

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

UNIVERSITÉ

ABDERAHAMANE MIRA DE BÉJAIA

Faculté de Technologies

Département Génie Électrique



MÉMOIRE DE FIN DE CYCLE

En vue de l'obtention du diplôme de Master en Électrotechnique

Option : Automatismes industriels

THÈME

Automatisation et supervision d'une bascule

PRECIA MOLEN avec l'API S7-300

Présenté par :

Gouiri yuba

Ait Hatrit Fodil

Encadré par :

Mr: ACHOUR A/Yazid

Mr: Megri Samir

Promotion 2017/2018

Remerciement

Nous remercions d'abord ALLAH le tout-puissant de nous avoir accordé la force, la volonté et la connaissance pour accomplir ce projet.

Nous tenons à exprimer nos profondes gratitude à notre promoteur Mr : A/Y. ACHOUR, pour avoir accepté de diriger ce travail, et témoigne toute notre reconnaissance pour ses conseils, ses orientations et sa patience.

Nous offrons notre gratitude à Mr : S.Megri, pour sa disponibilité, son aide, et d'avoir mis toute la documentation nécessaire, à notre disposition. Ainsi que tous le personnel de la direction technique de CEVITAL (unité raffinerie de sucre).

Mes vifs remerciements aux membres de jury de bien vouloir accepter d'évaluer mon travail

Dédicaces

Je tiens vivement, à dédier ce modeste travail :

*A mes très chers parents qui m'ont tant aidé tout au long de
mes études, que dieu le tout puissant erheme ma très cher mère ;*

A mes frères et sœurs ainsi que tous les membres de ma famille ;

A tous mes amis et camarades sans exception ;

A tous mes proches ;

A tous ceux qui aiment partager le savoir sans rien en retour.

GOVIRI YUBA

Dédicaces

*Je tiens vivement, à dédier ce modeste
travail :*

*A mes très chers parents qui m'ont tant aidé tout au long de mes
études, que dieu le tout puissant vous garde, et vous procure
santé, bonheur et longue vie à son adoration ;*

*A mes frères et sœurs ainsi que tous les membres de ma famille
spécialement a mon cousin IDIR et mon grand frère ZAHIR ;*

A tous mes proches ;

*A tous mes amis et camarades ; WALIDE, LAMINE,
ATMAN, BILALE, FAYCEL....*

AITHARIT FODIL

Sommaire

Sommaire

| | |
|-----------------------------|---|
| Introduction générale | 1 |
|-----------------------------|---|

Préambule

| | |
|---|---|
| I. Présentation du groupe CEVITAL | 3 |
|---|---|

| | |
|----------------------------------|---|
| II. Situation géographique | 3 |
|----------------------------------|---|

CHAPITRE 1 : PRESENTATION DE L'UNITE ET LA BASCULE

| | |
|------------------------|---|
| I.1 Introduction | 3 |
|------------------------|---|

| | |
|--|---|
| I.2 Unité de raffinage de sucre 3500 T | 3 |
|--|---|

| | |
|------------------------------------|---|
| I.2.1 Processus de raffinage | 3 |
|------------------------------------|---|

| | |
|---|---|
| I.2.1.1 Section 1 : Affinage et Refonte | 3 |
|---|---|

| | |
|---|---|
| I.2.1.2 Section 2 : Carbonatation | 3 |
|---|---|

| | |
|-------------------------------------|---|
| I.2.1.3 Section 3 : Filtration..... | 3 |
|-------------------------------------|---|

| | |
|---------------------------------------|---|
| I.2.1.4 Section 4 : Décoloration..... | 4 |
|---------------------------------------|---|

| | |
|--|---|
| I.2.1.5 Section 5 : Concentration..... | 4 |
|--|---|

| | |
|---|---|
| I.2.1.6 Section 6 : Cristallisation haut produit..... | 4 |
|---|---|

| | |
|----------------------------------|---|
| I.2.1.7 Section 7 : Séchage..... | 4 |
|----------------------------------|---|

| | |
|---|---|
| I.2.1.8 Section 8 : Stockage et maturation..... | 4 |
|---|---|

| | |
|-----------------------------------|---|
| I.2.1.9 Section 9 : Utilités..... | 5 |
|-----------------------------------|---|

| | |
|--|---|
| I.2.1.10 Section 10 : Conditionnement..... | 5 |
|--|---|

| | |
|--|---|
| I.3 Présentation de la bascule PRECIA MOLEN..... | 5 |
|--|---|

| | |
|--|---|
| I.4 Le rôle de la bascule au sein de l'unité | 6 |
|--|---|

| | |
|--|---|
| I.5 Instruments constituant la bascule PRECIA MOLEN..... | 6 |
|--|---|

| | |
|--|---|
| I.5.1 Sources d'alimentation en énergie..... | 6 |
|--|---|

| | |
|---------------------------|---|
| I.5.2 Pré-actionneur..... | 7 |
|---------------------------|---|

| | |
|------------------------|---|
| I.5.3 Actionneur | 8 |
|------------------------|---|

| | |
|---|---|
| A) Fonctions et caractéristiques des vérins pneumatiques..... | 8 |
|---|---|

| | |
|---------------------------------|---|
| B) Constitution d'un vérin..... | 8 |
|---------------------------------|---|

| | |
|--|----|
| C) Types de vérins utilisé dans la bascule | 9 |
| I.5.4 Les capteurs et les détecteurs utilisés dans la bascule..... | 9 |
| A) Détecteurs magnétiques | 10 |
| B) Capteur de force | 11 |
| C) Détecteur à lames vibrantes..... | 11 |
| I.6 Le cahier des charges..... | 12 |
| I.7 Conclusion | 13 |

CHAPITRE 2 : PROBLEMATIQUE ET ANALYSE FONCTIONNELLE

| | |
|---|----|
| II.1 Introduction | 14 |
| II.2 Problématiques | 14 |
| II.3 Solutions proposées | 14 |
| II.4 Cahier des charges modifié | 15 |
| II.5 a Description du langage GRAFCET | 16 |
| II.6 Présentation du logicielle AUTOMGEN | 16 |
| II.7 Les GRAFCETS réalisés | 17 |
| II.7.1 Tableau des variables | 17 |
| II.7.2 GRAFCET de Sécurité et de Mise en Marche | 18 |
| II.7.2 GRAFCET de Permutation entre les modes automatique/semi automatique | 19 |
| II.7.3 GRAFCET du fonctionnement automatique | 20 |
| II.7.4 GRAFCET du Fonctionnement semi automatique | 21 |
| II.8 Equation des GRAFCET | 21 |
| II.8.1 Equation des étapes du GRAFCET du fonctionnement automatique | 21 |
| II.8.2 Equation des étapes du GRAFCET du fonctionnement semi automatique | 21 |
| II.8.3 Equation des étapes du GRAFCET de sécurité et de mise en marche | 22 |
| II.8.4 Equation des étapes du GRAFCET de Permutation entre modes automatique/manuel | 22 |
| II.8.5 Equation des sortis des GRAFCET | 23 |
| II.9 Conclusion..... | 23 |

CHAPITRE 3 : L'API ET LES LOGICIELS ASSOCIES

| | |
|--|----|
| III.1 Introduction | 24 |
| III.2 Définition d'un système automatisé | 24 |
| III.3 Critères de choix de l'automate | 25 |
| III.4 Présentation de l'automate programmable industrielle utilisé | 25 |
| III.4.A Définition | 25 |
| III.4.B Constitution d'un Automate S7 300 | 26 |
| III.4.B.1 L'alimentation de l'automate | 26 |
| III.4.B.2 Présentation de la CPU 313 C | 27 |
| Contrôles visuel des LED de la CPU 313 | 28 |
| Le commutateur du mode de fonctionnement | 28 |
| L'interface multipoints | 28 |
| Les mémoires | 28 |
| Les modules entrées et sorties TOR | 29 |
| III.5 Présentation du logiciel STEP7 | 29 |
| III.5.1 Description du STEP7 | 29 |
| Lancement et création d'un projet sou STEP7 | 29 |
| Insertion d'une station | 30 |
| Configuration du matérielle | 30 |
| Structure d'un projet STEP7 | 31 |
| a). Bloc d'organisation (OB) | 32 |
| b). Bloc fonction (FC, SFC) | 32 |
| c). Bloc de données (DB) | 32 |
| d). Bloc fonctionnel (FB, SFB) | 32 |
| Création d'un programme en l'ADDER | 33 |
| III.5.2 Description de la S7 PLCSIM | 33 |
| III.6 Présentation du système WinCC flexible | 35 |
| III.7 Création ou chargement d'un projet | 36 |

| | | |
|---------|--|----|
| III.8 | Eléments de l'interface utilisateur de WinCC flexible | 37 |
| III.8.1 | Menus et barres d'outils | 38 |
| III.8.2 | Zone de travail | 38 |
| III.8.3 | Fenêtre de projet | 38 |
| III.8.4 | Fenêtre des propriétés | 39 |
| III.8.5 | Boîte à outils | 39 |
| III.8.6 | Bibliothèque | 39 |
| III.8.7 | Fenêtre des erreurs et avertissements | 39 |
| III.9 | WinCC flexible intégré à STEP 7 | 39 |
| III.10 | Créez et configurez des variables dans l'éditeur Variables | 40 |
| III.11 | Créer des liaisons | 41 |
| III.12 | La fenêtre permanente ProTool au modèle | 41 |
| III.13 | Utilisation d'objets | 42 |
| III.14 | Conclusion | 44 |

CHAPITRE 4 : PROGRAMME ET SUPERVESION

| | | |
|------|--|----|
| IV.1 | Introduction | 45 |
| IV.2 | Le rôle et le contenu des blocs utilisés dans le programme | 45 |
| A. | le bloc OB1 | 45 |
| B. | Le bloc FC 105 | 45 |
| C. | Le bloc FC1 (fonctionnement) | 45 |
| D. | Le bloc FC2 (automatique / semi-automatique) | 46 |
| E. | Le bloc FC4 (Alarmes) | 46 |
| F. | Le bloc FC3 (fonction pour WinCC) | 46 |
| G. | Le bloc DB1 | 46 |
| H. | Le bloc DB2 | 46 |
| IV.3 | Elaboration du programme sous STEP7 | 46 |
| a). | Normalisation de l'entrée analogique | 46 |

| | |
|---|----|
| b). Réseaux de mise en marche / arête et sécurité | 47 |
| c). Description de l'opération de comparaison | 48 |
| d). Réseau d'activation d'une action en mode automatique ou semi-automatique est remis à l'état initial | 48 |
| e). Réseau de déplacement de la valeur de pesé | 49 |
| f). Réseau de soustraction du résidu de vidange de la pesée d'un cycle | 50 |
| h). Réseau d'adition du poids totale des pesées | 51 |
| IV.3 Vue d'Accueil | 52 |
| IV.4 Vue du mode de fonctionnement Automatique..... | 52 |
| IV.5 Vue du mode de fonctionnement semi Automatique | 53 |
| IV.6 Vue des alarmes | 54 |
| IV.7 Configuration champ E/S | 55 |
| IV.8 Configuration d'un indicateur | 55 |
| IV.9 Conclusion | 56 |
| Conclusion générale | 57 |

Liste des Figures

LISTE DES FIGURES

| | |
|---|----|
| Figure (1) : Emplacement géographique de l'entreprise..... | 3 |
| Figure (I.1) : Balance de circuit ABS-XL (ABS I 400)..... | 7 |
| Figure (I.2) : Compresseur à vis..... | 8 |
| Figure (I.3) : Distributeur pneumatique 4/2 bistable a commande électrique..... | 8 |
| Figure (I.4) : Fonction générale des actionneurs..... | 9 |
| Figure (I.5) : Les éléments d'un vérin..... | 9 |
| Figure (I.6) : Vérin double effet..... | 10 |
| Figure (I.7) : Principe de fonctionnement des capteurs..... | 11 |
| Figure (I.8) : Fonctionnement d'un capteur magnétique..... | 11 |
| Figure (I.9) : le capteur de force..... | 12 |
| Figure (I.10) : Détecteur à lame vibrante (VLD)..... | 12 |
| Figure(I.11) : Détecteur à lames vibrantes Soliphant FTM50..... | 13 |
| Figure (II.1) : Déférent composant d'un GRAFCET..... | 16 |
| Figure (II.2): GRAFCET de sécurité et de mise en marche sous AUTOMGEN..... | 18 |
| Figure (II.3): GRAFCET de Permutation sous AUTOMGEN..... | 19 |
| Figure (II.4): GRAFCET du fonctionnement automatique sous AUTOMGEN..... | 20 |
| Figure (II.5): GRAFCET de fonctionnement semi automatique sous AUTOMGEN..... | 21 |
| Figure (III.1) : Constitution d'un Automate modulaire S7 300..... | 26 |
| Figure (III.2) : Alimentation 307 24 V / 5 A pour S7 300..... | 26 |
| Figure (III.3) : La CPU 313 C avec ses déférents éléments de commande et de signalisation...27 | |
| Figure (III.4) : Fenêtre d'assistance pour la créature de nouveau projet sous STEP7..... | 30 |
| Figure (III.5) : Insertion d'une station sous STEP7..... | 30 |
| Figure (III.6) : Configuration matériels sous STEP7..... | 31 |
| Figure (III.7) : Structure d'un projet sous STEP7..... | 32 |
| Figure (III.8) : Réalisation d'un réseau sous STEP7..... | 33 |
| Figure (III.9) : Fenêtre du commutateur de la CPU sous STEP7..... | 34 |
| Figure (III.10) : Fenêtre de l'entrée analogique dans PLCSIM..... | 34 |
| Figure (III.11) : La fenêtre de PLCSIM sous STEP7..... | 35 |
| Figure (III.12) : Fenêtre de création ou de chargement d'un projet WinCC flexible..... | 36 |
| Figure (III.13) : Fenêtre de sélection du pupitre sous WinCC..... | 36 |
| Figure (III.14) : Eléments de WinCC flexible sous WinCC..... | 37 |
| Figure (III.15) : Fenêtre pour l'intégration d'un projet WinCC flexible dans le projet STEP7..38 | |
| Figure (III.16) : Création et configuration des variables sous WinCC..... | 39 |

| | |
|---|-----------|
| Figure (III.17) : Création des liaisons sous WinCC..... | 40 |
| Figure (III.18) : Fenêtre modèle sous WinCC..... | 40 |
| Figure (III.19) : Configuration d'un bouton poussoir sous WinCC..... | 41 |
| Figure (III.20) : Fenêtre de représentation sous WinCC..... | 41 |
| Figure (III.21) : Fenêtre de mise en page sous WinCC..... | 42 |
| Figure (III.22) : Fenêtre de texte sous WinCC..... | 42 |
| Figure (III.23) : Configuration d'un événement pour un bouton sous WinCC..... | 42 |
| Figure (III.24) : Distribution d'une variable a un bouton sous WinCC..... | 43 |
| Figure (IV.1) : Réseaux de normalisation de l'entrée analogique sous STEP7..... | 47 |
| Figure (IV.2) : Réseau de mise en marche ou arrêt et de sécurité sous STEP7..... | 48 |
| Figure (IV.3) : Réseau de l'action de fermeture du vérin de grand débit sous STEP7..... | 49 |
| Figure (IV.4) : Réseau d'affectation de la valeur de l'entrée analogique sous STEP7..... | 50 |
| Figure (IV.5) : Réseau de l'opération de soustraction sous STEP7..... | 51 |
| Figure (IV.6) : Réseau de l'opération d'adition sous STEP7..... | 51 |
| Figure (IV.7) : Vue d'accueil sous WinCC..... | 52 |
| Figure (IV.8) : Vue du fonctionnement automatique sous WinCC..... | 53 |
| Figure (IV.9) : Vue du fonctionnement semi Automatique sous WinCC..... | 54 |
| Figure (IV.10) : Vue de l'état des alarmes sous WinCC..... | 54 |
| Figure (IV.11) : Configuration du champ E/S sous WinCC..... | 55 |
| Figure (IV.12) : Fenêtre générale sous WinCC..... | 56 |
| Figure (IV.13) : Fenêtre animation sous WinCC..... | 56 |

LISTE DES TABLEAUX

| | |
|-----------------------------|----|
| Tableau des variables | 17 |
|-----------------------------|----|

Introduction générale

Introduction générale

La croissance de la demande internationale et du marché local en matière de sucre sous ses différentes formes et couleurs, a poussé CEVITAL à augmenter sa production tout en garantissant la bonne qualité sur ses deux unités de raffinage, la 3500 et la 3000 tonnes.

La raffinerie de sucre 3500 tonnes est une unité qui traite le sucre dès son arrivée du port comme matière première, en passant par différentes étapes pour obtenir du sucre blanc.

La peser du produit est une tâche essentielle dans une entreprise de forte production telle que CEVITAL, c'est pour cela que la direction de la raffinerie 3500 tonnes ont installé plusieurs balances peseuses comme à l'entrée, sur les tapis qui transportent le produit d'une étape à une autre et à la section de stockage, où une balance de marque PRICIA MOLEN pèse la quantité de sucre raffiné par jour.

L'évolution, la concurrence, la complexité, la rapidité et les exigences du marché sont des difficultés que l'entreprise rencontre tous les jours dans ce monde industriel évoluant à chaque minute, pour faire face, les entreprises doivent faire preuve de souplesse, de dynamisme pour pouvoir améliorer la qualité et la quantité des produits, d'optimiser le potentiel de production, de réduire les coûts afin de conforter sa position sur le marché et de pouvoir répondre aux besoins des clients, pour ce la elles font de plus en plus recours à l'automatisation des systèmes de production.

De nombreux constructeurs d'automates programmables existent, ils offrent des gammes de grande diversité. La firme allemande SIEMENS offre l'une des plus grandes gammes de produits, telles que la série S7 200, S7 300, S7 400 et la série S7 1500.

Les problèmes qui nous ont été posés au sein de l'unité 3500 tonnes, concernent essentiellement le fonctionnement d'une balance peseuse situant dans la dernière étape de raffinage. Ses problèmes se présentent dans le fait de changer la platine de commande de la machine par un automate programmable industriel, de réaliser une supervision du fonctionnement de la balance au niveau de la salle de contrôle, pour faciliter le travail de l'opérateur, de régler le problème du manque d'information sur l'état des casques. Dans le premier chapitre nous décrivons l'unité de raffinage d'une manière générale, les différents composants de la balance peseuse et leur fonctionnement. Dans le deuxième chapitre, nous citerons les problèmes posés et on présentera l'analyse fonctionnelle réalisée pour le processus avec les GRAFCET conçus par le logiciel AUTOMGEN. Le chapitre trois, est

Introduction générale

consacré aux automates programmables industriels d'une manière générale et de l'automate S7-300 d'une manière particulière. Ensuite, on fera une présentation du logiciel de programmation STEP7 et description du logiciel de supervision WinCC flexible. Le chapitre quatre est dédié à la partie programmation, et l'élaboration d'une interface graphique, c'est-à-dire la supervision du système étudié.

Ce travail est finalisé par une conclusion générale qui récapitule le contenu du manuscrit.

PREAMBULE

I. Présentation du groupe CEVITAL :

CEVITAL est un ensemble industriel Algérien présidé par son fondateur l'homme d'affaires Algérien Issâd Rebrab, l'entreprise et d'un capital de 400 milliards de dinars Algériens et d'un chiffre d'affaire de 4 milliards de dollars en 2015, concentré en premier temps, dans le secteur de l'agroalimentaire : raffinage d'huile et de sucre et dérivés, conditionnement et distribution des produits destinés à l'alimentation humaine.

L'ensemble industriel a connu une croissance importante durant ces quinze dernières années, ce qui le rend un pôle économique national est un leader dans le domaine de l'agroalimentaire. Sa stratégie économique consiste à garantir le besoin du marché local et à pouvoir exporter vers le marché mondial surtout le marché européen.

2. Emplacement géographique :

CEVITAL possède un emplacement géographique très avantageux, à 3 Km du sud-ouest de la ville de Bejaia, elle est au voisinage du nouveau quai du port et à proximité de la route nationale 26 menant vers l'aéroport, ce qui la fait profiter de la facilité de circulation de ces produits et ces personnes, que ça soit de la voie maritime ou aérienne. La figure ci-dessous, montre l'emplacement exact de l'entreprise.



Figure (1) : Emplacement géographique de l'entreprise [1].

CHAPITRE 1

Présentation de l'unité et de la bascule PRECIA MOLEN

I.1 Introduction :

Dans ce chapitre nous allons présenter les différentes étapes et sections sur lesquelles passe le sucre ainsi que la bascule PRECIA MOLEN, son mode de fonctionnement, son rôle ainsi que les technologies et instruments utilisés.

I.2 Unité de raffinage de sucre 3500 T :

L'unité de raffinage de sucre 3500 tonnes est l'une des deux unités de CEVITAL, elle a été rénové en 2012 pour augmenter la capacité de production de 3000 à 3500 tonnes / jour.

I.2.1 Processus de raffinage :

Le processus de raffinage appliqué comporte plusieurs étapes réparties en dix sections où on traite le sucre dès son arrivée sous sa forme naturelle, jusqu'à sa sortie comme sucre cristallisé prêt à l'emballage.

I.2.1.1 Section 1 : Affinage et Refonte

***Affinage :** Après l'avoir pesé, le sucre roux sera mélangé à un liquide d'affinage saturé en sucre puis malaxé sans provoquer la fonte des cristaux (sous forme de pâte). La séparation de l'égout d'affinage du sucre se fait par une centrifugeuse discontinue, l'égout va être traité une autre fois pour retirer le maximum de sucre qu'il contient.

***Refonte :** La pâte obtenue dans le raffinage sera fondue à l'eau dans un fondoir pour obtenir un sirop.

I.2.1.2 Section 2 : Carbonatation

Le procédé de carbonatation est de nature chimique, il permet de décolorer le sirop après la fonte. Le sirop sera traité par une solution de chaux alimentaire (lait de chaux), le mélange barbotera avec du gaz carbonique qui insolubilise la chaux en carbonate de calcium afin de piéger les impuretés présentes dans le sucre.

I.2.1.3 Section 3 : Filtration

Le sirop issu de la carbonatation contient une suspension de carbonate de calcium. Cette dernière est séparée du mélange par une filtration par des filtres autonettoyants à bougies en toile. Le sirop filtré est envoyé à la décoloration, la boue résultante de la filtration est envoyée vers un filtre à presse pour récupérer le sucre résiduel, ce sucre récupéré et utilisé pour former le petit jus utilisé au début pour le raffinage et fonte, la boue résultante est utilisée comme engrais.

I.2.1.4 Section 4 : Décoloration

Le but de cette section est la décoloration du sirop pour lui enlever la couleur marronne, elle est réalisée en ajoutant de la résine au sirop filtré. Cette manière permet d'enlever les pigments colorant résiduels. Une station de nanofiltratio est utilisée pour épurer les rejets de la génération et permet de pousser la décoloration au maximum.

I.2.1.5 Section 5 : Concentration

La concentration est une procédure qui vient juste avant la cristallisation, son but est d'augmenter la densité du sirop par l'évaporation, pour qu'il soit prêt à être cristallisé rapidement.

Le sucre sous forme de sirop décoloré qui provient de la section (04) de décoloration est concentré en applique une pression de 6 bars, une foi la température du sirop atteint 100°C, il sera pompé vers un autre concentrateur à vide où on le met à une basse température pour éviter la caramélisation.

I.2.1.6 Section 6 : Cristallisation haut produit

La transformation du sirop en cristaux de sucre, se fait dans des enceintes équipées d'un système de cuisson de sirop appelées cuites, ce procédé de cristallisation se réalise dans les cuites sous vides (petite pression relativement à la pression atmosphérique : 200 Mbars) par ajout de sucre glacé (poudre de sucre) mélangé avec de l'alcool pour pouvoir l'introduire dans la cuite, et que l'alcool s'évapore instantanément, après la constitution des cristaux le produit passe par des turbines centrifugeuses essoreuses ce qui nous permet de séparer ses impuretés et d'avoir un sucre blanc prêt à être envoyé au séchage.

I.2.1.7 Section 7 : Séchage

Le sucre ainsi obtenu après centrifugation sera humide, alors il nécessite un séchage pour éliminer cette humidité. On utilise donc un tube sécheur et un refroidisseur, en sortant de la cristallisation, le sucre est humide à 0.05%, pour une bonne conservation on le refroidit dans un sécheur à lit fluidisant et on l'envoie vers les silos de maturation pour finaliser la déshumidification et le stocker.

I.2.1.8 Section 8 : Stockage et maturation

Dans cette section le sucre provenant du séchage est stocké dans le silo, et ce pour une durée minimale de 48 heures pour assurer la maturation avec de l'air conditionné qui élimine l'humidité résiduelle contenue dans les cristaux du sucre, il est ensuite acheminé vers le

conditionnement où il sera ensaché, convoyé dans un bateau pour exportation ou livré en vrac via des camions de grand tonnage.

I.2.1.9 Section 9 : Utilités

Cette section constitue toutes les utilités des autres sections, elle alimente les sections de l'eau chaude, de la vapeur, elle garantit la préparation du petit jus, la préparation du lait de chaux, de servir les turbines en eau de clairçage et toute utilité pour la raffinerie.

I.2.1.10 Section 10 : Conditionnement

Le sucre mûré soutiré du grand silo de maturation est acheminé par un convoyeur et passe par un casse –grugeant qui va éliminer toutes les mottes et morceaux de sucre, ensuite il sera criblé afin de le calibrer. Le sucre blanc est ensuite transporté par des convoyeurs qui alimenteront l'unité de conditionnement en sucre blanc où il sera conditionné sous forme de sacs de 50Kg, 10Kg, 1Kg ou des big-bags d'une tonne.

I.3 Présentation de la bascule PRECIA MOLEN

Le fonctionnement d'une balance électronique est basé sur l'utilisation de capteurs de force, suivis d'amplificateurs pour lui donner plus ou moins de sensibilité et qui génère un signal analogique qu'il transforme en un signal numérique par un convertisseur Analogique/Numérique (A/N).

La bascule totalisatrice à pesées non-constantes PRECIA MOLEN (séries **ABS-XL**) et de commande électropneumatique. C'est une bascule à trémie avec afficheur séparé. Elle se compose essentiellement d'une goulotte d'alimentation, un casque de remplissage, une benne peseuse et d'un casque de vidange. La goulotte d'alimentation est fixée sous la trémie amont. La benne peseuse se trouve entre la goulotte d'alimentation et la trémie avale. La balance de circuit **ABS-XL** a été conçue pour peser automatiquement dans un processus des produits granules secs faciles à infuser, tels que les (céréales le sucre...). Elle est associée au système d'instrument ABS I 400, elle remplit la fonction de totalisateur lourd discontinu non constante. Sa conception simple et robuste facilite son installation ainsi que sa maintenance.



Figure (I.1) : Balance de circuit **ABS-XL** (ABS I 400) [2].

I.4 Le rôle de la bascule au sein de l'unité

La bascule se situe dans la tour de remplissage du silo de stockage, son rôle est de peser la quantité du sucre produite par la raffinerie. La peser de la bascule permet d'avoir le rondement journalière de la raffinerie et la quantité produite annuellement.

I.5 Instruments constituant la bascule PRECIA MOLEN

Les parties principales dont se compose la bascule **PRECIA MOLEN** sont les suivantes :

I.5.1 Sources d'alimentation en énergie

Les sources d'alimentations en énergie sont les plus importantes dans l'industrie, elles permettent d'alimenter tous les pré-actionneurs et les actionneurs en énergie nécessaire de façon permanente.

La bascule peseuse est alimentée par une source pneumatique qui est le compresseur. Le dernier peut être à vis ou à piston, il est la source d'alimentation en énergie pneumatique, il alimente les circuits d'air comprimé, et leur pression de sortie est de l'ordre de 700 à 800 kPa (7 à 8 bars). Il garantit ainsi une pression de service minimal de 600 kPa (6 bars) au niveau des

vérins en cas de défauts d'étanchéité (points de fuite par lesquels l'air s'échappe de manière intempestive) et de pertes de charge dans les conduites [3].



Figure (I.2) : Compresseur à vis [4].

I.5.2 Pré-actionneur

Les pré-actionneurs servent à mettre en service ou hors-service un actionneur, une machine ou une installation. Ils permettent d'assurer, en toute sécurité, la bonne marche d'un équipement.

Le pré-actionneur utilisé pour la bascule, est le distributeur pneumatique qu'on utilise pour commander le départ, l'arrêt et la direction d'un débit d'air. Le distributeur aura pour rôle de réaliser les deux configurations possibles de branchements, en fonction d'un signal de commande extérieure. La bascule elle est équipée de trois distributeurs 4/2 bistable à commande électrique



Figure (I.3) : Distributeur pneumatique 4/2 bistable a commande électrique [5].

I.5.3 Actionneur

Les actionneurs sont des appareils de transformation de l'énergie. Ils permettent de transmettre l'énergie nécessaire pour le bon fonctionnement de la machine à partir de l'énergie disponible dans l'équipement (pneumatique, hydraulique ou électrique). Ils sont indispensables dans une machine, car ce sont eux qui créent l'action.

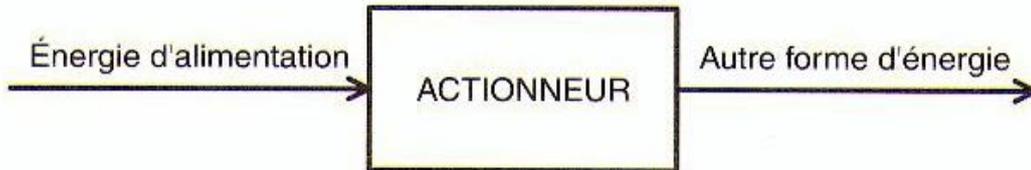


Figure (I.4) : Fonction générale des actionneurs [6].

A) Fonctions et caractéristiques des vérins pneumatiques

Le vérin pneumatique fait partie des actionneurs pneumatiques, il transforme l'énergie pneumatique en énergie mécanique.

L'énergie mécanique est produite sous forme d'un mouvement permettant de provoquer un déplacement ou de créer une force.

B) Constitution d'un vérin

Quel que soit le vérin, son type et son constructeur, il sera constitué des mêmes éléments. Le piston est solidaire de la tige qui peut se déplacer à l'intérieur du corps. Le dernier est délimité par le nez et le fond dans lequel sont aménagés les orifices d'alimentation en air comprimé. Les espaces vides qui peuvent être remplis d'air comprimé s'appellent les chambres.

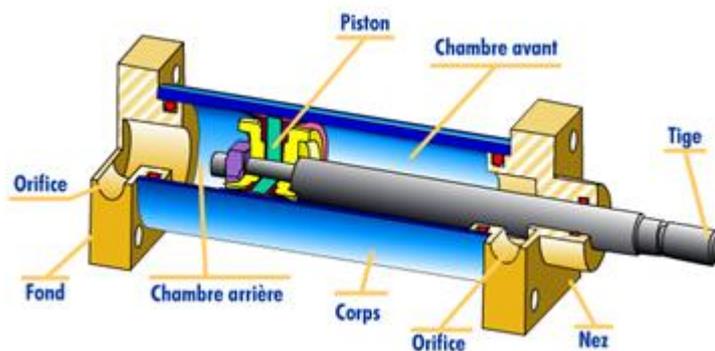


Figure (I.5) : Les éléments d'un vérin [7].

C) Types de vérins utilisé dans la bascule

- **Vérin double effet**

Les vérins à double effet sont alimentés en air comprimé des deux côtés, ils peuvent donc fonctionner dans les deux sens. La force transmise à la tige de piston pour l'aller est sensiblement supérieure à celle du retour, puisque la surface exposée à l'air comprimé du côté du piston est plus grande que du côté de la tige de piston.

Sur les vérins à double effet, chaque chambre de pression comporte un raccord. L'inversion du mouvement suppose donc l'échappement de l'air qui se trouve dans la chambre correspondante (côté piston ou côté tige de piston).



Figure (I.6) : Vérin double effet [3].

I.5.4 Les capteurs et les détecteurs utilisés dans la bascule

Les capteurs servent à détecter des informations et à les transmettre au module de traitement des signaux sous une forme exploitable, ils ont des usages très variés dans le domaine technique, se déclinent en plusieurs modèles et utilisent différents principes de fonctionnement. Il est donc important de les caractériser.

Les capteurs peuvent notamment être classés en fonction des critères suivants :

- principe de fonctionnement (optique, inductif, mécanique, fluide, ...);
- valeur mesurée (course, pression, distance, température, valeur de PH, intensité lumineuse, présence d'objets, ...);
- signal de sortie (analogique, numérique, binaire, ...).

