

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

Université Abderrahmane MIRA de Bejaia
Faculté de Technologie

Département d'électrotechnique



MÉMOIRE DE FIN D'ÉTUDE

En vue de l'obtention du diplôme de Master en
Commande Electrique / Electrotechnique Industrielle

THÈME

**Migration STEP5 vers STEP7 d'un
descenseur de pots HAMBА CEVITAL**

Réalisé par :

- Mr. AGGOUN Massil
 - Mr. OUAHRANI Lyes
-

Encadré par :

- Mr. ALKAMA Rezak
 - Mr. HITACHI Farid
-

Membres de jury :

- Mr. A.MELAHI
 - Mr. S.TARAFT
-

Promotion

2020

Remerciements

Nous remercions, Dieu, le tout puissant pour nous avoir donné la foi qui nous a guidé jusqu'à la réalisation et l'aboutissement de ce projet. Nous tenons à exprimer nos remerciements les plus distingués :

*A notre promoteur **Mr. ALKAMA** et à notre encadreur **Mr. HITACHI**. De nous avoir fait l'honneur d'assurer l'encadrement de notre travail, nous vous sommes très reconnaissants d'avoir veillé à son élaboration en ne ménageant aucunement votre temps et vos conseils.*

Nous tenons à remercier vivement messieurs les membres du jury d'avoir consacré de leur temps à la lecture de ce manuscrit, et d'accepter de juger et d'évaluer ce travail.

Par le biais de ce travail, nous exprimons notre profonde gratitude à toutes les personnes qui, de près ou de loin, nous ont aidées et accompagnées dans notre travail.

Nous voudrions aussi remercier nos chers parents et nos Familles qui nous ont Soutenus dans nos études.

Massil & Lyes

Dédicaces

A tous ceux qui nous sont chers...

Massil, Lyes

Table des matières

Sommaire :

<i>Introduction générale</i>	<i>1</i>
<i>Description du complexe CEVITAL</i>	<i>3</i>
1. Présentation générale de l'entreprise	3
2. Historique	3
3. Situation géographique	3
4. Activités et unités du complexe CEVITAL	4
5. Missions et objectifs	4
6. Différents organes constituant le complexe CEVITAL	5
7. Unité de Production de margarine	6
<i>Chapitre I : Automates programmables et logiciels associés</i>	<i>7</i>
I.1 Introduction	7
I.2 L'automatisme	7
I.2.1 Objectif de l'automatisation	7
I.2.2 Les systèmes automatisés	8
I.2.3.1 La partie commande (PC).....	8
I.2.3.2 La partie opérative (PO)	8
I.2.3.3 La partie relation (PR)	9
I.3 Généralités sur les automates programmables	9
I.3.1 Historique	9
I.3.2 Définition de l'automate programmable	9
I.3.3 Critères de choix d'un automate	10
I.3.4 Avantages des automates programmables	11
I.3.5 Traitement du programme automate	11
I.3.6 Langage de programmation pour API	12
I.3.6.1 Le langage LD (Ladder Diagram)	12
I.3.6.2 Le langage IL (Instruction List).....	12
I.3.6.3 Le langage FBD (Function Block Diagram).....	12
I.3.6.4 Le langage ST (Structured Text)	13
I.3.6.5 Le langage SFC (Sequential Function Chart), ou GRAFCET	13
I.3.7 Architecture des automates programmables.....	13
I.3.8 Structure interne des automates programmables	15
I.3.8.1 Le processeur :.....	15
I.3.8.2 Les mémoires.....	16
I.3.8.3 Les modules d'entrées/sorties.....	16
I.3.8.4 L'alimentation	17
I.3.8.5 Liaisons de communication	17
I.4 TIA Portal (Totally Integrated Automation)	17
I.4.1 Introduction	17
I.4.2 Description du logiciel TIA Portal	18
I.4.3 Les avantages du logiciel TIA portal	18
I.4.4 SIMATIC STEP 7	18
I.4.5 Vue du portail et vue du projet	18
I.4.6 Adressage des E/S	20
I.4.7 Les variables API	21
I.4.7.1 Adresses symboliques et absolue.....	21
I.4.7.2 Table des variables API.....	22
I.4.8 Liaison avec l'automate.....	22
I.4.9 Adresse Ethernet de la CPU	22

