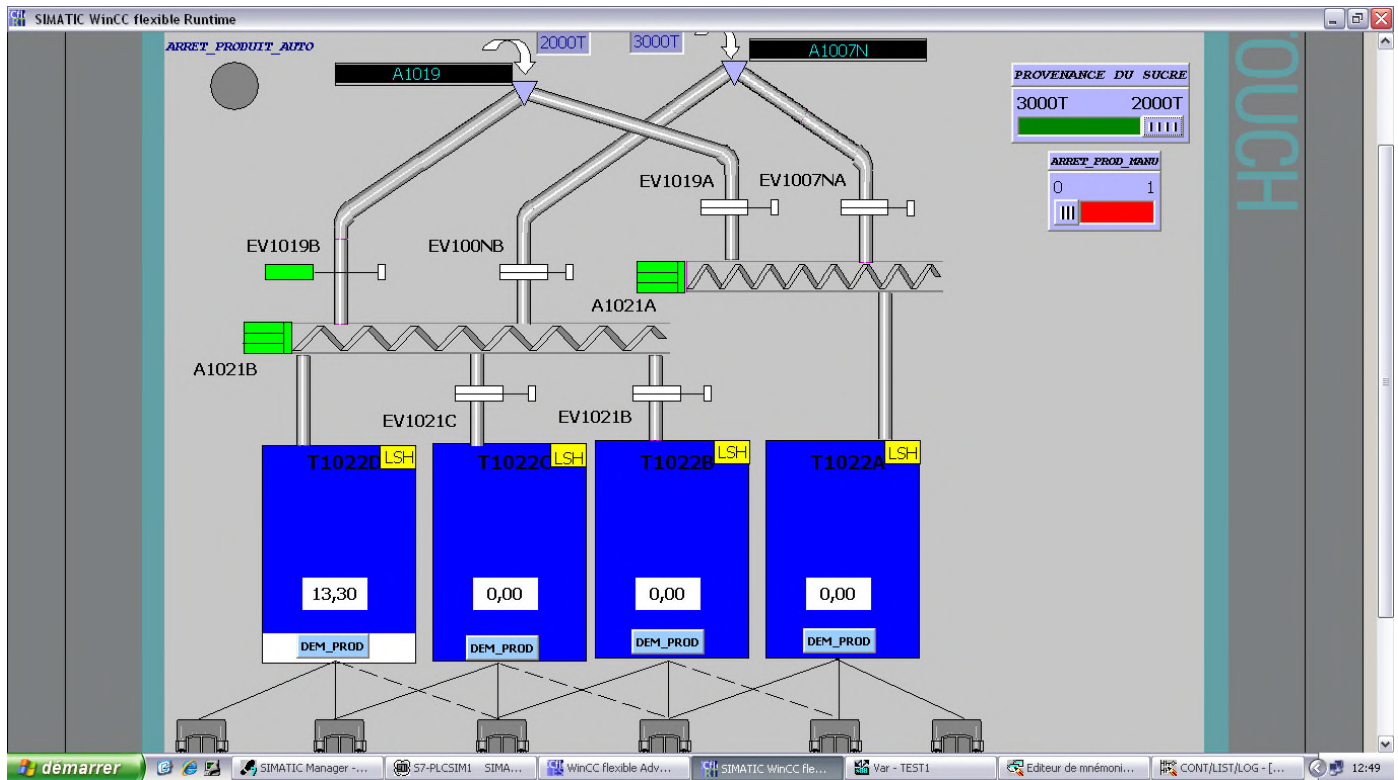


Figure 4.11: Visualisation l'ouverture d'EV1019B sur le pupitre

#### IV.5. Conclusion :

Dans ce chapitre nous avons procédé aux étapes pour simuler et superviser notre programme et nous avons visualisé l'ouverture d'une seule vanne la EV1019B, et le remplissage et le soutirage du sucre dans la trémie T1022D.

L'utilisation de langage step7, et le supervision WinCC, nous a permis de tester le programme élaboré vu que le STEP7 offre différentes possibilités de test telle que la visualisation du programme afin de remédier à d'éventuelles erreurs commises et les modifications appropriées avant de passer à l'implémentation dans l'automate.

*Encadré par : Mr. Hassen RECHAME*

*Mr. Hocine LEHOUCHE*

---

## *Conclusion Générale*

---



### ***Conclusion générale***

Dans les entreprises, les objectifs fondamentaux sont : maximiser les profits, augmenter le chiffre d'affaire, protéger la santé et la sécurité des employés, améliorer la productivité, remplir des responsabilités sociales. L'automatisation des systèmes de production est l'un des moyens souvent mis en œuvre pour y parvenir.

A travers cette étude du système de la section ensachage, la méthodologie de l'automatisation de processus industriels par API Siemens S7-300 a été présentée puis appliquée sur la commande des vannes, cette étude conduit au choix de la configuration optimale de l'API.

Le travail que nous avons effectué au sein de l'entreprise Cévitale Béjaïa, m'a permis d'acquérir des connaissances techniques et pratiques, et d'améliorer mes connaissances théoriques dans un vaste domaine de l'électronique, et de l'automatique et de me familiariser avec le milieu industriel.

L'objectif de ce travail était d'automatiser le remplissage des quatre trémies de sucre, selon leurs priorités établies, en utilisant la programmation sous SIMATIC Manager, et de superviser ensuite avec le WinCC flexible, et pour cela nous avons choisi un automate programmable Siemens S7-300, à cause des possibilités de simulation et de supervision qu'il offre.

Au début j'ai représenté l'entreprise brièvement l'entreprise CEVITAL, puis en donnant des généralités sur les systèmes automatisés, et nous avons enchaîné par les différents appareils et matériels constituant l'installation, et aussi on a pris le temps de présenter la station ensachage et le système à automatiser, et son mode de fonctionnement, en plus on a mentionné l'automate programmable industriel qu'on utilisera pour la programmation en donnant ses références.

Et par la suite, on a programmé les vannes selon le cahier de charges de l'installation pour le remplissage des trémies, et on a supervisé par une IHM. On a étudié en simulation le système de supervision WinCC flexible permettant à l'opérateur de connaître l'état d'avancement du processus en temps réel et d'intervenir directement sur le pupitre de commande depuis la salle de contrôle. Ce logiciel a permis de mettre en œuvre une supervision de l'installation étudiée d'une manière simple, efficace et facile à utiliser.



On a conclu notre travail par une simulation d'une seule vanne, et le remplissage, et le soutirage d'une seule trémie.

Suite au travail effectué au niveau de l'entreprise, les résultats obtenus sont très concluants.

Effectivement, ce travail a pour but de réduire voir d'éviter considérablement les arrêts de production, et il permet de faciliter la taches aux agents, par la mise en œuvre d'un programme et une supervision, qui permet de programmer toutes les vannes à l'ouverture et à la fermeture, plutôt de faire des demande de produit manuelle à chaque fois que le sucre manque dans les trémies

En effet, nous avons pu réaliser le projet qui nous a été confié et tester avec succès l'installation pour les différents cas des niveaux des trémies.

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.

*Encadré par : Mr. Hassen RECHAME*

*Mr. Hocine LEHOUCHE*

---

# ***Bibliographie***

---

- [1] **SIEMENS SIMATIC** « Description du logiciel Step7 », édition Siemens AG, Allemagne, 2010.
- [2] **William BOLTON**, « Les Automates programmables industriels », édition DUNOD, Paris, 2010.
- [3] **Information et formation, automatisation et entrainements**, Programmation niveau 1, édition Siemens AG, 2003.
- [4] **Paul BRARD, Gérard COLOMBARI** « Le GRAFCET, outil de description des automatismes séquentiels », édition Techniques de l'ingénieur, Paris, 1999.
- [5] **SIEMENS**, Simatic HMI WinCC flexible 2008 Runtime », Manuel d'utilisation, édition Siemens AG, 2007.
- [6] **Hû Jean-Louis**, « Les Automates Programmables », édition BRUN, 2002.
- [7] **F. Baudoin, M. Lavabre**, « Capteurs »: Principes et utilisations, Édition. Casteilla 2007.
- [8] **Documentation techniques CEVITAL**, « instrumentation», 2010.
- [9] **Micheal GROUT, Patrick SALAUN** «Instrumentation industriel », Spécification et installation des capteurs et des vannes de régulations, 2eme édition DUNOD, Paris, 2012.
- [10] **Patrick PROUVOST**, « Instrumentation et régulation - en 30 fiches », édition DUNOD, 2010.
- [11] **AUDER Simon**, « Automate programmable industriel », édition l'ELAN, liège 1991.
- [12] **Manuel Siemens Step7**, « Installation programmation», édition Siemens AG, 2010.
- [13] **Philippe LE BRUN** « Automates programmables industriels », édition Technologuepro, 1999.
- [14] **René HUSSON, Claude LUNG, Jean-François AUBRY, Jamal DAAFOUZ, Ddier WOLF**, « Automatique», de cahier des charges à la réalisation de système, Master école d'ingénieurs, édition DUNOD, 2007.

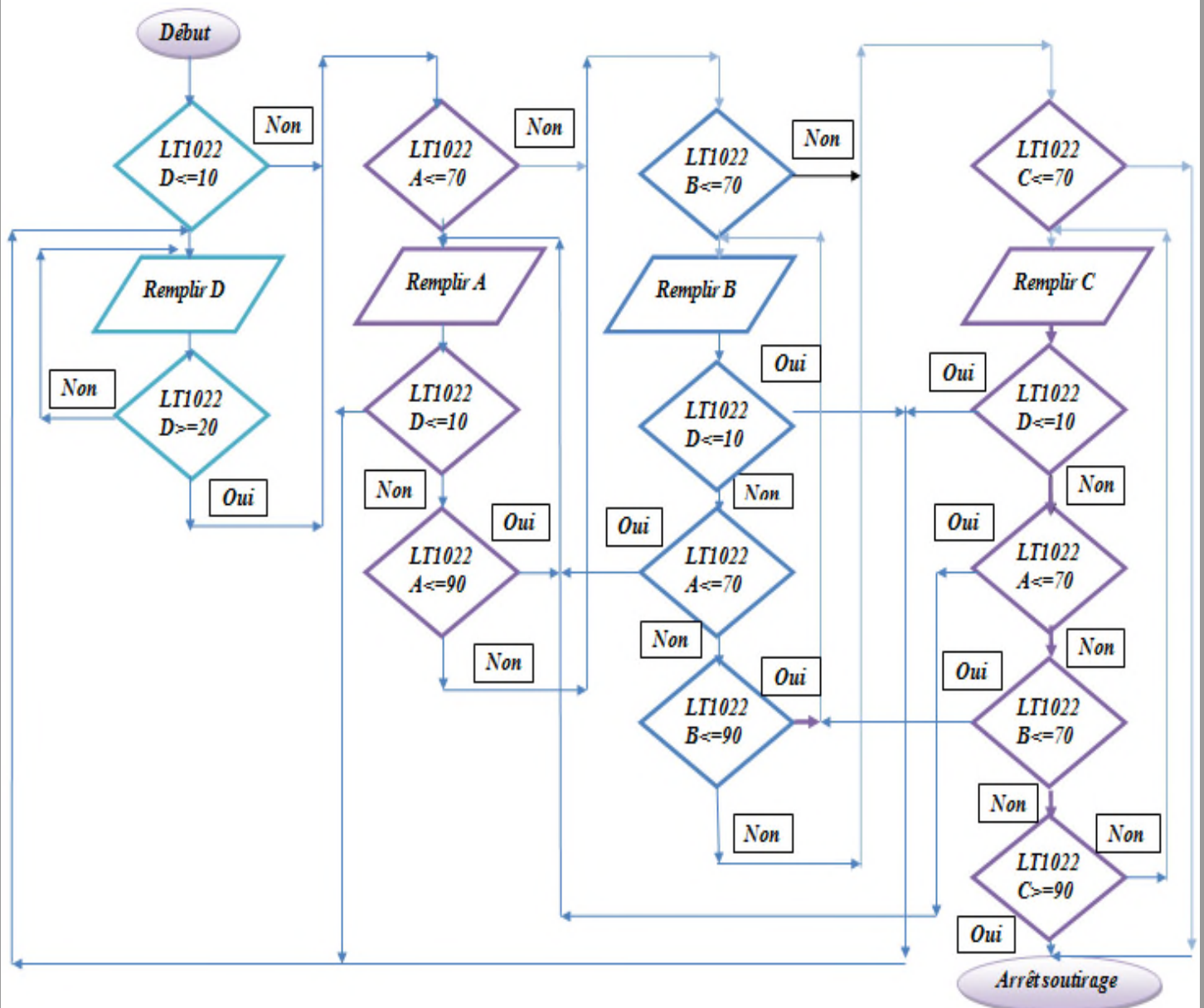
*Encadré par : Mr. Hassen RECHAME*

*Mr. Hocine LEHOUCHE*

---

## **A***nnexs*

---

Logigramme :

**Table des Mnémonique :**

Etat	Mnémonique	Opérande	Type de d	Commentaire
	A	VAT 1		
	ARRET_A1019 &A1007N	A 1.0	BOOL	ARRET DES TAPIS A1007N & A1019A
	ARRET_PROD	A 7.0	BOOL	ARRET DU PRODUIT
	ARRT_A1021A	E 5.1	BOOL	ARRET DU MOTEUR A1021A
	ARRT_A1021B	E 5.3	BOOL	ARRET DU MOTEUR A1021B
	CAD_LT1022A	M 150.0	BOOL	CADENCE DE LA TREMIE T1022A
	CAD_LT1022B	M 150.1	BOOL	CADENCE DE LA TREMIE T1022B
	CAD_LT1022C	M 150.2	BOOL	CADENCE DE LA TREMIE T1022C
	CAD_LT1022D	M 150.3	BOOL	CADENCE DE LA TREMIE T1022D
	CAD_VANNES	FC 9	FC 9	CADENCE DES VANNES
	CADENCE DES TREMIES	M 10.5	BOOL	
	CMD_EV1007NA	A 5.0	BOOL	commande de la vanne EV1007NA
	CMD_EV1007NB	A 4.2	BOOL	commande de la vanne EV1007NB
	CMD_EV1019A	A 4.6	BOOL	commande de la vanne EV1019A
	CMD_EV1019B	A 4.0	BOOL	commande de la vanne EV1019B
	CMD_EV1019C	A 4.4	BOOL	commande de la vanne EV1019C
	CMD_EV1021B	A 5.4	BOOL	commande de la vanne EV1021B
	CMD_EV1021C	A 5.2	BOOL	commande de la vanne EV1021C
	COMMANDE DE EV1007NA	FC 7	FC 7	OUVETURE ET FERMETURE DE EV1007NA
	COMMANDE DE EV1007NB	FC 3	FC 3	OUVETURE ET FERMETUTR EV1007NB
	COMMANDE DE EV1019A	FC 4	FC 4	OUVETURE ET FERMETUTR EV1019A
	COMMANDE DE EV1019B	FC 2	FC 2	OUVETURE ET FERMETUTR EV1019B
	COMMANDE DE EV1021B	FC 5	FC 5	OUVETURE ET FERMETUTR EV1021B
	COMMANDE DE EV1021C	FC 6	FC 6	OUVETURE ET FERMETUTR EV1021C
	DEF_FER_EV1007NA	M 30.7	BOOL	DEFAUT DE FERMETURE DE EV1007NA
	DEF_FER_EV1007NB	M 30.5	BOOL	DEFAUT DE FERMETURE DE EV1007NB
	DEF_FER_EV1019A	M 30.3	BOOL	DEFAUT DE FERMETURE DE EV1019A
	DEF_FER_EV1019B	M 30.1	BOOL	DEFAUT DE FERMETURE DE EV1019B
	DEF_FER_EV1021B	M 31.3	BOOL	DEFAUT DE FERMETURE DE EV1021B
	DEF_FER_EV1021C	M 31.1	BOOL	DEFAUT DE FERMETURE DE EV1021C
	DEF_OUV_EV1007NA	M 30.6	BOOL	DEFAUT D'OUVERTURE DE EV1007NA
	DEF_OUV_EV1007NB	M 30.4	BOOL	DEFAUT D'OUVERTURE DE EV1007NB
	DEF_OUV_EV1019A	M 30.2	BOOL	DEFAUT D'OUVERTURE DE EV1019A
	DEF_OUV_EV1019B	M 30.0	BOOL	DEFAUT D'OUVERTURE DE EV1019B
	DEF_OUV_EV1021B	M 31.2	BOOL	DEFAUT D'OUVERTURE DE EV1021B
	DEF_OUV_EV1021C	M 31.0	BOOL	DEFAUT D'OUVERTURE DE EV1021C
	DEM_PROD-LT1022A	M 20.0	BOOL	DEMANDE DE PRODUIT LT1022A
	DEM_PROD-LT1022B	M 20.1	BOOL	DEMANDE DE PRODUIT LT1022B
	DEM_PROD-LT1022C	M 20.2	BOOL	DEMANDE DE PRODUIT LT1022C
	DEM_PROD-LT1022D	M 20.3	BOOL	DEMANDE DE PRODUIT LT1022AD
	DISC_EV1007NA	A 10.2	BOOL	DISCORDONCE EV1007NA
	DISC_EV1007NB	A 10.3	BOOL	DISCORDONCE EV1007NB
	DISC_EV1019A	A 10.0	BOOL	DISCORDANCE EV1019A
	DISC_EV1019B	A 10.1	BOOL	DISCORDONCE EV1019B
	DISC_EV1021B	A 10.5	BOOL	DISCORDONCE EV1021B
	DISC_EV1021C	A 10.4	BOOL	DISCORDONCE EV1021A
	DPLC_EV1007NA	MW 508	WORD	LE DEPLACEMENT DE LA VANNE EV1007NA:FERMETURE & OUVERTURE
	DPLC_EV1007NB	MW 506	WORD	LE DEPLACEMENT DE LA VANNE EV1007NB:FERMETURE & OUVERTURE
	DPLC_EV1019A	MW 502	WORD	LE DEPLACEMENT DE LA VANNE EV1019A:FERMETURE & OUVERTURE
	DPLC_EV1019B	MW 500	WORD	LE DEPLACEMENT DE LA VANNE EV1019B:FERMETURE & OUVERTURE
	DPLC_EV1021B	MW 504	WORD	LE DEPLACEMENT DE LA VANNE EV1021B:FERMETURE & OUVERTURE
	DPLC_EV1021C	MW 510	WORD	LE DEPLACEMENT DE LA VANNE EV1021C:FERMETURE & OUVERTURE
	FDC_FER_EV1007NA	E 2.6	BOOL	FIN DE COURCE FERMETURE DE EV1007NA
	FDC_FER_EV1007NB	E 2.2	BOOL	FIN DE COURCE FERMETURE DE EV1007NB