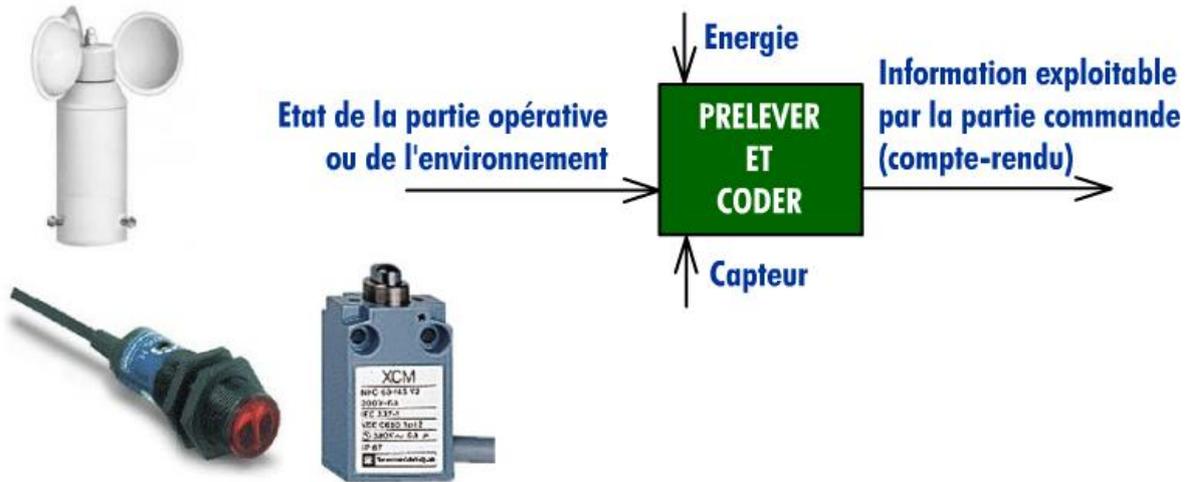


Figure (I.7) : Principe de fonctionnement des capteurs [7].

A) Détecteurs magnétiques

Les détecteurs magnétiques (aussi appelés interrupteurs à lame souple ou fin de course) sont de plus en plus employés dans les systèmes automatisés, ils sont directement fixés sur le corps d'un vérin dont le piston comporte un aimant.

Lorsque l'aimant passe à proximité du capteur, le contact électrique se ferme et l'information est donnée à la partie de la commande. Lorsque l'aimant s'éloigne du capteur, le contact s'ouvre et le circuit n'est plus établi, l'information disparaît.



Figure (I.8) : Fonctionnement d'un capteur magnétique [7].

B) Capteur de force

Un capteur de force est un dispositif utilisé pour convertir une force (par exemple un poids) appliquée sur un objet en un signal électrique. Le capteur est généralement construit en utilisant des jauges de déformation connectées en un pont approprié. Un amplificateur est nécessaire pour lire le signal délivré par le transducteur.

Le capteur de force, du type capacitif ou piézoélectrique, mesure des forces dynamiques (en traction/compression). La sortie peut être exprimée en différentes unités, telles le μg (balances analytiques), kg, t (pesée des camions, ...).



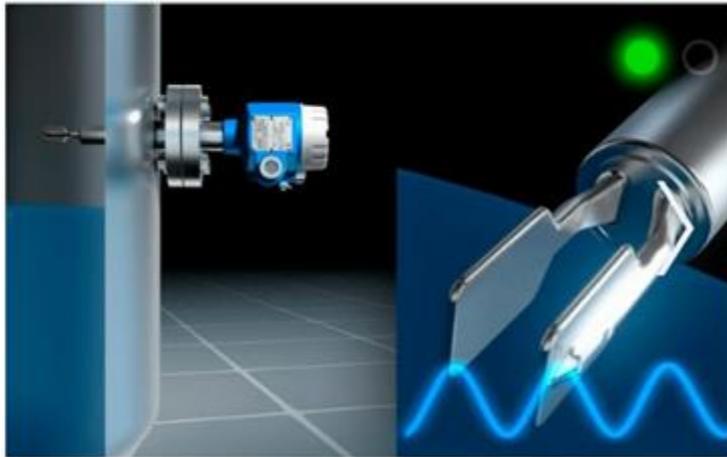
Figure (I.9) : le capteur de force [2].

C) Détecteur à lames vibrantes

Le barreau du détecteur est équipé d'un piézo-électrique. Les oscillations mécaniques générées sont converties en énergie électrique le produit entre en contact avec le barreau, l'oscillation mécanique s'interrompt, entraînant ainsi le changement d'état du contact. Disponible avec une fourche polie et un boîtier inox avec tri-clamp ou avec un boîtier aluminium et une fourche revêtue. Appareil sans entretien pour des applications typiques : céréales, farine, cacao, sucre, pigments en poudre, chaux. La bascule elle est équipée de deux types de détecteurs à lames vibrantes le **Soliphant FTM50** utilise comme détecteur de bourrage et le **VLD** utilise comme un détecteur de présence [8].



Figure (I.10) : Détecteur à lame vibrante (VLD) [8].



Figure(I.11) : Détecteur à lames vibrantes Soliphant FTM50 [9].

I.6 Le cahier des charges

Le cahier des charges c'est un tableau de bord qui définit et décrit l'ensemble des caractéristiques d'un projet et détaille l'enchaînement des séquences de fonctionnement du processus.

Le fonctionnement de la balance PRECIA MOLEN est tout simple, il se déroule en trois étapes, le remplissage, la peser et la vidange.

Au début du cycle le programme vérifie la présence des conditions initiales de démarrage, si elles sont remplies toutes, le cycle peut démarrer.

Quand le capteur de présence détecte le produit dans la trémie amont le casque de remplissage s'ouvre au maximum (grand débit et petit débit) pour admettre le produit de la trémie amont dans la benne peseuse, à la première consigne le casque se referme à 70% (fermeture grand débit), à la deuxième consigne le casque se ferme à 100% (fermeture petit débit) marquant la fin du cycle de remplissage. Ensuite le programme mémorise le poids du produit qui se trouve dans la benne peseuse (le poids brut de la pesée). Une fois le poids brut enregistré, le casque de vidange s'ouvre pour évacuer le produit dans la trémie aval. Quand la benne peseuse est complètement vide, le casque de vidange se referme. Le programme mémorise le poids du produit éventuellement encore présent dans la benne peseuse (la tare). Enfin il calcule le poids net de cette première pesée en déduisant la tare du brut.

Touts signale reçu par les entrées, arrêt d'urgence (AUR) ou détecteur de bourrage qui se trouve dans la benne peseuse (DBBP), le processus ce réinitialiser et coupe immédiatement l'alimentation électrique qui stop toutes opérations et signale une alarme. Si le signal vient de détecteur de bourrage qui se trouve dans la trémie aval (DBTAV), le cycle de remplissage se fait

si la banne n'est pas pleine, si non il stop toutes les opérations et signale une alarme. Dès que le circuit d'arrêt d'urgence ou de défaut est restauré, le processus reprend le fonctionnement.

L'action de remplissage elle se fait par deux vérins doubles effet et celle de vidange par un vérin double effet.

I.7 Conclusion :

Dans ce chapitre, on a présenté l'unité de raffinage de sucre et on a décrit le processus de raffinage appliqué au produit, la deuxième partie du chapitre contient la présentation de la bascule peseuse de marque PRECIA MOLEN, ses différentes parties et les différents instruments et technologies constituant la machine. On a aussi décrit le mode de fonctionnement dans un cahier des charges, ce qui nous facilitera la tâche pour présenter dans le deuxième chapitre notre problématique et la solution proposée sous forme du GRAFCET.

CHAPITRE 2

Problématique et analyse fonctionnelle

II.1 Introduction

L'automatisation de chaque système demande une étude et une analyse du mode de fonctionnement. L'analyse permet d'identifier les problèmes existants alors des modifications sont nécessaires sur le système. Après modification, une analyse fonctionnelle est obligatoire pour assurer le bon fonctionnement et le respect du cahier des charges. Pour cela on a présenté dans ce chapitre les différents problèmes du système étudié, aussi que les solutions et modifications appropriées sur la machine et sur sa commande. On a aussi réalisé une analyse fonctionnelle avec le GRAFCET.

II.2 Problématiques

Le système de pesé présenté dans le chapitre précédent, est commandé depuis la platine frontale du MASCON-3. Cette dernière comprend des éléments de commande, un écran LCD pour afficher les valeurs et la supervision et un buzzer (trembleur) pour l'alarme sonore en cas de défaut.

Cette platine qui est installée dans une salle d'armoires électriques, rend les tâches de commande et de supervision difficiles aux opérateurs, puisqu'ils sont obligés de se déplacer de la salle de contrôle à la salle des armoires pour modifier ou vérifier le fonctionnement.

Les deux casques (de remplissage et de vidange) sont équipés chaque un d'un capteur pour détecter l'ouverture ou la fermeture du casque. Le capteur est fixé près du casque pour indiquer l'ouverture et la fermeture de ce dernier. Mais par fois il ne s'ouvre pas ou ne se ferme pas complètement, ce que le capteur ne peut pas indiquer, ce qui fait que l'opérateur n'est pas informé de l'état exact du casque, ce manque d'information fait perdre du temps (si le casque est à moitié ouvert) et fausse les valeurs (si le casque est à moitié fermé).

Le système n'est pas équipé d'un mode manuel, ce qui oblige les opérateurs de se déplacer sur les lieux de la bascule pour intervenir, en cas d'un arrêt quelconque pour compléter le cycle.

II.3 Solutions proposées

Après étude et coopération avec l'ingénieure responsable à l'unité de raffinage, on a proposé des solutions aux problèmes cités, qui se présente comme suit :

Le premier problème cité qui est la supervision et la platine de commande située loin des opérateurs, on remplace par un API de marque SIEMENS et de réaliser un pupitre de

supervision, avec le logiciel WINCC FLIXIBLE, pour pouvoir le relier au système de supervision déjà existant.

Le deuxième problème qui est le manque d'information de l'état des casques (remplissage et vidange), durant le fonctionnement, on la résolu par l'ajout de capteurs de fin de course (fdc) de sortie et de rentrée pour chaque vérin (actionnant chaque casque), afin de connaître l'état exact du vérin dans chaque étape du processus de pesage.

La dernière solution c'est la conception d'un programme pour la gestion du fonctionnement manuel afin d'offrir à l'opérateur la possibilité de permuter entre les deux programmes (fonctionnel automatique, ou manuel) par le pupitre de supervision.

II.4 Cahier des charges modifié

Après proposition des solutions aux problèmes cités on a rédigé un autre cahier des charges, pour définir le nouveau fonctionnement de la bascule qui y est comme suit :

Si le système est en état initial (conditions initiales vérifiées, les vérins sont tous rentrés, les casques sont fermés et présence du sucre dans la trémie amont). Avec appuis sur le bouton marche dans le pupitre le cycle se lance au mode automatique, il commence en sortant les deux vérins de remplissage (grand et petit débit) pour charger la benne peseuse. Lorsque le poids pesé atteint la consigne du grand débit, le vérin du grand débit rentre pour fermer le casque de remplissage à grand débit, le remplissage continue au mode petit débit jusqu'à atteindre la consigne du petit débit, le vérin petit débit rentre pour fermer le casque de remplissage complètement. Une fois le casque fermé, la bascule pèse le poids complet, grâce aux quatre capteurs de force installés dans les quatre coins de la benne. Après la pesée, le vérin de vidange sort pour ouvrir le casque de vidange, une fois confirmé par le détecteur de bourrage que la trémie avale n'est pas pleine de sucre. Le casque de vidange ne se ferme pas jusqu'à ce que les capteurs de pesée affichent un poids inférieur ou égal à 100 Kg, une fois le vérin de vidange rentre et le casque fermé, la bascule pèse le résidu dans la benne, pour le soustraire du poids complet et affiche le poids final, et le cycle reprend.

La protection de la bascule se fait par l'arrêt d'urgence et le détecteur de bourrage dans la benne peseuse. Dans le cas où l'un des deux est vérifié le fonctionnement de la bascule est interrompu, le système est réinitialisé (les casques sont tous fermés et les vérins sont rentrés) et permute le fonctionnement vers le mode de semi-automatique. Une fois le problème est résolu l'opérateur doit finir le cycle en mode semi-automatique.

Le mode semi-automatique permet à l'opérateur d'effectuer toute action voulue séparément des autres actions (sorties ou rentrées de vérin), à partir du pupitre de supervision grâce aux boutons de commande.

II.5 a Description du langage GRAFCET

Le langage GRAFCET (Graphe Fonctionnel de Commande Etape Transition), est créé en 1977 par un groupe de travail l'AFCEC une association française. C'est un outil graphique de définition pour l'automatisme séquentielle, c'est un langage. Il peut être réalisé par un séquenceur ou être programmer sur automate ou ordinateur, c'est avant tout un outil de modélisation, il est sauvs utiliser pour la description du fonctionnement de l'automatisme contrôlant le procède.

Le GRAFCET peut être présenté par un ensemble :

- D'étapes auxquelles sont associées des actions
- Des transitions auxquelles sont associées des réceptivités
- Des liaisons orientées reliant les étapes aux transitions et les transitions aux étapes.

La figure II.1 représente un exemple de GRAFCET

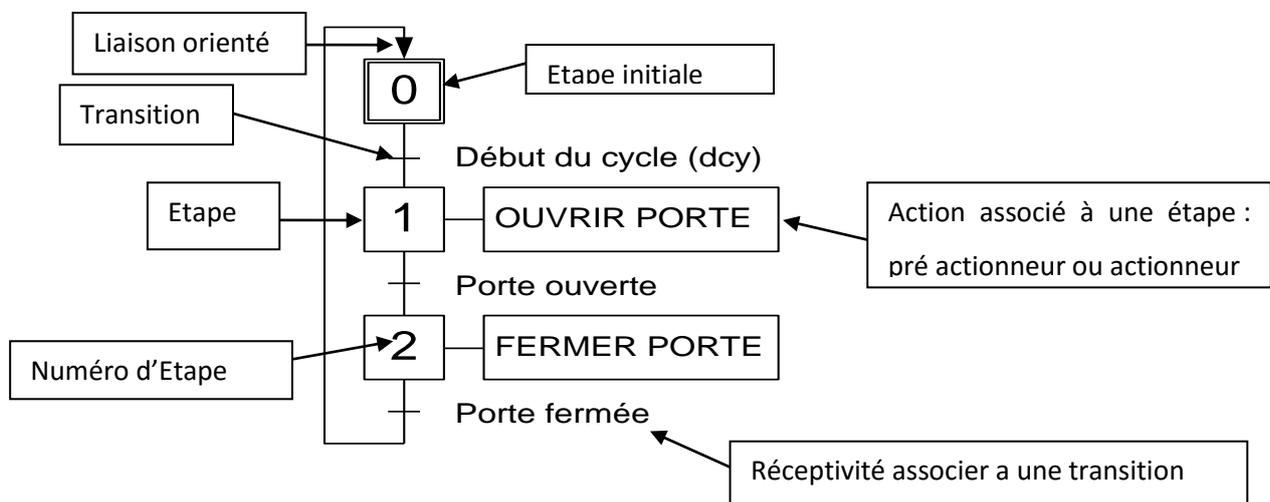


Figure (II.1) : Différents composants d'un GRAFCET [10]

II.6 Présentation du logiciel AUTOMGEN

Durant la réalisation de l'analyse fonctionnelle du système décrit dans le chapitre précédent, on a exploité un logiciel qui nous a permis de simuler les GRAFCETS réalisés, ce logiciel est l'AUTOMGEN.

AUTOMGEN est une famille des logiciels universels pour l'automatisation, avec pour principale fonctionnalité, la création d'application d'automatisme, pour automates et microprocesseurs, supervision et simulation en 3D grâce à AUTOSIM (un module de simulation pneumatique / électrique / hydraulique), il permet de créer des systèmes logiques séquentiel (décrit par un GRAFCET).

II.7 Les GRAFCETS réalisés

Le nouveau cahier des charges, formulé avec toutes les modifications proposées, nous a permis de faire l'analyse fonctionnelle de la machine et de réaliser un programme graphique (GRAFCET) sous AUTOMGEN.

Le programme graphique est constitué de quatre GRAFCET.

II.7.1 Tableau des variables

| symboles | commentaires |
|----------|---|
| fdcvgdr | fin de course vérin grand débit rentrée |
| fdcvpdr | fin de course vérin petit débit rentrée |
| fdcvvr | fin de course vérin vidange rentrée |
| fdcvgds | fin de course vérin grand débit sortie |
| fdcvpds | fin de course vérin petit débit sortie |
| fdcvvs | fin de course vérin vidange sortie |
| pstam | détecteur de présence de sucre dans la trémie amant |
| dbbp | détecteur de bourrage dans la bane peseuse |
| dbtav | détecteur de bourrage dans la trémie avale |
| vgdr | vérin grand débit rentrée |
| vpdr | vérin petit débit rentrée |
| vvr | vérin vidange rentrée |
| vgds | vérin grand débit sortie |
| vpds | vérin petit débit sortie |
| vvs | vérin vidange sortie |
| cfcgd | commande de fermeture de casque grand débit |
| cfcpd | commande de fermeture de casque petit débit |
| cfcv | commande de fermeture de casque vidange |
| cocgd | Commande d'ouverture de casque grand débit |

| | |
|--------------|--|
| cocpd | Commande d'ouverture de casque petit débit |
| cocv | Commande d'ouverture de casque vidange |
| ma | marche |
| aur | arrêt d'urgence |
| M (0, 3, 5) | mots système |
| f0 : {00} | désactive toutes les étapes du GRAFCET 0 sauf l'étape 0 qui sera activée |
| f200 : {200} | désactive toutes les étapes du GRAFCET 200 sauf l'étape 200 qui sera activée |
| f100 : {100} | désactive toutes les étapes du GRAFCET 100 sauf l'étape 100 qui sera activée |

II.7.2 GRAFCET de Sécurité et de Mise en Marche

C'est le GRAFCET qui gère de la mise en marche de la machine, et de l'arrêt en cas d'un défaut ou l'appuie sur le bouton d'arrêt d'urgence.

➤ **GRAFCET de mise en marche et de sécurité**

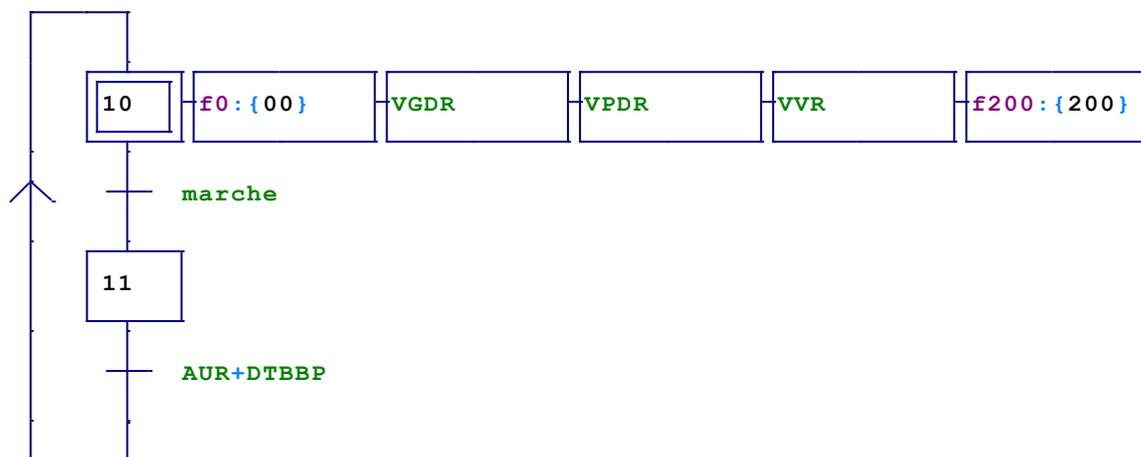


Figure (II.2): GRAFCET de sécurité et de mise en marche sous AUTOMGEN.

II.7.2 GRAFCET de Permutation entre les modes automatique/manuel

Il nous permet de choisir entre le mode de fonctionnement auto-manuel.

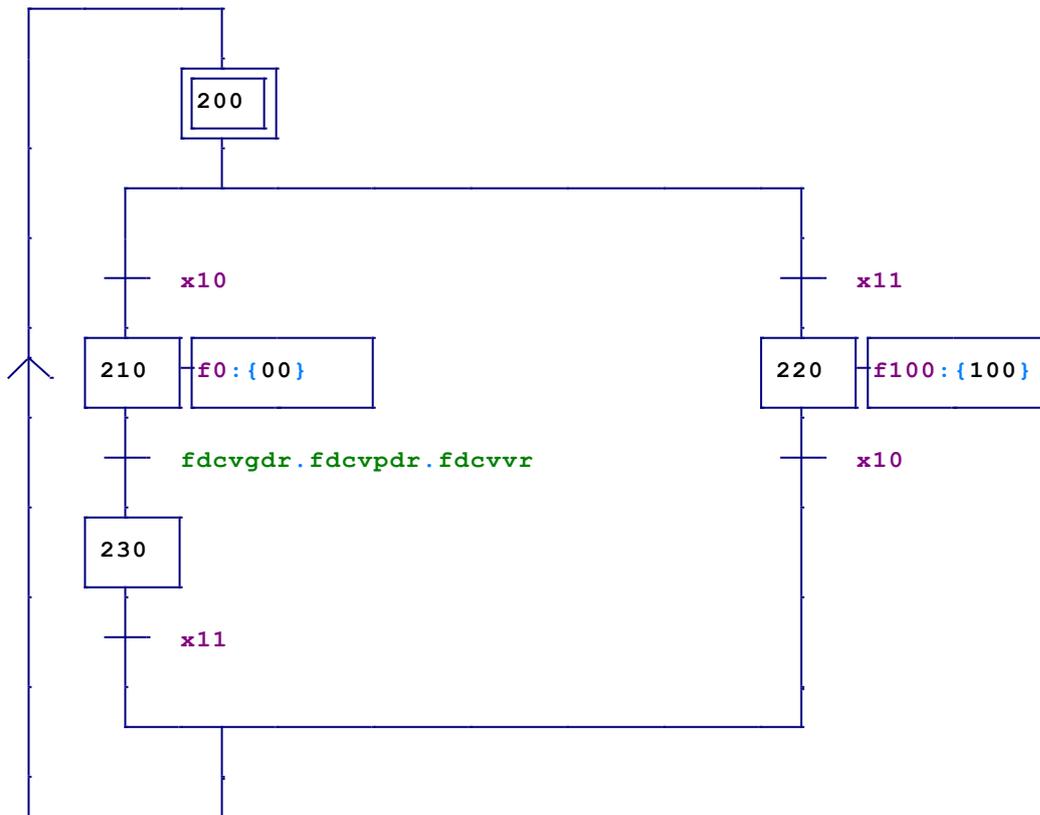


Figure (II.3): GRAFCET de Permutation sous AUTOMGEN.

II.7.3 GRAFCET du fonctionnement automatique

C'est le GRAFCET principal du fonctionnement normal de la machine.

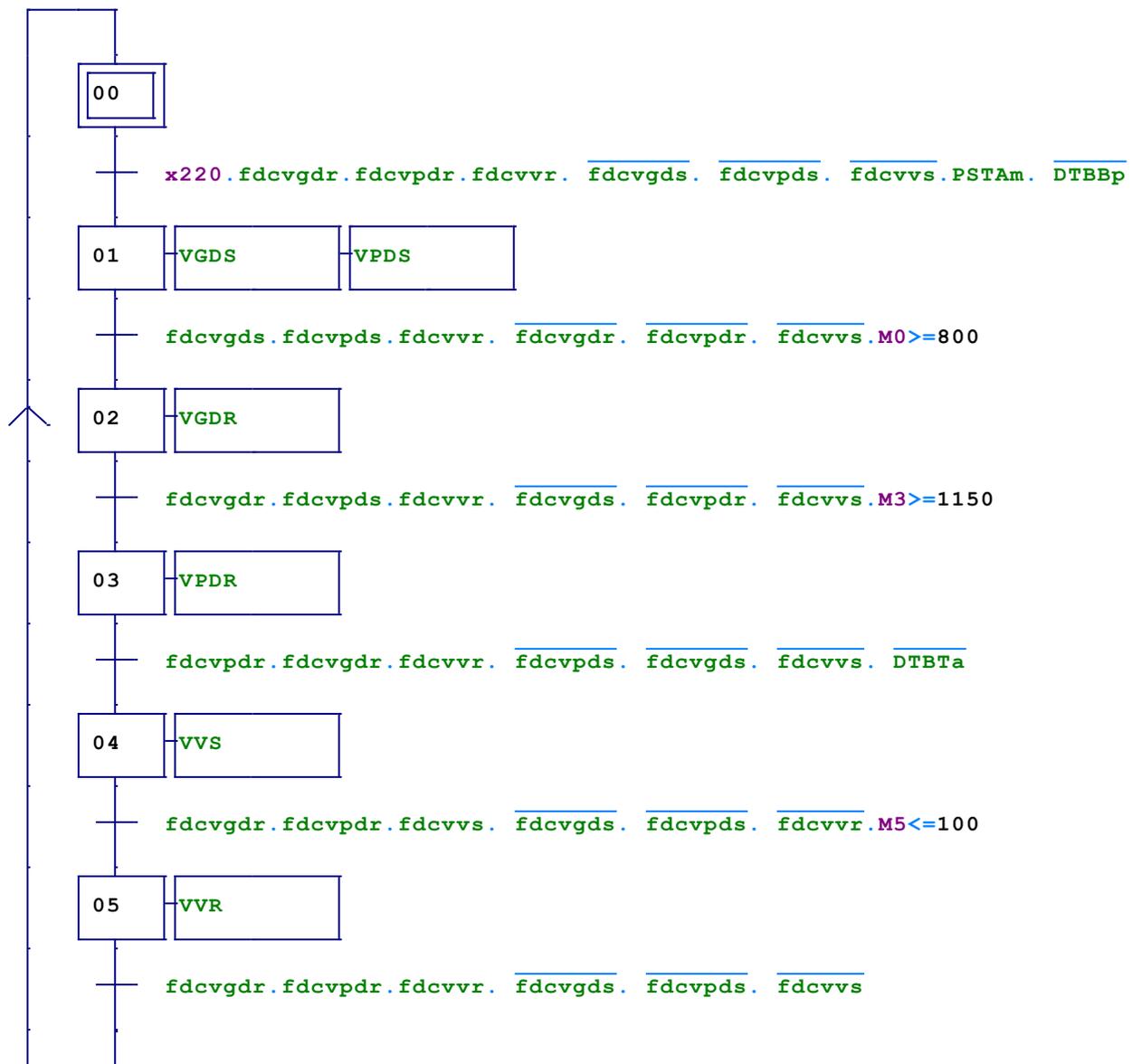


Figure (II.4): GRAFCET du fonctionnement automatique sous AUTOMGEN.

II.7.4 GRAFCET du Fonctionnement semi automatique

C'est le GRAFCET principal en mode semi-automatique. On utilise ce mode en cas de défaut ou d'arrêt d'urgence.

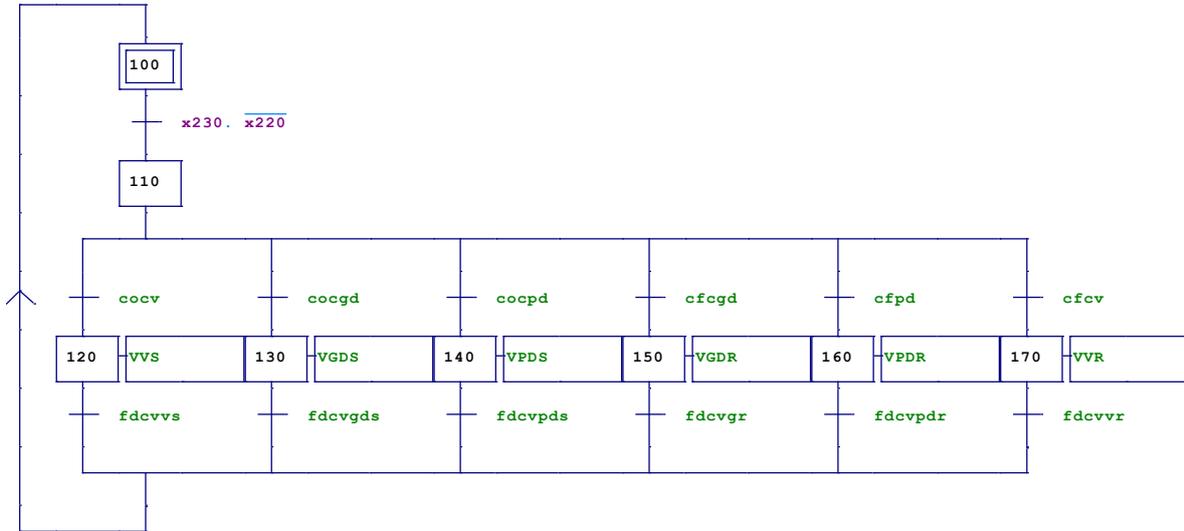


Figure (II.5): GRAFCET de fonctionnement semi automatique sous AUTOMGEN.

II.8 Equation des GRAFCET

II.8.1 Equation des étapes du GRAFCET du fonctionnement automatique

$$S_0 = \overline{fdcivr}.\overline{fdcivpr}.\overline{fcdvgdr}.\overline{fcdvvs}.\overline{fcdvpds}.\overline{fcdvgds}.x_5$$

$$R_0 = x_1$$

$$S_1 = \overline{fcdvgdr}.\overline{fcdvpr}.\overline{fcdivr}.\overline{fcdvgds}.\overline{fcdvpds}.\overline{fcdvvs}.\overline{pstam}.\overline{dbbp}.x_{11}.x_{220}.x_0$$

$$R_1 = x_2$$

$$S_2 = \overline{fcdvgds}.\overline{fcdvpds}.\overline{fcdivr}.\overline{fcdvgdr}.\overline{fcdvpr}.\overline{fcdvvs}.\overline{comparateur}.x_1$$

$$R_2 = x_3$$

$$S_3 = \overline{fcdvgdr}.\overline{fcdvpds}.\overline{fcdivr}.\overline{fcdvgds}.\overline{fcdvpr}.\overline{fcdvvs}.\overline{comparateur}.x_2$$

$$R_3 = x_4$$

$$S_4 = \overline{fcdivr}.\overline{fcdvgdr}.\overline{fcdvpr}.\overline{fcdvvs}.\overline{fcdvgds}.\overline{fcdvpds}.\overline{dbtav}.x_3$$

$$R_4 = x_5$$

$$R_5 = \overline{fcdvgdr}.\overline{fcdvpr}.\overline{fcdvvs}.\overline{fcdvgds}.\overline{fcdvpds}.\overline{fcdivr}.\overline{comparateur}.x_4$$

$$R_5 = x_0$$

II.8.2 Equation des étapes du GRAFCET du fonctionnement semi automatique

$$S_{100}=(fdecvvs.x_{120})+(fdecvgds.x_{130})+(fdecvpd.x_{140})+(fdecvvr.x_{150})+(fdecvgdr.x_{160})+(fdecvpdr.x_{170})$$

$$R_{100}=x_{110}$$

$$S_{110}=x_{230}.\overline{x_{220}}.x_{11}$$

$$R_{110}=x_{120}+x_{130}+x_{140}+x_{150}+x_{160}+x_{170}$$

$$S_{120}=cocv.x_{110}$$

$$R_{120}=x_{100}$$

$$S_{130}=cocgd.x_{110}$$

$$R_{130}=x_{100}$$

$$S_{140}=cocpd.x_{110}$$

$$R_{140}=x_{100}$$

$$S_{150}=cfcv.x_{110}$$

$$R_{150}=x_{100}$$

$$S_{160}=cfcgd.x_{110}$$

$$R_{160}=x_{100}$$

$$S_{170}=cfcpd.x_{110}$$

$$R_{170}=x_{100}$$

II.8.3 Equation des étapes du GRAFCET de sécurité et de mise en marche

$$S_{10}=(au+dbbp).x_{11}$$

$$R_{10}=x_{11}$$

$$S_{11}=ma.x_{10}$$

$$R_{11}=x_{10}$$

II.8.4 Equation des étapes du GRAFCET de Permutation entre modes automatique/manuel

$$S_{200}=(x_{11}.x_{230})+(x_{10}.x_{220})$$

$$R_{200}=x_{210}+x_{220}$$

$$S_{210}=x_{10}.x_{200}$$

$$R_{210}=x_{230}$$

$$S_{220}=x_{11}.x_{200}$$

$$R_{220}=x_{200}$$

$$S_{230} = fdcv\text{gdr}.fdcv\text{pdr}.fdcv\text{vr}.x_{210}$$

$$R_{230} = x_{200}$$

II.8.5 Equation des sortis des GRAFCET

$$vgds = x_1 + x_{130}$$

$$vpds = +x_1 x_{140}$$

$$vvs = x_4 + x_{120}$$

$$vgdr = x_2 + x_{10} + x_{160}$$

$$vpdr = x_3 + x_{10} + x_{170}$$

$$vvr = x_5 + x_{10} + x_{150}$$

II.9 Conclusion

Dans ce chapitre nous avons décrit la problématique et les solutions, proposées pour améliorer le fonctionnement de la machine. On a rédigé aussi un nouveau cahier des charges. On a défini les différentes parties d'un GRAFCET (les étapes, les transitions, les conditions de transition, les liaisons orientées, les actions associées à des étapes). Nous avons présenté le logiciel de simulation utilisé pour réaliser les GRAFCET. Pour finir on a réalisé le fonctionnement de la machine avec le langage GRAFCET.