Sommaire

I.4.10 WinCC sur TIA portal.....	22
I.5 Conclusion.....	23
Chapitre II : Etude et présentation du descenseur de pots HAMBА BK 8006-M.....	24
II.1 Introduction.....	24
II.2 Description de la machine.....	26
II.3 Principe de fonctionnement.....	27
II.4 Branchement aux sources d'énergies.....	28
II.5 Déroulements des menus sur l'OP.....	29
II.6 Procédure de mise en marche.....	30
II.6.1 Vérification avant la mise en marche.....	30
II.6.2 Procédure de mise en marche.....	30
II.6.3 Procédure de changement de format.....	30
II.6.4 Procédure de mise en marche du CIP.....	30
II.6.5 Procédure de mise à l'arrêt.....	31
II.6.6 Ajustement remplissage.....	31
II.7 Problématique.....	31
II.8 Conclusion.....	31
Chapitre III : Etude fonctionnelle en vue d'une migration.....	32
III.1 Introduction.....	32
III.2 Considérations et problèmes à traiter avant la migration.....	32
III.3 Présentation du S5-100U.....	33
III.3.1 Description de l'automate S5-100U.....	33
III.3.1.1 Module d'alimentation :.....	33
III.3.1.2 Unité central (CPU) :.....	33
III.3.1.3 Le bus interne :.....	34
III.3.1.4 Les mémoires :.....	34
III.3.1.5 Les modules d'entrées /sorties :.....	34
III.3.2 Elaboration d'un programme S5.....	35
III.3.3 Les différents modes de représentation.....	35
III.3.4 Stratégie de la firme SIEMENS pour la série S5.....	35
III.3.5 Solution pour remplacer SIMATIC S5.....	36
III.4 Présentation du S7-300.....	36
III.4.1 CPU.....	37
III.4.1.1 LED de visualisation d'état et de défaut.....	38
III.4.1.2 Commutateur de mode de fonctionnement.....	38
III.4.1.3 Pile de sauvegarde ou accumulateur.....	38
III.4.1.4 Carte mémoire.....	39
III.4.1.5 Interface MPI (interface multipoint).....	39
III.4.2 Module d'alimentation.....	39
III.4.3 Module d'entrées/sorties.....	39
III.4.3 Organisation du projet Step 7.....	39
III.5 Raisons de choix de SIEMENS S7-300.....	41
III.6 Comparatif.....	42
III.6.1 Mise en parallèle.....	42
III.6.1.1 Fonctions et blocs fonctionnels.....	43
III.6.1.2 Blocs de données.....	43
III.6.1.3 Blocs système.....	43
III.6.1.4 Blocs d'organisation.....	44
III.6.2 Types de données.....	44
III.6.3 Transposition des blocs à la conversion.....	45

III.7 Avantages de la modernisation :	45
III.8 Conclusion :	46
Chapitre IV : Migration du S5 vers S7	47
IV.1 Introduction	47
IV.2 Conversion du S5 vers S7	48
IV.2.1 Préparation de la conversion	48
IV.2.1.1 Fichier requis	48
IV.2.1.2 Vérification des opérandes	48
IV.2.1.3 Les phase de conversion	48
VI.2.1.4 Fichiers générés	49
IV.2.1.4 Interprétation des messages	49
IV.2.2 Convention	52
IV.2.3 Vérification et retouche du programme converti	56
IV.2.3.1 Préparation.....	56
IV.2.3.2 Compilation	56
IV.2.3.3 Vérification de la cohérence :	56
IV.2.3.4 Compilation du fichier source :	56
IV.2.3.5 Correction des erreurs :	56
IV.2.3.6 Réalisation des vérifications et retouches	57
IV.2.4 Configuration des matériels sur HW Config	62
IV.2.4.1 Paramétrage	63
IV.2.4.2 Manipulations de base pour la configuration matérielle	63
IV.2.4.3 Organisation de projet créé	64
IV.3 Migration S7 vers TIA PORTAL	65
IV.3.1 Condition de Migration :	65
IV.3.2. Marche à suivre pour faire la migration :	66
IV.3.3 Résultats de la migration	67
IV.4 Conclusion	70
Conclusion générale	71
BIBLIOGRAPHIE	72
ANNEXES	I