55	FDC_FER_EV1019A	E	2.4	BOOL	FIN DE COURSE FERMETURE E EV1019A
56	FDC_FER_EV1019B	E	2.0	BOOL	FIN DE COURSE FERMETURE DE EV1019B
57	FDC_FER_EV1021B	E	3.2	BOOL	FIN DE COURSE FERMETURE DE EV1021B
58	FDC_FER_EV1021C	E	3.0	BOOL	FIN DE COURSE FERMETURE DE EV1021C
59	FDC_OUV_EV1007NA	E	2.7	BOOL	FIN DE COURSE OUVERTURE DE EV1007NA
60	FDC_OUV_EV1007NB	E	2.3	BOOL	FIN DE COURSE OUVERTURE DE EV1007NB
61	FDC_OUV_EV1019A	E	2.5	BOOL	FIN DE COURSE OUVERTURE DE EV1019A
62	FDC_OUV_EV1019B	E	2.1	BOOL	FIN DE COURSE OUVERTURE DE EV1019B
63	FDC_OUV_EV1021B	E	3.3	BOOL	FIN DE COURSE OUVERTURE DE EV1021B
64	FDC_OUV_EV1021C	E	3.1	BOOL	FIN DE COURSE OUVERTURE DE EV1021C
65	LSH_LT1022A	E	0.5	BOOL	LSH DE LA TREMIE LT1022A
66	LSH_LT1022B	E	0.6	BOOL	LSH DE LA TREMIE LT1022B
67	LSH_LT1022C	E	0.7	BOOL	LSH DE LA TREMIE LT1022C
68	LSH_LT1022D	E	1.0	BOOL	LSH DE LA TREMIE LT1022D
69	LT1022A	MD	10	REAL	NIVEAU LT1022A
70	LT1022B	MD	14	REAL	NIVEAU LT1022B
71	LT1022C	MD	18	REAL	NIVEAU LT1022C
72	LT1022D	MD	22	REAL	NIVEAU LT1022D
73	MEMO_EV1007NA	M	41.7	BOOL	Mémorisation EV1007NA
74	MEMO_EV1007NB	M	41.5	BOOL	Mémorisation EV1007NB
75	MEMO_EV1019A	M	41.6	BOOL	Mémorisation EV1019A
76	MEMO_EV1019B	M	41.4	BOOL	Mémorisation EV1019B
77	MEMO_EV1021B	M	42.1	BOOL	Mémorisation EV1021B
78	MEMO_EV1021C	M	42.0	BOOL	Mémorisation EV1021C
79	MISE A L'ECHELLE	FC	1	FC 1	BLOC TRAITAMT DES ENTREES ANALOGIQUE" MISE A L'ECHELLE DES TREMIES
80	MRCH_A1021A	E	5.0	BOOL	MARCHE DU MOTEUR A1021A
81	MRCH_A1021B	E	5.2	BOOL	MARCHE DU MOTEUR A1021B
82	RESET_EV1007NA	M	40.7	BOOL	Conditions fermeture de la vanne EV1007NA
83	RESET_EV1007NB	M	40.3	BOOL	Conditions fermeture de la vanne EV1007NB
84	RESET_EV1019A	M	40.5	BOOL	Conditions fermeture de la vanne EV1019A
85	RESET_EV1019B	M	40.1	BOOL	Conditions fermeture de la vanne EV1019B
86	RESET_EV1021B	M	41.3	BOOL	Conditions fermeture de la vanne EV1021B
87	RESET_EV1021C	M	41.1	BOOL	Conditions fermeture de la vanne EV1021C
88	SCALE	FC	105	FC 105	MISE A L'ECHELLE DES NIVEAUX DES TREMIES
89	SELECT_PROD	M	100.0	BOOL	BOUTON DE SELECTION DE LA 2000T
90	SET_EV1007NA	M	40.6	BOOL	Conditions d'ouverture de la vanne EV1007NA
91	SET_EV1007NB	M	40.2	BOOL	Conditions d'ouverture de la vanne EV1007NB
92	SET_EV1019A	M	40.4	BOOL	Conditions d'ouverture de la vanne EV1019A
93	SET_EV1019B	M	40.0	BOOL	Conditions d'ouverture de la vanne EV1019B
94	SET_EV1021B	M	41.2	BOOL	Conditions d'ouverture de la vanne EV1021B
95	SET_EV1021C	M	41.0	BOOL	Conditions d'ouverture de la vanne EV1021C
96	Stop_soutirage	M	50.5	BOOL	demande de fermeture des vannes de soutirage des silos de la 2000T et 3000T
97	T_OUV_MANU	E	0.4	BOOL	OUVERTURE MANUELLE
98	TEMPO_T1022A	T	11	TIMER	Durée ouverture manuelle vers T1022A
99	TEMPO_T1022B	T	12	TIMER	Durée ouverture manuelle vers T1022B
100	TEMPO_T1022C	T	13	TIMER	Durée ouverture manuelle vers T1022C
101	TEMPO_T1022D	T	14	TIMER	Durée ouverture manuelle vers T1022D
102	TEMPORISATEUR_2	T	2	TIMER	LE TEMPS DE DEF AUT D'OUVERTURE EV1019B
103	TEMPORISATEUR_20	T	20	TIMER	LE TEMPS DE DEF AUT D'OUVERTURE EV1007NB
104	TEMPORISATEUR_3	T	3	TIMER	LE TEMPS DE DEF AUT DE FERMETURE EV1019B
105	TEMPORISATEUR_30	T	30	TIMER	LE TEMPS DE DEF AUT DE FERMETURE EV1007NB
106	TEMPORISATEUR_4	T	4	TIMER	LE TEMPS DE DEF AUT D'OUVERTURE EV1019A
107	TEMPORISATEUR_40	T	40	TIMER	LE TEMPS DE DEF AUT D'OUVERTURE EV1021B
108	TEMPORISATEUR_5	T	5	TIMER	LE TEMPS DE DEF AUT D'OUVERTURE EV1019A
109	TEMPORISATEUR_50	T	50	TIMER	LE TEMPS DE DEF AUT DE FERMETURE EV1021B
110	TEMPORISATEUR_60	T	60	TIMER	LE TEMPS DE DEF AUT D'OUVERTURE EV1021C
111	TEMPORISATEUR_70	T	70	TIMER	LE TEMPS DE DEF AUT DE FERMETURE EV1021C
112	TEMPORISATEUR_80	T	80	TIMER	LE TEMPS DE DEF AUT D'OUVERTURE EV1007NA
113	TEMPORISATEUR_90	T	90	TIMER	LE TEMPS DE DEF AUT DE FERMETURE EV1007NA
114	TEST1	VAT	2		LE TESTE DU PROGRAMME AVEC DES VALEURS CHOISIS
115					

Pour obtenir de l'aide, appuyez sur F1.

NUM

**OBI :**

OBI : "Main Program Sweep (Cycle)"

Commentaire :

Réseau 1: Titre :

faire appeller toutes les fonction du systeme

```
CALL "MISE A L'ECHELLE"          FC1          -- BLOC TRAITAMT DES ENTREES ANALOGIQUE"MISE A L'ECHELLE DES TREMIES
CALL "COMMANDE DE EV1019B"       FC2          -- OUVERTURE ET FERMETUTR EV1019B
CALL "COMMANDE DE EV1007NB"      FC3          -- OUVERTURE ET FERMETUTR EV1007NB

CALL "COMMANDE DE EV1019A"       FC4          -- OUVERTURE ET FERMETUTR EV1019A
CALL "COMMANDE DE EV1021B"       FC5          -- OUVERTURE ET FERMETUTR EV1021B
CALL "COMMANDE DE EV1021C"       FC6          -- OUVERTURE ET FERMETUTR EV1021C
CALL "COMMANDE DE EV1007NA"      FC7          -- OUVERTURE ET FERMETURE DE EV1007NA
CALL FC      8
CALL "CAD_VANNES"                FC9          -- CADENCE DES VANNES

CALL FC      10
```



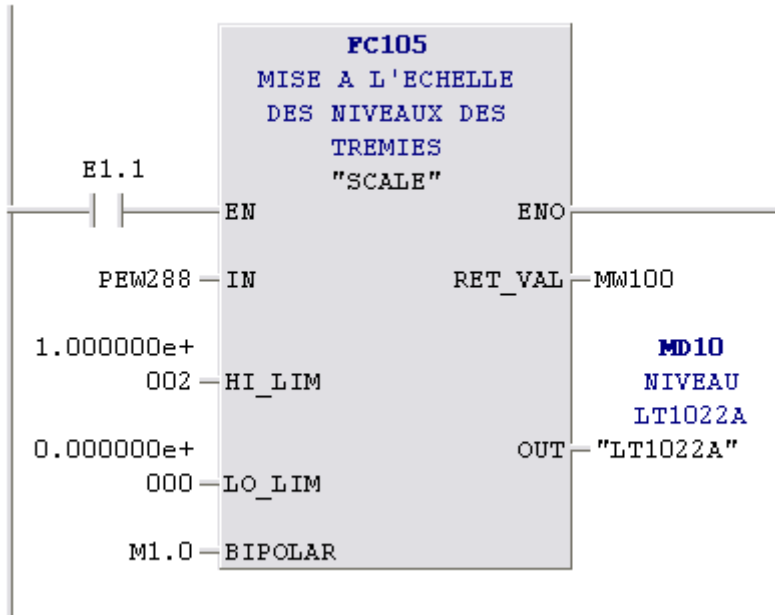
**FC1:**

FC1 : Titre :

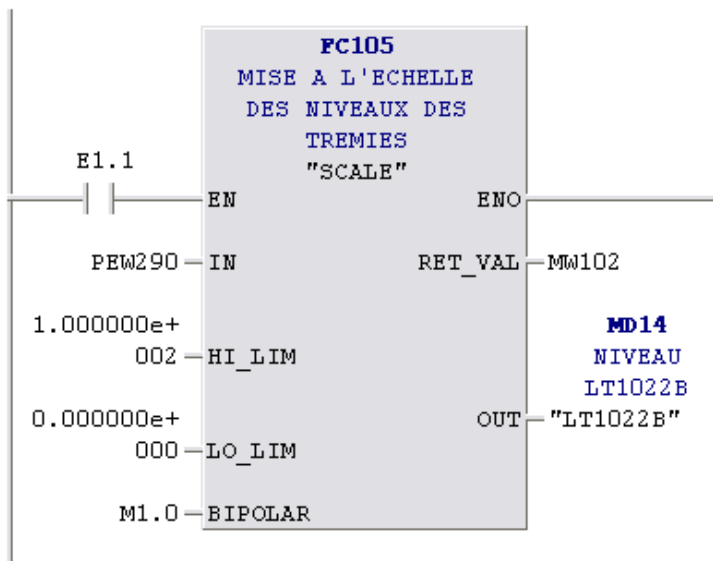
mise a l'echelle des niveaux des tremies

**Réseau 1** : Titre :

mise à l'echelle du niveau LT1022A

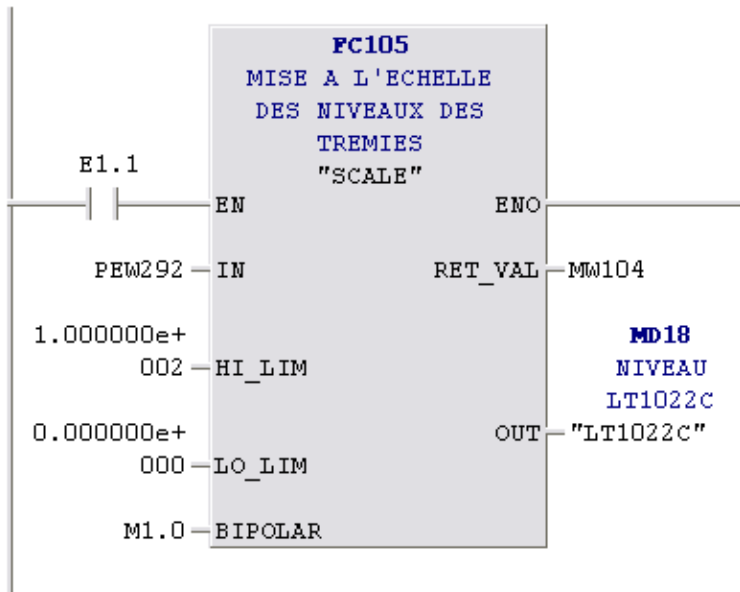
**Réseau 2** : Titre :

mise à l'echelle du niveau LT1022B



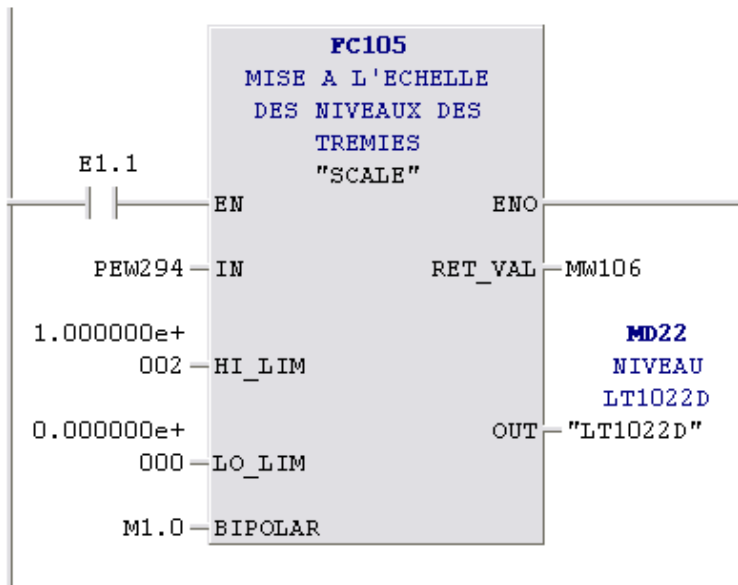
## Réseau 3 : Titre :

mise à l'échelle du niveau LT1022C



## Réseau 4 : Titre :

mise à l'échelle du niveau LT1022D



**FC2 :**

FC2 : Titre :

OUVERTURE ET FERMETURE DE EV1019B

**Réseau 1 : OUVERTURE EV1019B**

OUVERTURE DE EV1019B:

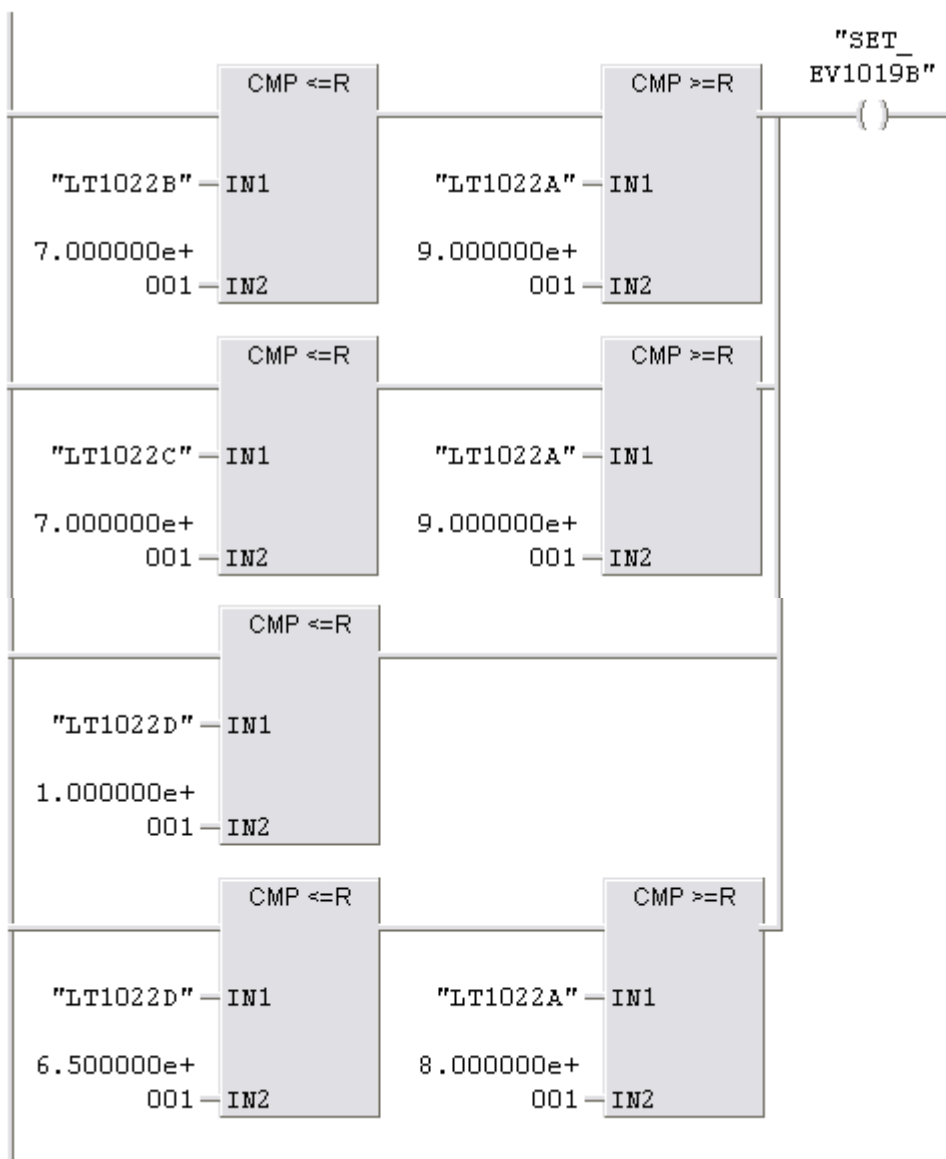
S'IL Y'A SELECTION DU PRODUIT DE LA 2000T ET QUE

SI LE NIVEAU DE LA TREMIE LT1022B&lt;=70% ET QUE LE NIVEAU DE LT1022A&gt;=90%

OU LE NIVEAU DE LA TREMIE LT1022C&lt;=70% ET QUE LE NIVEAU DE LT1022A&gt;=90%

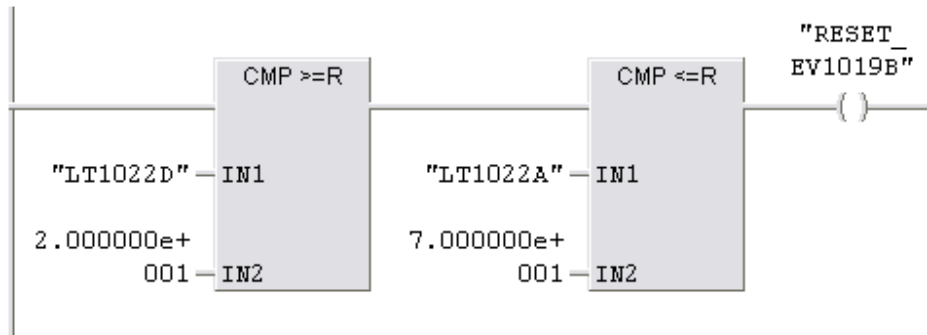
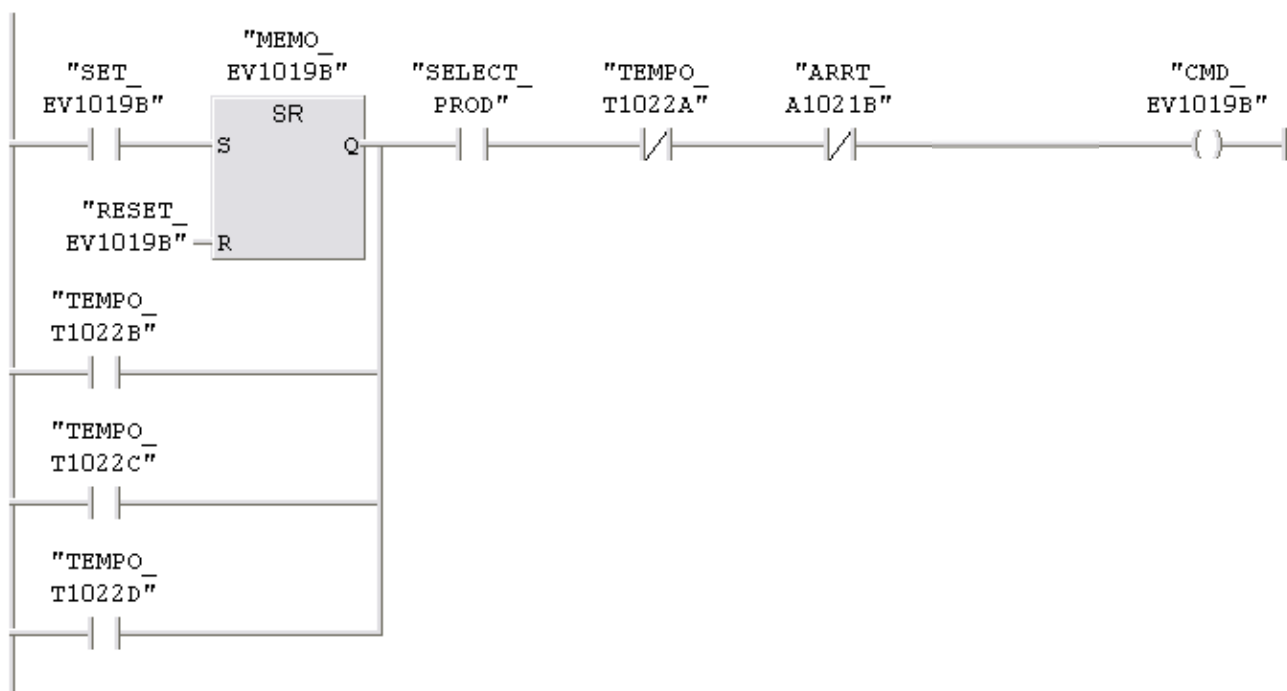
OU LE NIVEAU DE LT1022D&lt;=10%