CHAPITRE 3

L'API et les logiciels associés

III.1 Introduction :

Dans le domaine de l'automatisation des systèmes industriels l'automate programmable industriel (**API**), est l'outil maître car il est responsable de l'exécution du programme implémenté dans sa mémoire, ce dernier gère l'enchaînement des actions pour réaliser le processus industriel voulu. La large implication de l'**API** dans les systèmes automatisés est reliée à ses caractéristiques (robustesse, diversité des systèmes de commande auxquels il peut être employé, vitesse de réalisation et répétition des actions) qui le rend mieux adapté pour les milieux industriels.

Actuellement, il existe plusieurs marques et formes d'**API** sur le marché, cela est dû à la diversité d'utilisation et à la forte demande du marché industriel, les marques de constructeurs les plus réputées dans le monde industriel grâce à leurs professionnalismes et leur fiabilité sont ABB, TOSHIBA, SIEMENS...etc

La diversité des marques d'**API** a causé la multitude des logiciels de programmation des automates puisque, chaque constructeur développe son propre logiciel de programmation pour sa marque entière, ou pour une certaine gamme, par exemple l'**API** de marque SIEMENS avec le quel on à travailler est programmable avec le logiciel STEP7, associé au logiciel de supervision réservé à la marque, qui est le Win CC flexible.

Ce chapitre est destiné à la description de l'automate utilisé (**S7 300**) et les logiciels associés pour pouvoir réaliser l'automatisation de la machine décrite dans le premier chapitre.

III.2 Définition d'un système automatisé :

L'automatisation d'un système consiste à transformer toutes tâches réalisées par des opérateurs humains dans la partie de commande à un programme, qui est exécuté par l'API pour commander l'ensemble des actions à effectuer par les actionneurs.

Les systèmes automatisés sont composés de trois parties principales : partie opérative, partie relation (dialogue) et partie commande, ces trois parties s'échangent les informations grâce aux capteurs et pré-actionneurs.

L'automatisation permet d'augmenter la productivité et d'améliorer la sécurité ainsi que la qualité du produit.

III.3 Critères de choix de l'API :

Le choix de l'API revient au premier lieu à l'utilisateur, après avoir réalisé le cahier des charges de son système, il choisit l'API qui satisfait ces besoins économiques et techniques, pour notre cas on a pris compte des critères suivants :

- ✓ Le choix de l'entreprise CEVITAL : puisque l'entreprise a choisi la marque SIEMENS, vu à la confiance acquise par rapport à cette marque.
- ✓ La maîtrise des logiciels de programmation : le logiciel de programmation des API de SIEMENS est le STEP 7, un logiciel sur le quel les ingénieurs de CEVITAL en fait plusieurs formations ce qui a facilité sa maîtrise.
- ✓ Le nombre et la type des entrées/ sorties : après la réalisation du cahier des charges on a pu déterminer le nombre d'entrées/ sorties ainsi que leurs type, ce alors on a choisit un **API** avec des entrées et sorties tout ou rien (**ETOR/STOR**) et des entrées analogiques (**EANA**).
- ✓ La fiabilité et la robustesse : d'après l'expérience de l'entreprise CEVITAL la marque SIEMENS produisent des API de grandes robustesses et de fiabilité.

Après avoir tenu compte de tous ces critères, on a choisi l'API **S7 300** de la marque SIEMENS, équipé d'une **CPU 313 C**.

III.4 Présentation de l'API utilisé :

III.4.A Définition :

L'automate programmable industriel (**API**) est un dispositif électronique programmable destiné à la commande de processus industriels par un traitement séquentiel [11]. Dans son unité centrale de traitement, il exécute un nombre d'instructions introduites dans sa mémoire sous forme de programme, il est branché directement aux capteurs et pré-actionneurs grâce aux entrées/ sorties dont il est équipé. Il diffère des autres outils informatiques par son adaptation aux environnements industriels et ses langages de programmation qui sont développées spécialement pour le traitement des fonctions d'automatisation.

Le **S7 300** est un automate modulaire de la marque SIEMENS et il est utilisé dans plusieurs branches de l'industrie.

III.4.B Constitution d'un API S7 300 :

L'API S7 300 est constitué d'une alimentation (PS), d'une unité centrale de traitement (CPU), des modules d'entrées/ sorties (SM), des modules de communication (CP) et des modules spécifiques destinés à des fonctions particulières (IM et FM).

La figure ci-dessous représente les différents composants d'un API :

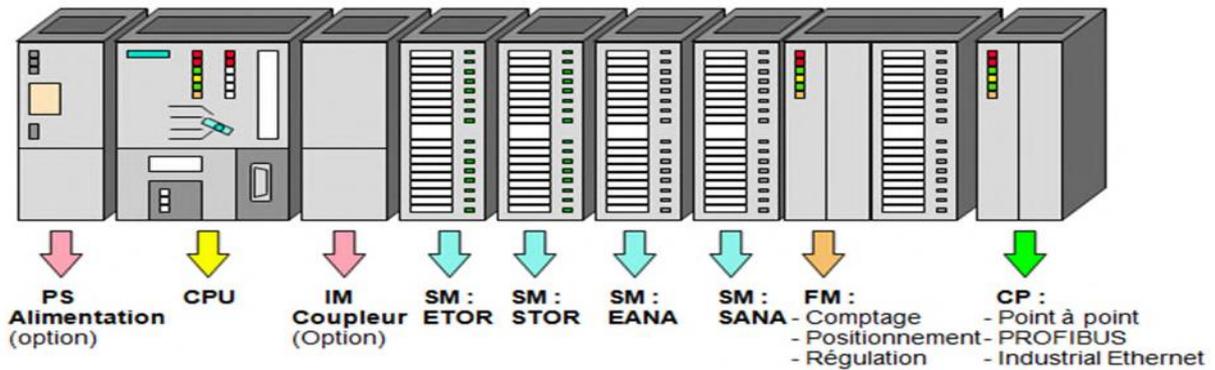


Figure (III.1) : Constitution d'un Automate modulaire S7 300 [12].

III.4.B.1 L'alimentation de l'API :

L'alimentation fournit une tension continue nécessaire pour les composants du rack afin de fonctionner avec de bonnes performances, elle peut être de 5 V, 12 V ou 24 V. Il est préférable d'ajouter un onduleur pour éviter les risques de coupure du réseau, une ou plusieurs alimentations peuvent être ajoutées pour les châssis d'extension en cas de nécessité.

L'alimentation de notre automate est PS 307 24 V / 5 A, une alimentation stabilisatrice d'entrée 120 / 230 V à courant alternatif et de sortie 24 V / 5 A à courant continu. La figure suivante représente l'alimentation choisie :

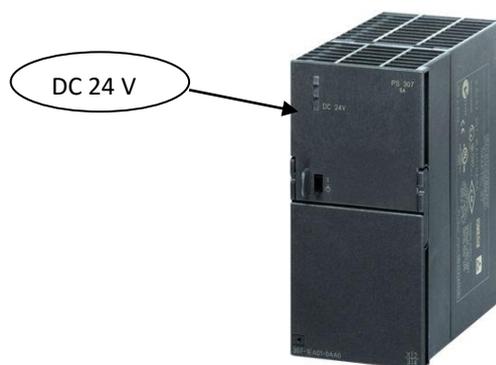


Figure (III.2) : Alimentation 307 24 V / 5 A pour S7 300. [13]

La LED de diagnostic (DC 24 V) de la face avant, permet de visualiser l'état de l'alimentation comme suit [14] :

- ✓ LED allumée : présence d'une tension de 24 V ;
- ✓ LED clignotante : surcharge coté circuit de sortie ;
- ✓ LED éteinte : court –circuit coté circuit de sortie.

III.4.B.2 Présentation de la CPU 313 C :

Le rôle de la CPU est l'exécution du programme utilisateur, l'alimentation de la busse interne du S7 300 avec 5 V et la communication par l'interface multipoints (**MPI**) avec d'autres correspondants d'un réseau **MPI**. Une mémoire est intégrée dans la **CPU**, elle enregistre le programme à exécuter et les images, des entrées/sorties.

La **CPU 313 C** est équipée aussi des modules intégrés d'entrées/ sorties **TOR** (24 entrées et 16 sorties) et aussi d'un module entrées/sorties analogiques a 16 bits (5 entrées et 2 sorties). La CPU, nous donne la possibilité d'ajouter d'autres modules en cas de nécessité.

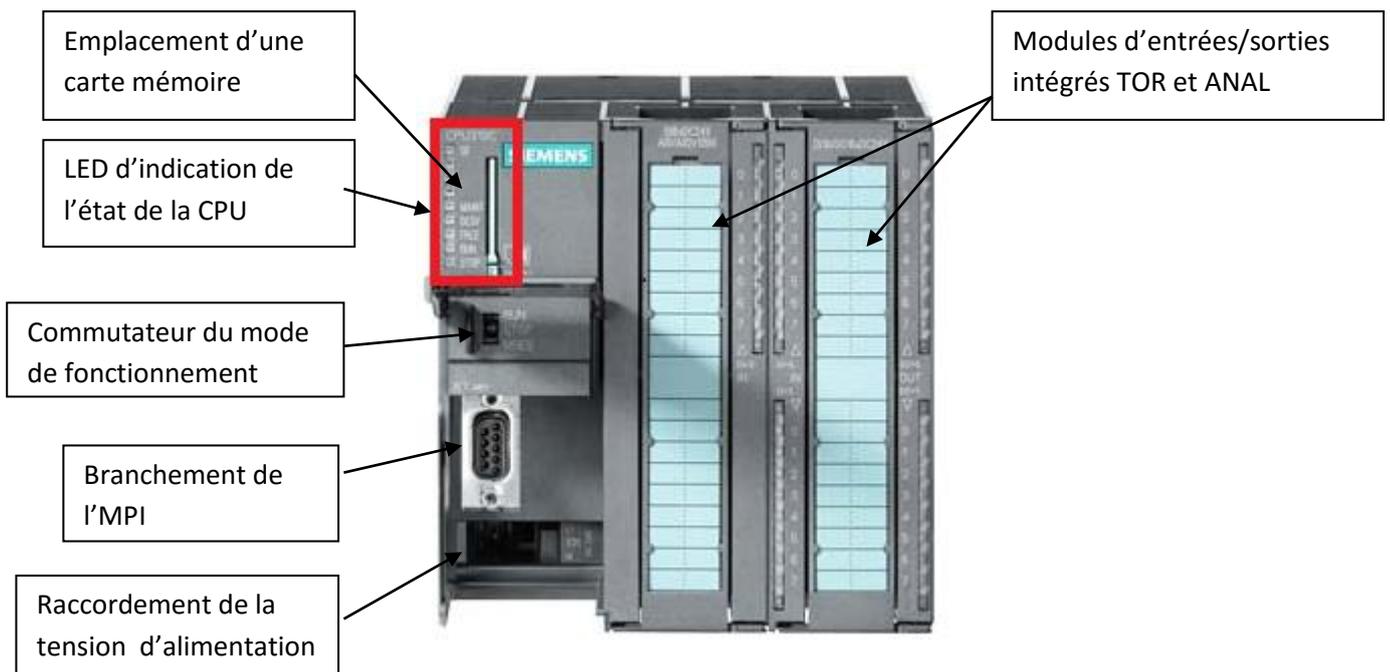


Figure (III.3) : La CPU 313 C avec ses différents éléments de commande et de signalisation [13].

➤ **Contrôles visuel des LED de la CPU 313 [14] :**

- ✓ LED SF : Signalisation groupée des défauts (erreur dans le programme ou défaut sur un module diagnostic)
- ✓ LED BATF : Défaillance de la pile (pile déchargée ou absente) ;
- ✓ LED DC 24 V : Témoin de présence de la tension d'alimentation interne 5 V ;
- ✓ LED FRCE : Elle s'allume en cas de commande de forçage permanente active ;
- ✓ LED RUN : Clignote à la mise en route de la CPU et allumage continue en mode RUN ;
- ✓ LED STOP : Allumage continue en mode STOP, clignotement lent lorsqu'un effacement général est requis et clignotement rapide lors de l'effacement.

➤ **Commutateur du mode de fonctionnement :**

Le commutateur et un interrupteur à trois positions (RUN, STOP et M-RES), il permet de gérer le mode de fonctionnement et l'arrêt de la CPU [14].

- ✓ RUN : la CPU est mise en marche et traite le programme utilisateur ;
- ✓ STOP : la mise en arrêt de la CPU (elle ne traite aucun programme) ;
- ✓ M-RES : le positionnement du commutateur sur M -RES permet le lancement de l'effacement général de la CPU.

➤ **Interface multipoints :**

L'interface multipoints (**MPI**) est celui de communication de la CPU avec un PC ou une PG-MPI, afin de pouvoir la programmer ou pour la communication dans un sous-réseau MPI. La vitesse de transmission par défaut pour la CPU 313 est de 17805 K bauds, cette dernière envoie automatiquement à l'interface MPI ses paramètres de bus réglés ainsi une console de programmation peut avoir les bonnes paramètres et se connecter automatiquement à un sous-réseau MPI. La communication MPI suit un protocole propre à SIEMENS [14].

➤ **Les mémoires :**

Les zones de mémoires vive (**RAM**) sont destinées à recevoir les informations issues des capteurs (image des entrées) et les informations générées par le processeur destinées aux sorties de l'API (images des sorties). Elle sert aussi à l'enregistrement du programme utilisateur.

Des mémoires **RAM** non volatiles sont réservées pour sauvegarder des données en cas de coupure d'alimentation.

Une mémoire morte (**ROM**) est réservée pour contenir le système d'exploitation de la CPU.

➤ **Les modules entrées/sorties TOR :**

Les modules d'entrées/sorties (**SM**) sont des interfaces d'entrées et de sorties pour les signaux de l'API, grâce à eux ce dernier reçoit les signaux des capteurs et envoie des signaux aux prés-actionneurs ou aux actionneurs après le traitement du programme.

La **CPU 313 C** est équipée d'un module d'entrées/sorties TOR intégré qui contient 24 entrées/16 sorties et d'un module d'entrées e/sorties ANAL qui contient 5 entrées /2 sorties. Le nombre des entrées et sorties intégrées dans la CPU est suffisant pour notre projet, pour ce la, on n'a pas ajouté des modules d'extensions.

III.5 Présentation du logiciel STEP7 :

III.5.1 Description du STEP7 :

STEP7 est un progiciel de la firme SIEMENS destiné à la programmation et la configuration des systèmes d'automatisation du SIMATIC S7-300 ou 400. Il fait partie de la gamme SIMATIC. C'est un logiciel qui permet de programmer avec trois langages de programmation, le langage à contact (CONT) qu'on a utilisé dans notre travail, liste des instructions (LIST) et le logigramme (LOG).

Le logiciel STEP7 permet de configurer le matériel utilisé, de réaliser le programme et de le charger dans le système cible. Les paragraphes suivants décrivent les différentes étapes à suivre pour exploiter le STEP7 pour l'automatisation d'un système.

➤ **Lancement et création d'un projet sous STEP7 :**

Le lancement du logiciel STEP7 se fait par un double click sur l'icône du logiciel. Lors du lancement du logiciel, ce dernier nous donne la possibilité de créer un projet à l'aide de la fenêtre d'assistance, cette dernière facilite aux débutants de créer un projet, car au simple click sur le bouton « suivant », on lance un projet qui permet de démarrer directement la programmation. Un click sur l'icône « Nouveau » dans la barre des tâches, permet aussi de créer un projet STEP7.

La figure suivante représente la fenêtre d'assistance de STEP7.

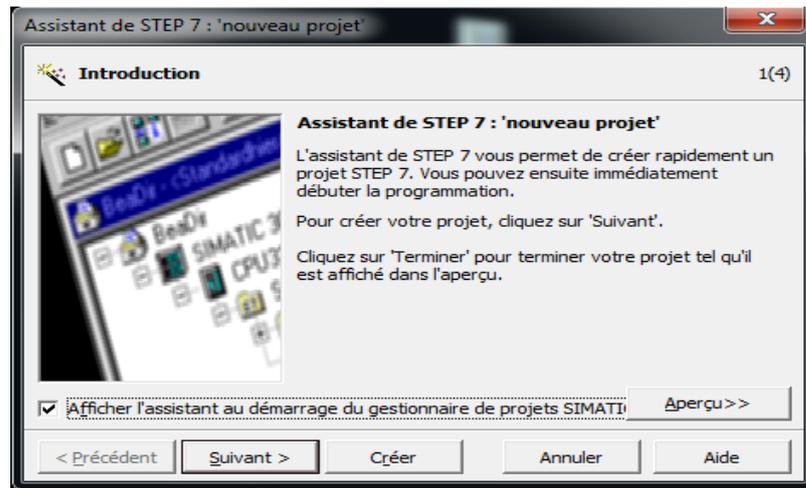


Figure (III.4) : Fenêtre d'assistance pour la création de nouveau projet sous STEP7.

➤ **Insertion d'une station :**

Une fois le projet ouvert, on devrait insérer une station SIMATIC 300, cette action est réalisée de deux manières, le premier est par un click sur l'icône du projet créer avec le bouton droit de la souris on sélectionne « Insérer un nouvel objet », puis on choisit la « station SIMATIC 300 ». La deuxième manière est de cliquer dans la barre d'outils sur « Insertion », puis station et on choisit le SIMATIC 300.

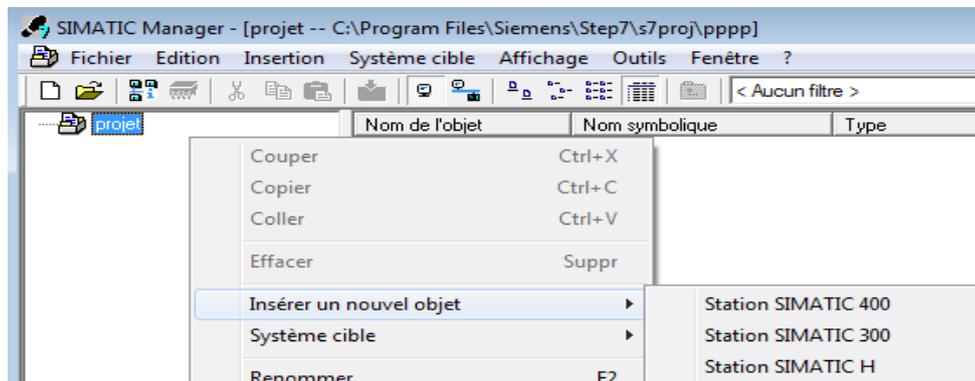


Figure (III.5) : Insertion d'une station sous STEP7.

➤ **Configuration du matériel :**

Un double click sur l'icône de la station nous affiche dans la deuxième partie de la fenêtre une icône nommée Matériel, un click sur cette dernière ouvre la fenêtre **HW config**, dans cette dernière fenêtre en cliquant sur « affichage » on ouvre le catalogue du matériel, ce dernier contient une liste de matériels qu'on peut utiliser (les CPU 300 et 400, les différents modules IM FM et SM, RACK-300). Une sélection du RACK-300 (profile support) ouvre une fenêtre nommée **(0) UR**, cette dernière représente le châssis du RACK du S7 300. Se dernier est représenté sous forme de lignes numéroté de 1 à 11 qui représente les emplacements existants dans un châssis réel du S7 300. On click sur la première ligne réservée à l'alimentation, nous

affiche les différentes alimentations du catalogue, une sélection d'une alimentation l'intègre directement dans le RACK. La deuxième ligne est réservée à la CPU, de la même manière que l'alimentation on accède à la liste des CPU du catalogue et une sélection de la CPU l'intègre directement dans le RACK. Pour notre cas lors de l'intégration de la CPU, les entrées /sorties intégrées dans la CPU sont aussi affichées. Les lignes restantes sont réservées pour l'extension des modules d'entrées ou sorties et pour les modules de communication. La figure III.5 présente un RACK configuré.

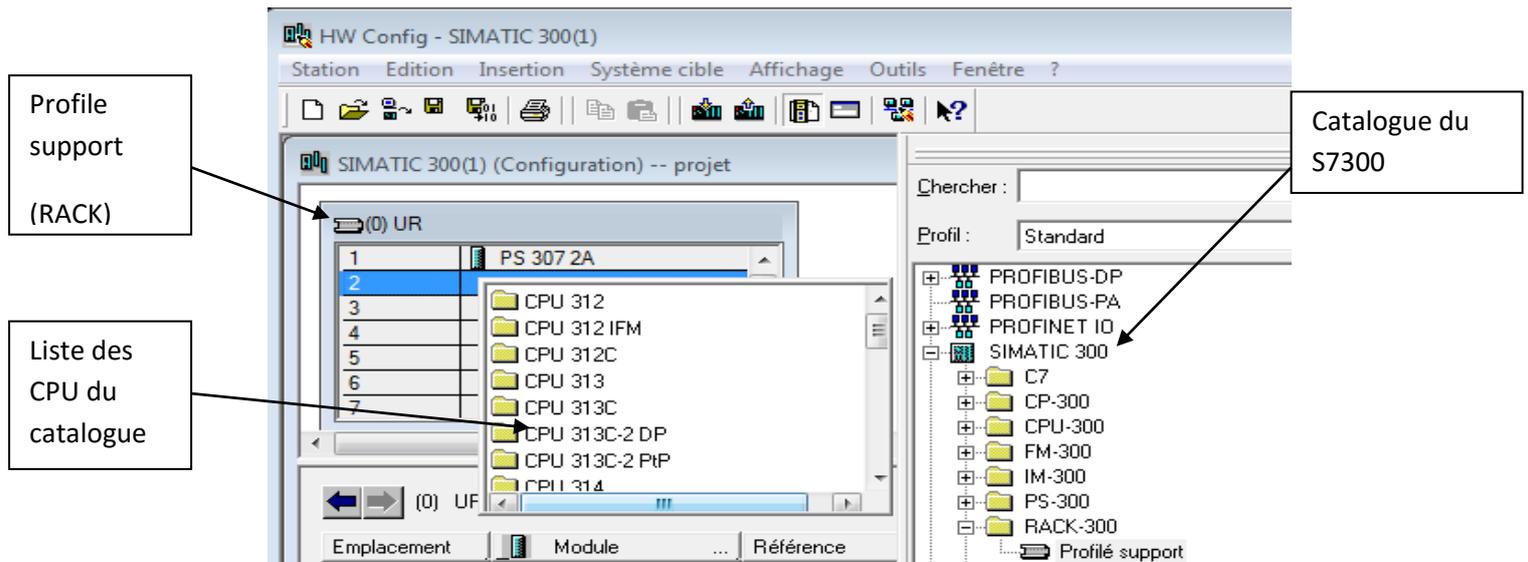


Figure (III.6) : Configuration matériels sous STEP7.

L'action enregistrer est compiler, permet d'enregistrer est de compiler la configuration matériels réalisée.

➤ **Structure d'un projet STEP7 :**

Dans un projet, les données sont mémorisées sous forme d'objets. La présentation arborescente des objets rend compte de la hiérarchie logique du projet. Le niveau 1 de la hiérarchie est l'icône du projet associé à une base de données dans laquelle, elles sont mémorisées toutes les informations relatives au projet, comme les informations relatives à la structure matérielle [14].

La deuxième partie de la fenêtre du projet contient les blocs de programme, le système d'automatisation utilise les différents types de blocs dans lesquelles, on peut mémorisés le programme utilisateur et les données correspondantes, selon les exigences du projet. Les différents types de blocs sont :

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

Bloc fonction (FC, SFC) : Une fonction (FC) assure une fonctionnalité spécifique du programme, elle est souvent réservée pour les fonctions récurrentes et complexe (pour effectuer des calculs). Les fonctions SFC sont paramétrables, intégrées au système d'exploitation de la CPU, dont le numéro de la fonctionnalité est définie de manière fixe.

Bloc de données (DB) : les blocs de données (DB) sont des zones de données du programme utilisateur, dans lesquels les données utilisateur sont gérées de manière structurée.

Bloc fonctionnel (FB, SFB) : Ils sont réservés pour des fonctions récurrentes et plus complexes que celle des blocs de fonction, on les utilise par exemple pour les fonctions de régulation

L'ajout d'un bloc quelconques se fait par un clique du bouton droit de la souris et la sélection d'ajouté objet, puis on choisie le bloc voulu.

La figure suivante présente la structure d'un projet.

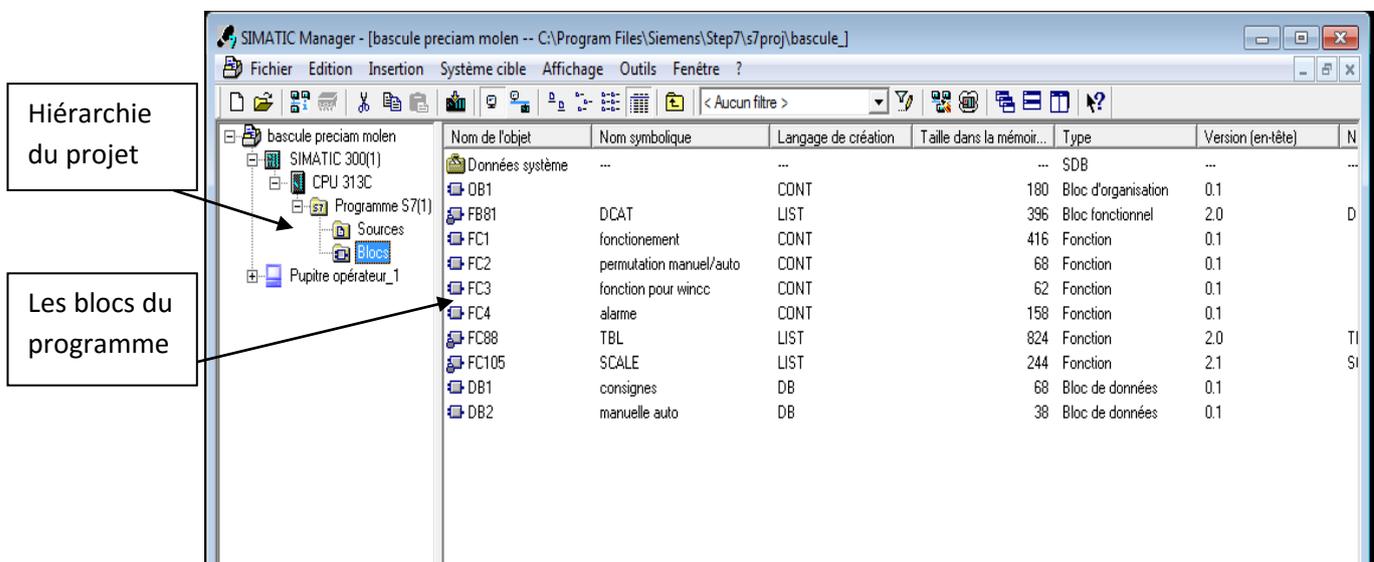


Figure (III.7) : Structure d'un projet sous STEP7.

Un clique par le bouton droit de la souris sur l'icône de la station, nous donne la possibilité d'intégrer un projet de supervision WinCC flexible, on choisit dans la liste des stations affichés, une station SIMATIC HMI.

➤ **Création d'un programme en LADDER :**

Par un clic, sur le bloc OB ou FC, on ouvre une fenêtre pour choisir le langage de programmation voulue. Cette action ouvre une deuxième fenêtre dans laquelle le programme sera rédigé. Dans la barre d'outils de la dernière fenêtre, des icônes qui nous permettent d'ajouter ou de dessiner un réseau comme suit :

 Ajout d'un réseau,
  contacte normalement ouvert,
  contacte normalement fermé,
  bobine normale,
  chargement d'un programme à la CPU.

Sur la partie gauche de la fenêtre, une liste d'éléments de programmation qui sert à réaliser des blocs ou des opérations sur les différentes variables ou entrées du programme, telle que l'addition ou la soustraction, les déplacements et la mise à l'échelle des entrées analogiques et l'appel des blocs dans l'OB.

La figure suivante représente la fenêtre de rédaction du programme :

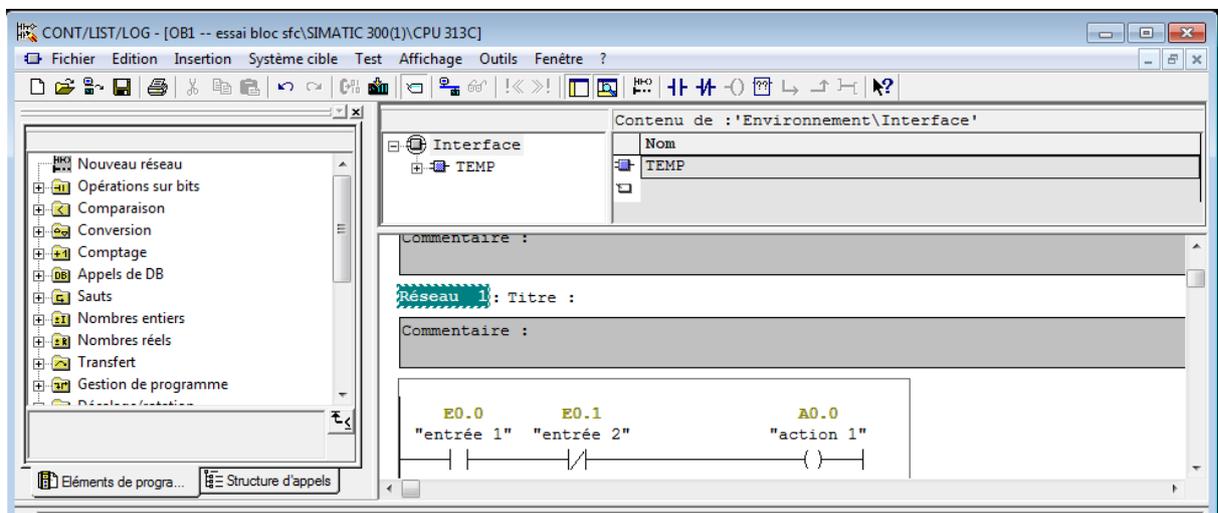


Figure (III.8) : Réalisation d'un réseau sous STEP7.

III.5.2 Description de S7 PLCSIM :

L'application S7-PLCSIM permet de simuler le fonctionnement d'un API S7. La simulation permet de tester les programmes sans la connectée à une CPU physique. Elle permet de visualiser et de modifier les variables du programme.

La mise en route de S7- PLCSIM, s'effectue de deux manière, soit par le cliquer sur l'icône de lancement dans STEP7 manager , soit par la fenêtre démarrer de Windows, puis programme et on choisi S7-PLCSIM.

Après le lancement de PLCSIM et le chargement des blocs du programme, une fenêtre nommée CPU s'affiche, elle représente le commutateur de l'état de la CPU.

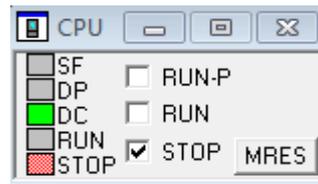


Figure (III.9) : Fenêtre du commutateur de la CPU sous STEP7.

Tout les entrées et sorties qui ont été utilisées dans le programme à tester doivent être insérées avec le menu (Insertion → entrées F2), (Insertion → sorties F3), ou par un cliquer sur les icônes d'entrées, ou de sorties, ou de mémentos ou des temporisations, ou des compteurs sur la barre d'outils de PLCSIM. Les entrées et sorties sont paramétrées selon leurs mode (TOR ou ANAL).

Pour exploiter une entrée analogique dans PLCSIM, on doit d'abord la configurée selon le type d'entrée. Une fois l'entrée est affichée, elle s'affiche par défaut comme une entrée TOR, en introduisant dans la zone d'adresse le symbole de l'entrée EANA qui est le **PEW**, et l'adresse de l'entrée (par défaut l'adresse est la 256), l'entrée est devenue de type EANA. Dans la zone du type d'entrée, une liste est affichée (Binaire, décimal, entier, caractère, entier défiler, décimal défiler), il suffit de choisir le type à utiliser. Dans la zone de valeur, on introduit la valeur maximale et la valeur minimale (dans notre cas en décimal, la valeur min est 0 et la valeur max est 27648)

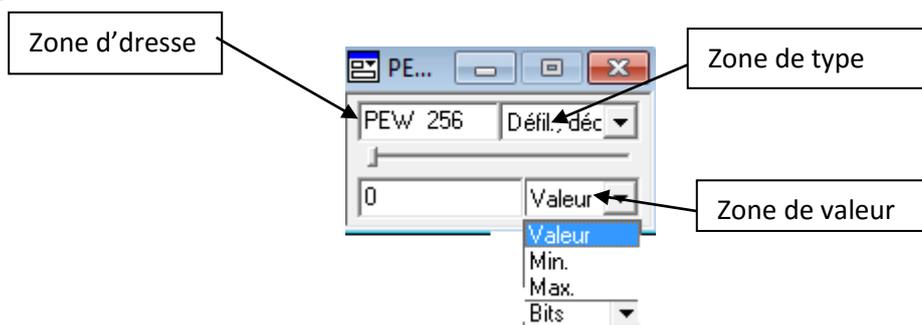


Figure (III.10) : Fenêtre de l'entrée analogique dans PLCSIM.

Les entrées et sorties TOR n'admettent comme configuration que le changement de bit d'adresse. Par contre les mémentos admettent une configuration sur leur type (bit, mot ou double mot). La figure suivante présente la fenêtre de PLCSIM avec des entrées (EB) et sorties (AB) TOR du bit d'adresse 1.x et un memento (MB) du type bits (d'adresse 1.x).

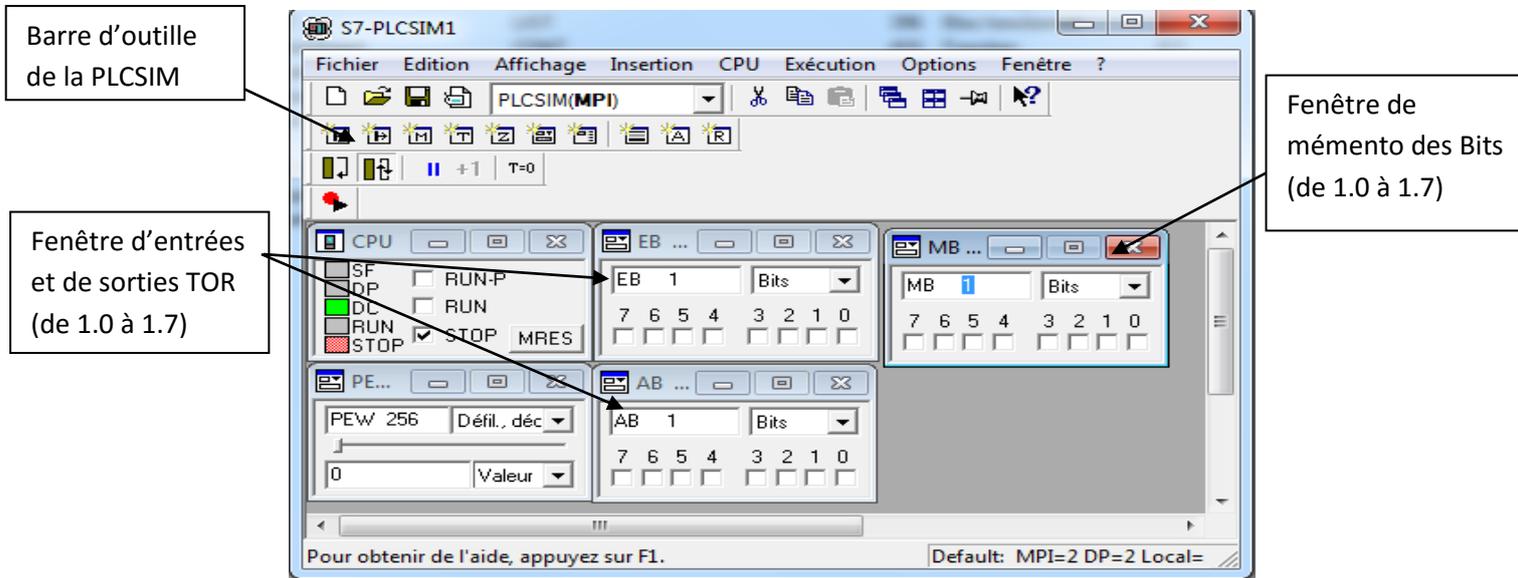


Figure (III.11) : La fenêtre de PLCSIM sous STEP7.

III.6 Présentation du système WinCC flexible :

SIMATIC WinCC est un système d'acquisition et de contrôle des données (SCADA) (anglais : Supervisory Control And Data Acquisition), ainsi qu'une interface homme-machine développée par SIEMENS. Les SCADA sont particulièrement utilisés dans la surveillance des processus industriels et des infrastructures. SIMATIC WinCC peut être utilisé avec SIEMENS PCS7 et Teleperm. WinCC est conçu pour fonctionner sur des systèmes Windows. Il utilise Microsoft SQL Server pour gérer les connexions. Il est également accompagné de VBScript et d'applications d'interface en langage C [15].

III.7 Création ou chargement d'un projet :

Lorsque vous démarrez WinCC flexible, choisissez la commande "Nouveau" pour créer un projet, pour charger un projet existant, choisissez la commande "Ouvrier", ou dans le menu "Projet".

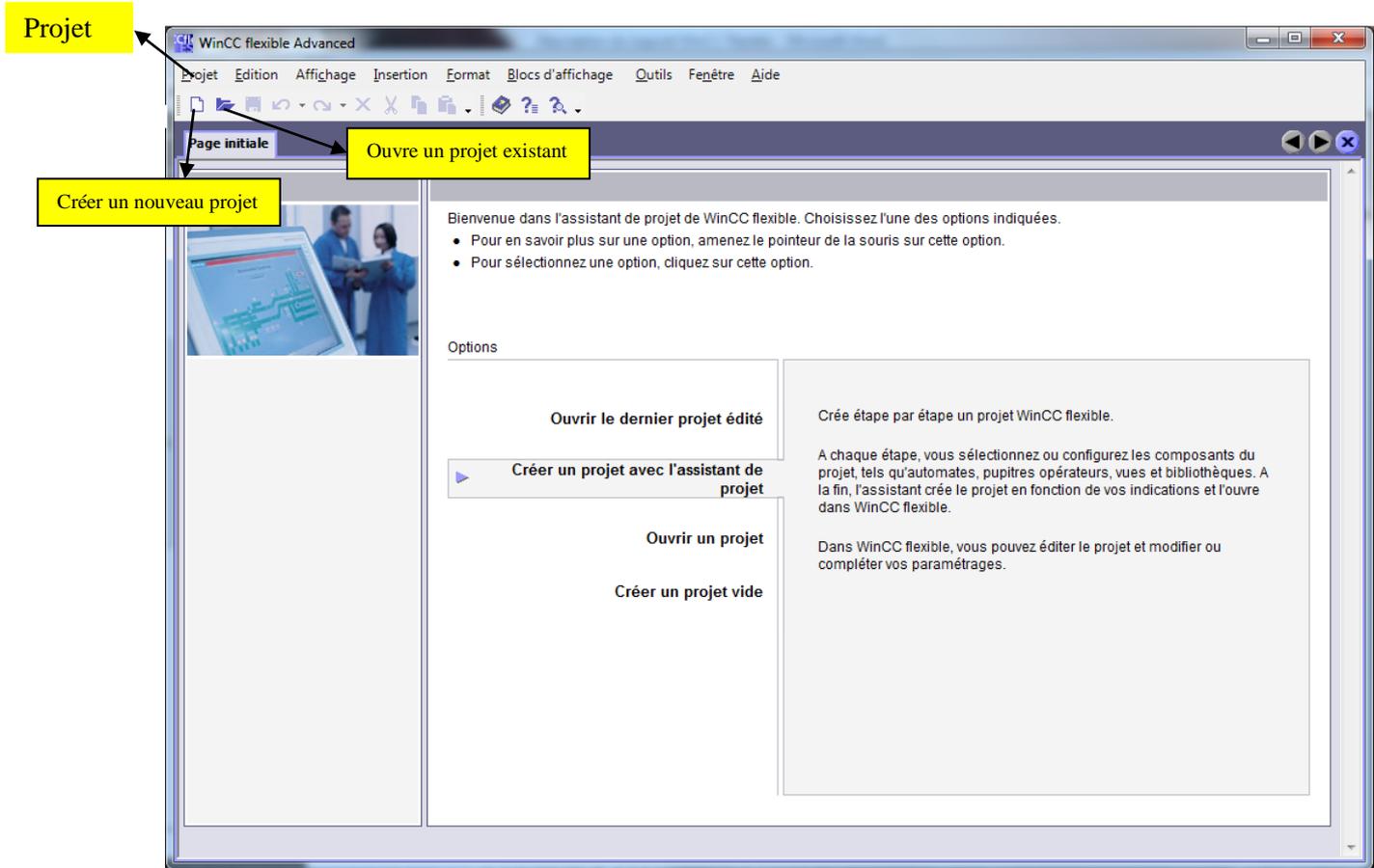


Figure (III.12) : Fenêtre de création ou de chargement d'un projet WinCC flexible.

Un assistant vous guide tout au long des étapes nécessaires pour créer un nouveau projet. L'assistant vous demande par exemple, d'entrer un nom pour le projet et de sélectionner un pupitre opérateur.

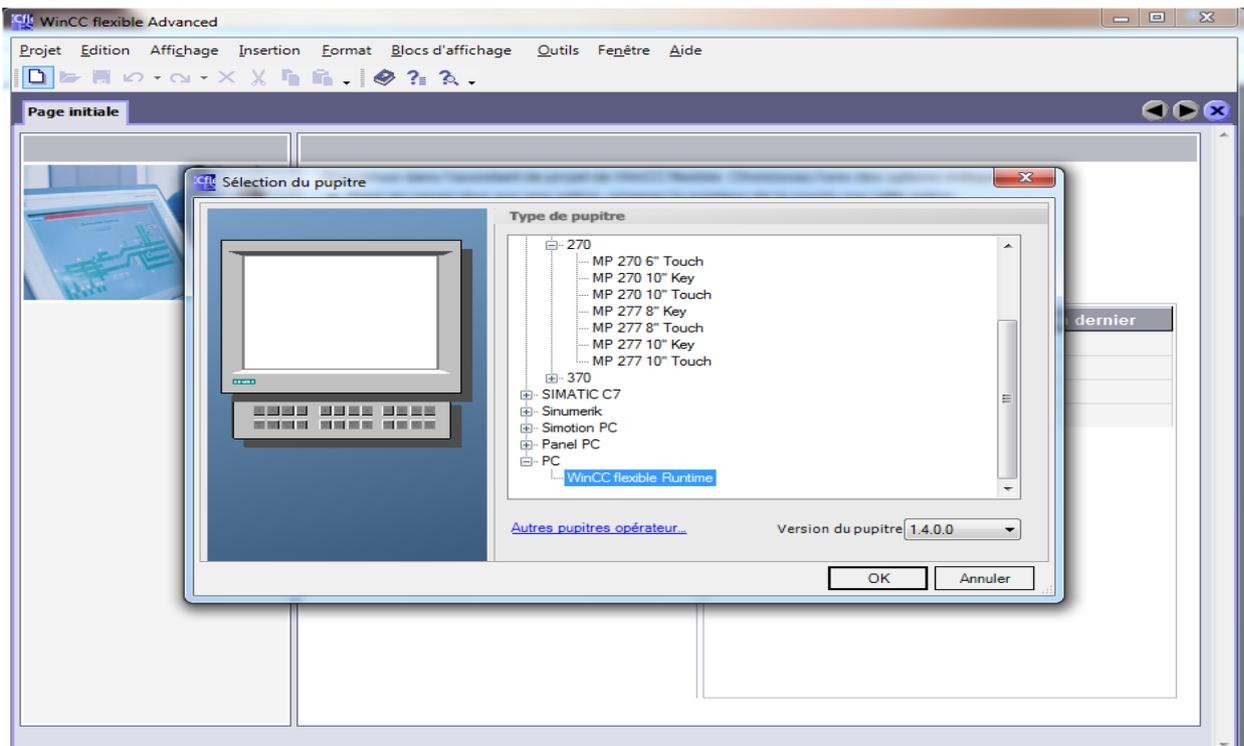


Figure (III.13) : Fenêtre de sélection du pupitre sous WinCC.

III.8 Eléments de l'interface utilisateur de WinCC flexible :

L'environnement de travail de WinCC flexible se compose de plusieurs éléments, certains sont liés à des éditeurs particuliers et uniquement visibles lorsque cet éditeur est activé.

WinCC flexible se compose des éléments suivants :

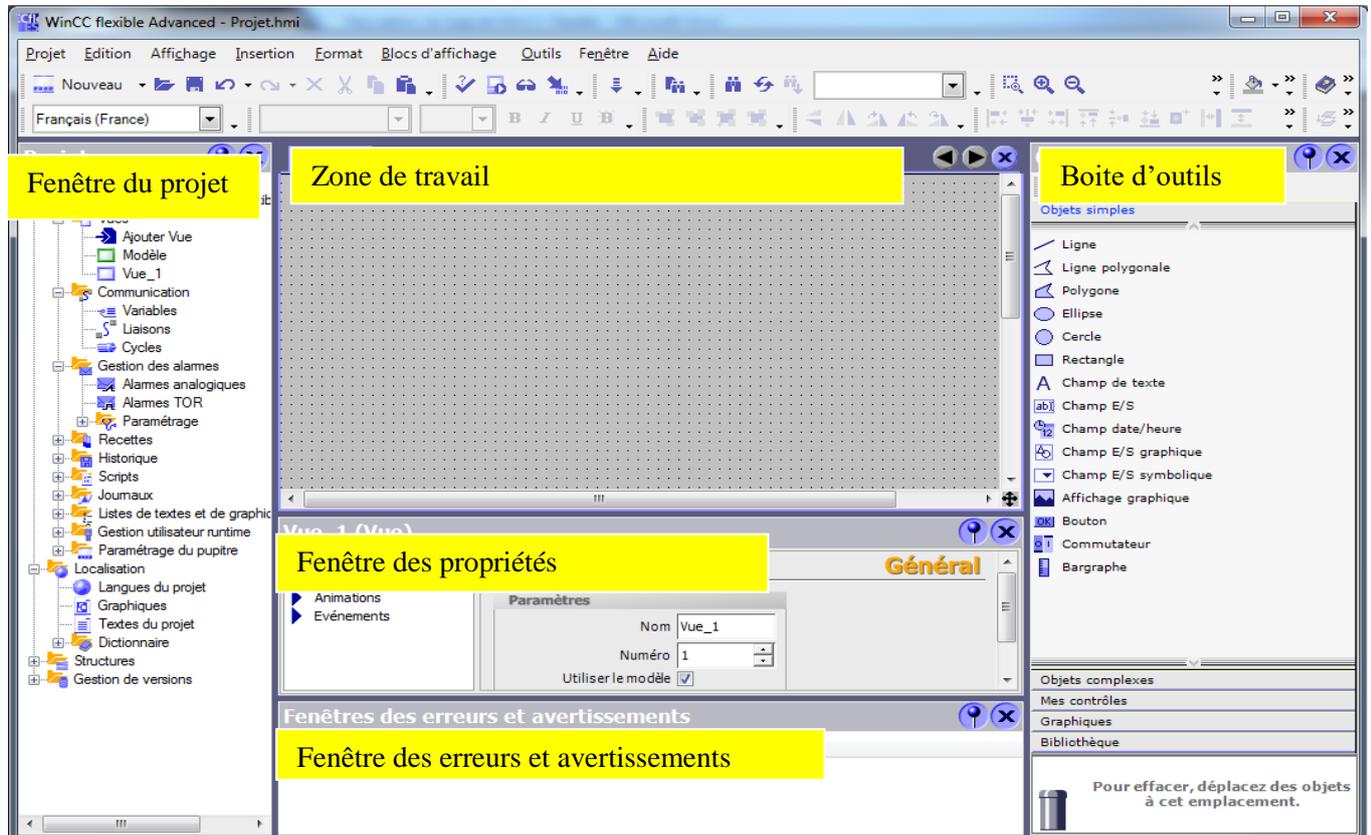


Figure (III.14) : Eléments de WinCC flexible sous WinCC.

III.8.1 Menus et barres d'outils :

Les menus et barres d'outils vous donnent accès à toutes les fonctions disponibles sous WinCC flexible. Lorsque vous positionnez le pointeur de la souris sur une fonction, vous obtenez une info-bulle [16].

III.8.2 Zone de travail :

La zone de travail sert à éditer les objets du projet. Tous les éléments de WinCC flexible sont disposés autour de la zone de travail. A l'exception de la zone de travail, vous pouvez disposer et configurer, déplacer ou masquer par exemple tous les éléments [16].

III.8.3 Fenêtre du projet :

Tous les éléments et tous les éditeurs disponibles d'un projet sont affichés sous forme d'arborescence dans la fenêtre du projet, et peuvent être ouverts à partir de cette fenêtre. Sous chaque éditeur se trouvent les dossiers, dans lesquels un stockage structuré des objets est possible. Pour les vues, les recettes, les scripts, les journaux et les dictionnaires personnalisés,

vous pouvez en outre accéder directement aux objets configurés. Dans la fenêtre de projet, vous pouvez accéder aux paramètres du pupitre, à la localisation et à la gestion de versions [16].

III.8.4 Fenêtre des propriétés :

La fenêtre des propriétés vous permettent d'éditer les propriétés des objets, par exemple la couleur des objets de vue. Elle n'est disponible que dans certains éditeurs [16].

III.8.5 Boîte à outils :

La fenêtre d'outils vous propose un choix d'objets que vous pouvez insérer dans vos vues, par exemple des objets graphiques et éléments de commande. La fenêtre d'outils contient en outre des bibliothèques d'objets et collections de blocs d'affichage prêts à l'emploi [16].

III.8.6 Bibliothèque :

La bibliothèque fait partie de la fenêtre d'outils. La bibliothèque vous donne accès aux objets de vue préconfigurés. Les objets de la bibliothèque permettent d'augmenter la quantité d'objets de vue disponibles et d'améliorer votre productivité lors de la configuration par la réutilisation d'objets préconfigurés. Elle est le lieu central d'enregistrement des objets fréquemment utilisés tels que les objets graphiques et variables [16].

III.8.7 Fenêtre des erreurs et avertissements :

La fenêtre des erreurs et avertissements affiche les alarmes système générées par exemple lors du test d'un projet [16].

III.9 WinCC flexible intégré à STEP 7 :

Dans la fenêtre 3 (intégrer dans les projets STEP7), on choisie le projet STEP7 que nous voulons intégrer.

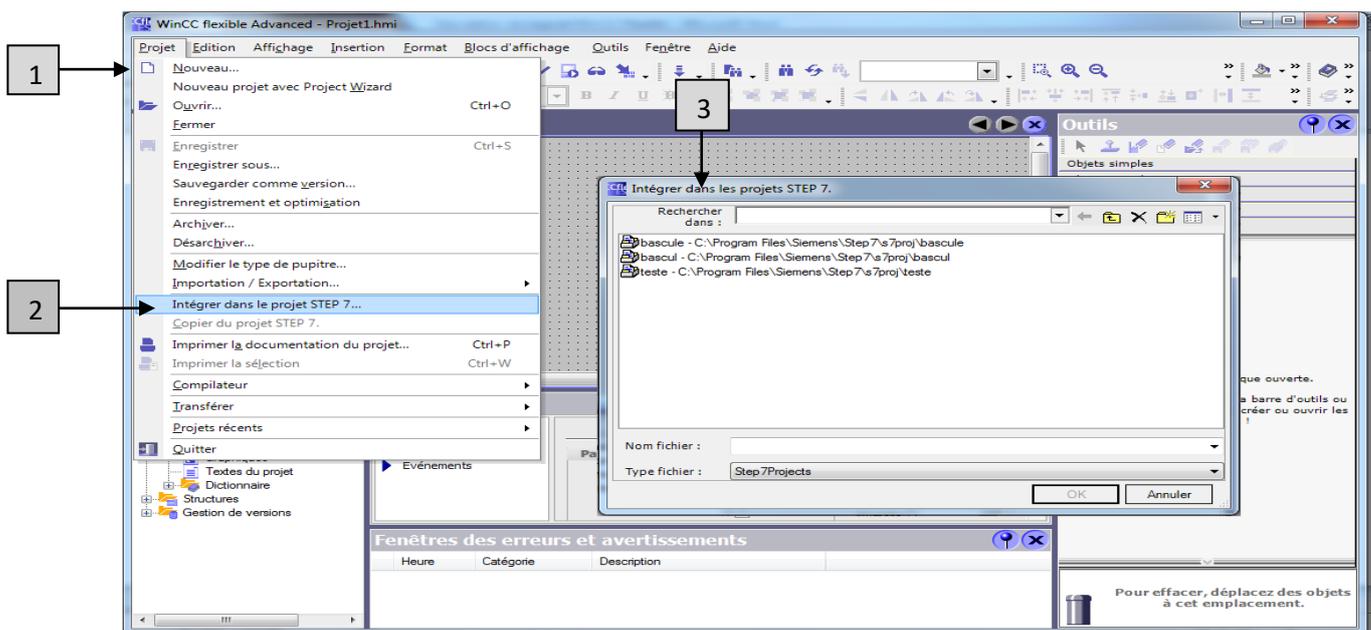


Figure (III.15) : Fenêtre pour l'intégration d'un projet WinCC flexible dans le projet STEP7.

III.10 Création et configuration des variables dans l'éditeur Variables :

Vous ouvrez l'éditeur variable en double-cliquant dans la fenêtre de projet, sur l'entrée "Variables". L'éditeur s'affiche alors dans la zone de travail. Pour configurer les variables il se fait de définir le nom de la variable, sélectionner la liaison à l'automate, le type de donnée et définir l'adresse.

Une variable externe est l'image d'une cellule-mémoire définie de l'automate. L'accès en lecture et en écriture à cette cellule-mémoire est possible aussi bien à partir du pupitre opérateur que de l'automate.

Lorsque vous créez une variable externe dans WinCC flexible, l'adresse saisie doit être identique à celle du programme d'automatisation. De cette manière, le pupitre et l'automate accèdent au même emplacement de mémoire.

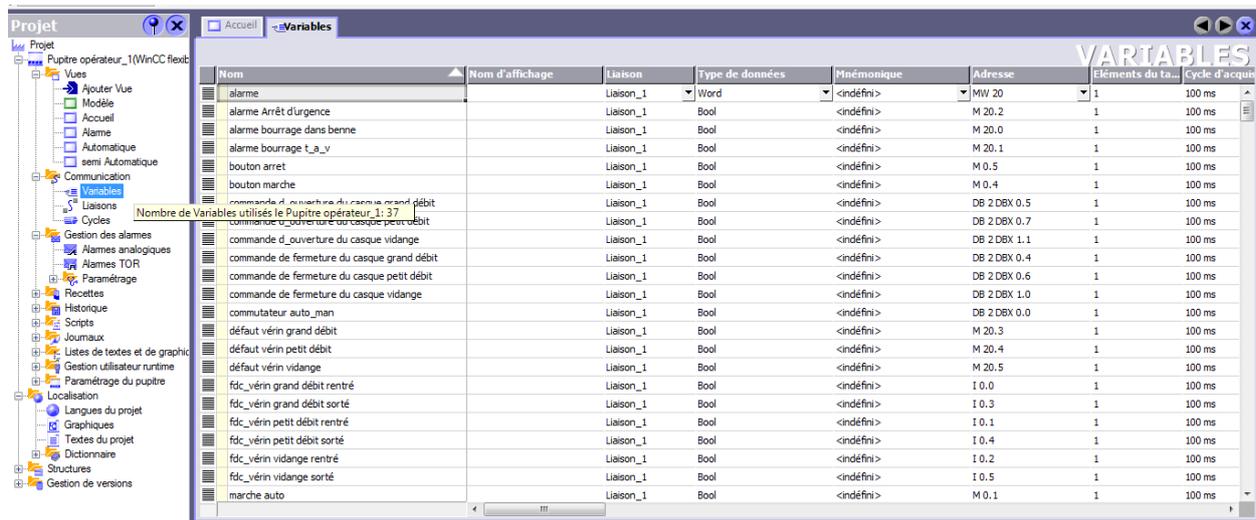


Figure (III.16) : Création et configuration des variables sous WinCC.

III.11 Création des liaisons :

Un double clic sur la fenêtre de projet du pupitre opérateur, exactement sur "Communication > Liaisons", ce dernier permet de définir les paramètres. Dans la colonne "Pilote de communication" de la zone de travail, "Telemecanique Uni-Telway" est sélectionné. A présent, vous pouvez entrer ou modifier les paramètres du protocole dans la fenêtre des propriétés.

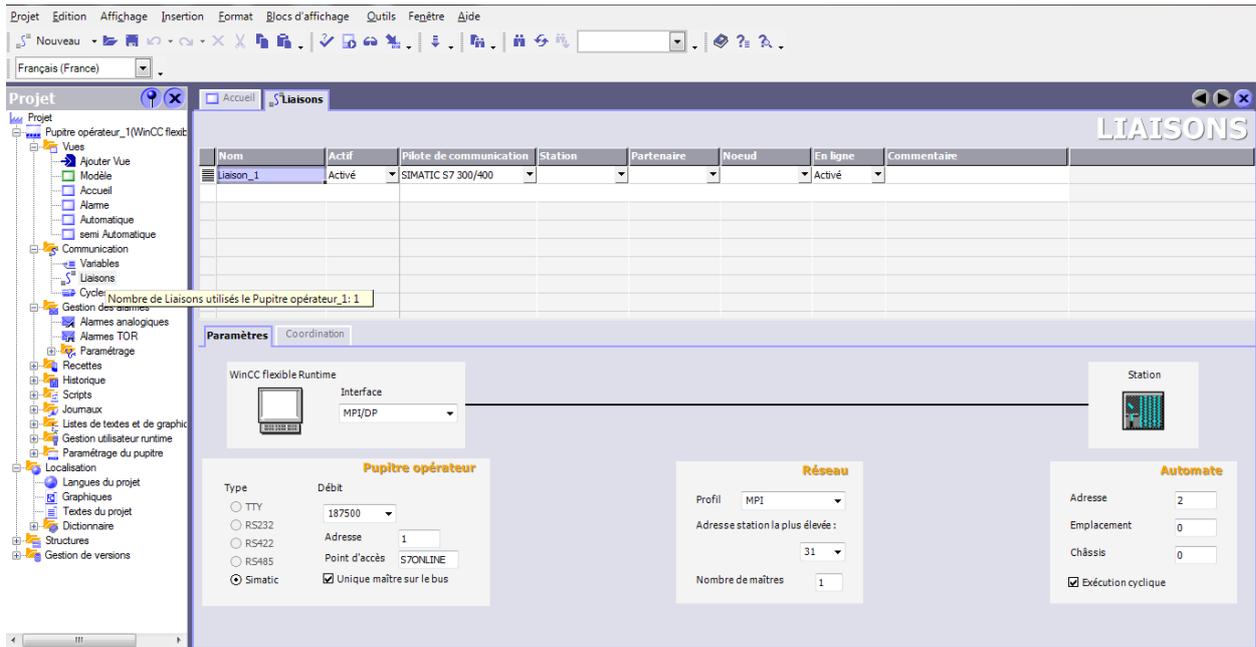


Figure (III.17) : Création des liaisons sous WinCC.

III.12 La fenêtre permanente ProTool au modèle :

Dans WinCC flexible, les objets et les touches de fonction qui sont utilisés dans plusieurs vues peuvent être stockés de manière centrale, dans un modèle. Contrairement à la fenêtre permanente, le modèle n'exige pas sa propre zone de vue exclusive, mais il se trouve pour ainsi dire sous la vue.

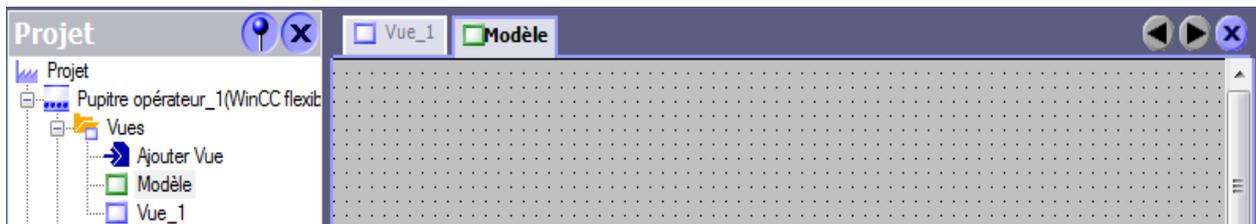


Figure (III.18) : Fenêtre modèle sous WinCC.

III.13 Utilisation d'objets :

Les objets sont des éléments graphiques qui permettent de configurer la présentation des vues de process du projet.

Exemple de configuration d'un bouton poussoir (montré sur la figure III.18).

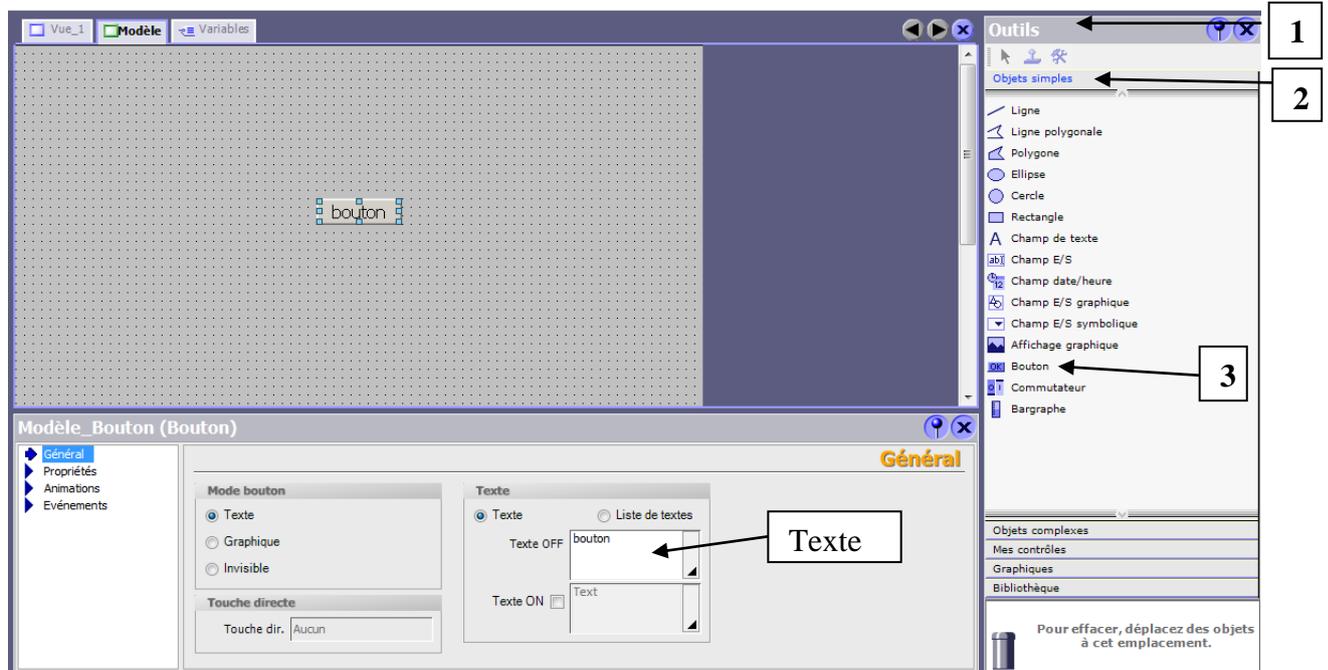


Figure (III.19) : Configuration d'un bouton poussoir sous WinCC.