OU LE NIVEAU DE LA TREMIE LT1022D&lt;=65% ET QUE LE NIVEAU DE LT1022A&gt;=80%



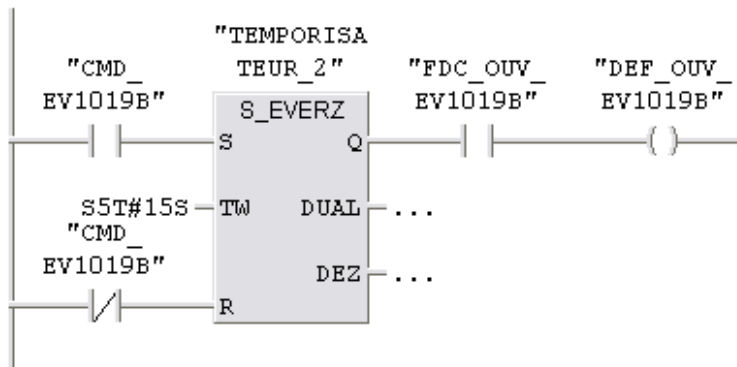
**Réseau 2 : FERMETUTR EV1019B**

On ferme la vanne que si la trémie T1022A manque de sucre ou le circuit de la 3000T est sélectionné

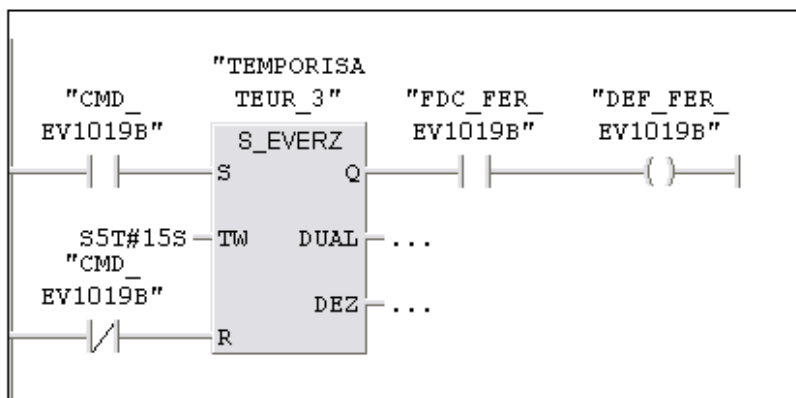
**Réseau 3 : arrêt soutirage des silos de maturation**

**Réseau 4** : GESTION DES DISCORDANCE DE LA VANNE EV1019B (DEFAULT D'OUVERTURE)

SI LE FIN DE COURSE D'OUVERTURE DE LA VANNE EV1019B N'EST PAS ATTEINT APRÈS 15 SECONDES DONC Y'A UN PROBLEME D'OUVERTURE QUI VA ETRE SIGNALER

**Réseau 5** : GESTION DES DISCORDANCE DE LA VANNE EV1019B (DEFAULT DE FERMETURE)

SI LE FIN DE COURSE DE FERMETURE DE LA VANNE EV1019B N'EST PAS ATTEINT APRÈS 15 SECONDE DONC Y'A UN PROBLEME DE FERMETURE QUI VA ETRE SIGNALER

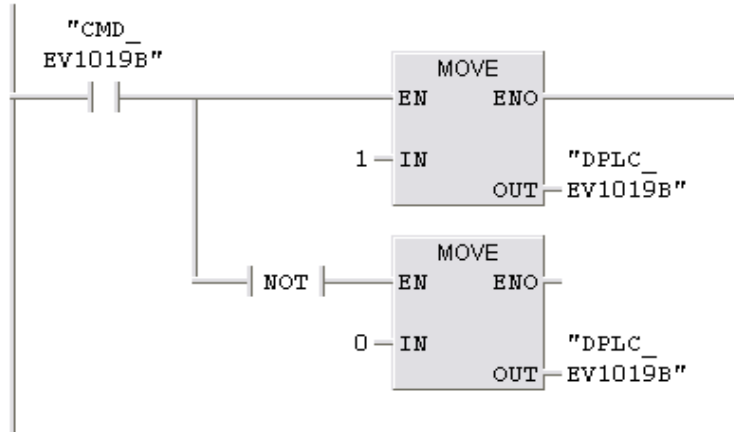
**Réseau 6** : DISCORDONCE EV1019B

Commentaire :



**Réseau 7** : Titre :

LE DEPLACEMENT DE LA VANNE EV1019B A LA FERMETURE ET A L'OUVERTURE

**FC3**:

FC3 : OUVERTURE ET FERMETURE DE LA VANNE EV1007NB

OUVERTURE ET FERMETURE DE LA VANNE EV1007NB

**Réseau 1** : OUVERTURE EV1007NB

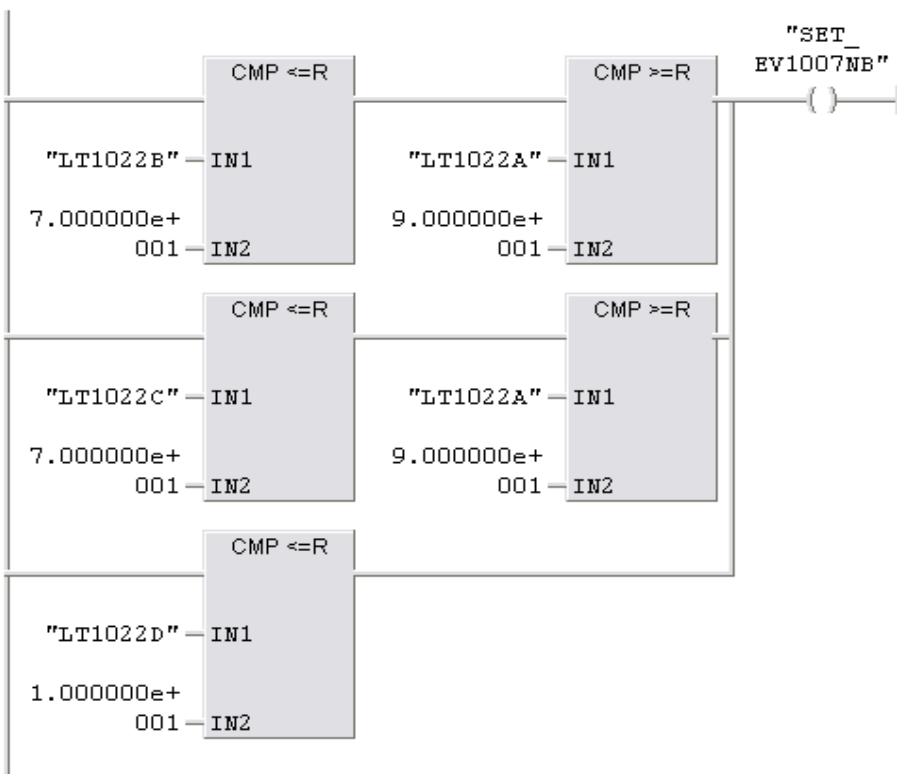
OUVERTURE DE EV1019B:

S'IL Y'A SELECTION DU PRODUIT DE LA 3000T ET QUE

SI LE NIVEAU DE LA TREMIE LT1022B<=70% ET QUE LE NIVEAU DE LT1022A>=90%

OU LE NIVEAU DE LA TREMIE LT1022C<=70% ET QUE LE NIVEAU DE LT1022A>=90%

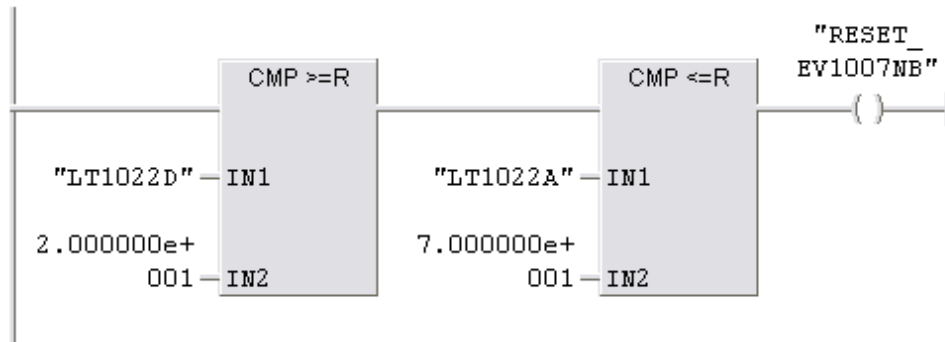
OU LE NIVEAU DE LT1022D<=10%



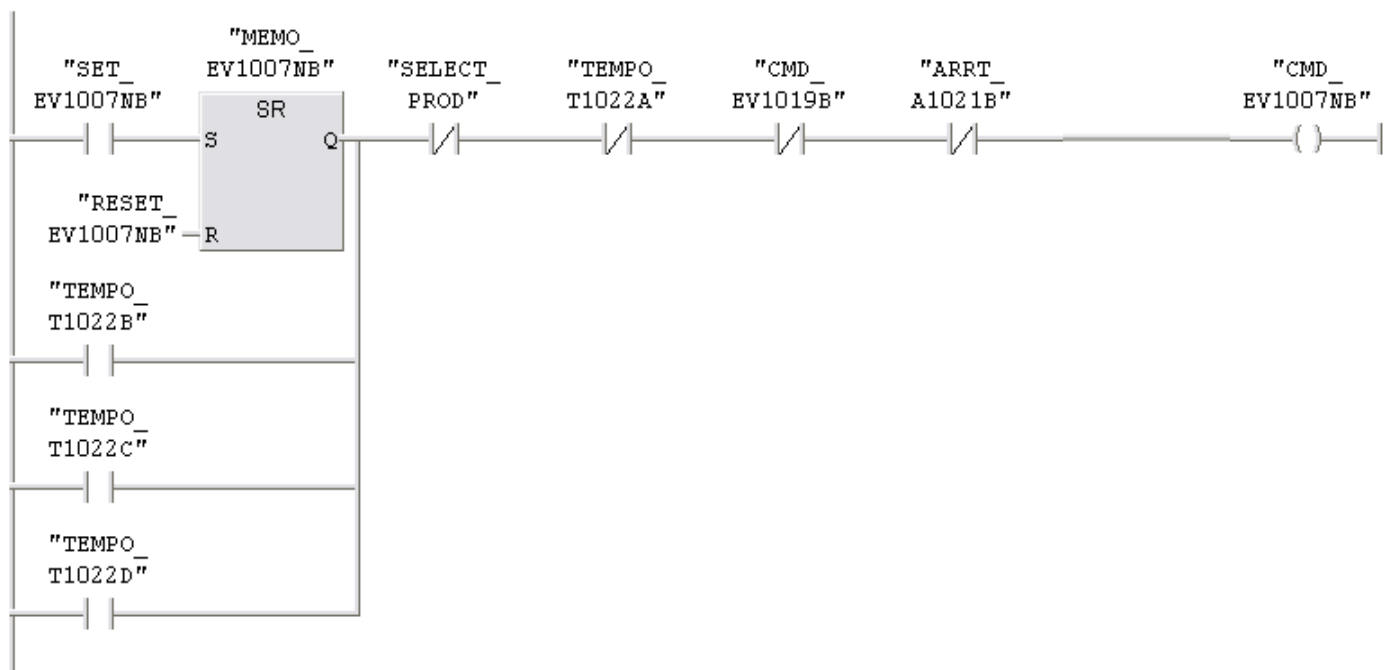
**Réseau 2 : FERMETURE EV1007NB**

FERMETURE DE EV1007NB

la vanne EV1007NB se ferme que si la trémie T1022A manque de produit ou le circuit de la 2000T est sélectionné.

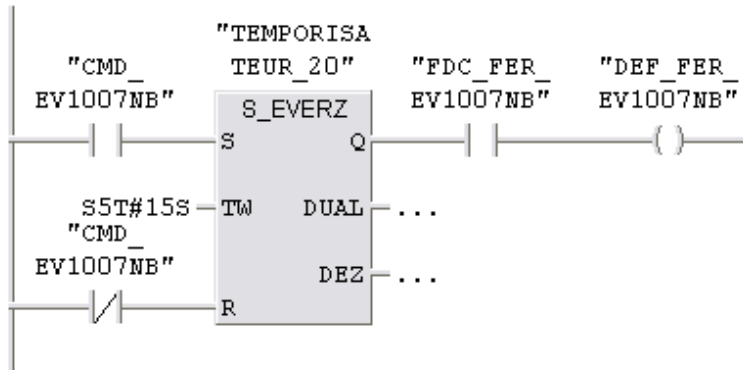
**Réseau 3 : OUVERTURE EV1007NB**

SI LA VANNE EV1019B EST FERMÉE OU BIEN LA CONDITION DU DEUXIEME RESEAU

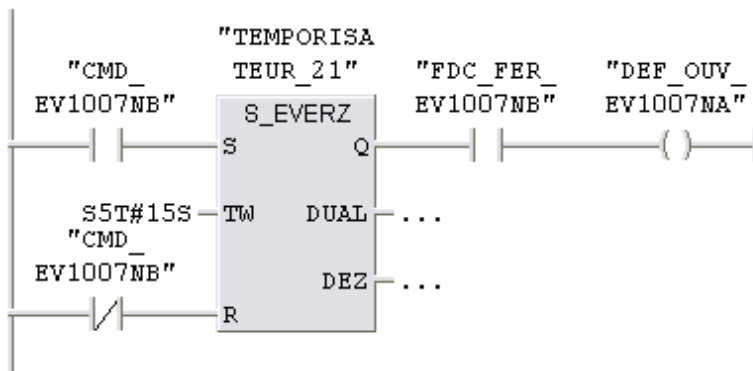


**Réseau 4 : GESTION DE DISCORDANCE DE LA VANNE EV1007NB (DEFAULT D'OUVERTURE)**

SI LE FIN DE COURSE D'OUVERTURE DE LA VANNE EV1007NB N'EST PAS ATTEINT APRÈS 15 SECONDE DONC Y'A UN PROBLEME D'OUVERTURE QUI VA ETRE SIGNALER

**Réseau 5 : GESTION DES DISCORDANCE DE LA VANNE EV1007NB (DEFAULT DE FERMETUR)**

SI LE FIN DE COURSE DE FERMETURE DE LA VANNE EV1007NB N'EST PAS ATTEINT APRÈS 15 SECONDE DONC Y'A UN PROBLEME DE FERMETURE QUI VA ETRE SIGNALER

**Réseau 6 : DISCORDONCE EV1007NB**

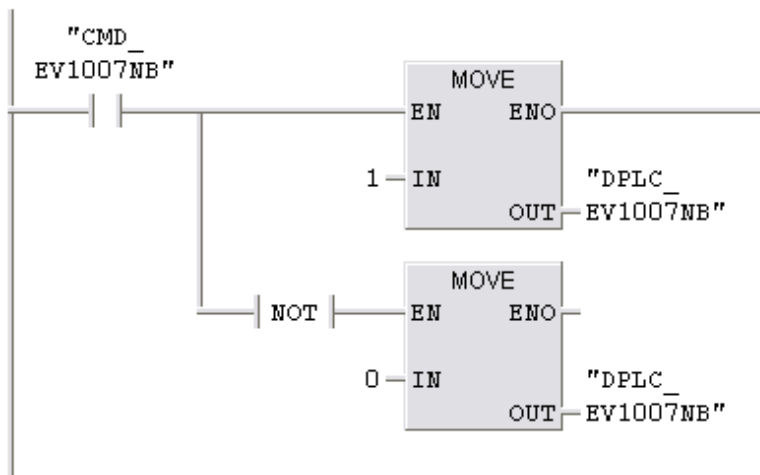
Commentaire :





**Réseau 7** : Titre :

LE DEPLACEMENT DE LA VANNE EV1007NB A LA FERMETURE ET A L'OUVERTURE

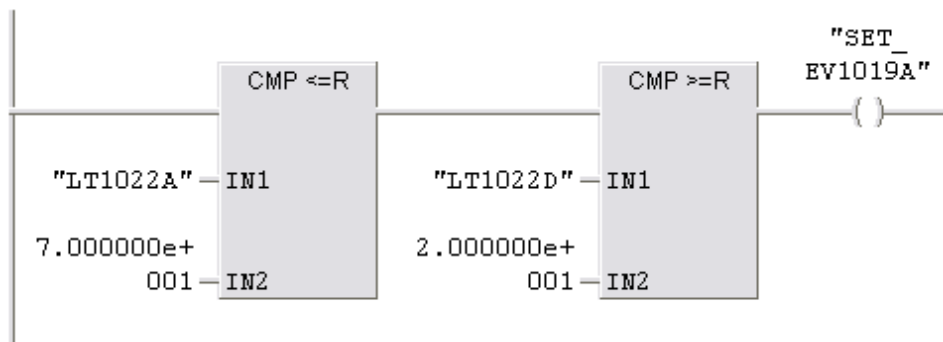
**FC4:**

## FC4 : Titre :

OUVERTURE ET FERMETURE DE LA VANNE EV1019A

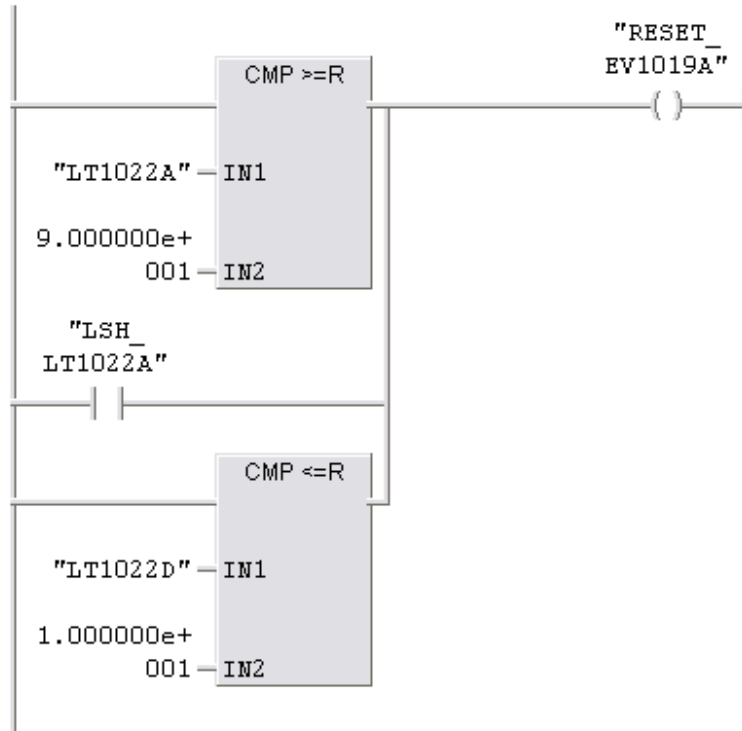
**Réseau 1** : OUVERTURE EV1019A

SI LE NIVEAU DU PRODUIT DANS LA TREMIE LT1022A <= 70% ET QUE LE NIVEAU DANS LA TREMIE LT1022D >= 20% ET QUE LE BOUTON DE SELECTION DE PRODUIT DE LA 2000T EST SELECTIONNÉ