Etape 1 : Dans la fenêtre outils, on sélectionne objets simples puis bouton, on clique sur la zone de travail pour le ramener.

Etape 2 : Dans la fenêtre des propriétés, on procède comme suit :

1. Général → Texte, on écrit le texte qu'on veut avoir sur le bouton (comme dans la figure (III.18)) ;
2. Dans Propriétés, on configure la représentation pour choisir les couleurs, la mise en page pour la position et la taille, et le texte pour le style.

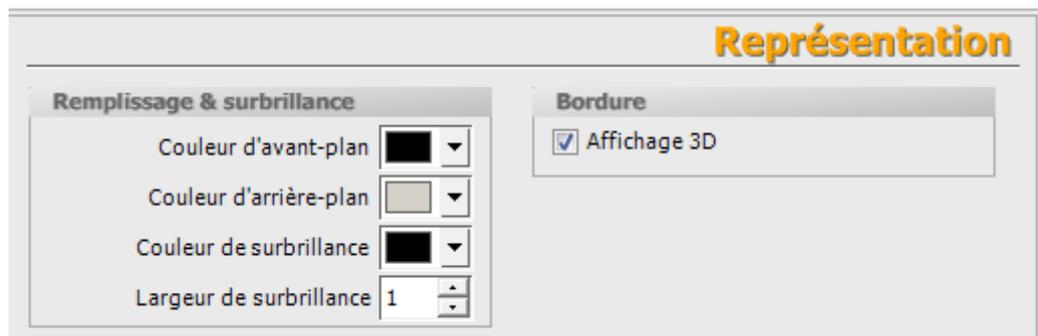


Figure (III.20) : Fenêtre de représentation sous WinCC.

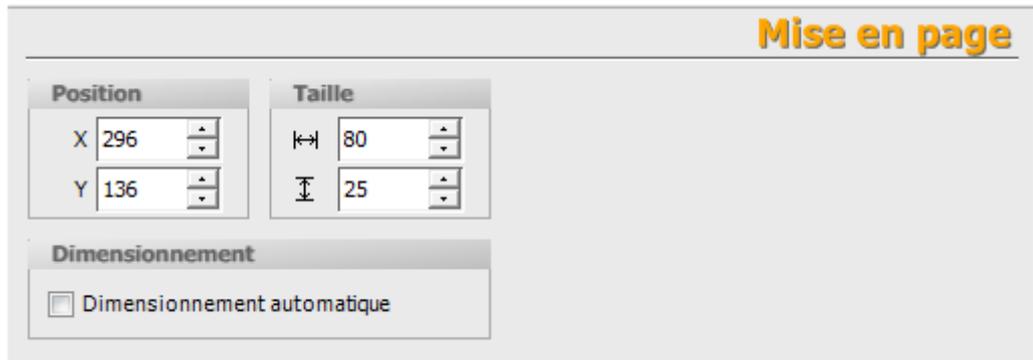


Figure (III.21) : Fenêtre de mise en page sous WinCC.

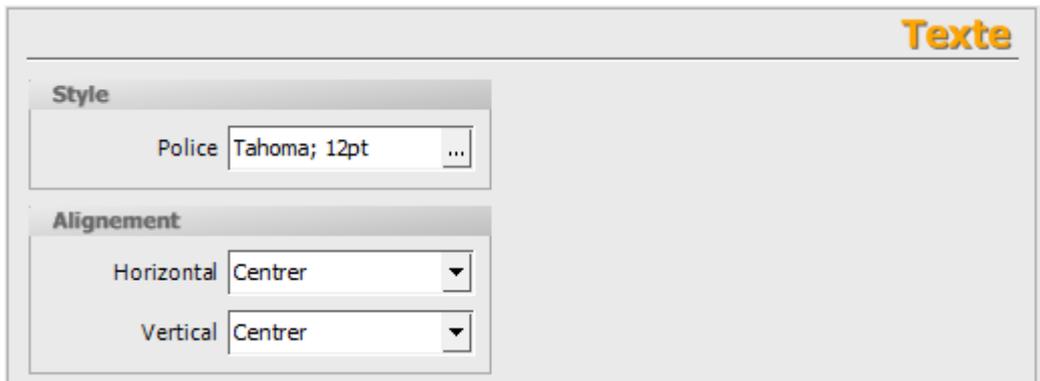


Figure (III.22) : Fenêtre de texte sous WinCC.

3. Dans Evénements, on choisit le mode de fonctionnement, pour un bouton poussoir c'est presser et relâcher ;
 - Presser → Edition de bits → Positionner Bit → ok, Apprêt, on choisie la variable.
 - Relâcher → Edition de bits → RAZbit → ok, Apprêt on choisie la même variable que pour presser.

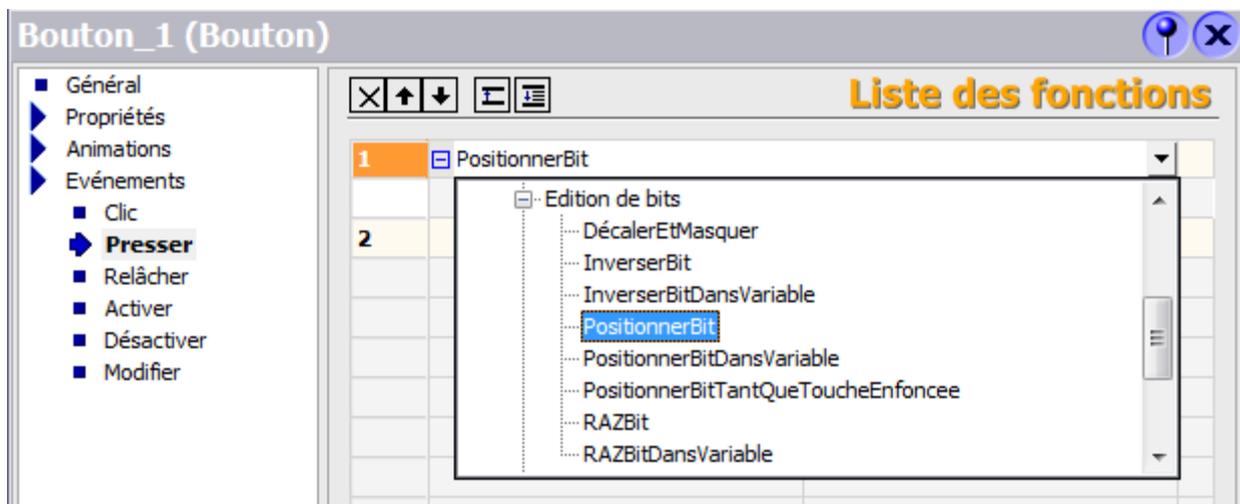


Figure (III.23) : Configuration d'un événement pour un bouton sous WinCC.

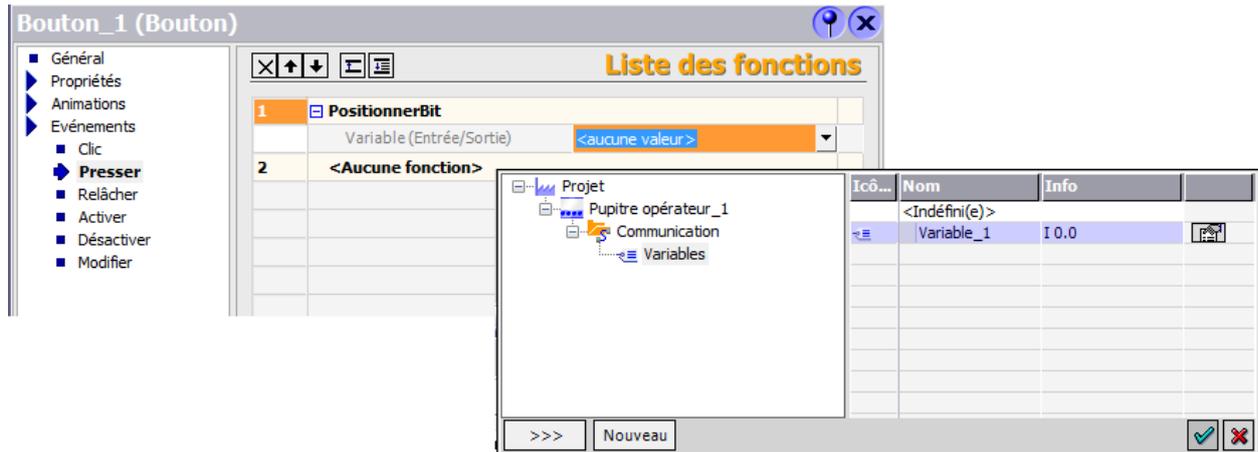


Figure (III.24) : Distribution d'une variable a un bouton sous WinCC.

III.14 Conclusion :

Dans ce chapitre, nous avons présenté un aperçu général sur l'API S7-300, ainsi que l'architecture détaillée de celui qu'on a utilisé. Par la suite, on a mis en avant ses caractéristiques techniques pour une meilleure exploitation pendant sa programmation.

Nous avons aussi présenté les étapes à suivre pour paramétrer et configurer le logiciel Step7 et le WinCC flexible.

CHAPITRE 4

Programme et supervision

IV.1 Introduction :

La programmation est un outil indispensable des API, cars comme tous les systèmes à base de microprocesseur l'API fonction sur la base d'un programme qui lui définit les tâches à exécuter pour réaliser un processus.

Dans notre cas pour pouvoir piloter la bascule peseuse, nous avons réalisé un programme qui sera intégré dans l'automate programmable. Pour ce la, nous utilisons un logiciel de conception de programme pour les systèmes d'automatisation SIMATIC STEP7, que nous avons décrit dans le chapitre précédent. Ce programme est associé à une représentation graphique de la bascule, réalisé avec le logiciel WinCC flexible, pour pouvoir superviser le fonctionnement de la machine. Une validation du programme par simulation avec l'application PLCSIM est réalisée.

Dans ce chapitre nous présentons les différentes fonctions (FC) et les blocs de données que nous avons utilisées dans le programme ainsi que leurs rôles. Nous allons décrire les différentes étapes et configuration réaliser pour les différentes entrées et sorties, les différentes opérations logiques et arithmétiques existant dans notre programme. Nous décrivons aussi la représentation graphique réalisée et la manière de piloter la machine avec cette dernière.

IV.2 Rôle et contenu des blocs utilisés dans le programme :

A. Bloc OB1 :

Le bloc OB1 est réservé pour l'appelle des blocs fonctions et du bloc FC 105 de la mise à l'échelle de l'entrée analogique. On l'a seulement utilisé pour l'appelle des blocs et non pas pour contenir le programme, puisque nous avons plusieurs programme qui réalise des fonctions séparées, ce qui donne un programme structuré.

B. Le bloc FC 105 :

Le bloc FC est intégré au système d'exploitation de la CPU. La fonction FC 105 réalise la mise à l'échelle de l'entrée analogique des quatre capteurs de force montés en série sur la machine.

C. Le bloc FC1 :

Cette fonction est créée pour contenir le programme qui réalise le fonctionnement de la bascule peseuse. Elle contient onze réseaux : Un pour la mise en marche ou arrêt ainsi que la sécurité, un pour réaliser l'addition des pesées, l'autre pour la soustraction du poids résiduel, deux pour déplacer des valeurs de pesée d'une mémoire de donnée à une autre pour ne pas l'écraser par la valeur suivante et les utilisées dans les opérations d'addition et de soustraction. Le reste des réseaux réalisent les actions automatiques ou semi-automatiques du processus de pesé et les actions de mise à l'état initial de la machine.

D. Le bloc FC2 :

Cette fonction est créée pour contenir les réseaux qui font le changement du mode automatique vers le mode semi-automatique.

E. Le bloc FC4 :

Ce bloc contient les réseaux qui activent ou désactivent les alarmes d'erreur, lors du fonctionnement, comme l'alarme d'arrêt d'urgence ou les alarmes de bourrage ou de défaut des vérins.

F. Le bloc FC3 :

Cette fonction est réservée pour contenir les réseaux nécessaires dans la supervision tel que l'allumage des voyants lumineux et l'animation de la représentation graphique de la bascule.

G. Le bloc DB1 :

Le bloc de données DB1 contient les différentes données réelles de pesée tel que les consignes de pesé pour le fonctionnement à grand débit et à petit débit ainsi que la consigne de fin de vidange. On trouve aussi les données de pesé totale et de la pesé instantané de la bascule.

H. Le bloc DB2 :

Il contient les différentes données booliennes du programme tel que les données de commande semi-automatique.

IV.3 Elaboration du programme sous STEP7 :

a) Normalisation de l'entrée analogique :

Les valeurs analogiques ne peuvent être traitées dans la CPU que sous forme numérique. Cette opération de conversion est assurée par le convertisseur analogique numérique (CAN) dans le module d'entrées analogique. La conversion se fait de manière séquentielle dans le cas où nous avons plusieurs entrées.

Le résultat de la conversion est stocké dans la mémoire du résultat jusqu'à ce qu'elle soit écrasée par la prochaine valeur. Cette dernière est lue par l'opération de chargement (L PEW...), par défaut l'adresse de la valeur est L PEW 256.

Le module analogique codifie la plage du courant (4 mA à 20 mA) délivré par les capteurs en une plage de valeurs numériques de 0 et 27648. Ces dernières valeurs sont alors converties en grandeurs physiques (entre 0 et 1300 Kg), cette opération est la normalisation de l'entrée analogique. Cette dernière opération est réalisée par la fonction standard FC 105 qui est fournie avec STEP7, dans la bibliothèque STANDAR LIBRARY. La figure IV.1 représente le réseau de normalisation de l'entrée analogique L PEW 256.

Réseau 1 : Bloc de mise a l'échelle de l'entrée analogique

Commentaire :

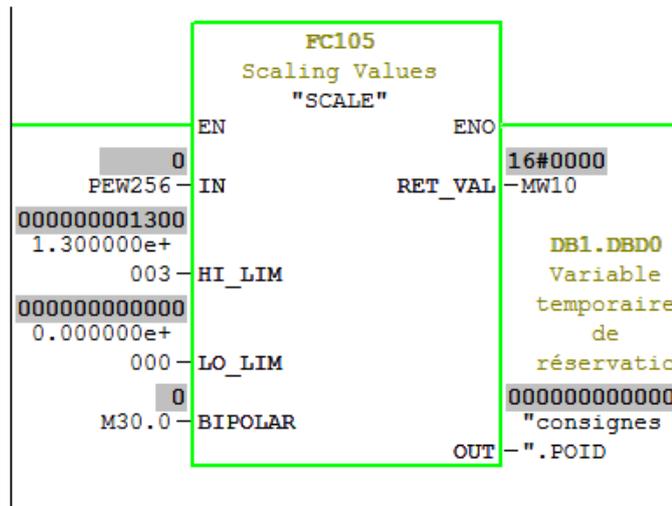


Figure (IV.1) : Réseaux de normalisation de l'entrée analogique sous STEP7.

La configuration de la fonction FC105 est comme suit :

- ✓ **EN** : l'entrée de validation de la fonction (mise en fonction) dans notre cas l'entrée est tout jour vrai.
- ✓ **IN** : C'est l'entrée qui lit la valeur analogique codifiée par le module d'entrée, dans notre cas l'adresse est par défaut PEW 256.
- ✓ **HI LIM** : la valeur limite supérieur de la conversion et elle est donnée en réelle avec une virgule flottante. Dans notre cas c'est la valeur 1300 Kg.
- ✓ **LO LIM** : la valeur limite inferieur de la conversion. Et elle est de même type que la valeur supérieure. Dans notre cas c'est la valeur 0 Kg.
- ✓ **Out** : la valeur normalisée qui est délivré à la sortie OUT sous forme de nombre réel.
- ✓ **RET-VAL** : Cette sortie délivre la valeur 0 en cas de traitement du programme sans erreur.
- ✓ **BIPOLAIR** : l'entrée BIPOLAIR détermine si les valeurs négatives doivent également être converties, dans notre cas, on n'a pas de valeurs négatives.

b) Réseaux de mise en marche / arête et sécurité :

Ce réseau gère la mise en marche est l'arrêt du système, il gère aussi les deux défauts du premier niveau, détecteur de bourrage de la benne peseuse (DBBP) est l'arrêt d'urgence (AUR), puisque l'activation de ces deux défauts remet le système à l'état initial.

Le réseau est constitué d'une bascule set est reset (SR), elle est mise a un par le contacteur normalement ouvert marche et elle est remise à zéro par l'un des deux contacteurs

montés en parallèle (le premier est normalement fermé suivis d'un contacteur NOT et le deuxième est un contacteur normalement ouvert), un contacteur normalement fermé (ARRET) est monté à la sortie de la bascule, suivis d'une bobine normale qui mène à un le memento M_{0.0}.

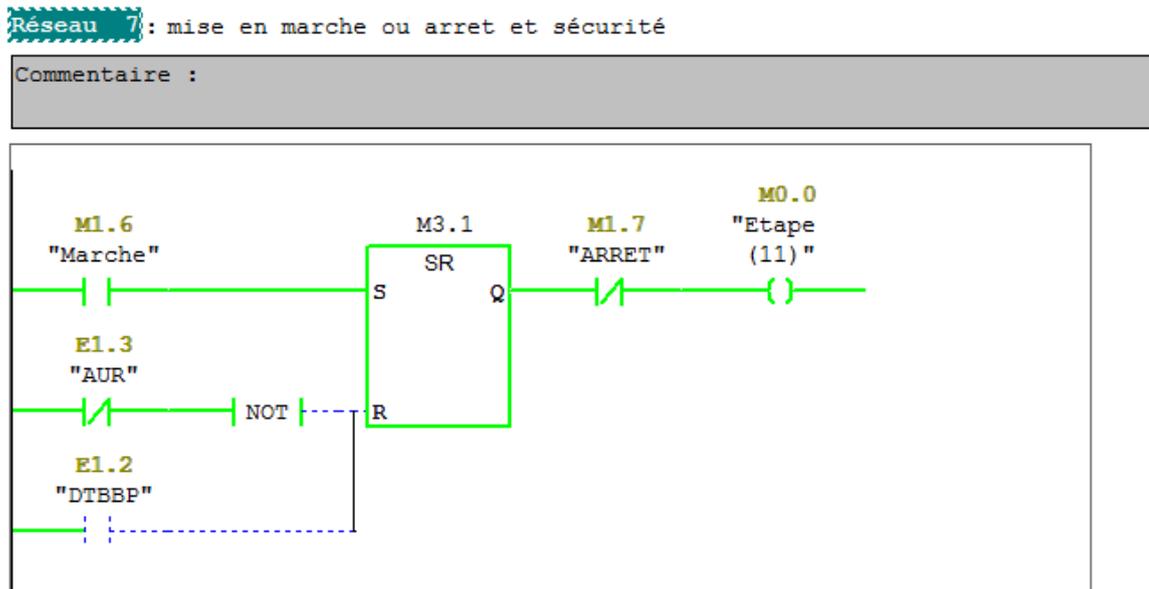


Figure (IV.2) : Réseau de mise en marche ou arrêt et de sécurité sous STEP7.

c) Description de l'opération de comparaison :

Cette opération compare entre deux valeurs réelles (IN1 et IN2) selon le type de comparaison que nous voulons réaliser, si la comparaison est vraie le résultat logique est mis à un. Ce dernier est exploité pour activer une action ou un memento.

d) Réseau d'activation d'une action en mode automatique ou semi-automatique et réinitialisation :

Ce réseau active l'action d'entrée du vérin de grand débit avec les trois manières possible décrite dans le fonctionnement. La première branche du réseau contient des contacteurs normalement ouvert et ceux normalement fermé qui représentent les conditions d'activation de l'action dans le mode automatique. Cette branche active la fonction de comparaison pour comparer entre la valeur de l'entrée et la consigne de grand débit, une fois la comparaison est vérifiée la sortie A_{1.6} est mise à un.

La deuxième branche contient le memento de mise en mode semi automatique (M_{0.4}) avec un contacteur normalement ouvert, qui représente le bouton de commande de l'entrée du vérin dans la partie de supervision. La fermeture des trois contacteurs active la sortie de l'action de fermeture d'adresse A_{1.6}.

La troisième branche est celle de remise a l'état initial, elle s'active lorsque un défaut est détecté, ce qui remis le memento de remise à l'état initiale à un pour activer la sortie A 1.6.

La figure IV.3 donne le réseau numéro trois qui active l'entrée du vérin de grand débit.

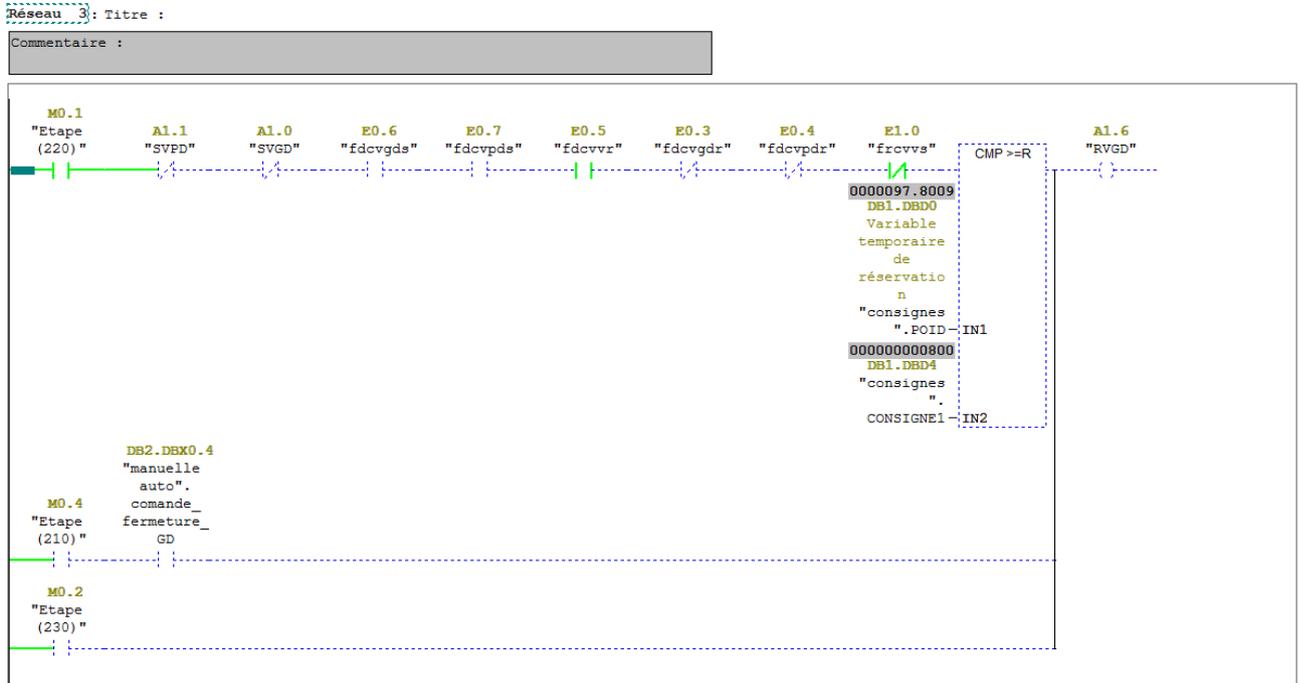


Figure (IV.3) : Réseau de l'action de fermeture du vérin de grand débit sous STEP7.

e) Réseau de déplacement de la valeur de la pesé :

Dans notre programme, on a utilisé deux réseaux pour le déplacement, l'un est pour déplacer la valeur de la pesé avant la vidange du sucre, la seconde est pour déplacer la valeur de la pesé après la vidange (le résidu) à des adresses déférentes de celle du départ pour les exploiter dans les opérations d'adition et de soustraction.

L'opération d'affectation est activée par l'entrée de validation **EN**, qui est dans notre cas toujours validée. La valeur indiquée dans l'entrée **IN** est copiée à l'adresse précise dans la sortie **OUT** qui est montré par la figure (IV.4). Le mot de l'adresse DB1.DBD0 est affecté à l'adresse DB1.DBD20. L'opération MOVE ne permet de copier que des octets, des mots ou des doubles mots. La figure IV.4 présente le réseau d'affectation de la valeur de la pesé avant la vidange.

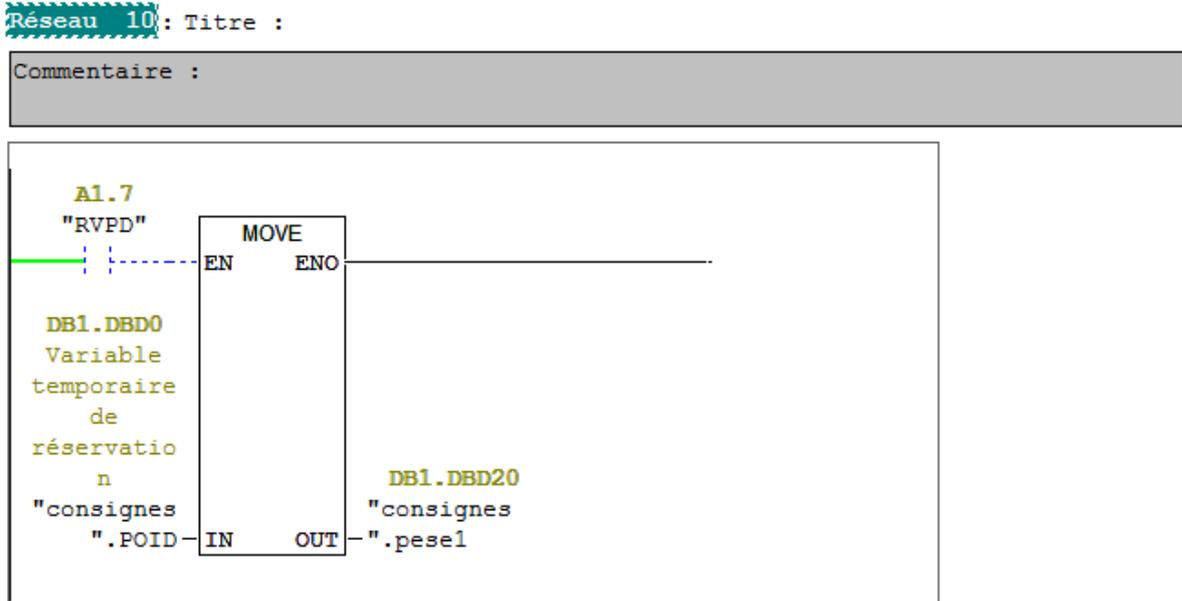


Figure (IV.4) : Réseau d'affectation de la valeur de l'entrée analogique sous STEP7.

f) Réseau de soustraction du résidu de vidange de la pesée d'un cycle :

Ce réseau réalise l'opération de soustraction du poids résiduel après l'action de vidange de celui de la pesée avant la vidange. Cette opération permet d'avoir le poids exacte transmitt a la section de stockage.

La fonction de soustraction est activée si l'état de signal à l'entrée de validation (EN) est à un, dans notre réseau l'opération est activée par le front descendant de la action de rentré du vérin de vidange.

Cette opération soustrait l'entrée IN2 de l'entrée IN1 qui range le résultat dans la sortie OUT. Si le résultat est hors de la plage autorisée pour un nombre réel, les bits DEB et DM du mot d'état sont mis à un et la sortie du bloc ENO est mis à zéro. La figure IV.5 représente ce réseau se soustraction.

Réseau 8 : soustraction du poids réséduelle

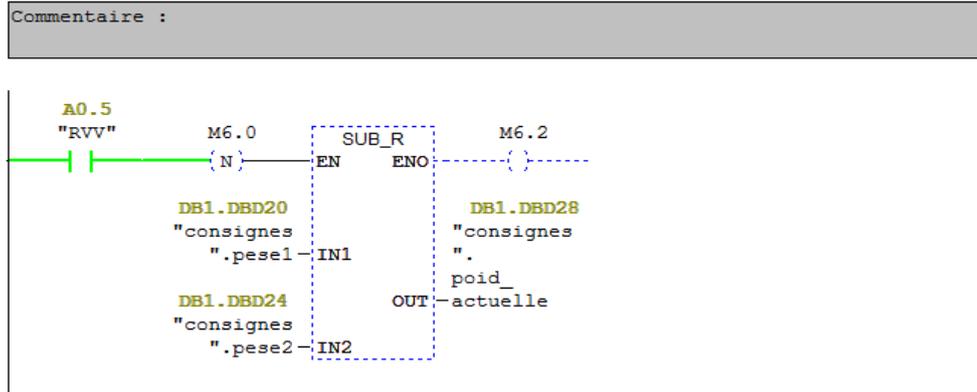


Figure (IV.5) : Réseau de l'opération de soustraction sous STEP7.

g) Réseau d'adition du poids total des pesés :

Ce réseau réalise l'opération d'adition des pesés finaux après avoir soustrait le poids résiduel en sommant le poids de la dernière pesé avec le poids total pesé au par avant.

La fonction d'adition est activée si l'entrée de validation (EN) est à un. Dans notre réseau l'opération est activée par le front descendant du contacteur normalement ouvert du memento M_{6.2} qui est activé par la sortie de la fonction de soustraction. Cette condition d'activation, nous garantie que l'opération d'adition ne sera pas réalisée si la soustraction n'est pas achevée.

Cette opération réalise l'adition entre les deux entrées IN1 et IN2 et range le résultat dans la sortie OUT. La figure IV.6 présente le réseau qui commande l'adition.

Réseau 9 : adition des poids pesée

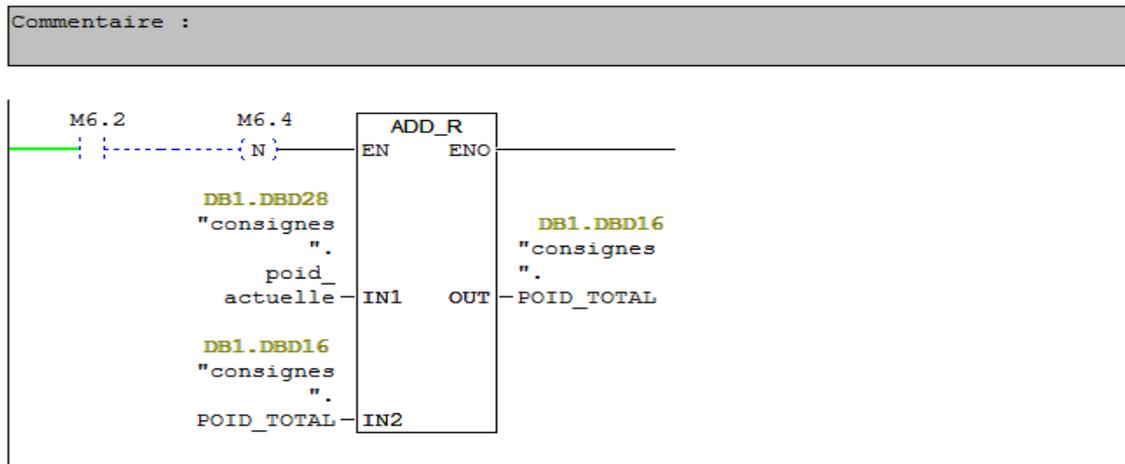


Figure (IV.6) : Réseau de l'opération d'adition sous STEP7.

IV.3 Vue d'Accueil :

La vue d'accueil contient le logo de croupe CEVITAL et celui de l'université de Béjaïa et une photo de la machine PRECIA MOLAN, elle a aussi deux boutons qui permettent à l'opérateur de choisir l'accès vers les deux vues principales pour les deux modes de fonctionnement (Automatique / semi Automatique).



Figure (IV.7) : Vue d'accueil sous WinCC.

IV.4 Vue du mode de fonctionnement Automatique :

Cette vue contient la représentation graphique de la machine, des indicateurs, un bouton marche et arrêt. Cette vue permet à l'opérateur de :

- Mise en marche et l'arrêt de la machine avec les boutons (MAR et AR) ;
- Surfé entre les vues ;
- Visualisation de l'ensemble des équipements de la machine.
- Visualisation du déroulement du processus en temps réel, ainsi que les mesures des capteurs de force, configurées avec un champ d'entre/sorte (E/S).
- Visualisation de l'état des composants de la machine.

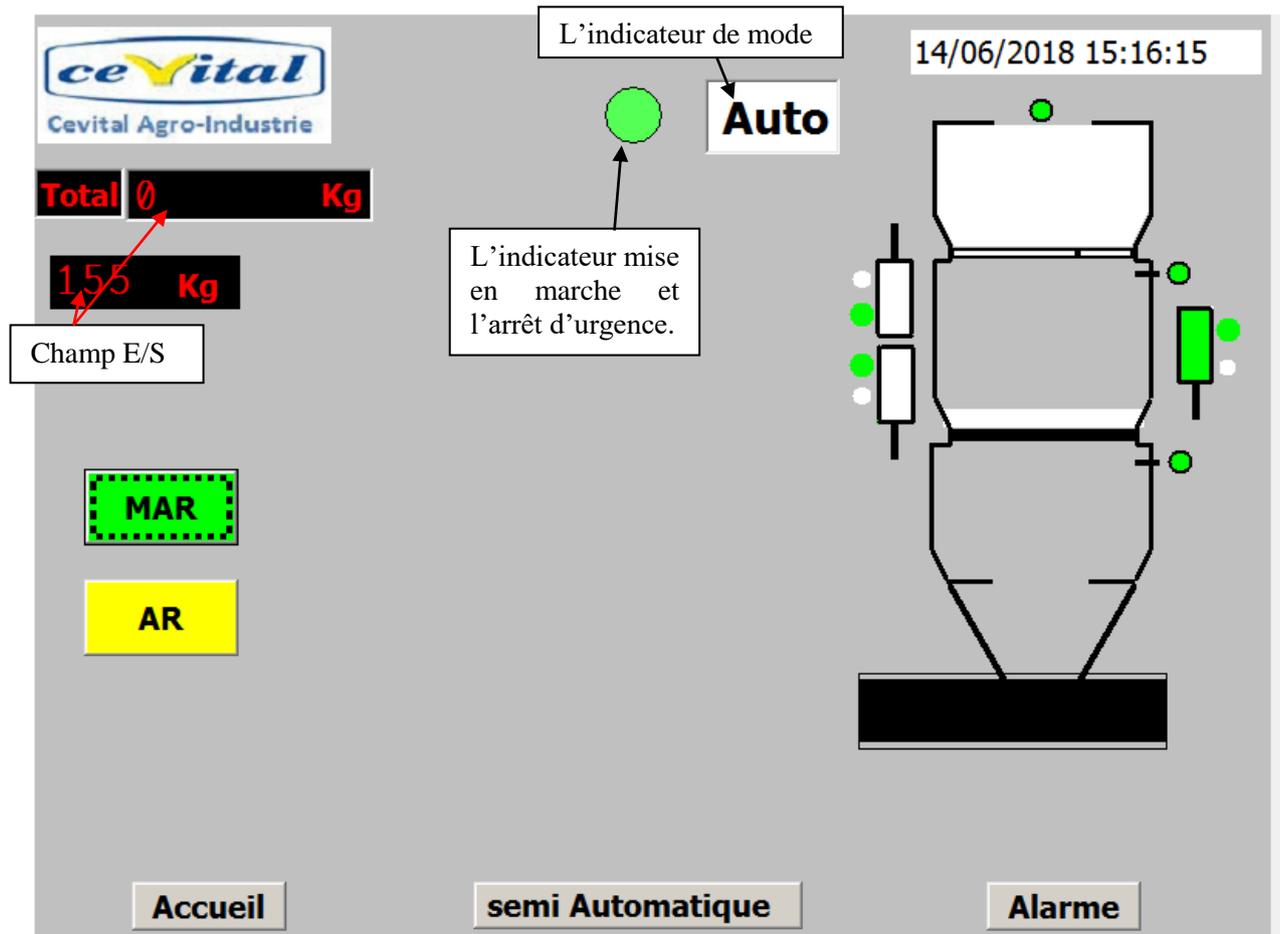


Figure (IV.8) : Vue du fonctionnement automatique sous WinCC.

IV.5 Vue du mode de fonctionnement semi Automatique :

Cette vue contient le même contenu que celle de la figure IV.7 mais elle a en plus des boutons de commande. Cette vue permet à l'opérateur de :

- Mise en marche et l'arrêt de la machine avec les boutons (MAR et AR).
- Surfé entre les vues ;
- Visualisation de l'ensemble des équipements de la machine ;
- Visualisation du déroulement du processus en temps réel, ainsi que les mesures des capteurs de force, configurées avec un champ E/S ;
- Visualisation de l'état des composants de la machine ;
- Commandé l'ouverture et la fermeture des casques.

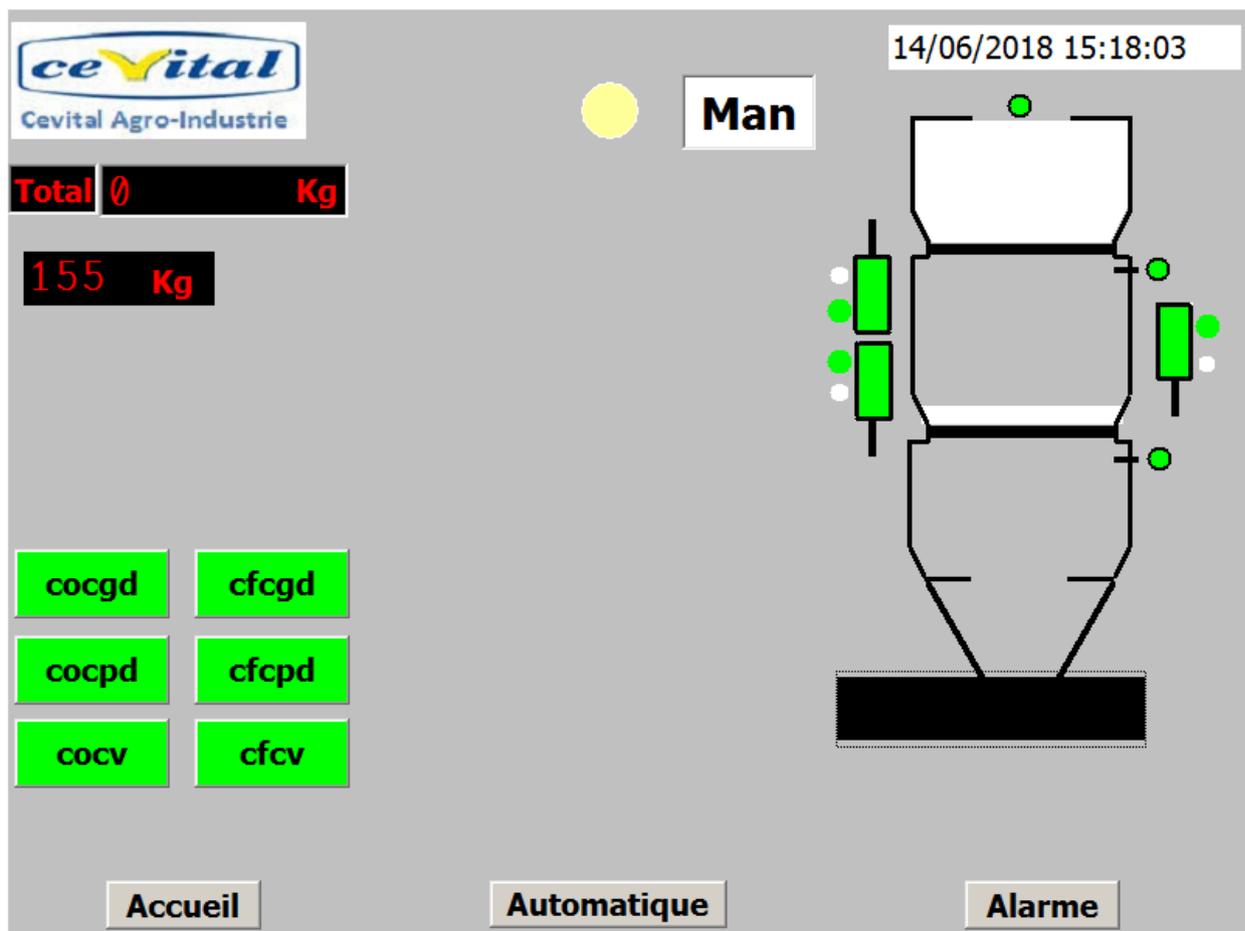


Figure (IV.9) : Vue du fonctionnement semi Automatique sous WinCC.

IV.6 Vue des alarmes :

Les alarmes se déclenchent lorsque certains bits sont mis à un dans la mémoire de l'API.

Pour cela, nous avons configuré des alarmes Toute Ou Rien (TOR) dans WinCC flexible.

Cette vue permet à l'opérateur de consulter et voir tous les défauts.

| N° | Heure | Date | Etat | Texte | GR |
|-----|----------|------------|------|---------------------------------------|----|
| ! 3 | 19:34:18 | 10/06/2018 | A | défaut vérin vidange | 0 |
| ! 2 | 19:34:17 | 10/06/2018 | A | défaut vérin petit débit | 0 |
| ! 1 | 19:34:16 | 10/06/2018 | A | défaut vérin grand débit | 0 |
| ! 5 | 19:34:01 | 10/06/2018 | A | bouffage dans la trémie aval (DBTAV) | 0 |
| ! 4 | 19:33:58 | 10/06/2018 | A | bouffage dans la benne peseuse (DBBP) | 0 |
| ! 6 | 19:33:54 | 10/06/2018 | A | arrêt d'urgence (AUR) | 0 |

Figure (IV.10) : Vue de l'état des alarmes sous WinCC.

IV.7 Configuration du champ E/S

La configuration du champ se fait selon les séquences suivantes :

Dans fenêtre outils → objets simples → champ E/S → on click sur la zone de travail, en suite, on ouvre la fenêtre des propriétés ou on effectue les configurations suivantes :

- Général : On configure le type de champ d'E/S, la variable de processus et le format.
- Propriétés : On configure la couleur du texte, l'arrière plan, la position, la taille le style de la police.....

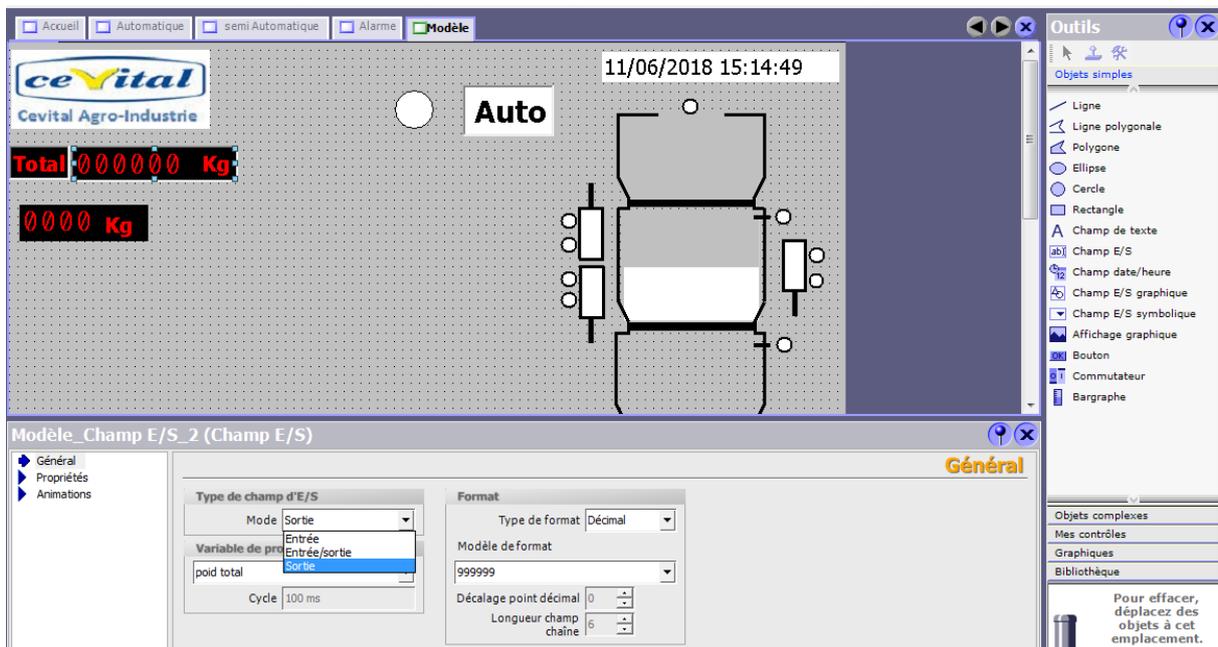


Figure (IV.11) : Configuration du champ E/S sous WinCC.

IV.8 Configuration d'un indicateur

Le fonctionnement de l'indicateur de mode est comme suit :

Si le processus fonctionne sans défaut l'indicateur indique un champ de texte « Auto », si un défaut apparaît ou une alarme est déclenchée, le processus cesse de fonctionner en mode Auto et il se bascule en mode semi Auto ou l'indicateur affiche un champ de texte « Man », pour permettre à l'opérateur d'accéder aux commandes. La configuration de cette partie est comme suit :

- On crée deux champs de texte et on les superpose ;
- On écrit les textes « Auto » et « Man » dans les champs de texte ;

La figure IV.11 présente une vue générale de WinCC.

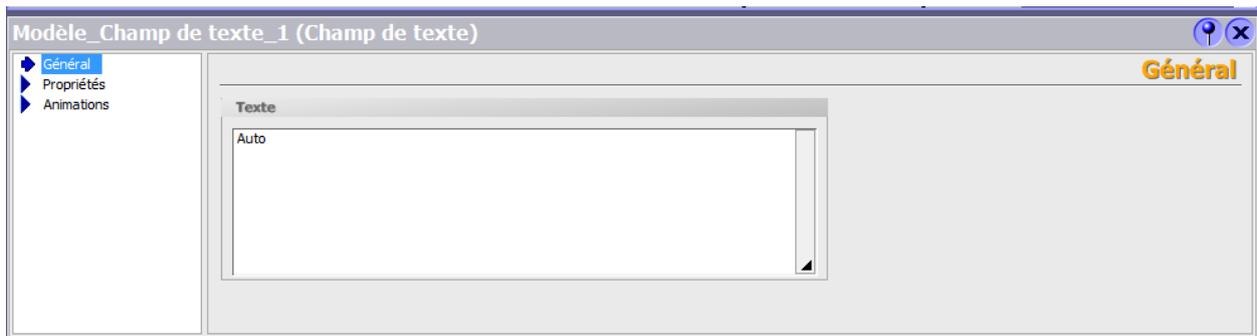


Figure (IV.12) : Fenêtre générale sous WinCC.

- Pour les animations, on choisit la fonction visibilité et on configure la variable de l'état de l'objet et le type. La figure IV.12 représente la fenêtre des propriétés afin de créer les animations.

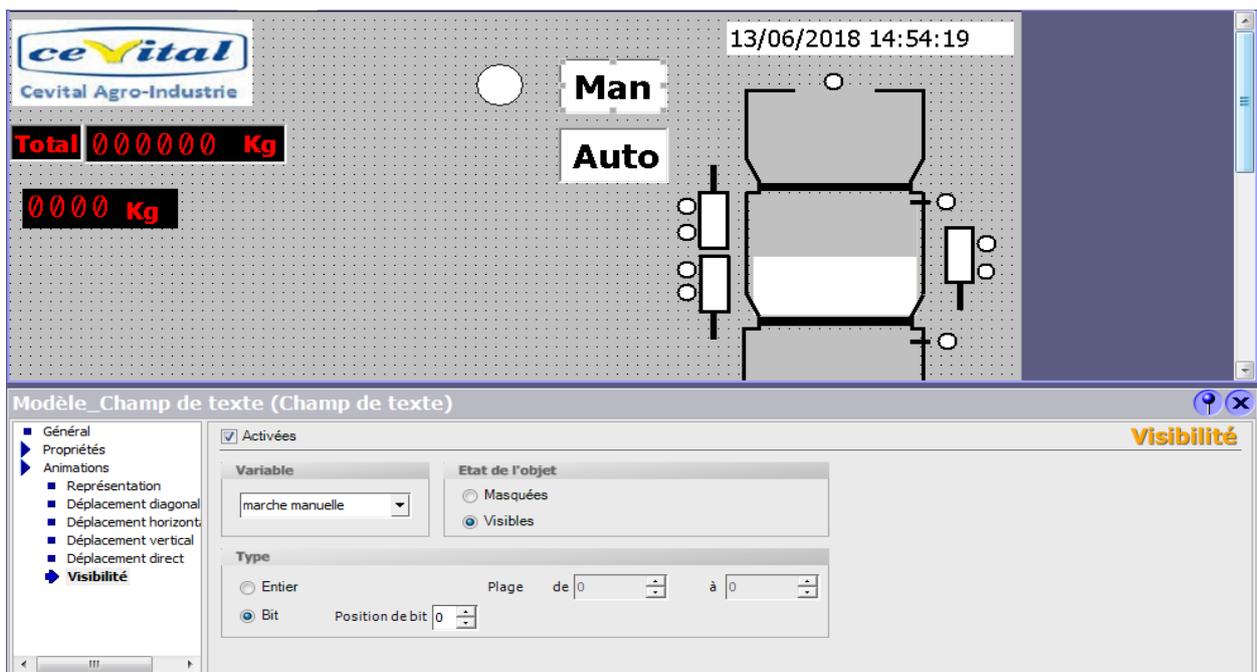


Figure (IV.13) : Fenêtre animation sous WinCC.

IV.9 Conclusion :

Dans ce chapitre, nous avons présent les différentes étapes de notre programme et de l'interface Homme Machine (IHM) pour le contrôle et la commande de la bascule peseuse, et on a donné aussi le rôle et le fonctionnement de chaque fonction (FC) et bloc de données (BD) utilisés lors de la programmation. Nous avons aussi présenté les étapes pour créer des vues afin de commander la machine selon le mode choisi (Auto/Manu).

Conclusion générale

Conclusion générale :

Notre travail concerne l'automatisation et supervision d'une bascule PRECIA MOLAN à CEVITAL, et d'apporter quelques améliorations, en utilisant l'automate programmable S7-300, les logiciels STEP7 et WinCC Flexible.

Pour atteindre l'objectif de notre projet, nous avons commencé par prendre connaissance de notre machine ainsi que l'identification des différents éléments du système, et on a abordé la problématique suite aux différents soucis rencontrés par les opérateurs, pour ce la un cahier des charges a été élaboré.

Afin d'automatiser la bascule, l'étude et l'élaboration de son analyse fonctionnelle ainsi que sa modélisation par des GRAFCETS ont été effectuées.

Nous avons un programme qui commande le fonctionnement de la bascule à l'aide du logiciel STEP7, et nous avons conçu un pupitre de supervision, en utilisant le logiciel WinCC Flexible, qui est un moyen utile pour gérer le fonctionnement du système et diagnostiquer les pannes qui peuvent surgir.

Références Bibliographique

Références Bibliographique

- [1] www.Google Maps.com. Consulté le 01 avril 2018.
- [2] www.preciamolen.com. Consulté le 20 avril 2018.
- [3] Bases de la technique d'automatisation Livre technique. Auteurs : F. Ebel, S. Idler, G. Prede, D. Scholz. Situation : 01/2008
- [4] <https://topcompresseur.fr/informations-compresseur/considerations-techniques/>. Consulté le 12 mai 2018.
- [5] <http://www.air-techniques.fr/distributeur-a-commande-electrique-rappel-differentiel-bm-01-310-2.html>. Consulté le 12 mai 2018.
- [6] Memotech maintenance industrielle. Auteurs : F.Castellazzi, Y.Gangloff, D.Cogniel. Editions : CASTEILLA-25, rue Monge-75005 PARIS
- [7] www.guidedesautomatismes.com. Auteurs : Thierry Schanen. Editions : Lycée Etienne MIMARD Saint Etienne 2001/2007
- [8] <http://www.iteca.fr/index.php/fr/instru/detect-niveau/detect-lame-vibrante>. Consulté le 12 mai 2018.
- [9] <https://www.fr.endress.com/fr/instrumentation-terrain-sur-mesure/mesure-detection-niveau/vibronique-detecteur-niveau-FTM50>. Consulté le 12 mai 2018.
- [10] Www.cours-electromecanique.com
- [11] Introduction aux automatismes industriels. Auteurs : Y. Lecourtier et B. Saint-Jean Edition : Masson, Paris, 1985.
- [12] Automates programmables industriels. Auteurs : William Bolton. Traduction d'Hervé Soulard. Edition Dunod, Paris, 2010, 2015.
- [13] <https://groupe-mb.scene7.com/is/image/groupemb/img>. Consulté le 13 juin 2018.
- [14] Formation automatisation et SIMATIC S7, SIEMENS AG2010. Edition A5.4 pour STEP7 version 5.4.
- [15] <https://fr.wikipedia.org/wiki/WinCC>. Consulté le 15 juin 2018.
- [16] SIMATIC HMI WinCC flexible 2008 Compact / Standard / Advanced Manuel d'utilisation, pack de documentation référencé 6AV6691-1AB01-3AC0.

Annexe

Propriétés de la table des mnémoriques

Nom : Mnémoriques
 Auteur :
 Commentaire :
 Date de création : 07/05/2018 11:25:51
 Dernière modification : 23/06/2018 12:20:12
 Dernier filtre sélectionné : Tous les mnémoriques
 Nombre de mnémoriques : 54/54
 Dernier tri : Opérande ordre croissant

| Etat | Mnémorique | Opérande | Type de données | Commentaire |
|------|-------------------------|----------|-----------------|---|
| | RVV | A 0.5 | BOOL | vérin de vidange rentre |
| | SVGD | A 1.0 | BOOL | vérin de grand débit sotr |
| | SVPD | A 1.1 | BOOL | vérin petit débit sort |
| | SVV | A 1.3 | BOOL | vérin védange sort |
| | RVGD | A 1.6 | BOOL | vérin grand débit rentre |
| | RVPD | A 1.7 | BOOL | vérin petit débit rentre |
| | consignes | DB 1 | DB 1 | bloc de données de consigne de pesé |
| | manuelle auto | DB 2 | DB 2 | bloc de de données depermutation et de commande semi auto |
| | fdcvgdr | E 0.3 | BOOL | fin de course rentrant du vérin grand débit |
| | fdcvpdr | E 0.4 | BOOL | fin de course rentrant du vérin petit débit |
| | fdcvvr | E 0.5 | BOOL | fin de course rentrant du vérin de vidange |
| | fdcvgds | E 0.6 | BOOL | fin de course sortant du vérin grand débit |
| | fdcvpds | E 0.7 | BOOL | fin de course sortant du vérin petit débit |
| | frcvvs | E 1.0 | BOOL | fun de course sortant du vérin de vidange |
| | PSTAm | E 1.1 | BOOL | capteur de présance de sucre dans la trémi amant |
| | DTBBP | E 1.2 | BOOL | détecteur de bourage de la benne peuseuse |
| | AUR | E 1.3 | BOOL | arrêt d'urgence |
| | DBTAV | E 1.4 | BOOL | détecteur de bourage de la trémi avale |
| | ew1 | EW 1 | WORD | |
| | DCAT | FB 81 | FB 81 | Discrete Control Alarm Timer |
| | fonctionnement | FC 1 | FC 1 | bloc du programme de fonctionnement |
| | permutation manuel/auto | FC 2 | FC 2 | bloc de permutation auto-man |
| | fonction pour wincc | FC 3 | FC 3 | bloc des fonction de supervision |
| | alarme | FC 4 | FC 4 | bloc des alarmes |
| | défaux des verins | FC 5 | FC 5 | |
| | alarmes et témoin | FC 6 | FC 6 | |
| | TBL | FC 88 | FC 88 | Table |
| | SCALE | FC 105 | FC 105 | Scaling Values |
| | Etape (11) | M 0.0 | BOOL | étape de mise en marche |
| | Etape (220) | M 0.1 | BOOL | étape de mise en marche avec mode auto |
| | Etape (230) | M 0.2 | BOOL | étape de remise a létat initiale |
| | Etape (00) | M 0.3 | BOOL | etape initiale du grafcet auto |
| | Etape (210) | M 0.4 | BOOL | étap de mise en marche avec mode semi auto |
| | Etape (03) | M 0.6 | BOOL | étape pour défirencier entre deux réseux de même condition d'activation |
| | cocgd | M 1.0 | BOOL | commande d'ouverture du casque grand débit |
| | cocpd | M 1.1 | BOOL | commande d'ouvertura du casque petit débit |
| | cfcgd | M 1.2 | BOOL | commande de fermiture du casque grand débit |
| | cocv | M 1.3 | BOOL | commande d'ouverture du casque de vidange |
| | cfcpd | M 1.4 | BOOL | commande de fermiture du casque petit débit |
| | cfcv | M 1.5 | BOOL | commande de fermiture du casque de vidange |
| | Marche | M 1.6 | BOOL | mémnto de lise en marche |
| | ARRET | M 1.7 | BOOL | mémemento d'arrêt |
| | ouverture du casque gd | M 5.0 | BOOL | memento déanimation du casque grand débit |
| | ouverture du casque pd | M 5.1 | BOOL | mémemento déanimation du casque petit débit |

| Etat | Mnémorique | Opérande | Type de données | Commentaire |
|-------------|-----------------------|-----------------|------------------------|--|
| | ouverture du casque v | M 5.2 | BOOL | méméto d'animation du casque de vidange |
| | alarme AUR | M 20.0 | BOOL | méméto d'alarme d'arrêt d'urgence |
| | alarme bourrage B_P | M 20.1 | BOOL | méméto du défaut de bourrage de la benne |
| | alarme bourrage T_A_V | M 20.2 | BOOL | méméto du défaut de la trémie avale |
| | défaut_vgd | M 20.3 | BOOL | méméto du défaut du vérin grand débit |
| | défaut_vpd | M 20.4 | BOOL | méméto du défaut du vérin petit débit |
| | défaut_vv | M 20.5 | BOOL | méméto du défaut du vérin de vidange |
| | pesée 1 | MD 0 | DWORD | |
| | consigne pesée 1 | MD 1 | DWORD | |
| | CONSIGNE GRAND DEBIT | MW 1 | WORD | |

OB1 - <offline>

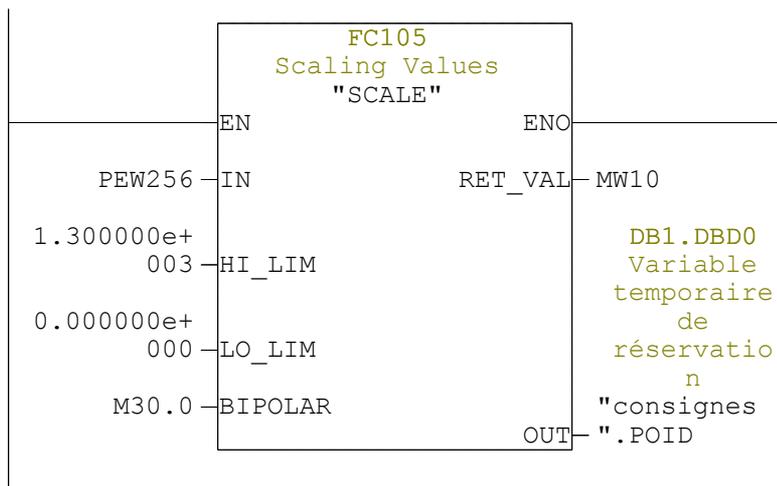
""

Nom : **Famille :**
Auteur : **Version :** 0.1
Version de bloc : 2
Horodatage Code : 03/06/2018 13:40:07
Interface : 15/02/1996 16:51:12
Longueur (bloc/code /données locales) : 00296 00144 00034

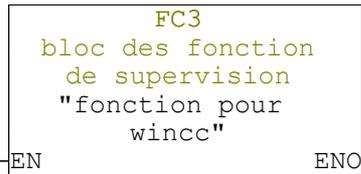
| Nom | Type de données | Adresse | Commentaire |
|----------------|-----------------|---------|---|
| TEMP | | 0.0 | |
| OB1_EV_CLASS | Byte | 0.0 | Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1) |
| OB1_SCAN_1 | Byte | 1.0 | 1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1) |
| OB1_PRIORITY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

Bloc : OB1 "Main Program Sweep (Cycle)"

Réseau : 1 Bloc de mise a l'échelle de l'entrée analogique



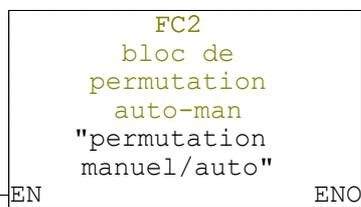
Réseau : 2



Réseau : 3



Réseau : 4 Réseau de sécurité



Réseau : 5



FC1 - <offline>

"fonctionnement" bloc du programme de fonctionnement

Nom : **Famille :**

Auteur : **Version :** 0.1

Version de bloc : 2

Horodatage Code : 23/06/2018 12:32:22

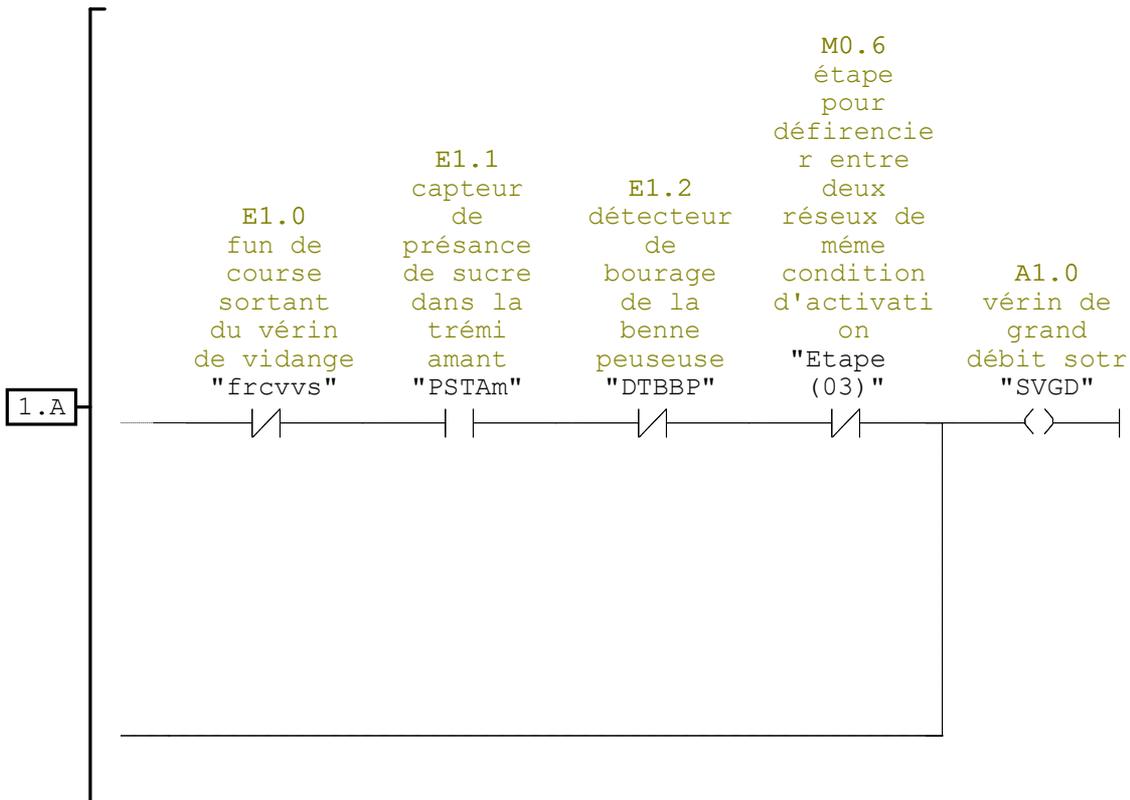
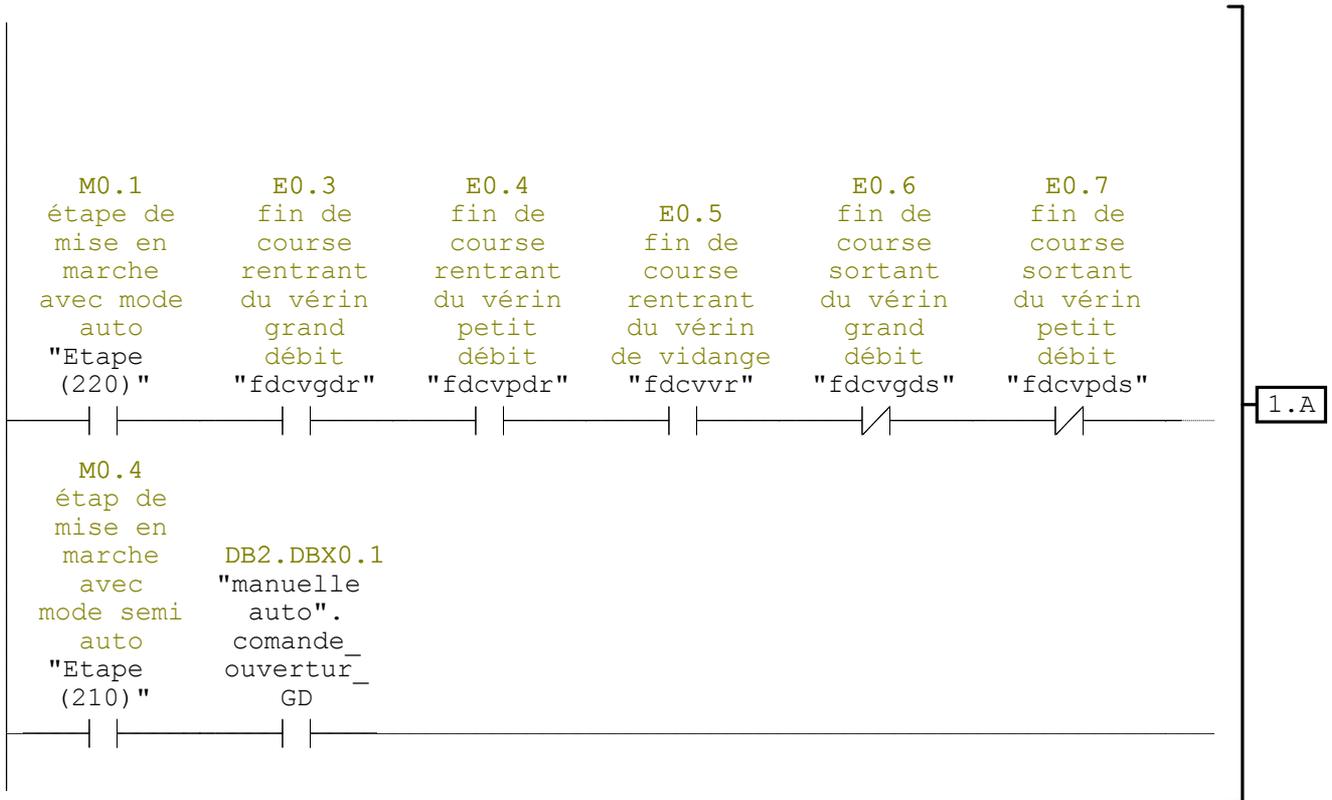
Interface : 27/05/2018 14:19:01