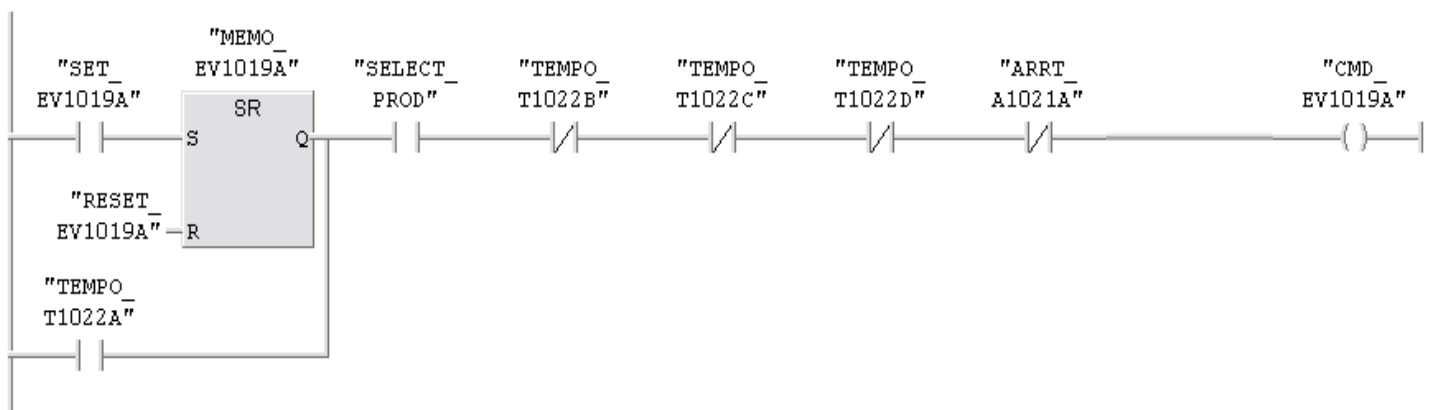


**Réseau 2 : FERMETUTR EV1019A**

SI LE NIVEAU DU PRODUIT DANS LA TREMIE LT1022A >= 90% OU LE NIVEAU DANS LA TREMIE LT1022D >= 10% OU IL Y'A LSH DANS LT1022A ET QUE LA SELECTION DU PRODUIT EST LA 3000T

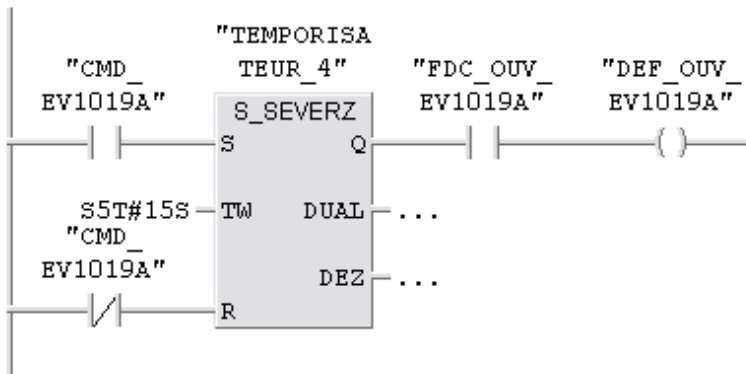
**Réseau 3 : Mémorisation EV1019A**

Commentaire :

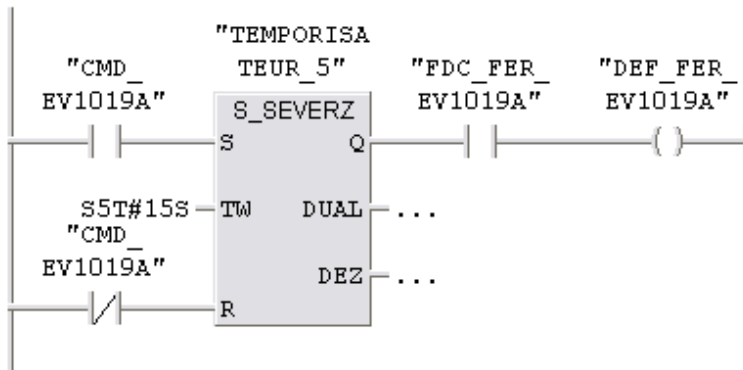


**Réseau 4 : GESTION DE DICORDANCE DE LA VANNE EV1019A (DEFAULT D'OUVERTURE)**

SI LE FIN DE COURSE D'OUVERTURE DE LA VANNE EV1019A N'EST PAS ATTEINT APRÈS 15 SECONDE DONC Y'A UN PROBLEME D'OUVERTURE QUI VA ETRE SIGNALER

**Réseau 5 : GESTION DE DICORDANCE DE LA VANNE EV1019A (DEFAULT DE FERMETURE)**

SI LE FIN DE COURSE DE FERMETURE DE LA VANNE EV1019A N'EST PAS ATTEINT APRÈS 15 SECONDE DONC Y'A UN PROBLEME DE FERMETURE QUI VA ETRE SIGNALER

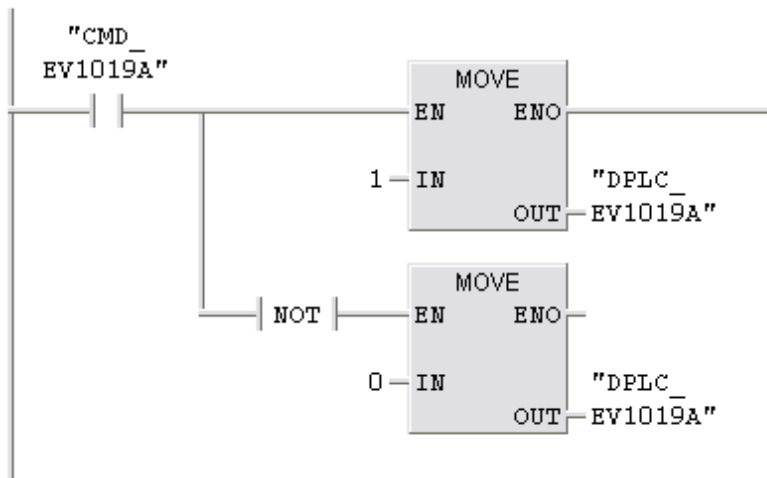
**Réseau 6 : DISCORDANCE EV1019A**

Commentaire :



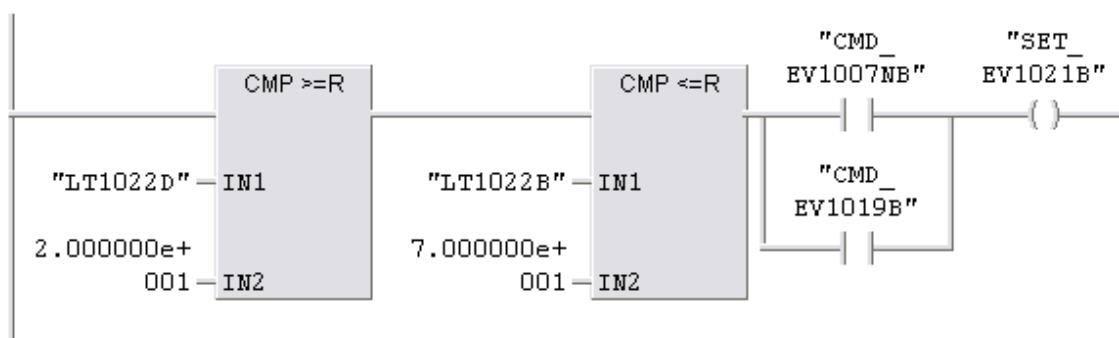
Réseau 7 : Titre :

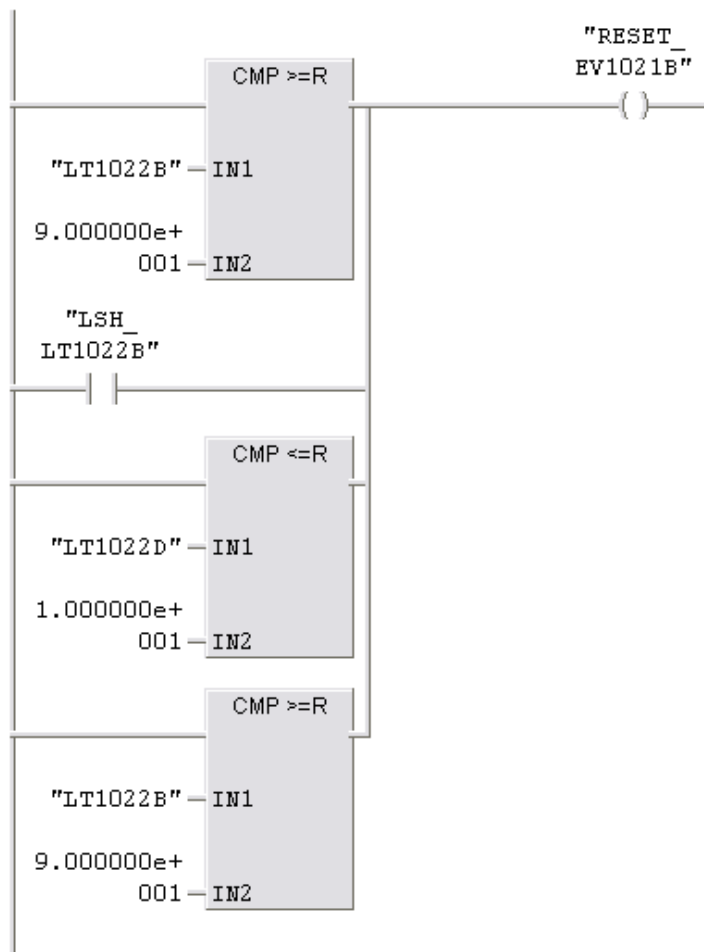
LE DEPLACEMENT DE LA VANNE EV1019A A LA FERMETURE ET A L'OUVERTURE

**FC5:**

FC5 : Titre :

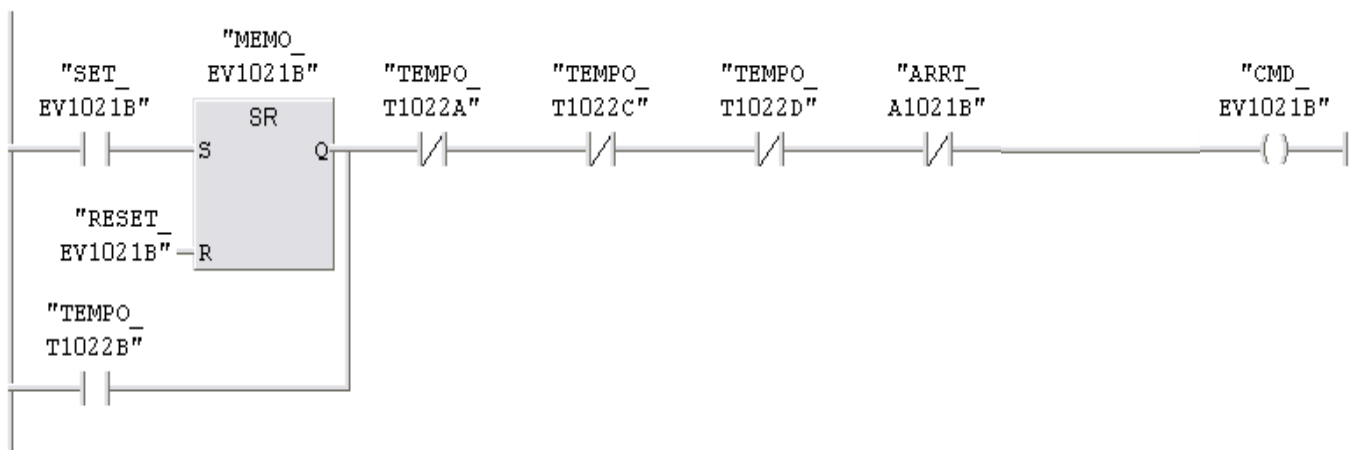
OUVERTURE ET FERMETURE DE LA VANNE EV1021B

**Réseau 1:** OUVERTURE EV1021BSI LE NIVEAU DU PRODUIT DANS LA TREMIE LT1022D  $\geq 20\%$  ET QUE LE NIVEAU DANS LA TREMIE LT1022B  $\leq 70\%$  ET QUE LA SELECTION DU PRODUIT EST LA 2000T**Réseau 2:** FERMETURE EV1021BSI LE NIVEAU DU PRODUIT DANS LA TREMIE LT1022B  $\geq 90\%$  OU LE NIVEAU DANS LA TREMIE LT1022D  $\leq 10\%$  OU LE NIVEAU DE LA TREMIE LT1022B  $\geq 90\%$  OU IL Y'A LSH DANS LA TREMIE LT1022B ET QUE LA SELECTION DU PRODUIT EST LA 3000T



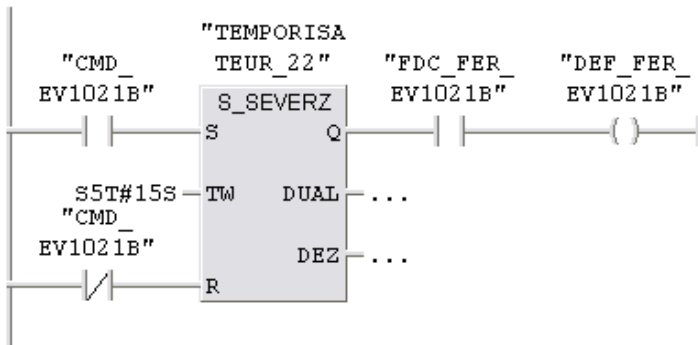
Réseau 3 : Mémorisation EV1021B

Commentaire :



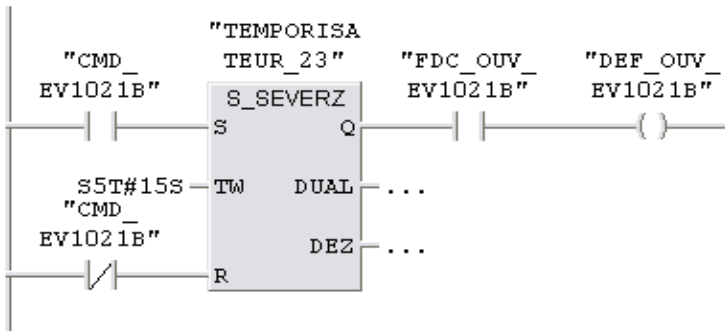
Réseau 4 : GESTION DE DICORDANCE DE LA VANNE EV1021B (DEFAULT DE FERMETURE)

SI LE FIN DE COURSE DE FERMETURE DE LA VANNE EV1021B N'EST PAS ATTEINT APRÈS 15 SECONDE DONC Y'A UN PROBLEME DE FERMETURE QUI VA ETRE SIGNALER



**Réseau 5 :** GESTION DE DICORDANCE DE LA VANNE EV1021B (DEFAULT D'OUVERTURE)

SI LE FIN DE COURSE D'OUVERTURE DE LA VANNE EV1021B N'EST PAS ATTEINT APRÈS 15 SECONDE DONC Y'A UN PROBLEME D'OUVERTURE QUI VA ETRE SIGNALER



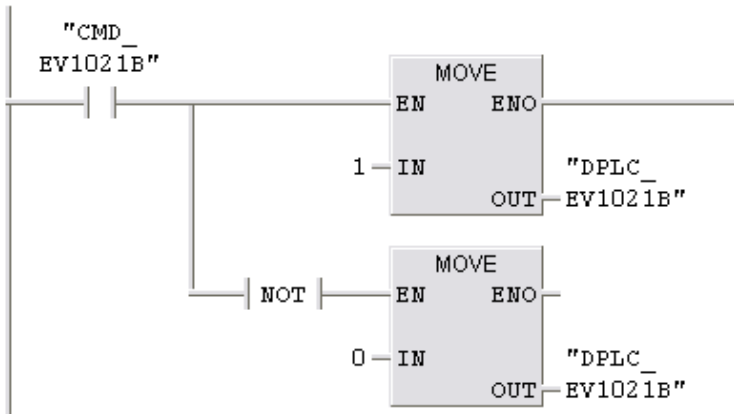
**Réseau 6 :** DISCORDANCE EV1021B

Commentaire :



**Réseau 7 :** Titre :

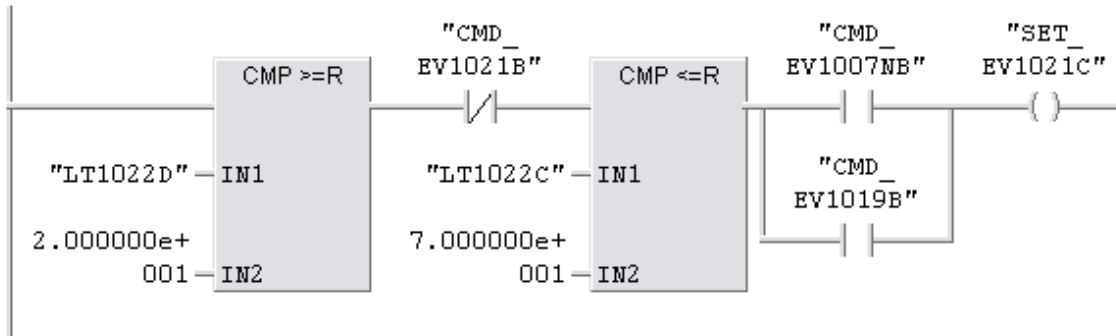
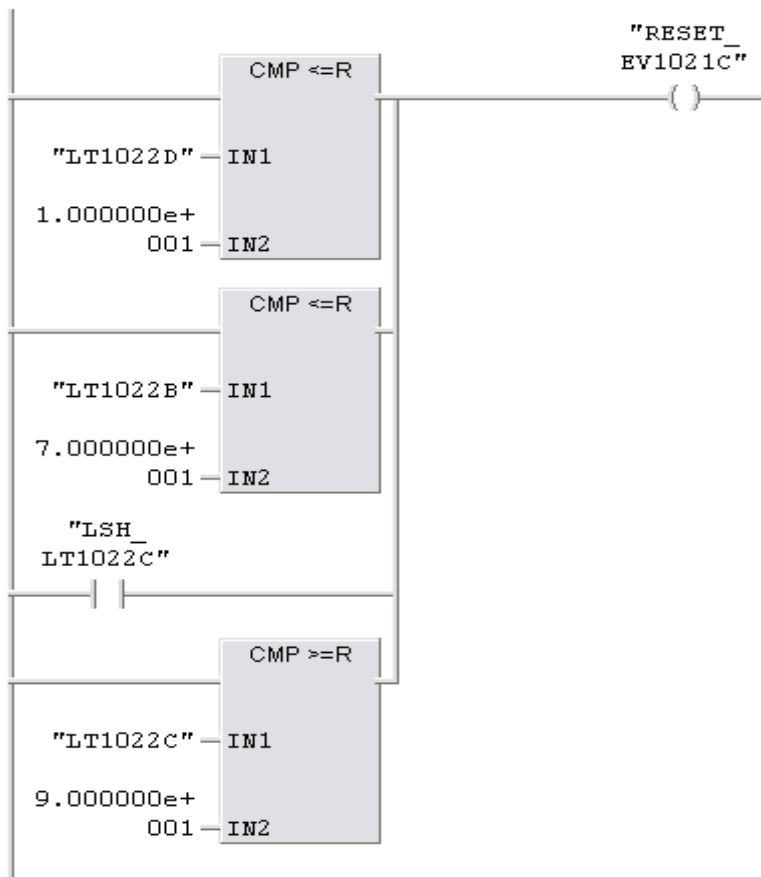
LE DEPLACEMENT DE LA VANNE EV1021B A LA FERMETURE ET A L'OUVERTURE



**FC6:**

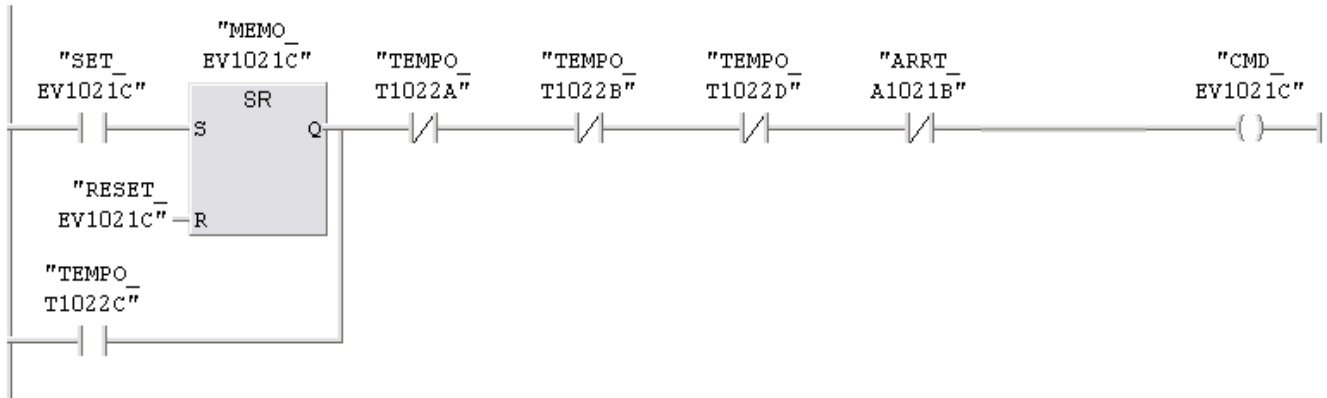
FC6 : Titre :

OUVERTURE ET FERMETURE DE LA VANNE EV1021C

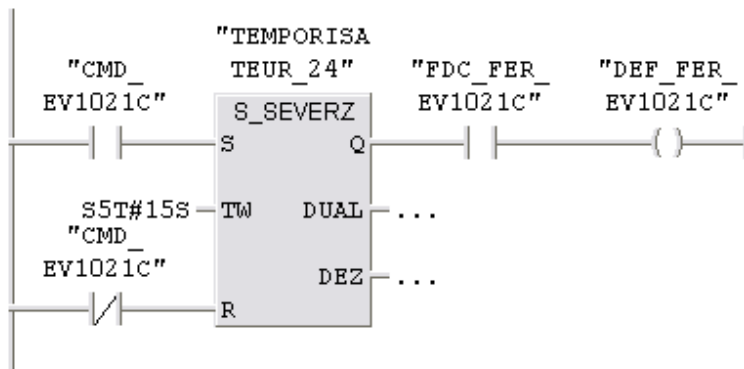
**Réseau 1** : OUVERTURE EV1021CSI LE NIVEAU DU PRODUIT DANS LA TREMIE LT1022D $\geq$ 20% ET QUE LE NIVEAU DANS LA TREMIE LT1022B $\geq$ 70% ET QUE LA SELECTION DU PRODUIT EST LA 2000T**Réseau 2** : FERMETUTR EV1021CSI LE NIVEAU DU PRODUIT DANS LA TREMIE LT1022B $\geq$ 70% OU LE NIVEAU DANS LA TREMIE LT1022D $\leq$ 10% OU LE NIVEAU DE LA TREMIE LT1022C $\geq$ 90% OU IL Y'A LSH DANS LA TREMIE LT1022C ET QUE LE NIVEAU DANS LT1022B $\leq$ 70% ET QUE LA SELECTION DU PRODUIT EST LA 3000T

**Réseau 3 : Mémorisation EV1021C**

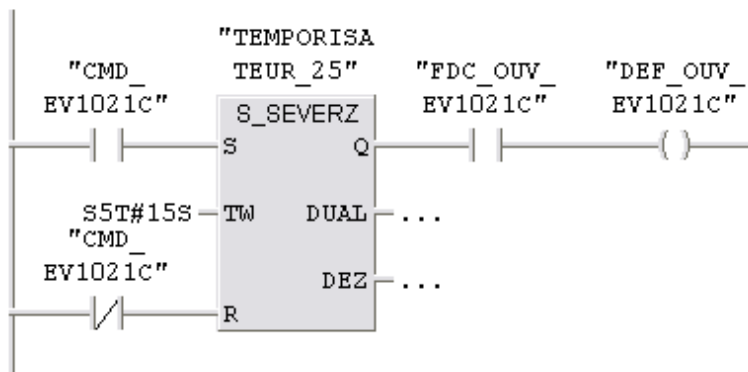
Commentaire :

**Réseau 4 : GESTION DE DICORDANCE DE LA VANNE EV1021C (DEFAULT DE FERMETURE)**

SI LE FIN DE COURSE DE FERMETURE DE LA VANNE EV1021C N'EST PAS ATTEINT APRÈS 15 SECONDE DONC Y'A UN PROBLEME DE FERMETURE QUI VA ETRE SIGNALER

**Réseau 5 : GESTION DE DICORDANCE DE LA VANNE EV1021BC (DEFAULT D'OUVERTURE)**

SI LE FIN DE COURSE D'OUVERTURE DE LA VANNE EV1021C N'EST PAS ATTEINT APRÈS 15 SECONDE DONC Y'A UN PROBLEME D'OUVERTURE QUI VA ETRE SIGNALER



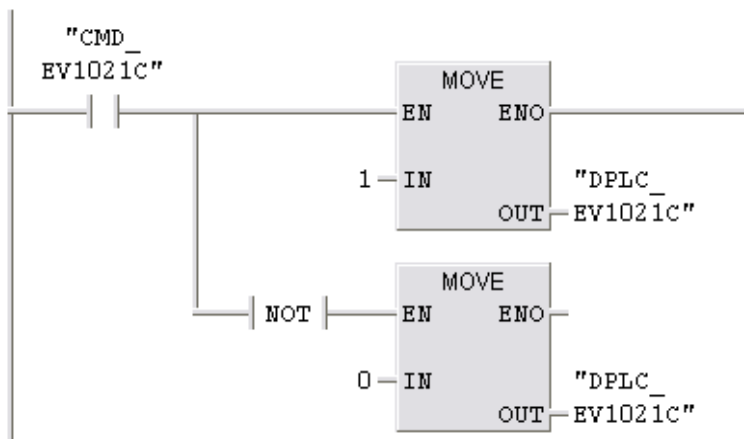


**Réseau 6 : DISCORDONCE EV1021A**

Commentaire :

**Réseau 7 : Titre :**

LE DEPLACEMENT DE LA VANNE EV1021C A LA FERMETURE ET A L'OUVERTURE

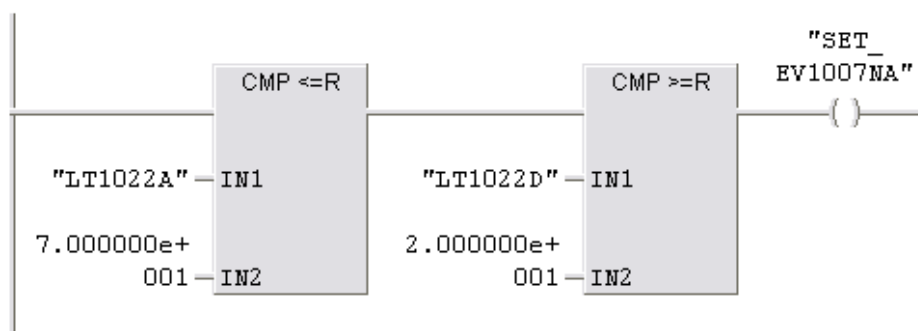
**FC7 :**

FC7 : Titre :

OUVERTURE ET FERMETURE DE LA VANNE EV1007NA

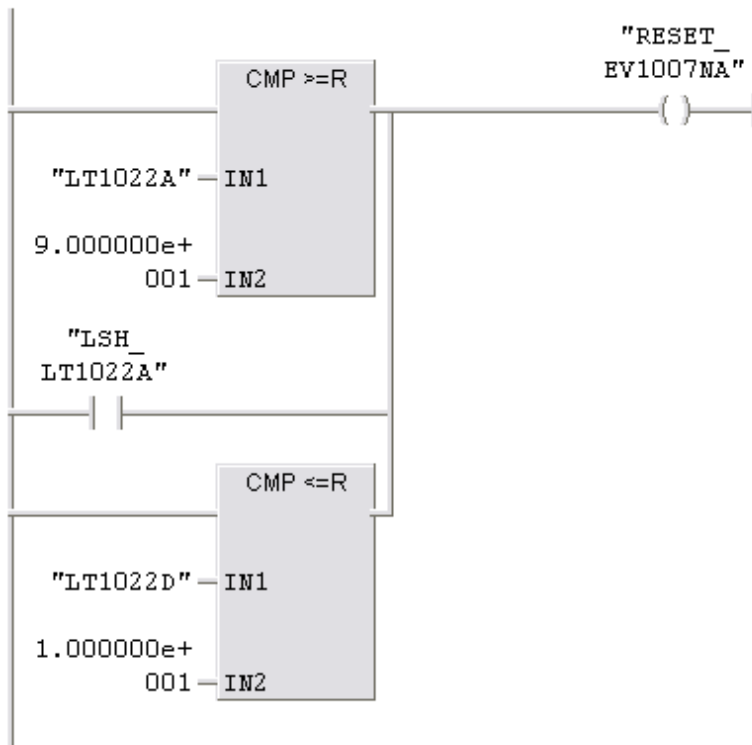
**Réseau 1 :** OUVERTURE EV1007NA

SI LE NIVEAU DU PRODUIT DANS LA TREMIE LT1022A &lt;= 70% ET QUE LE NIVEAU DANS LA TREMIE LT1022D &gt;= 20% ET QUE LE BOUTON DE SELECTION DE PRODUIT DE LA 3000T EST SELECTIONNÉ

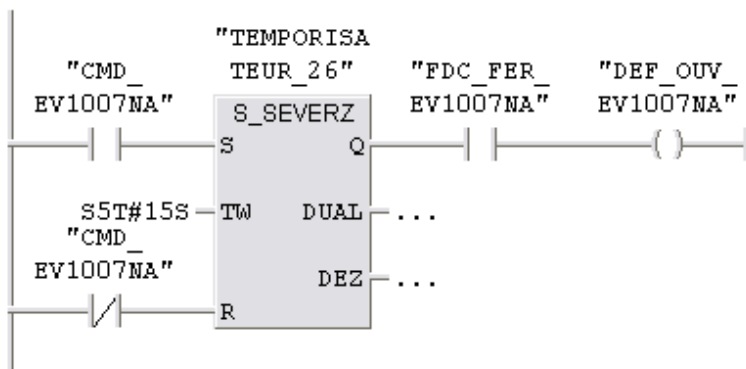


**Réseau 2 : FERMETUTR EV1007NA**

SI LE NIVEAU DU PRODUIT DANS LA TREMIE LT1022A >= 90% OU LE NIVEAU DANS LA TREMIE LT1022D >= 10% OU IL Y'A LSH DANS LT1022A ET QUE LA SELECTION DU PRODUIT EST LA 2000T

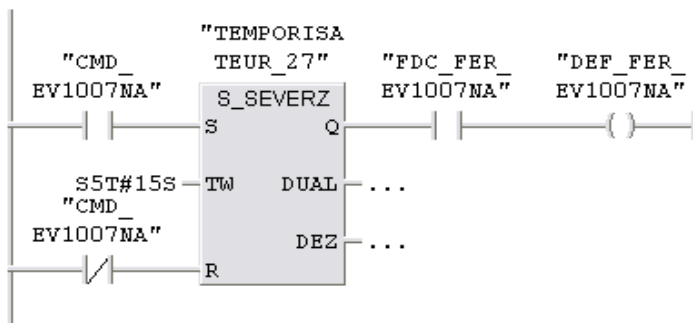
**Réseau 3 : GESTION DE DICORDANCE DE LA VANNE EV1007NA (DEFAULT D'OUVERTURE)**

SI LE FIN DE COURSE D'OUVERTURE DE LA VANNE EV1007NA N'EST PAS ATTEINT APRÈS 15 SECONDE DONC Y'A UN PROBLEME D'OUVERTURE QUI VA ETRE SIGNALER

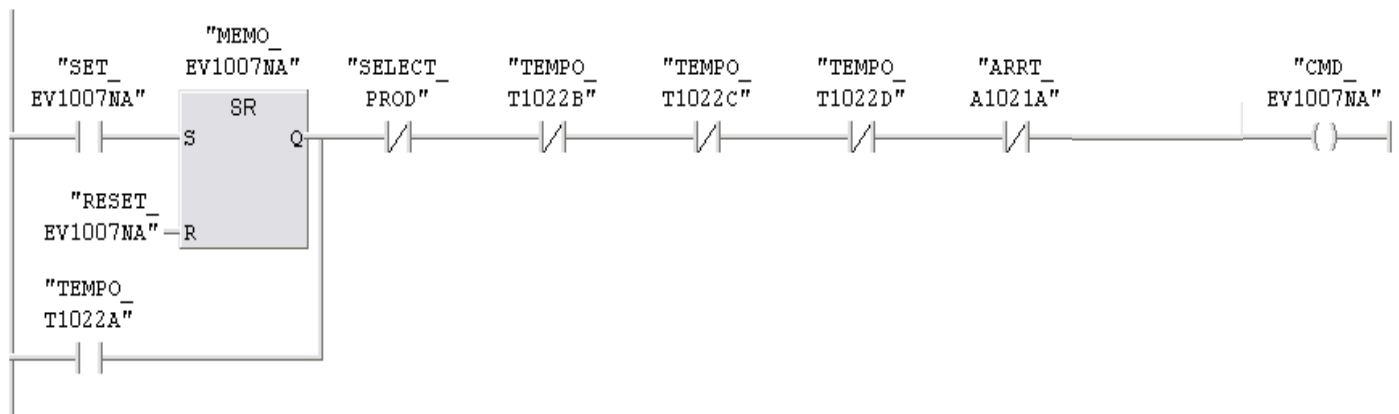


**Réseau 4 : GESTION DE DISCORDANCE DE LA VANNE EV1007NA (DEFAULT DE FERMETURE)**

SI LE FIN DE COURSE DE FERMETURE DE LA VANNE EV1007NA N'EST PAS ATTEINT APRÈS 15 SECONDE DONC Y'A UN PROBLEME DE FERMETURE QUI VA ETRE SIGNALER

**Réseau 5 : Mémorisation EV1007NA**

Commentaire :

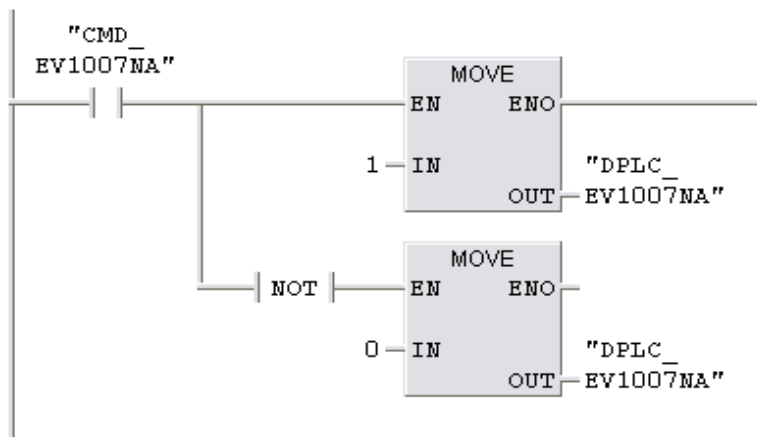
**Réseau 6 : DISCORDANCE EV1007NA**

Commentaire :



Réseau 7 : Titre :

LE DEPLACEMENT DE LA VANNE EV1007NA A LA FERMETURE ET A L'OUVERTURE

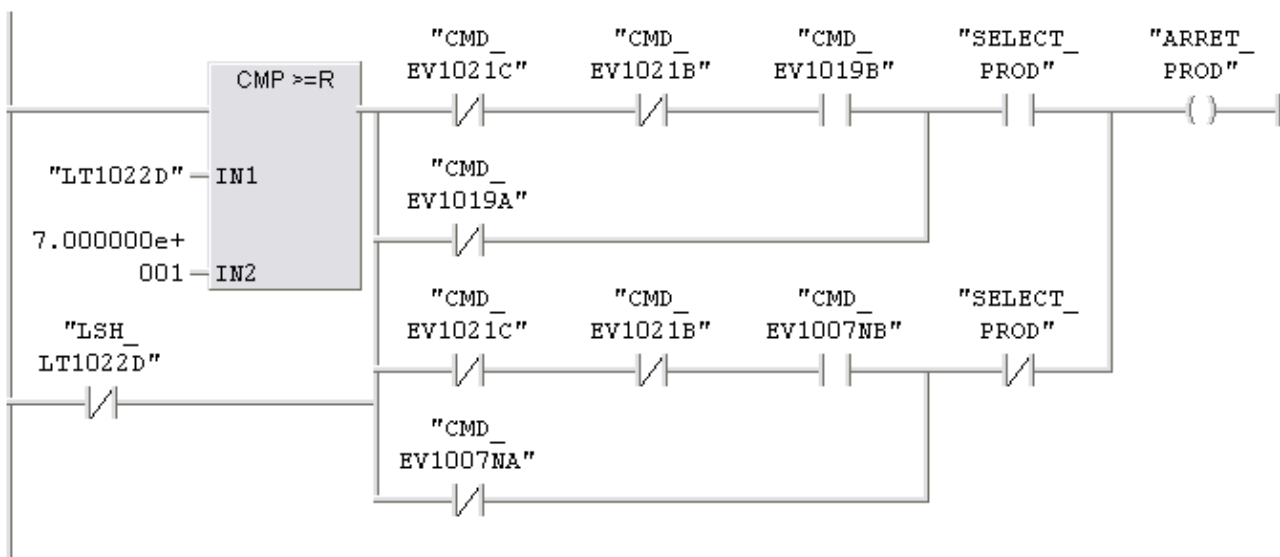
**FC8:**

FC8 : Titre :