Longueur (bloc/code /données locales) : 00506 00386 00000

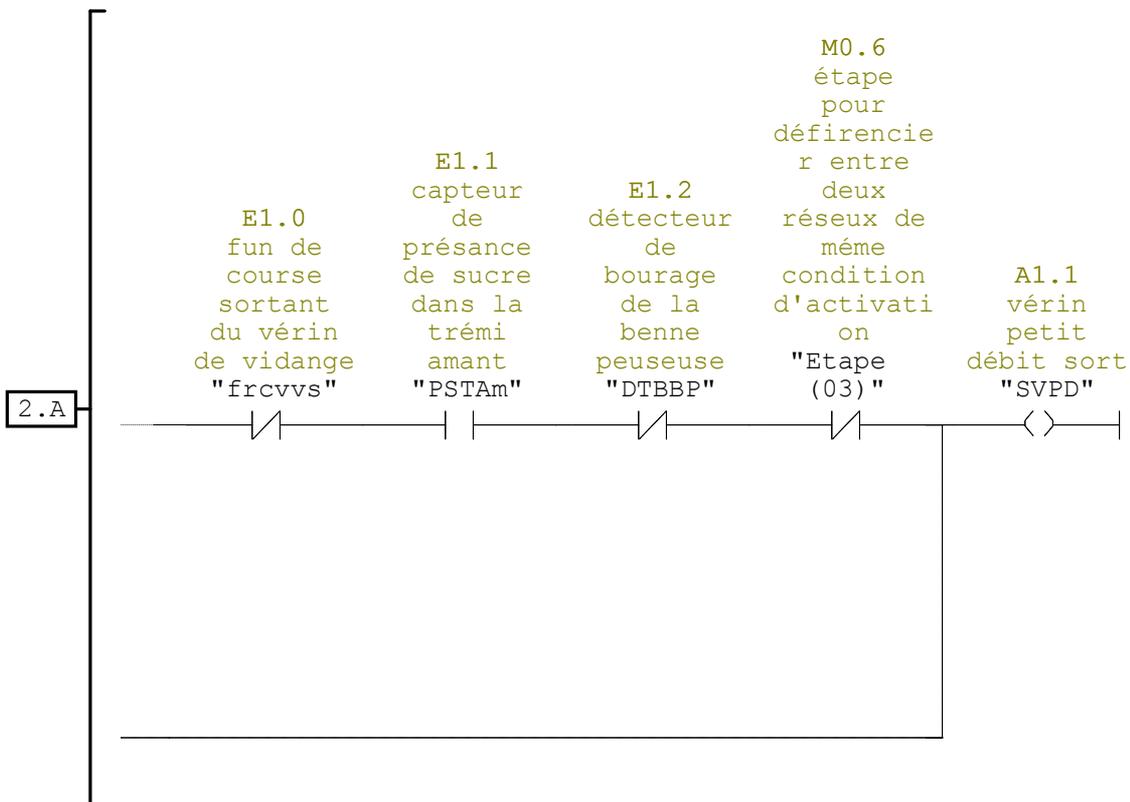
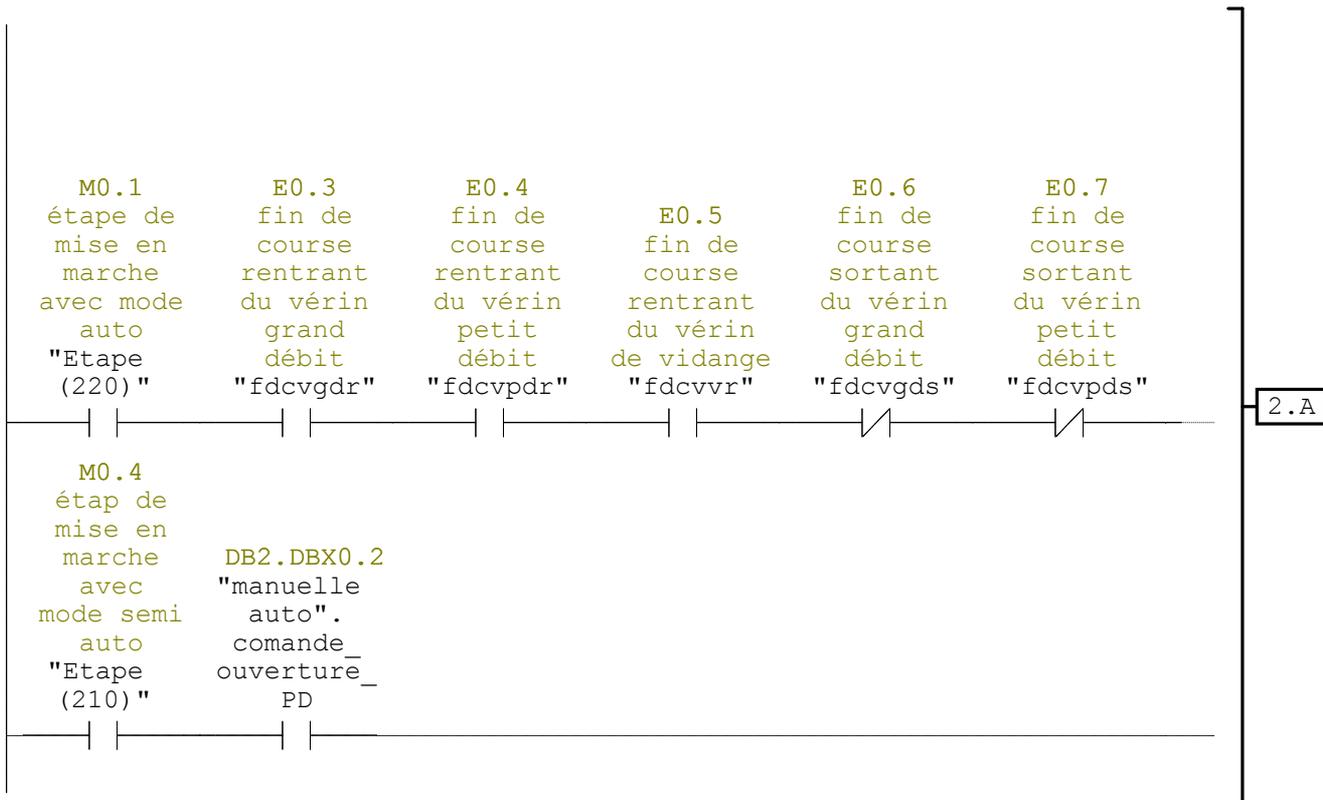
| Nom | Type de données | Adresse | Commentaire |
|---------|-----------------|---------|-------------|
| IN | | 0.0 | |
| OUT | | 0.0 | |
| IN_OUT | | 0.0 | |
| TEMP | | 0.0 | |
| RETURN | | 0.0 | |
| RET_VAL | | 0.0 | |

Bloc : FC1

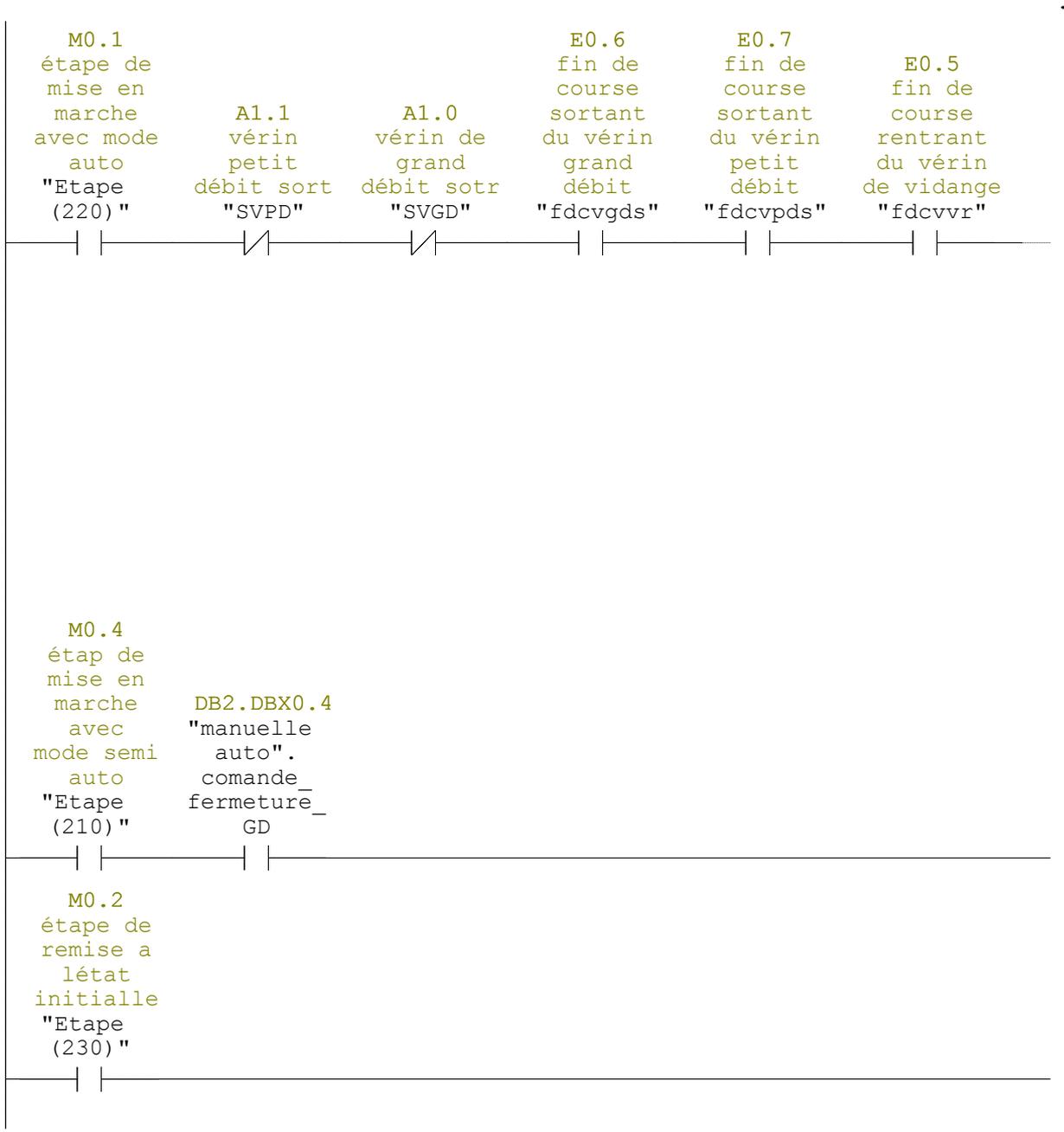
Réseau : 1 sortie du vérin grand débit



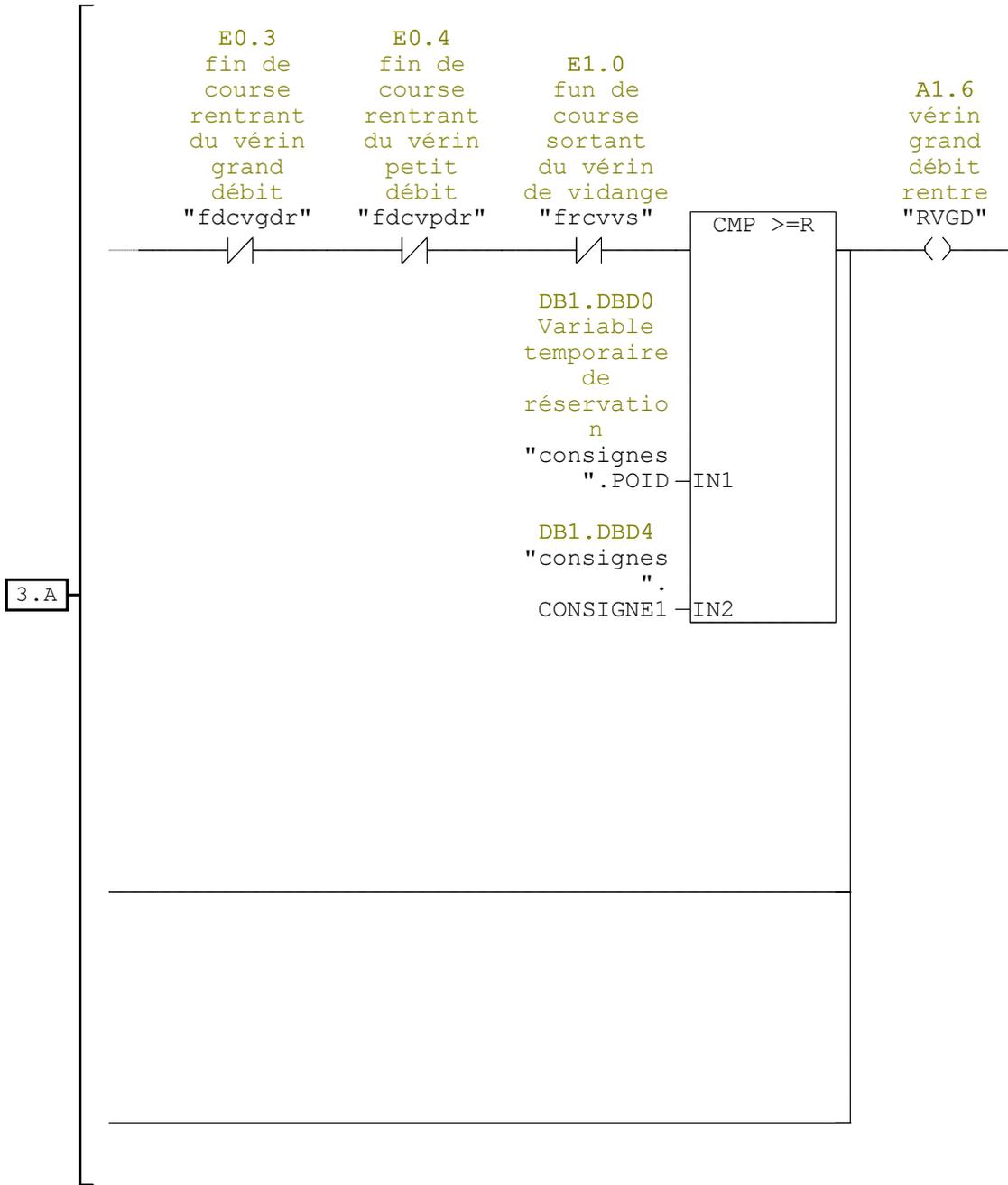
Réseau : 2 sortie du vérin petit débit



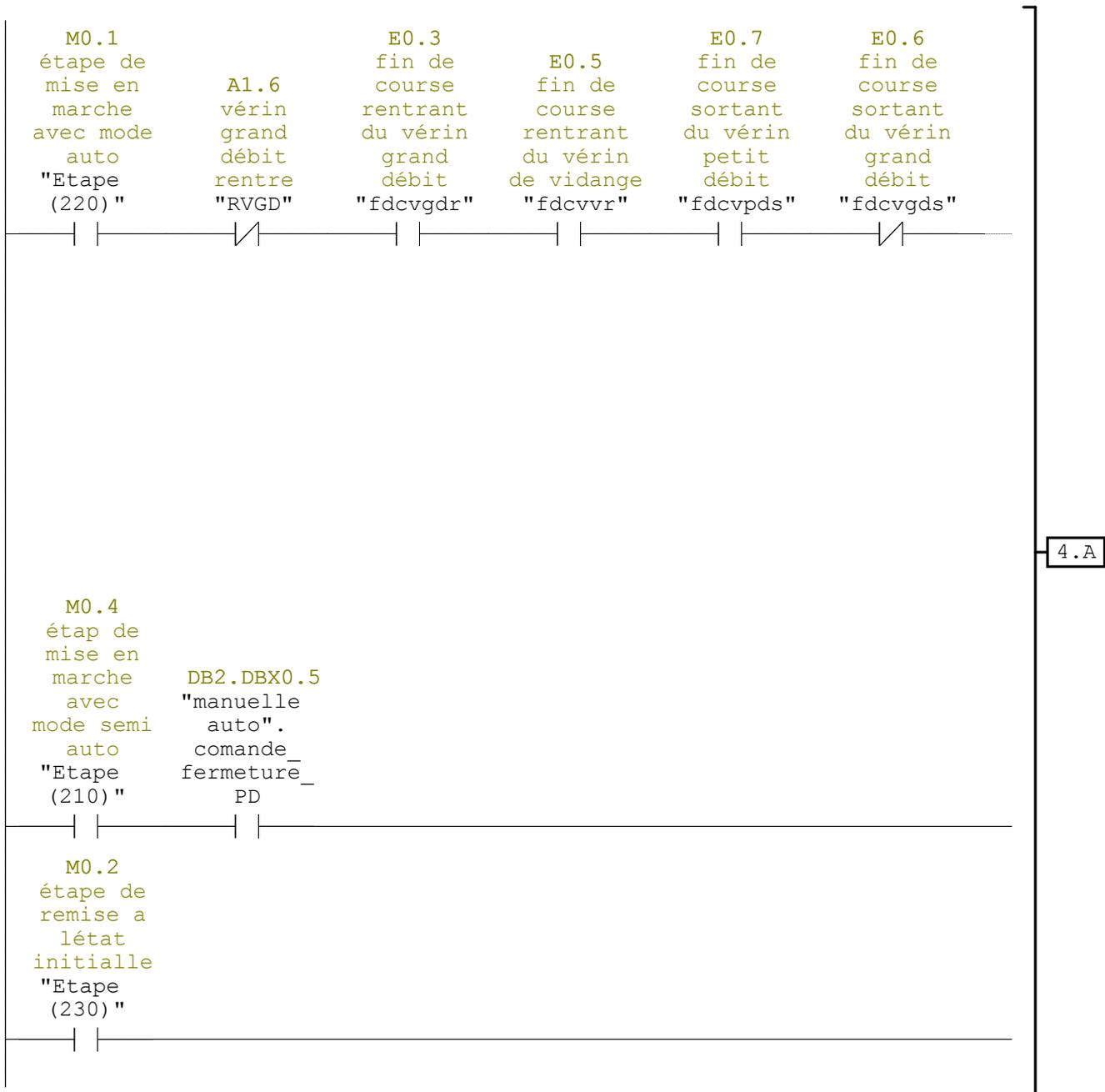
Réseau : 3 rentrée du vérin grand débit

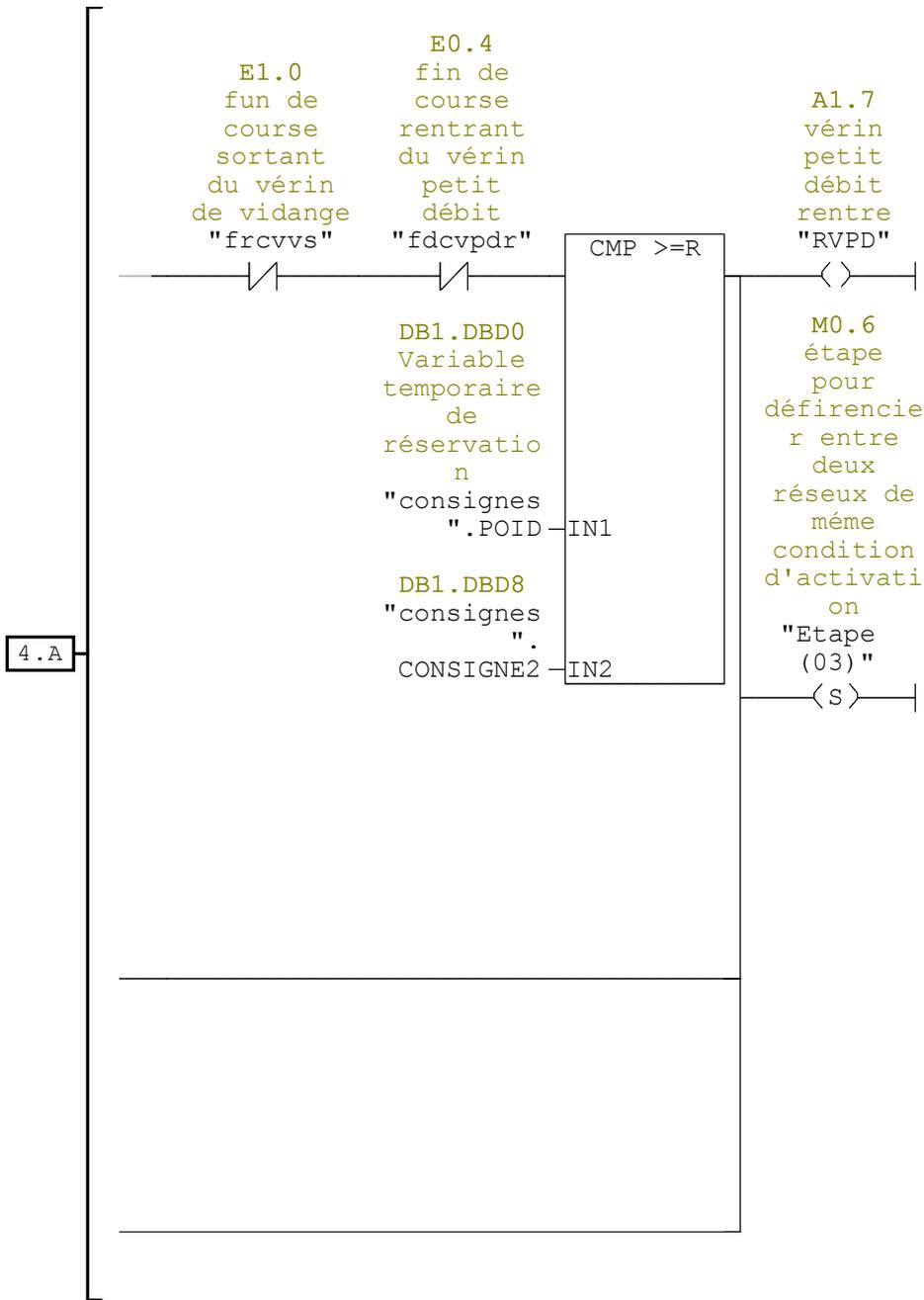


3.A

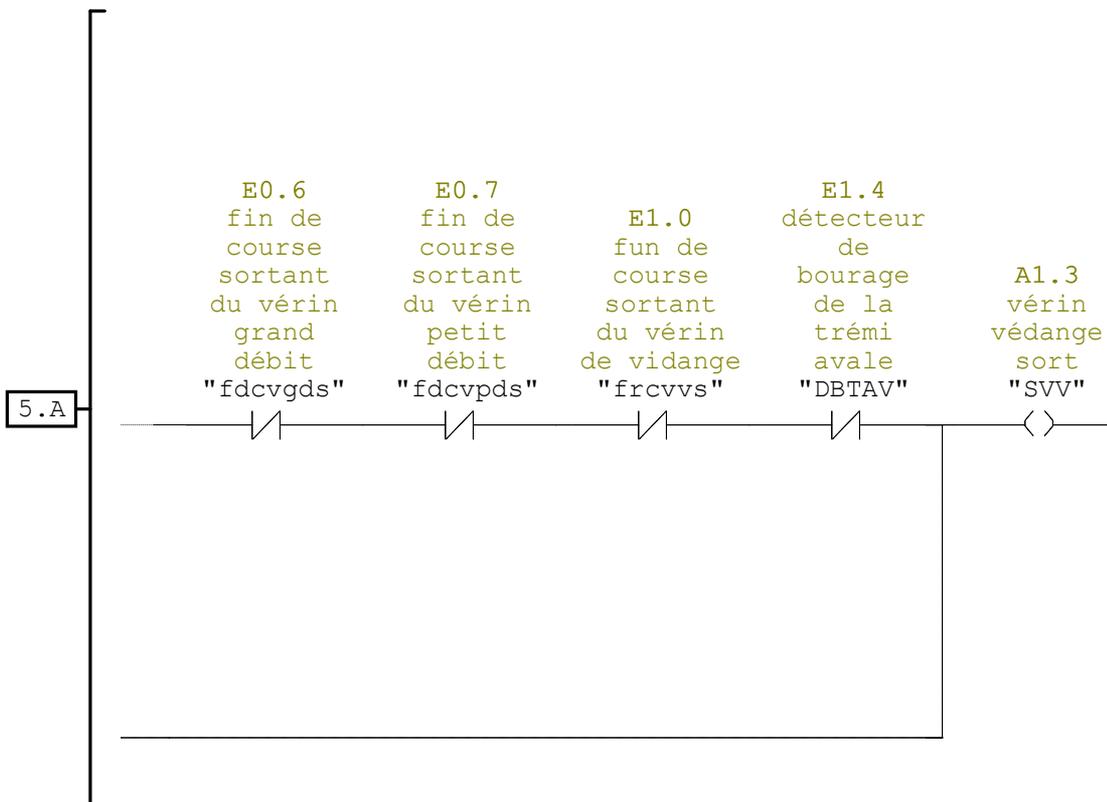
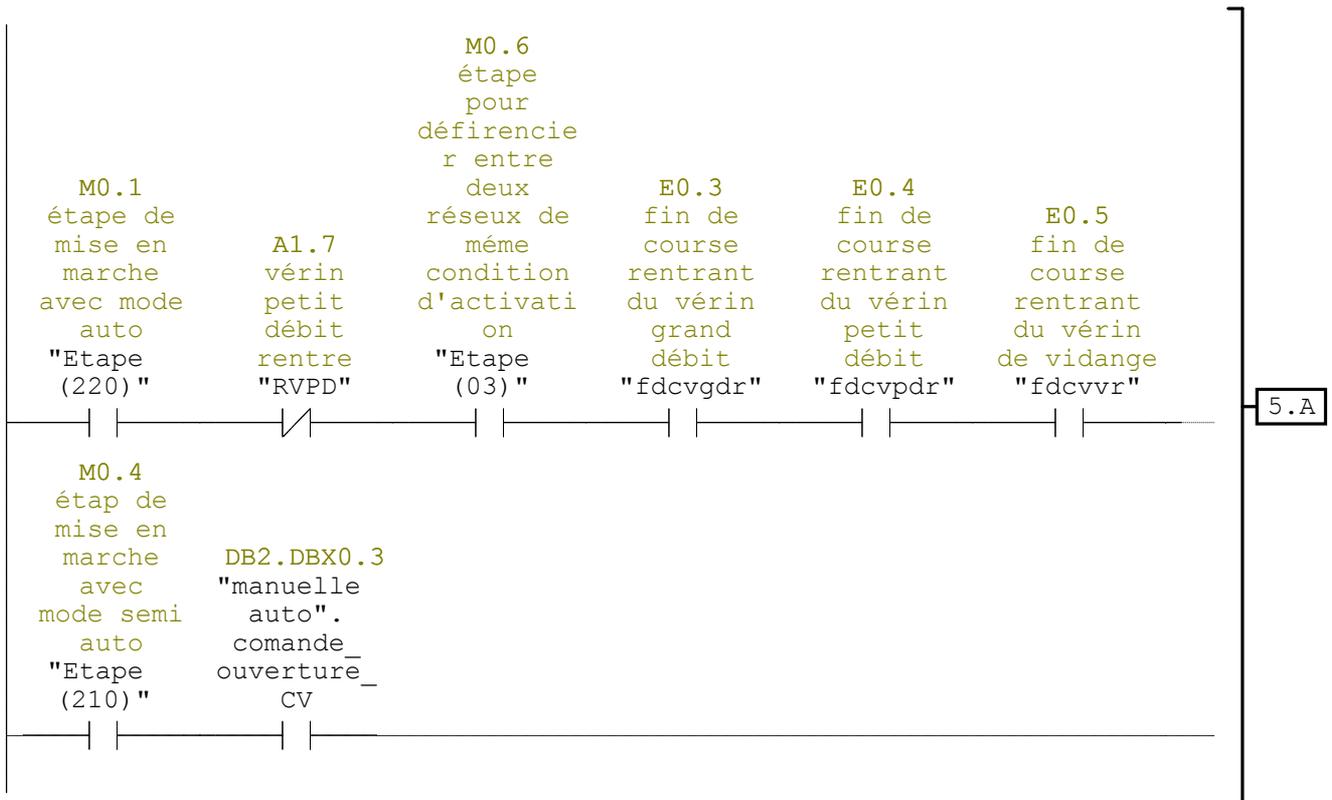