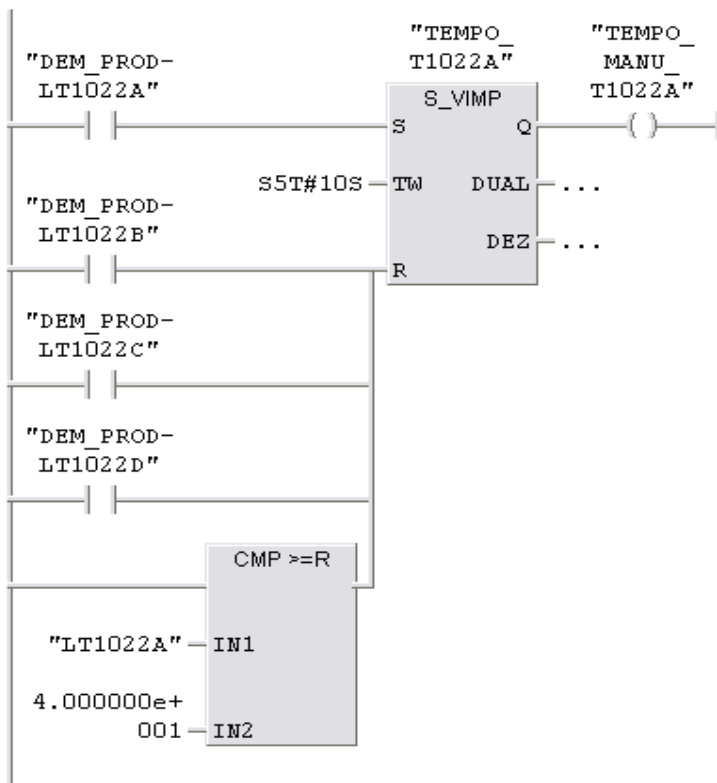
Commentaire :

**Réseau 1:** arrêt soutirage

ARRÊT DE PRODUIT DANS LE CAS OU TOUTES LES TREMIES SONT PLEINE ET QUE LA T1022D EST A 70% IL FAUT ARRÊTER LA DEMANDE DE PRODUIT POUR QUE LE CIRCUIT PUISSE SE VIDER DANS T1022D

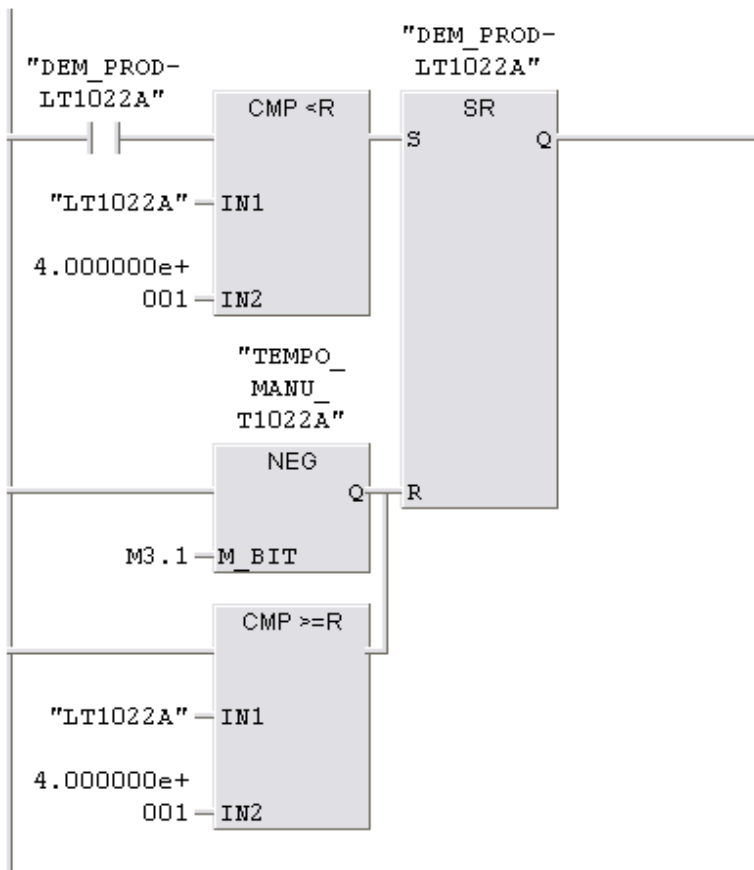
**Réseau 2:** Appel de produit T1022A

Demande manuelle de produit vers la T1022A



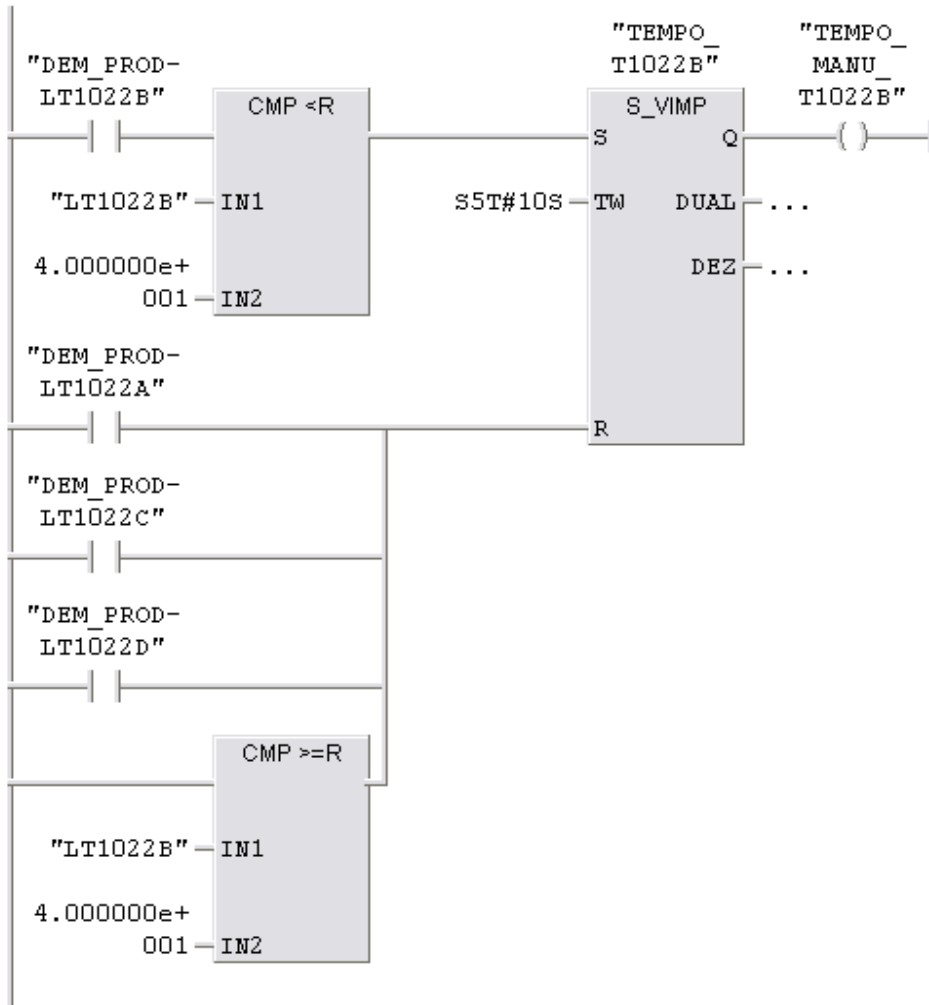
**Réseau 3 : DEMANDE DE PRODUIT LT1022A**

Commentaire :



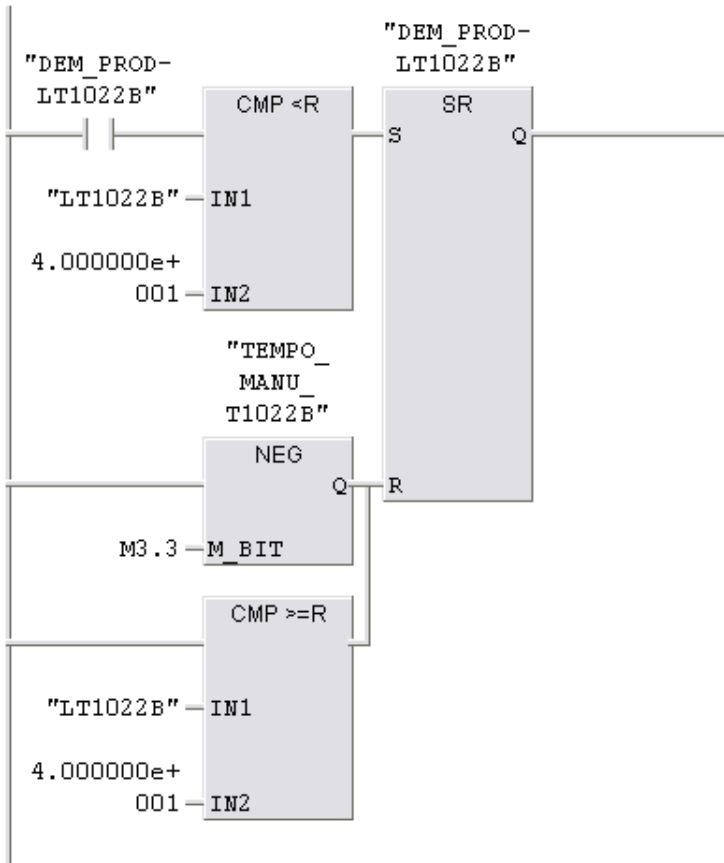
**Réseau 4 : Appel de produit T1022B**

Demande manuelle de produit vers la T1022B



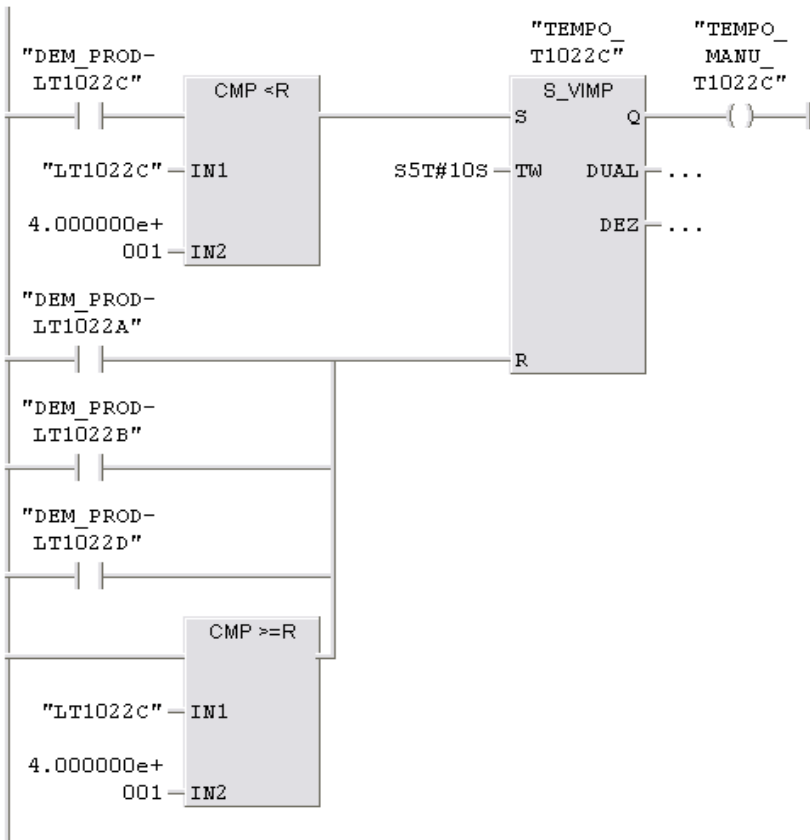
**Réseau 5 : DEMANDE DE PRODUIT LT1022C**

DEMANDE DE PRODUIT LT1022C



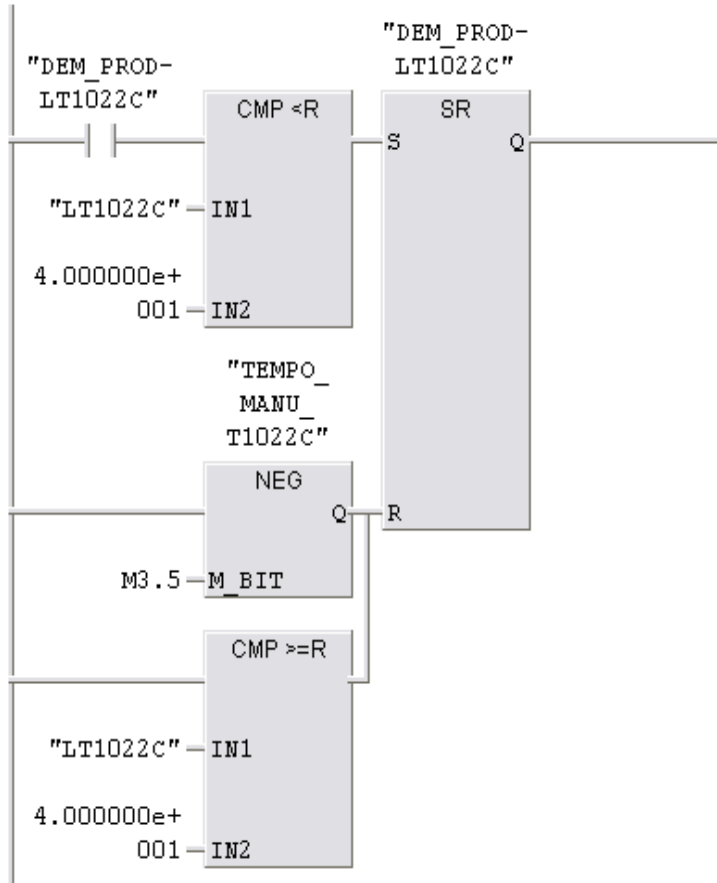
Réseau 6 : Appel de produit T1022C

Demande manuelle de produit vers la T1022C



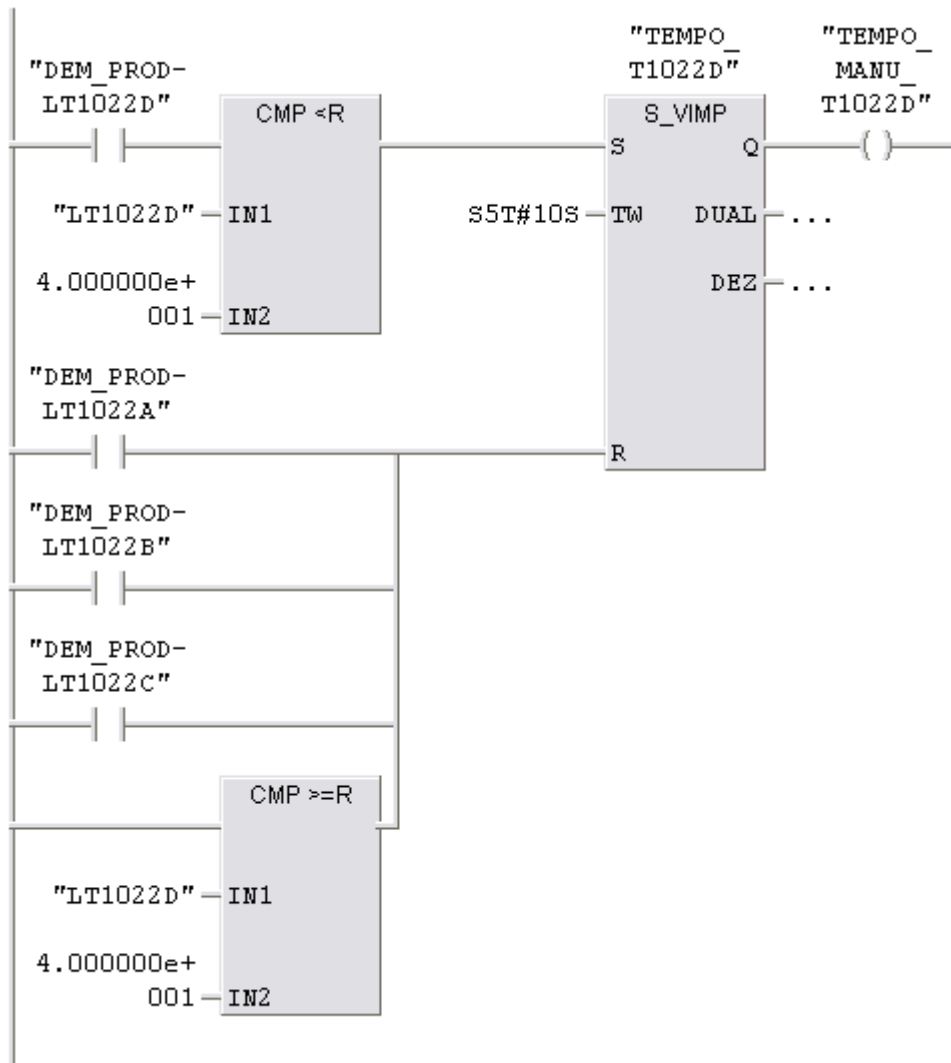
**Réseau 7 : DEMANDE DE PRODUIT LT1022C**

Commentaire :

**Réseau 8 : Appel de produit T1022D**

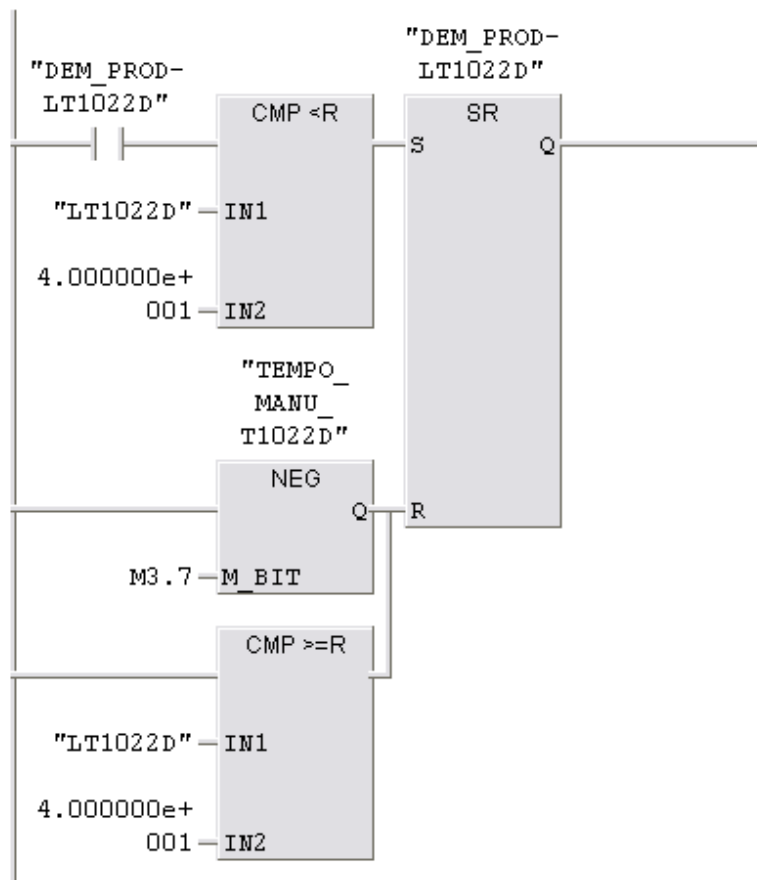
Demande manuelle de produit vers la T1022D





**Réseau 9 : DEMANDE DE PRODUIT LT1022D**

Commentaire :

**Réseau 10 : ARRET DU PRODUIT**

Commentaire :



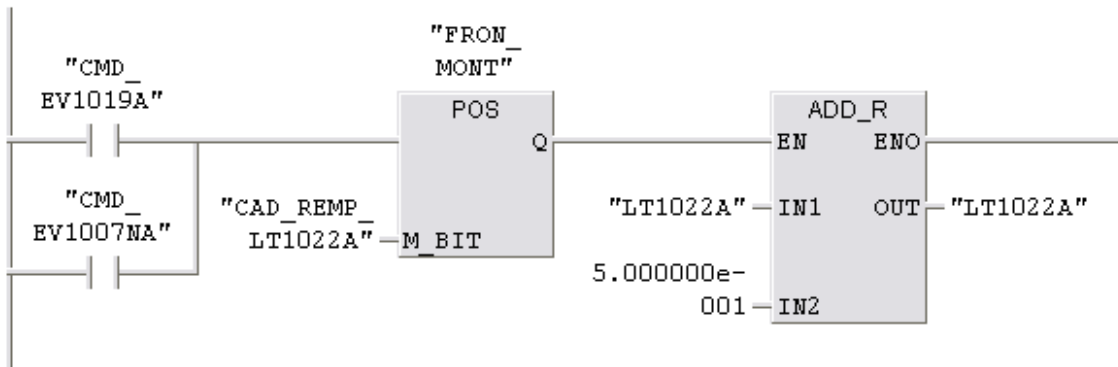
**FC9:**

FC9 : Titre :