Réseau : 4 rentrée du vérin petit débit

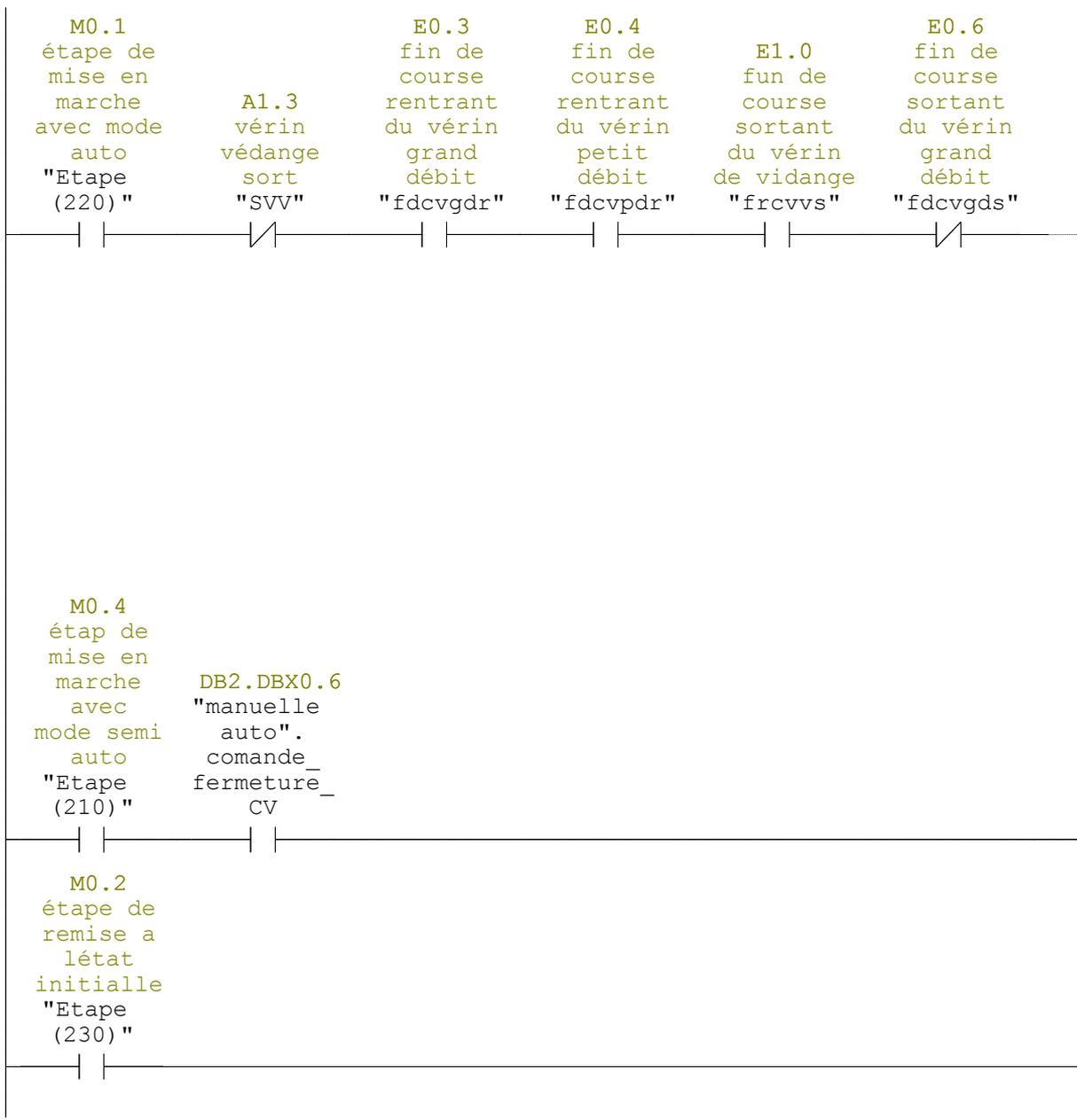




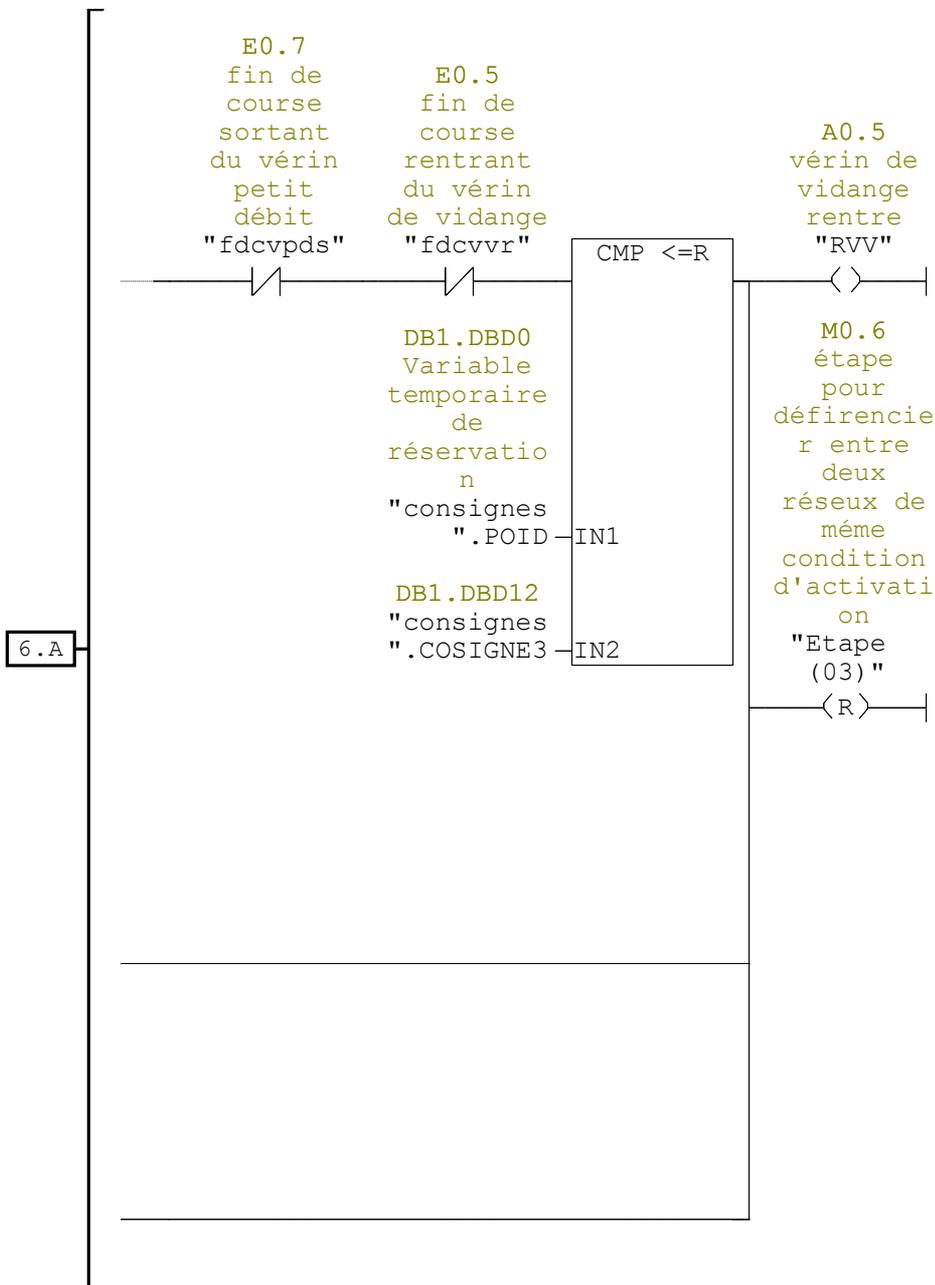
Réseau : 5 sortie du vérin de vidange



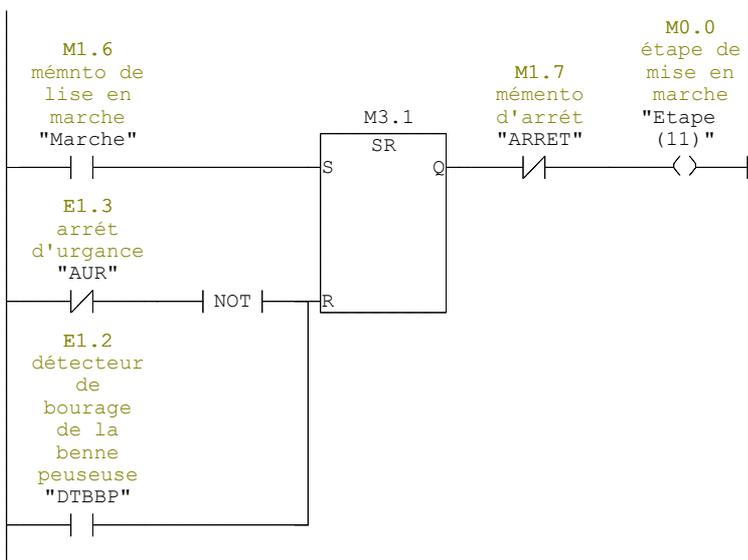
Réseau : 6 rentrée du verin de vidange



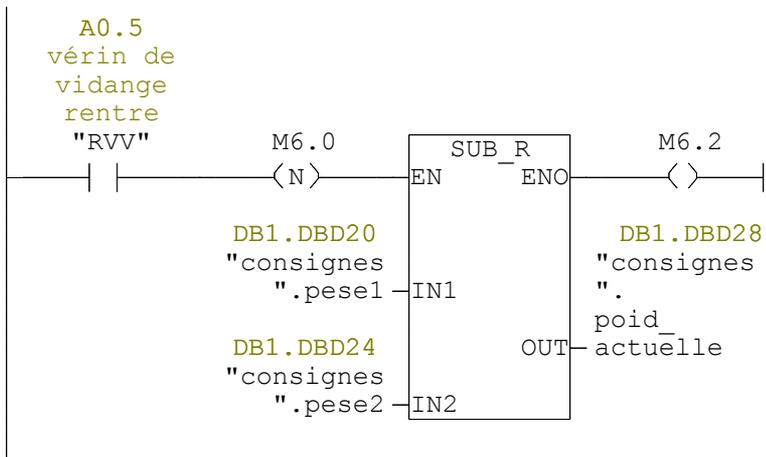
6.A



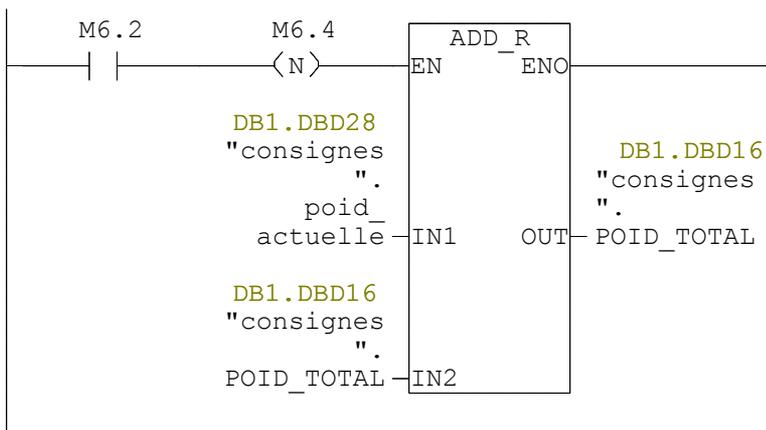
Réseau : 7 mise en marche ou arrêt et sécurité



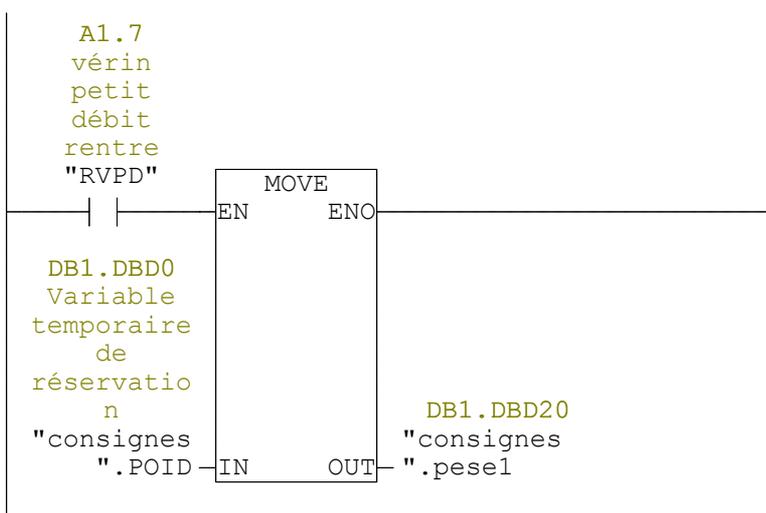
Réseau : 8 soustraction du poids réséduelle



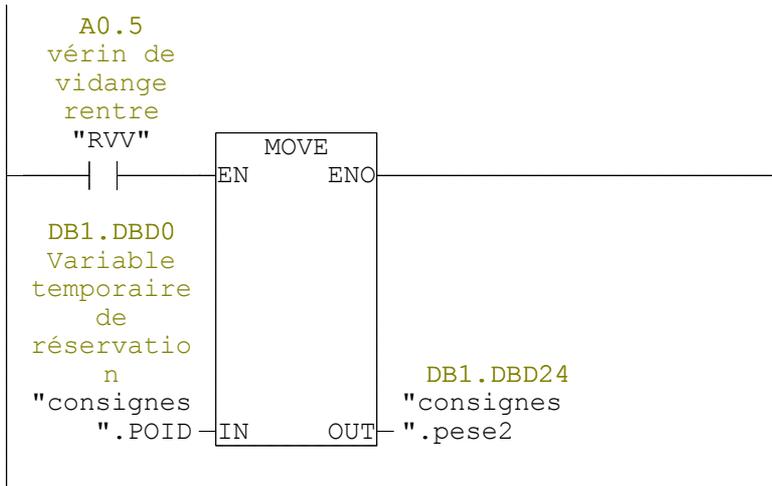
Réseau : 9 adition des poids pesée



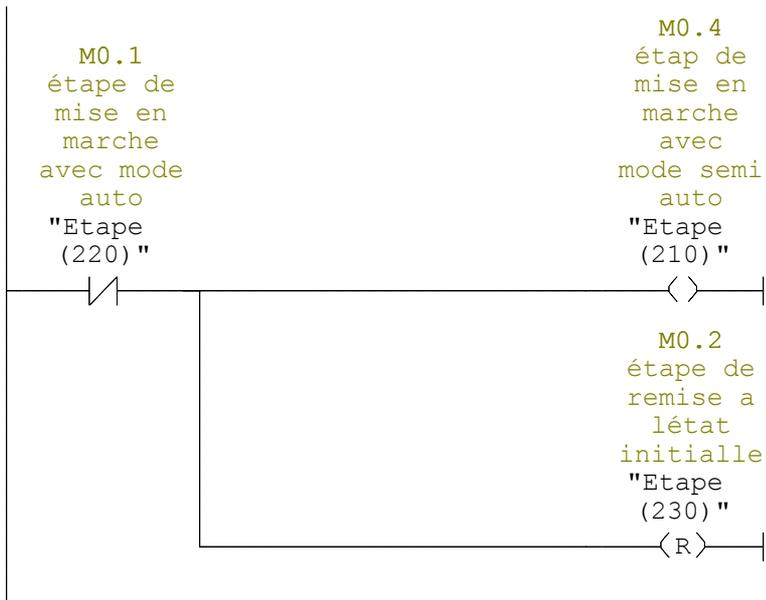
Réseau : 10 déplcement de la donnée de la pesé du cycle



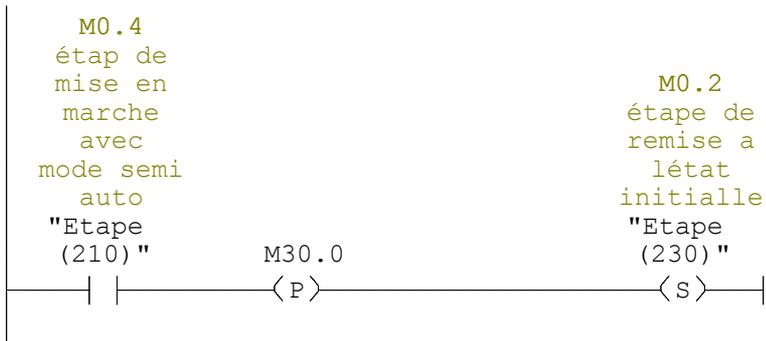
Réseau : 11 déplacement de la donnée de pesé après vidange



Réseau : 2 selection manuelle



Réseau : 3



FC3 - <offline>

"fonction pour wincc" bloc des fonction de supervision
Nom : **Famille :**
Auteur : **Version :** 0.1
 Version de bloc : 2
Horodatage Code : 07/06/2018 12:09:49
 Interface : 27/05/2018 15:15:59
Longueur (bloc/code /données locales) : 00164 00058 00000

| Nom | Type de données | Adresse | Commentaire |
|---------|-----------------|---------|-------------|
| IN | | 0.0 | |
| OUT | | 0.0 | |
| IN_OUT | | 0.0 | |
| TEMP | | 0.0 | |
| RETURN | | 0.0 | |
| RET_VAL | | 0.0 | |

Bloc : FC3

Réseau : 1



Réseau : 2



Réseau : 3

| | |
|------------|------------|
| A1.1 | M5.1 |
| vérin | memento |
| petit | déanimatio |
| débit sort | n du |
| "SVPD" | casque |
| | petit |
| | débit |
| | "ouverture |
| | du |
| | casque pd" |
| | <S> |

Réseau : 4

| | |
|--------|------------|
| A1.7 | M5.1 |
| vérin | memento |
| petit | déanimatio |
| débit | n du |
| rentre | casque |
| "RVPD" | petit |
| | débit |
| | "ouverture |
| | du |
| | casque pd" |
| | <R> |

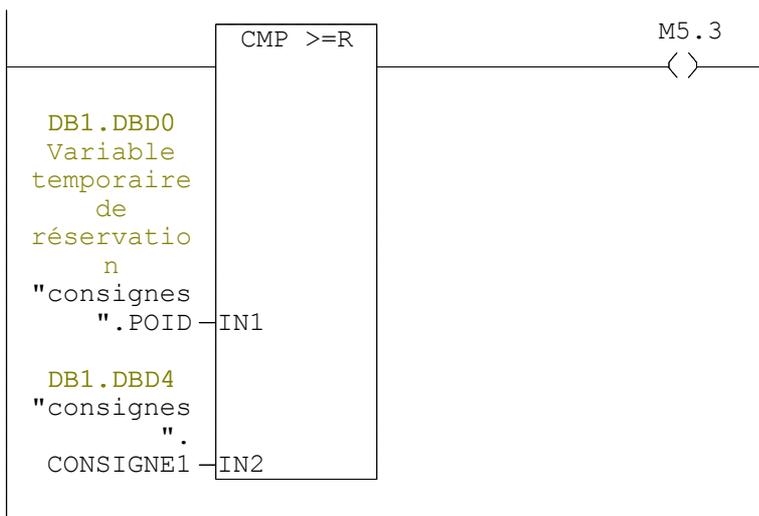
Réseau : 5

| | |
|---------|------------|
| A1.3 | M5.2 |
| vérin | memento |
| védange | d'animatio |
| sort | n du |
| "SVV" | casque de |
| | vidange |
| | "ouverture |
| | du |
| | casque v" |
| | <S> |

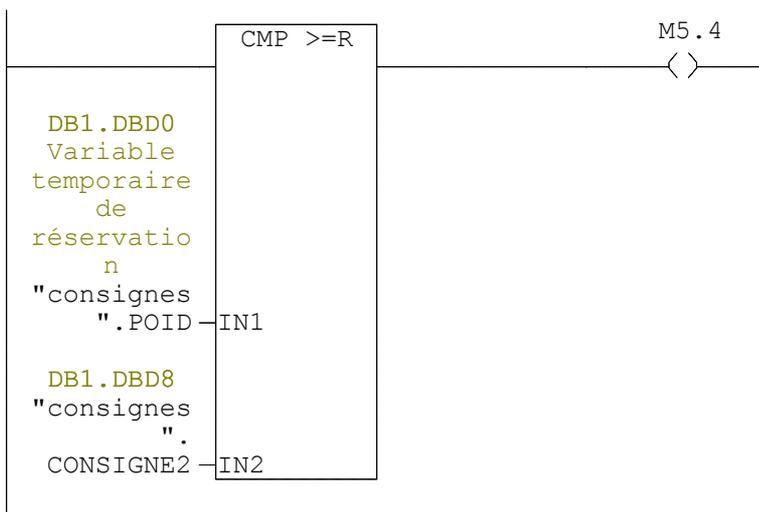
Réseau : 6



Réseau : 7



Réseau : 8



FC4 - <offline>

"alarme" bloc des alarmes

Nom : Famille :

Auteur : Version : 0.1

Version de bloc : 2

Horodatage Code : 23/06/2018 12:24:57

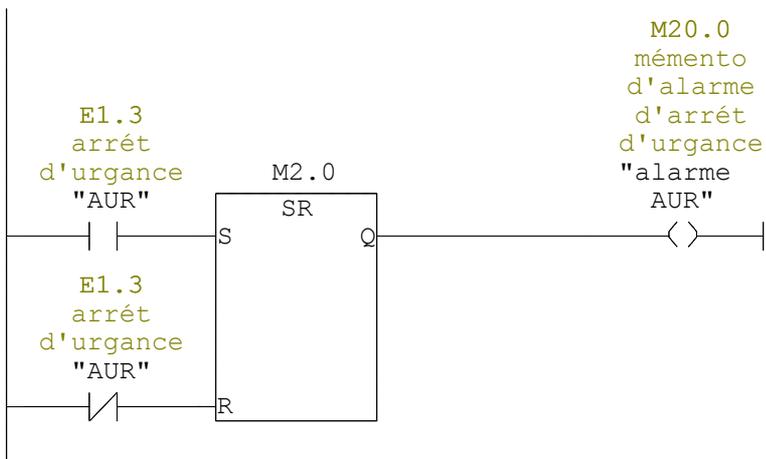
Interface : 27/05/2018 15:21:00

Longueur (bloc/code /données locales) : 00232 00122 00000

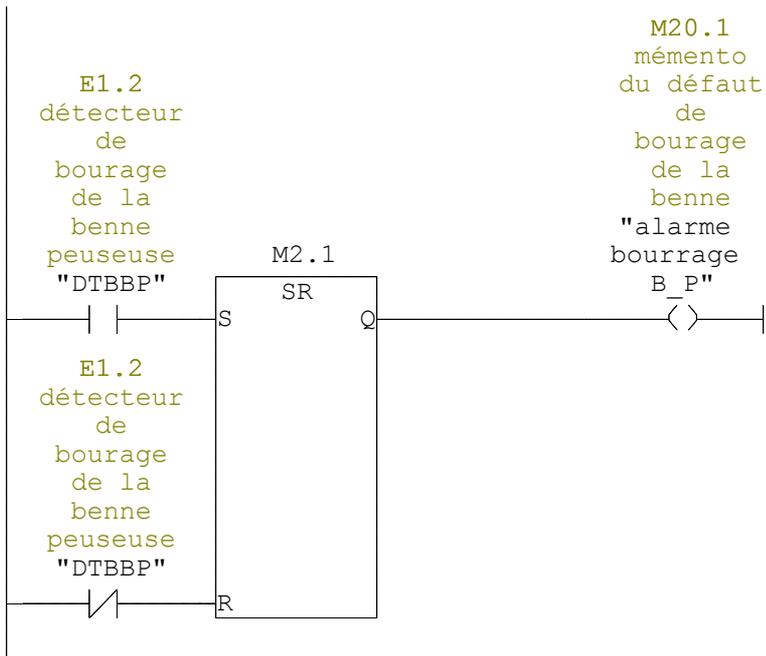
| Nom | Type de données | Adresse | Commentaire |
|---------|-----------------|---------|-------------|
| IN | | 0.0 | |
| OUT | | 0.0 | |
| IN_OUT | | 0.0 | |
| TEMP | | 0.0 | |
| RETURN | | 0.0 | |
| RET_VAL | | 0.0 | |

Bloc : FC4

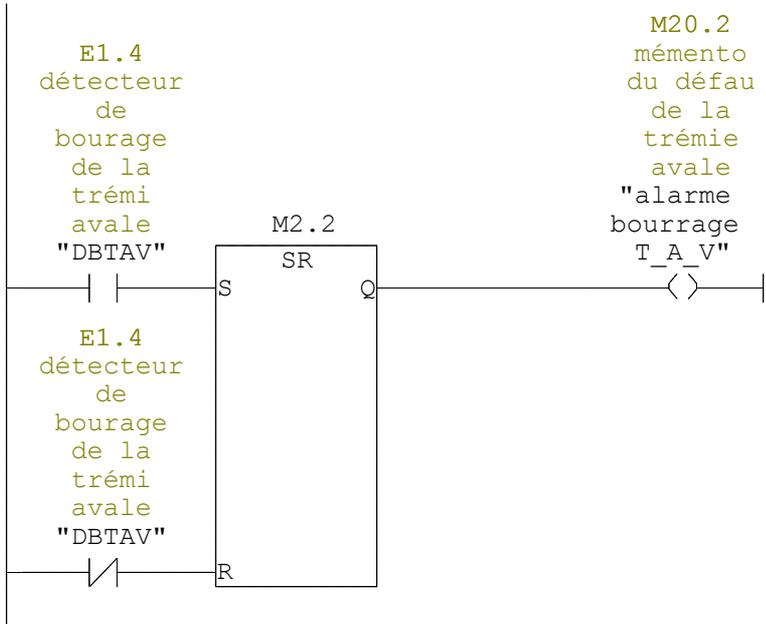
Réseau : 1 alarme d'arrêt d'urgence



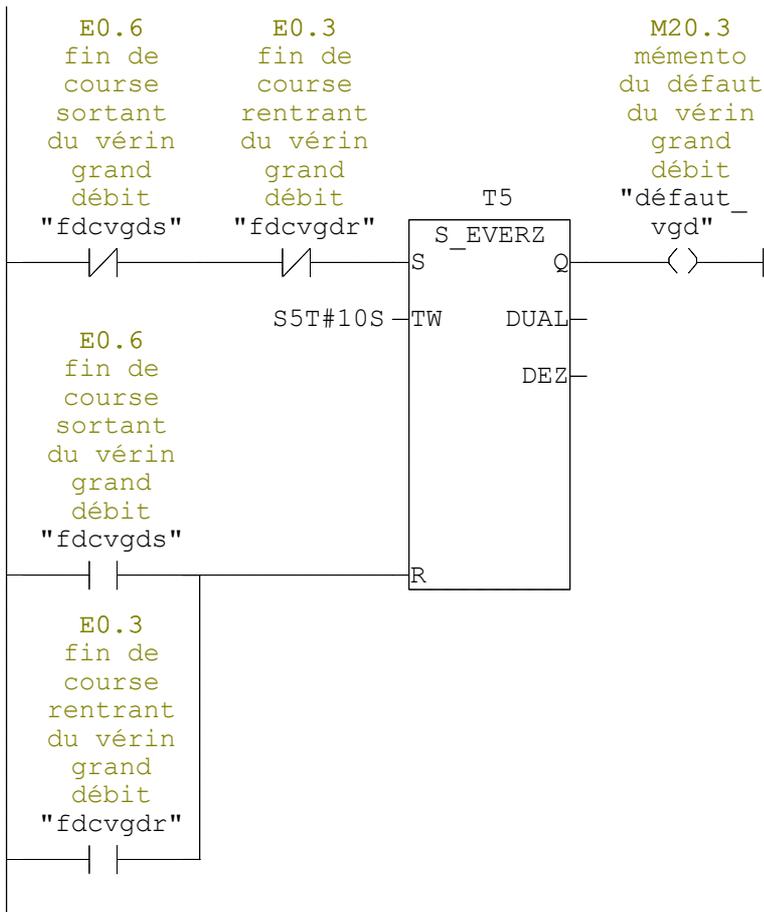
Réseau : 2 alarme de bourage de la benne peuseuse



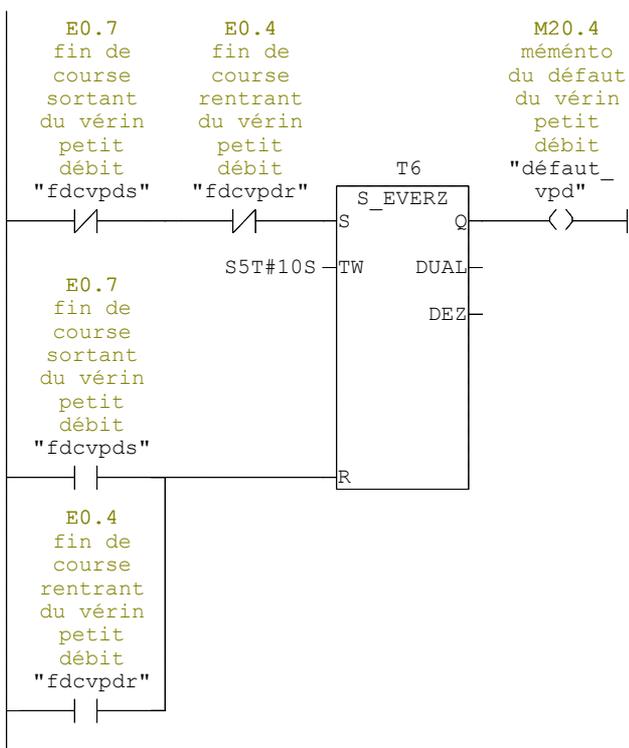
Réseau : 3 alarme de bourage de trémie avale



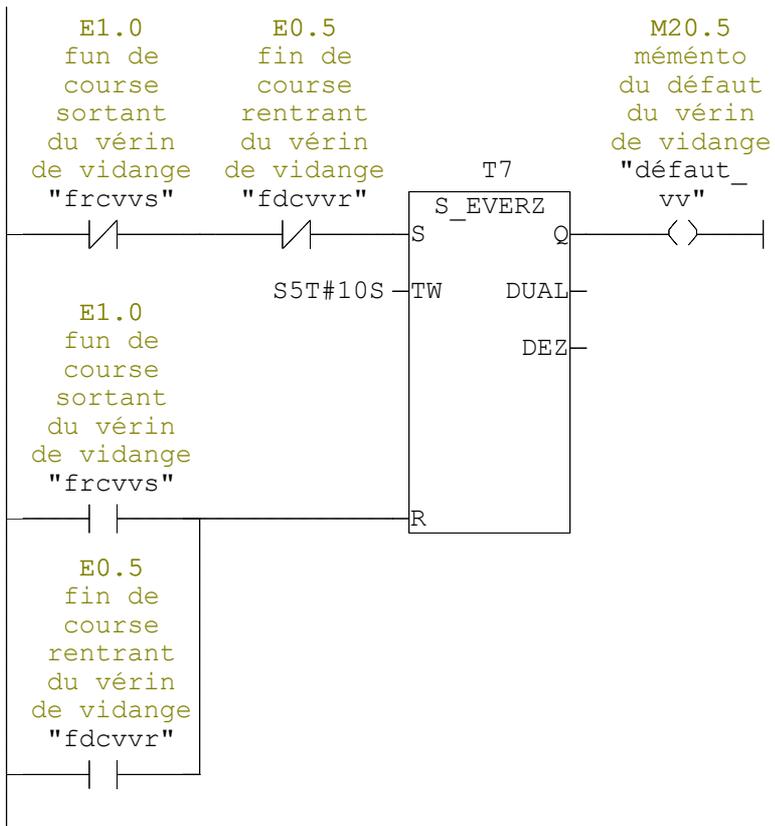
Réseau : 4 défaut du vérin du grand débit



Réseau : 5 défaut du vérin du petit débit



Réseau : 6 défaut du vérin de vidange



DB1 - <offline> - Vue des déclarations

"consignes" bloc de données de consigne de pesé

Bloc de données (DB) global 1

Nom : **Famille :**
Auteur : **Version :** 0.1
Version de bloc : 2
Horodatage Code : 19/05/2018 13:28:48
Interface : 15/05/2018 13:27:43
Longueur (bloc/code /données locales) : 00148 00032 00000

Bloc : DB1

| Adresse | Nom | Type | Valeur initiale | Commentaire |
|---------|---------------|------------|-----------------|------------------------------------|
| 0.0 | | STRUCT | | |
| +0.0 | POID | REAL | 0.000000e+000 | Variable temporaire de réservation |
| +4.0 | CONSIGNE1 | REAL | 8.000000e+002 | |
| +8.0 | CONSIGNE2 | REAL | 1.150000e+003 | |
| +12.0 | COSIGNE3 | REAL | 1.000000e+002 | |
| +16.0 | POID TOTAL | REAL | 0.000000e+000 | |
| +20.0 | pesel | REAL | 0.000000e+000 | |
| +24.0 | pese2 | REAL | 0.000000e+000 | |
| +28.0 | poid_actuelle | REAL | 0.000000e+000 | |
| =32.0 | | END_STRUCT | | |

DB2 - <offline> - Vue des déclarations

"manuelle auto" bloc de de données depermutation et de commande semi auto

Bloc de données (DB) global 2

Nom : **Famille :**
Auteur : **Version :** 0.1
Version de bloc : 2
Horodatage Code : 07/06/2018 00:23:41
Interface : 07/06/2018 00:23:41
Longueur (bloc/code /données locales) : 00104 00002 00000

| |
|-------------------|
| Bloc : DB2 |
|-------------------|

| Adresse | Nom | Type | Valeur initiale | Commentaire |
|---------|----------------------|------------|-----------------|------------------------------------|
| 0.0 | | STRUCT | | |
| +0.0 | permutatio nman auto | BOOL | FALSE | Variable temporaire de réservation |
| +0.1 | comande ouvertur GD | BOOL | FALSE | |
| +0.2 | comande ouverture PD | BOOL | FALSE | |
| +0.3 | comande ouverture CV | BOOL | FALSE | |
| +0.4 | comande fermeture GD | BOOL | FALSE | |
| +0.5 | comande fermeture PD | BOOL | FALSE | |
| +0.6 | comande fermeture CV | BOOL | FALSE | |
| =2.0 | | END_STRUCT | | |