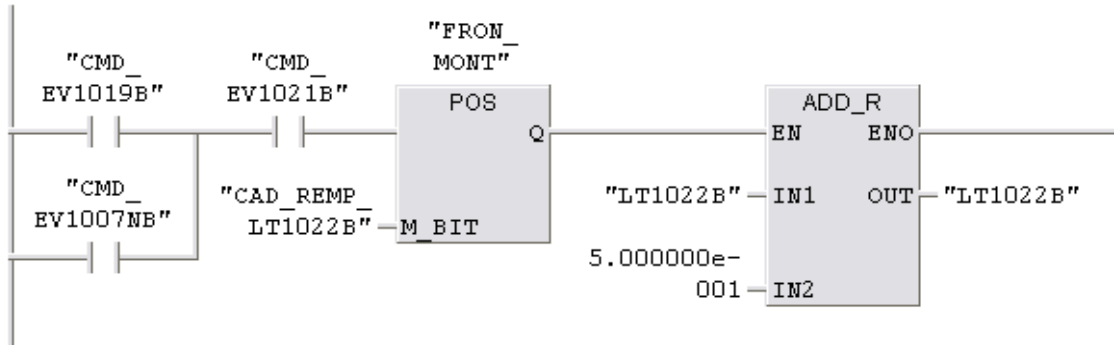
Commentaire :

**Réseau 1** : Titre :

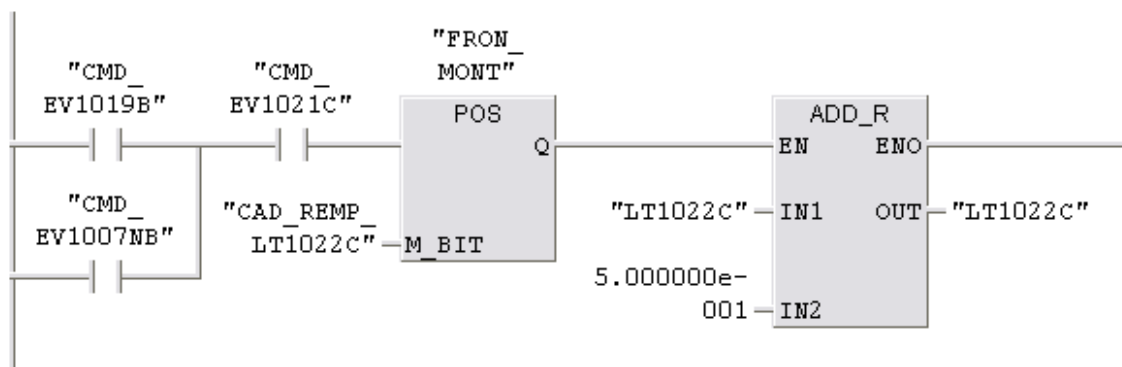
LE REMPLISSAGE DE LA TREMIE T1022A

**Réseau 2** : Titre :

LE REMPLISSAGE DE LA TREMIE T1022B

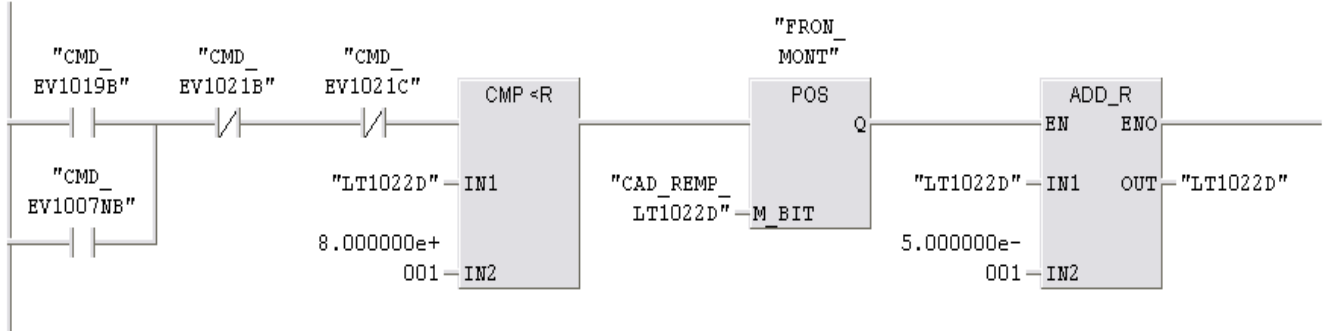
**Réseau 3** : Titre :

LE REMPLISSAGE DE LA TREMIE T1022C



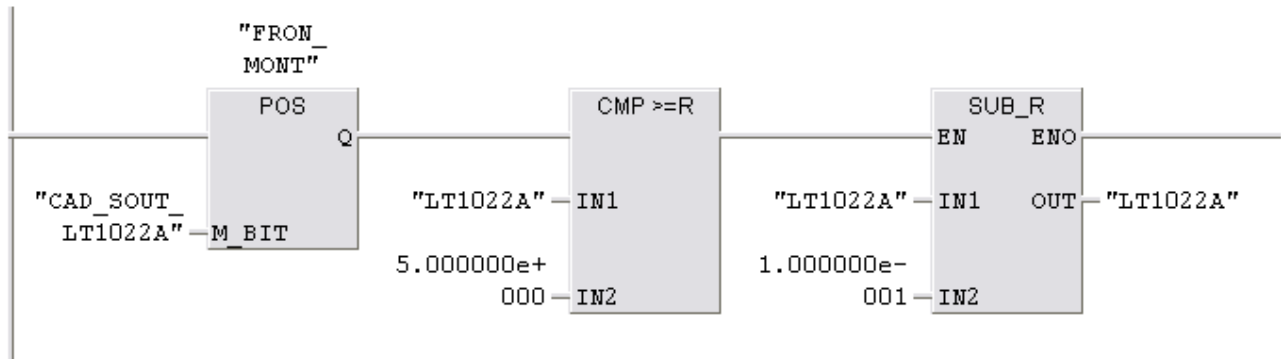
Réseau 4 : Titre :

LE REMPLISSAGE DE LA TREMIE T1022D



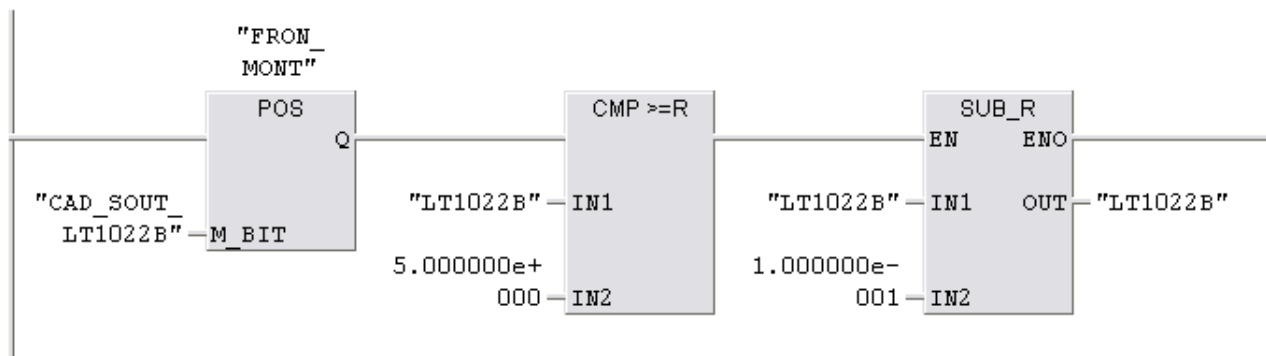
Réseau 5 : Titre :

LE SOUTIRAGE DE LA TRÉMIE T1022A



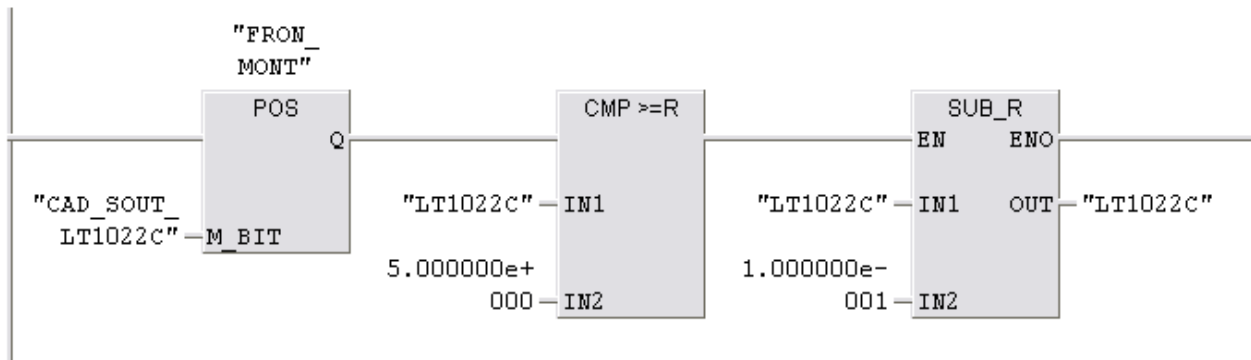
Réseau 6 : Titre :

LE SOUTIRAGE DE LA TRÉMIE T1022B



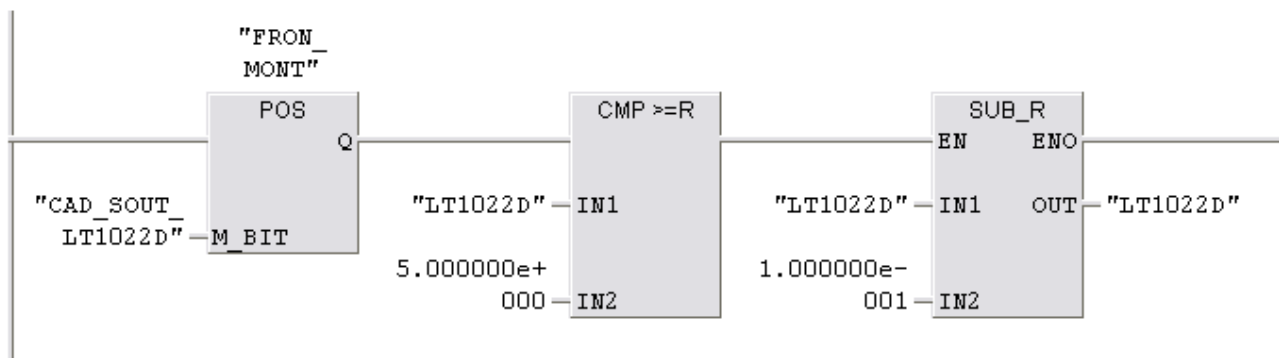
Réseau 7 : Titre :

LE SOUTIRAGE DE LA TRÉMIE T1022C



Réseau 8 : Titre :

LE SOUTIRAGE DE LA TRÉMIE T1022A

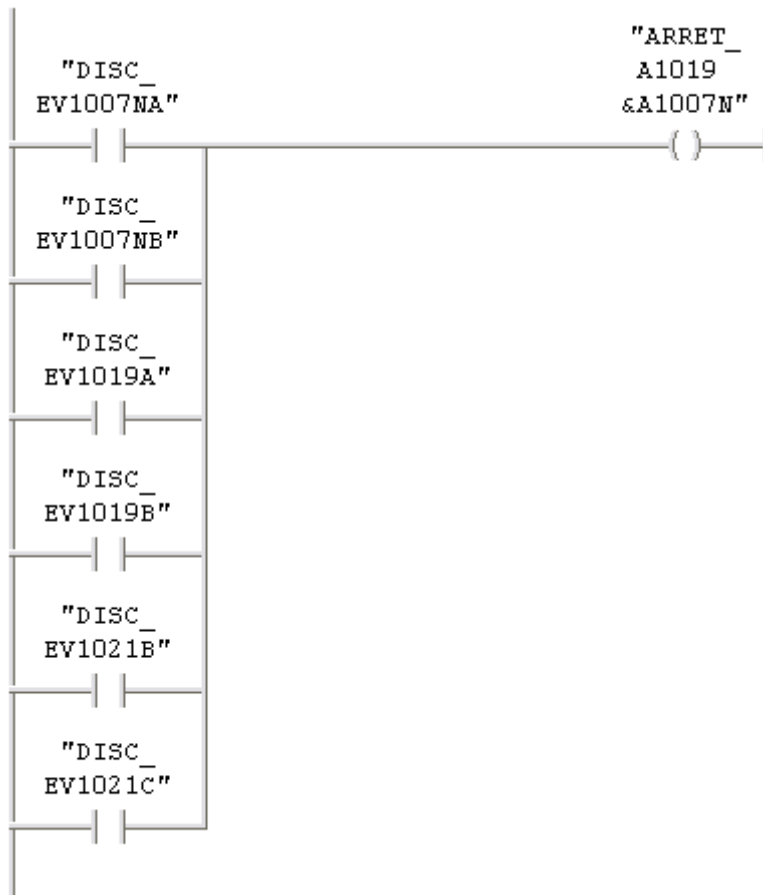
**FC10 :**

FC10 : Titre :

Commentaire :

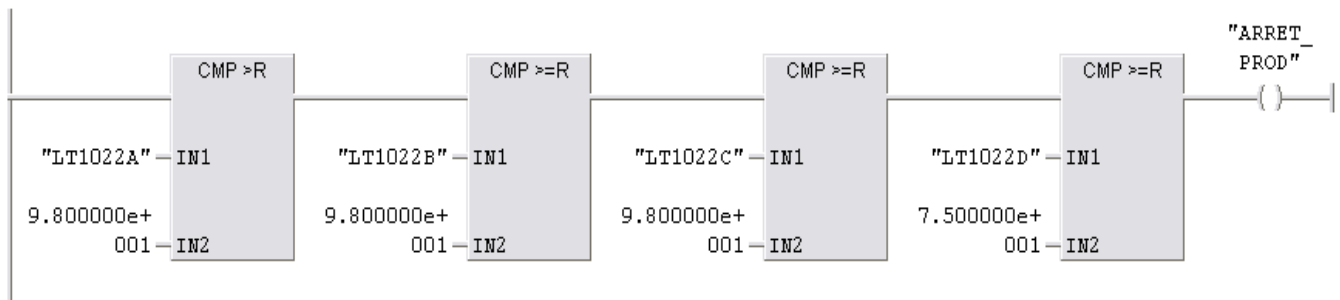
**Réseau 1** : ARRET PRODUIT

S'IL Y'A DISCORDANCE D'UNE SEULE VANNE, IL Y'AURA UN ARRET DU PRODUIT



**Réseau 2 : ARRET DU PRODUIT**

SI TOUTES LES TREMIES T1022A & T1022B & T1022C SONT PLEINNES ET QUE LA TREMIE T1022D EST > = 75% IL Y'AURA UN ARRET PRODUIT



**Réseau 3 : ARRET ET MARCHÉ DES MOTEURS A1021A & A1021B**

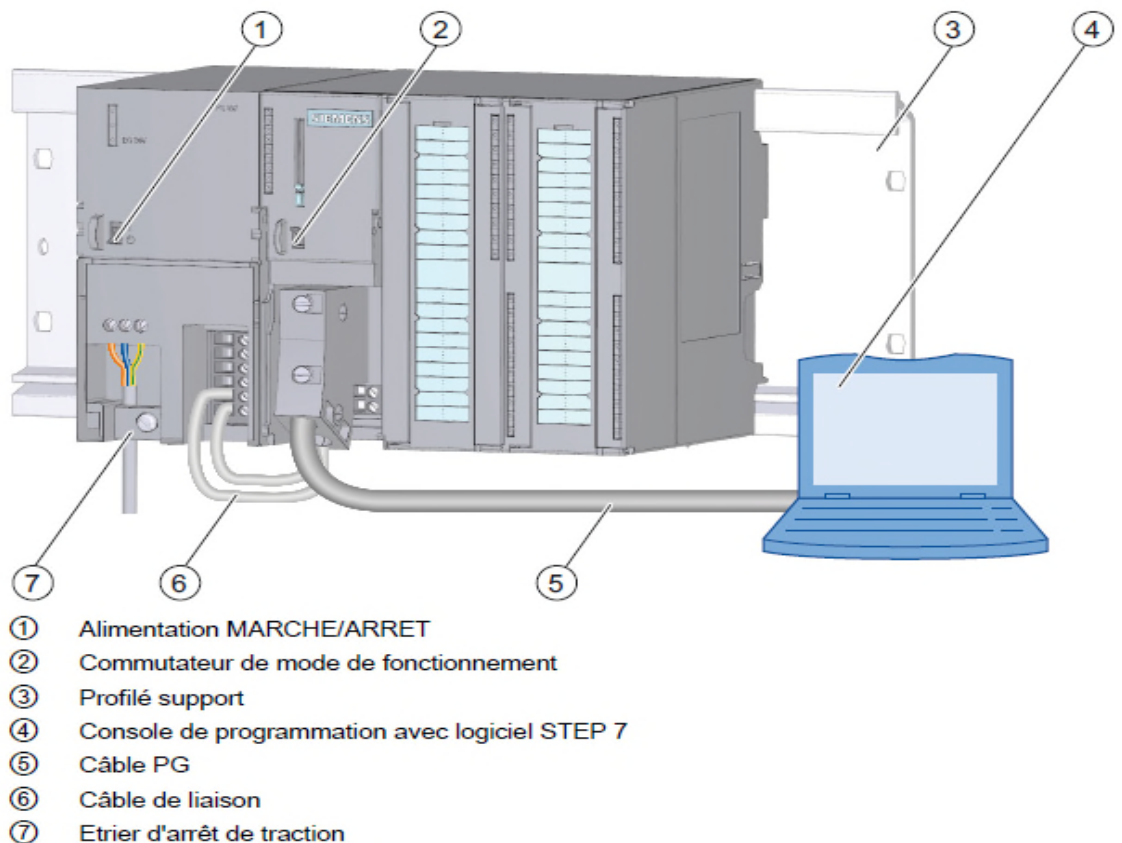
MARCHE DU MOTEUR A1021A



**Réseau 4** : ARRET ET MARCHE DES MOTEURS A1021A & A1021B

MARCHE DU MOTEUR A1021B



**Automate siemens S7-300 :****Différences essentielles par rapport au S7-300**

- ❑ Mémoire supérieure et nombre supérieur de : sorties/entrées/ mémentos/temporisations/compteurs
- ❑ Adresses des modules d'entrée/sortie paramétrables
- ❑ Possibilité de raccorder un châssis d'extension du S5 et d'utiliser des modules CP/IP du S5
- ❑ Plus de fonctions système, par ex. la communication programmée par blocs
- ❑ Taille des blocs jusqu'à 64 Ko et nombre de DB doublé
- ❑ Démarrage et redémarrage
- ❑ Comparaison de la configuration prévue/réelle au démarrage
- ❑ Possibilité de retrait des modules sous tension
- ❑ Plusieurs mémoires image
- ❑ Priorités des OB paramétrables
- ❑ Plusieurs OB pour les alarmes cycliques, de processus, horaires
- ❑ Imbrication des blocs jusqu'à 16 niveaux
- ❑ Taille de la pile L paramétrable par niveau d'exécution
- ❑ 4 accumulateurs
- ❑ Mode multiprocesseur



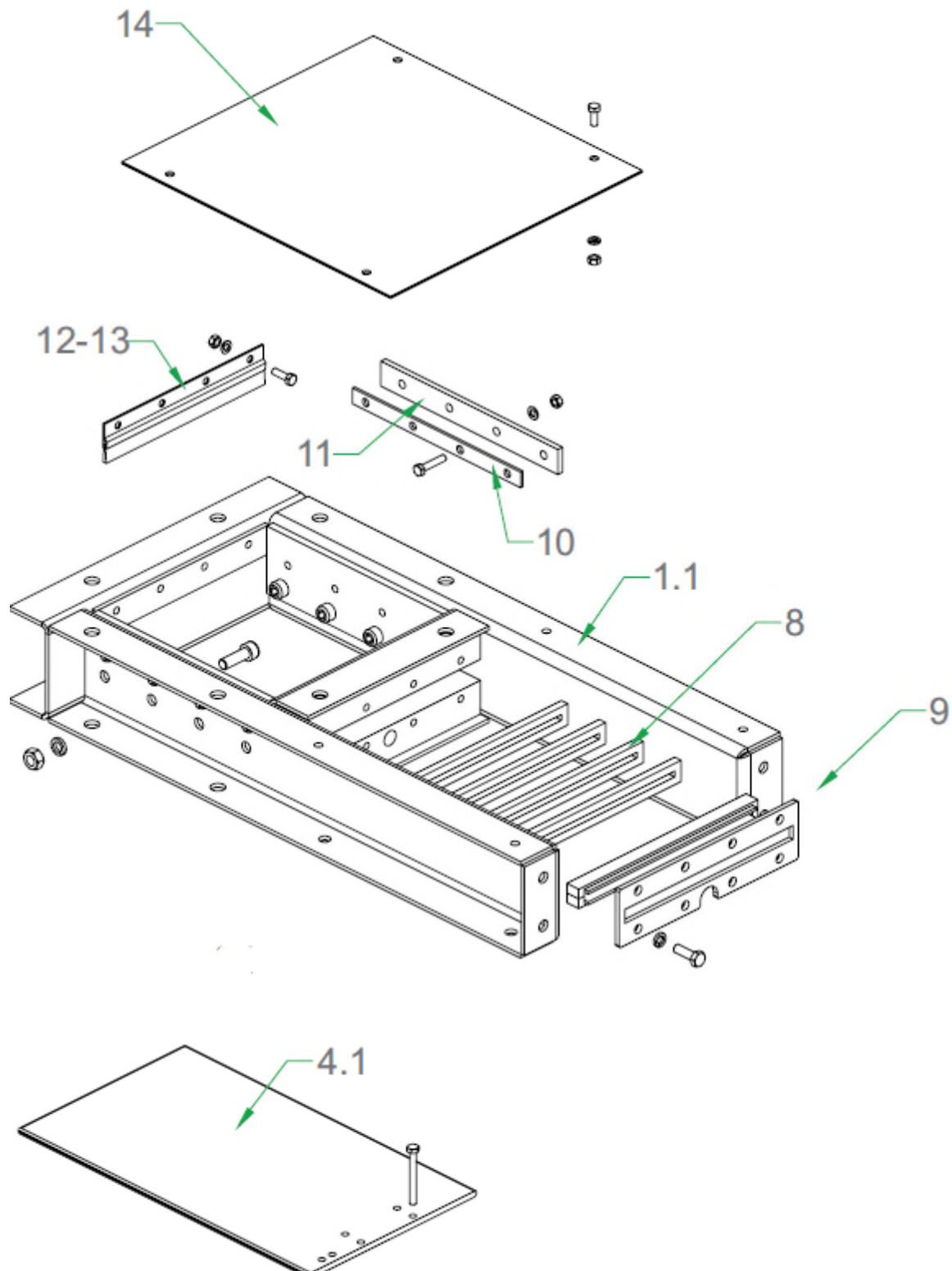
## Caractéristiques techniques des CPU du S7-300 (1)

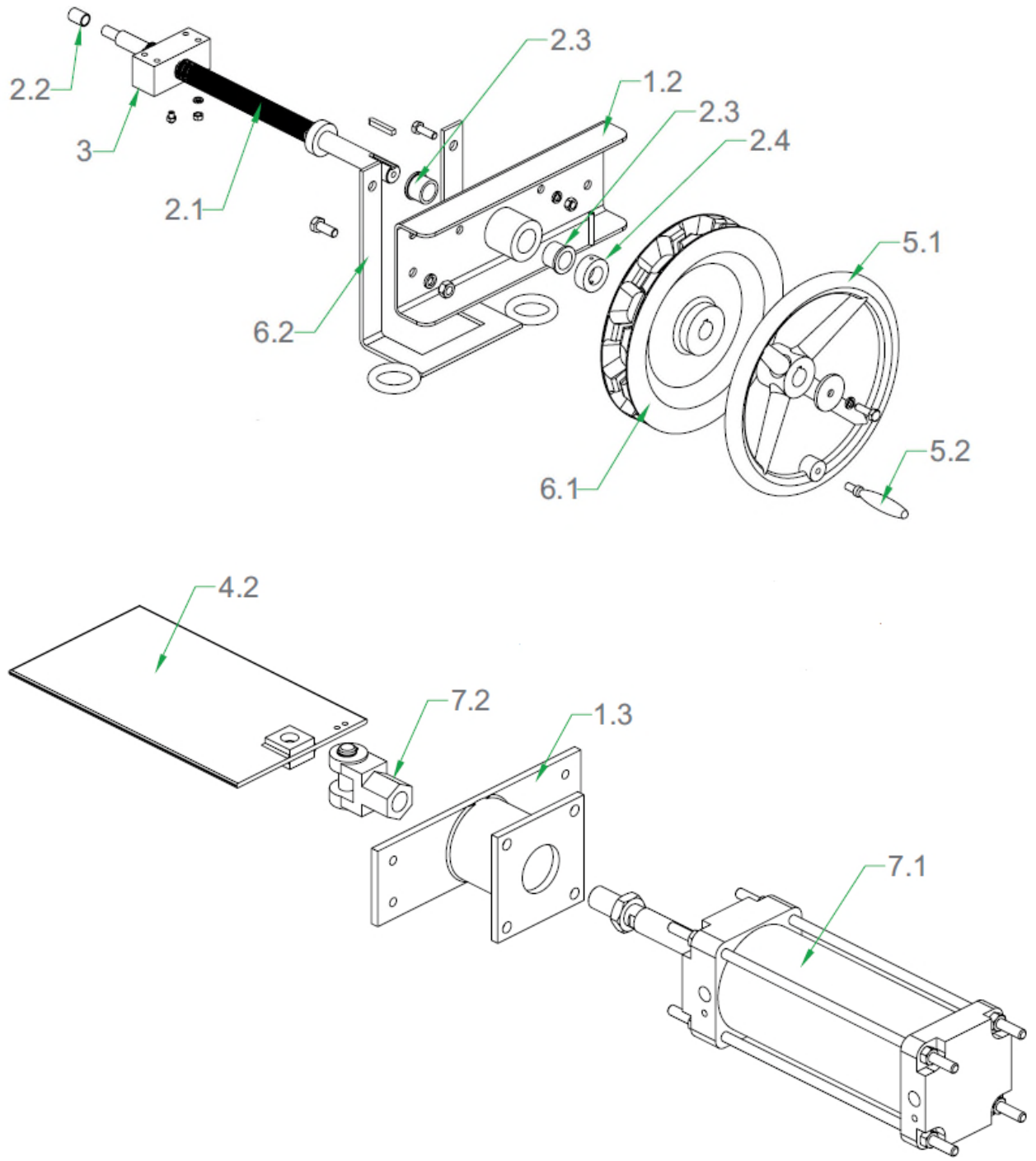
CPU	312 IFM	313	314	314 IFM	315	315-2 DP	316-2 DP	318-2 DP
Temps d'exécution en µs								
Instruction binaire	0,6 - 1,2	0,6 - 1,2	0,3 - 0,6	0,3 - 0,6	0,3 - 0,6	0,3 - 0,6	0,3 - 0,6	0,1
Opération sur mot	2,0	2,0	1,2	1,2	1,0	1,0	1,0	0,1
Entier (+/-)	3,0	3,0	2,0	2,0	2,0	2,0	2,0	0,1
Réel (+/-)	60,0	60,0	50,0	50,0	50,0	50,0	50,0	0,6
Mémoire utilisateur								
Mémoire de travail	6 Ko	12 Ko	24 Ko	32 Ko	48 Ko	64 Ko	128 Ko	512 Ko
Mém. chargem. intégr.	20 Ko	20 Ko	40 Ko	48 Ko	80 Ko	96 Ko	192 Ko	64 Ko
Mém. chargem. ext.	-	4 Mo	4 Mo	(4 Mo)	4 Mo	4 Mo	4 Mo	4 Mo
Opérandes								
Mémentos (bits)	1024	2048	2048	2048	2048	2048	2048	8192
Mémentos de cadence	8	8	8	8	8	8	8	8
Temporisations	64	128	128	128	128	128	128	512
Compteurs	32	64	64	64	64	64	64	512
Types de blocs/Nombre								
FB	32	128	128	128	192	192	256	1024
FC	32	128	128	128	192	192	512	1024
DB	63	127	127	127	255	255	511	2047
Taille de la mémoire image (Entrée/sorties)	32 octets chacune	128 octets chacune	128 octets chacune	124 octets chacune	128 octets chacune	128 octets chacune	128 octets chacune	256 octets chac. (2048)
Espace maxi. d'adress. des périphéries	32 chacun	32 octets chacun	768 octets chacun	752 octets chacun	768 octets chacun	1024 octets chacun	1024 octets chacun	8192 octets chacun
Interfaces	MPI	MPI	MPI	MPI	MPI	MPI, DP	MPI, DP	MPI/DP, DP

## Caractéristiques techniques des CPU du S7-300 (2)

CPU	312 IFM	313	314	314 IFM	315	315-2 DP	316-2 DP	318-2 DP
Blocs d'organisation	N° OB	N° OB	N° OB	N° OB	N° OB	N° OB	N° OB	N° OB
Cycle libre	1	1	1	1	1	1	1	1
Alarmes horaires	-	10	10	10	10	10	10	10,11
Alarmes temporisées	-	20	20	20	20	20	20	20,21
Alarmes cycliques	-	35	35	35	35	35	35	32,35
Alarmes de processus	40	40	40	40	40	40	40	40,41
Traitement d'arrière-plan	-	-	-	-	-	-	-	90
Démarrage	100	100	100	100	100	100	100	100,102
Erreur, asynchrone	-	80-82, 85, 87	80-82, 85, 87	80-82, 85, 87	80-82, 85, 87	80-82, 85, 87	80-82,85, 87	80-82, 85, 87
Erreur, synchrone	-	121,122	121,122	121,122	121,122	121,122	121,122	121,122
Données locales (octets)	512	1536	1536	1536	1536	1536	1536	4096(8192)
Longueur de bloc maxi.	8 Ko	8 Ko	8 Ko	8 Ko	16 Ko	16 Ko	16 Ko	64 Ko
Profondeur d'imbrication des blocs/niveau d'exéc.	8	8	8	8	8	8	8	20
Communication :								
Nombre maxi. de liaisons statiques/dynamiques	4/2	4/4	4/8	4/8	4/8	4/8	4/8	32
Communication par donn. globales via MPI :								
cercles GD par CPU	4	4	4	4	4	4	4	8
Emission Paquet GD par cercle GD	1	1	1	1	1	1	1	1
Réception Paquets GD par cercle GD	1	1	1	1	1	1	1	2
Taille maxi. des données utiles d'un paquet	22 octets	22 octets	22 octets	22 octets	22 octets	22 octets	22 octets	54 octets



**Vanne guillotine :**



<i>REPRESENTATION</i>	<i>N°</i>	<i>DISIGNATION PIECES DE RECHARGES</i>
<i>1.1</i>	<i>1</i>	<i>châssis</i>
<i>1.2</i>	<i>1</i>	<i>Traverse commande manuelle</i>
<i>1.3</i>	<i>1</i>	<i>Traverse commande pneumatique</i>
<i>2.1</i>	<i>1</i>	<i>Vis de commande</i>
<i>2.2</i>	<i>1</i>	<i>Bague bronze</i>
<i>2.3</i>	<i>2</i>	<i>coussinet</i>
<i>2.4</i>	<i>1</i>	<i>Rondelle de maintien de la vis</i>
<i>3</i>	<i>1</i>	<i>Ecrou avec graisseur</i>
<i>4.1</i>	<i>1</i>	<i>Trappe commande manuelle inox</i>
<i>4.2</i>	<i>1</i>	<i>Trappe commande pneumatique inox</i>
<i>5.1</i>	<i>1</i>	<i>Volant de manœuvre à main</i>
<i>5.2</i>	<i>1</i>	<i>Poignée</i>
<i>6.1</i>	<i>1</i>	<i>Volant de manœuvre à chaîne</i>
<i>6.2</i>	<i>1</i>	<i>Guide de chaîne</i>
<i>7.1</i>	<i>1</i>	<i>vérin pneumatique</i>
<i>7.2</i>	<i>1</i>	<i>Chape pour vérin pneumatique</i>
<i>8</i>	<i>4</i>	<i>Tresse d'étanchéité</i>
<i>9</i>	<i>1</i>	<i>Presse-étoupe</i>
<i>10</i>	<i>2</i>	<i>Contre plaque pour joints latéraux</i>
<i>11</i>	<i>2</i>	<i>Joint latéral</i>
<i>12</i>	<i>1</i>	<i>Brosse d'étanchéité</i>
<i>13</i>	<i>1</i>	<i>Plat de serrage brosse</i>
<i>14</i>	<i>1</i>	<i>Capot</i>



**Détecteur de niveau :****Gamme de mesure :**

Levelflex FMP50				
Groupe de produit	Coefficient diélectr. ( $\epsilon_r$ )	Liquides typiques	Gamme de mesure	
			Sondes à câble Sondes à tige	Sondes à câble Sondes à câble
1	1,4...1,6	Gaz liquéfiés, par ex. N <sub>2</sub> , CO <sub>2</sub>	sur demande	
2	1,6...1,9	<ul style="list-style-type: none"> <li>▪ Gaz liquide, par ex. propane</li> <li>▪ Solvant</li> <li>▪ Fréon</li> <li>▪ Huile de palme</li> </ul>	4 m (13 ft)	12 m (39 ft)
3	1,9...2,5	Huiles minérales, carburants	4 m (13 ft)	12 m (39 ft)
4	2,5...4	<ul style="list-style-type: none"> <li>▪ Benzène, styrène, toluène</li> <li>▪ Furane</li> <li>▪ Naphtalène</li> </ul>	4 m (13 ft)	12 m (39 ft)
5	4...7	<ul style="list-style-type: none"> <li>▪ Chlorobenzène, chloroforme</li> <li>▪ Vernis cellulosique</li> <li>▪ Isocyanate, aniline</li> </ul>	4 m (13 ft)	12 m (39 ft)
6	> 7	<ul style="list-style-type: none"> <li>▪ Solutions aqueuses</li> <li>▪ Alcools</li> <li>▪ Acides, bases</li> </ul>	4 m (13 ft)	12 m (39 ft)

**Sortie :**

Codage du signal	FSK $\pm 0,5$ mA via le signal de courant
Vitesse de transmission des données	1200 baud
Séparation galvanique	Oui



**Capteur LSH :****Généralité :**

Fabricant	Endress+Hauser
Appareil	Détecteur de niveau
Désignation de l'appareil	Minicap FTC 260, FTC 262
Documentation technique Version	TI 287F/fr 04.00
Caractéristiques techniques	Selon DIN 19259

**Domaine d'application :**

Produit	Solides en vrac, granulométrie max. 30 mm Densité min. 200 g/l, $\epsilon_r \geq 1,6$
---------	--

**Principe de fonctionnement et construction :**

Principe de mesure	Capacitif
Construction	Compacte
Modèle	Avec sonde à tige (FTC 260) ou sonde à câble (FTC 262)
Transmission du signal	Binaire

**Entrée :**

Grandeur de mesure	Seuil de niveau
Gamme de mesure	FTC 260: $\epsilon_r \geq 1,6$ , FTC 262: $\epsilon_r \geq 1,5$

**Sortie :**

Electronique	DC Sortie transistor	AC / DC Sortie relais
Signal de sortie	Type de contact : PNP $I_{max}$ 200 mA – Protection contre les surcharges et les courts-circuits – Tension résiduelle au transistor pour $I_{max} < 2,9$ V	Type de contact : inverseur sans potentiel $U_{\sim max}$ 253 V, $I_{\sim max}$ 4 A $P_{\sim max}$ 1000 VA, $\cos \varphi = 1$ $P_{\sim max}$ 500 VA, $\cos \varphi > 0,7$ $I_{\dots max}$ 4 A jusqu'à $U_{\dots}$ 30 V, $I_{\dots max}$ 0,2 A jusqu'à $U_{\dots}$ 253 V
Temporisation de commutation lors du recouvrement ou du découvrement de la sonde	FTC 260: 0,5 s FTC 262: 0,8 s	FTC 260: 0,5 s FTC 262: 0,8 s
Signal de panne	< 100 $\mu$ A	Relais retombé
Catégorie de surtension	Catégorie III selon EN 61010	
Classe de protection selon EN 61010-1	FTC 260: classe II, FTC 262: classe I	



**Précision de mesure :**

Conditions de référence (pour réservoir en matière synthétique)	Température ambiante 23°C, Température de service 23 °C, pression de service $p_{abs} = 1$ bar Produit : coefficient diélectrique $\epsilon_r = 2,6$ , conductivité $< 1 \mu S$ , Réglage sensibilité commutateur : C	
	FTC 260	FTC 262
Hystérésis	Horizontale 4 mm, Verticale 7 mm	Verticale 5 mm
Point de commutation	Horizontal : milieu de la sonde - 5mm Vertical : 40 mm au dessus de l'extrémité de sonde	Vertical : 35 mm au dessus de l'extrémité de sonde
Temps de réponse	Commutation correcte après 1,5 s max.	Commutation correcte après 2 s max.
Dérive à long terme	Horizontale 3 mm, Verticale 6 mm	Verticale 6 mm
Effet de la température du produit	En fonction du produit	En fonction du produit

CONVOYEUR A VIS Ø1000 x 12100 mm

N°130050-010-0000

<b>POSITION</b> .....	Horizontale
<b>DEBIT</b> T/h-m <sup>3</sup> /h.....	300 T/h – 353 m <sup>3</sup> /h
Alimentation.....	Transport
<b>PRODUIT</b> .....	Sucre blanc sec
Densité - granulométrie.....	0,85 – NC
Humidité - température.....	Sec – Ambiante
<b>VITESSE</b> rotor.....	42 tr/mn
<b>AUGE</b> type - mat - épaisseur.....	En U – inox 304L – ép. 6 mm
<b>ETANCHEITE</b> sur arbre.....	Presses étoupes inox 304L
<b>ROTOR</b> Ø – pas - matière.....	Ø1000 mm – 630 pas à gauche – inox 304L ép. 8 mm
Tourillons.....	Extrêmes et intermédiaires en inox 316L
<b>PALIER</b> S.....	Paliers à semelle en 2 parties + paliers intermédiaires avec supports intermédiaires
<b>GROUPE DE COMMANDE</b> .....	« SEW USOCOME » à engrenage cylindrique
Puissance installée.....	37 kW – 380/660V – 50 Hz – IP 65 - position M1
Type.....	R147/II2GD EDRE225M4/3GD/AL
<b>EQUIPEMENT COMPLEMENTAIRE</b> .....	Accouplement élastique « TSCHAN » type E415
	Contrôleur de rotation et son support