Liste des figures :

Figure 1 : Situation géographique du complexe Cevital.	3
Figure 2 : Organigramme du complexe Cevital.	5
Figure I.1 : Structure d'un système automatisé.....	8
Figure I.2 : Fonctionnement cyclique d'un automate.	11
Figure I.3 : Automate Programmable Industriel SIEMEN.	14
Figure I.4 : Structure interne d'un API.	15
Figure I.5 : Vue du portal.....	19
Figure I.6 : Vue du projet.....	20
Figure I.7 : Adressage des E/S.	21
Figure I.8 : Vue SIMATIC HMI.....	23
Figure II.1 : schéma remplisseuse BK 8006-M.	25
Figure II.2 : vue sur machine HAMBAL.	26
Figure II.3 : vue sur station de pots.....	27
Figure II.4 : vue sur station de pots (couvercles).....	28
Figure II.6 : Déroulements des menus sur l'OP.....	29
Figure III.1 : API S5-100U.....	33
Figure III.2 : Cycle de vie de série S5.	36
Figure III.3 : API S7-300.....	37
Figure III.4 : CPU 315-2DP.	37
Figure III.5 : Structure hiérarchique des objets du projet STEP 7.....	40
Figure III.6 : Différentes gammes des SIMATIC S5 et des nouveaux SIMATIC S7.	42
Figure III.7 : Blocs S5 et S7 remplissant des fonctions similaires.	45
Figure IV.1 : Vue d'ensemble des possibilités de migration.....	47
Figure IV.2 : Ouverture du convertisseur S5.	52
Figure IV.3 : Ouverture du fichier S5 à convertir.	53
Figure IV.4 : Choix du fichier à convertir.....	53
Figure IV.5 : Menu de conversion.	54
Figure IV.6 : Menu fin de conversion.....	55
Figure IV.7 : Messages d'erreurs et avertissements.....	55
Figure IV.8 : SIMATIC Manager création d'un nouveau projet.	57
Figure IV.9 : SIMATIC Manager création du projet	58
Figure IV.10 : Insertion d'un nouveau programme S7.	58
Figure IV.11 : Insertion d'une source externe.	59
Figure IV.12 : Choix du fichier .AWL.....	60
Figure IV.13 : Vue sur fenêtre SIMATIQUE Manager une fois le fichier source insérer.	60

Liste des figures

Figure IV.14 : Vue sur sources LIST	61
Figure IV.15 : Résultats de compilation.	62
Figure IV.16 : projet créé sur STEP7.....	64
Figure IV.17 : Résultat de la vérification de cohérence des blocs.	65
Figure IV.18 : Migration vers TIA Portal.	66
Figure IV.19 : Migration en cours.....	67
Figure IV.20 : Fin de la migration sans erreurs.	67
Figure IV.21 : Résultat de la migration.....	68
Figure IV.22 : Table des variables sur TIA Portal.	68
Figure IV.23 : Configuration matériel sur TIA Portal.	69
Figure IV.24 : Chargement des blocs, mise en ligne et simulation.....	69

Liste des tableaux :

Tableau III.1 : Les positions du commutateur.....	38
Tableau III.2 : Mise en parallèle des blocs STEP 5 et blocs STEP 7.....	43
Tableau III.3 : Types de données dans S5 et S7.....	44
Tableau IV.1 : Liste des messages d'erreur, leurs significations et les mesures à prendre pour y remédier.....	51
Tableau IV.2 : Avertissements, signification et réaction conseillée.....	52

Introduction générale

Introduction générale

L'automatisation est la priorité absolue dans les industries modernes, dans un monde industriel où les performances demandées ne se limitent pas à l'augmentation de la productivité, l'amélioration de la qualité du produit et la diminution des coûts de production, mais aussi à l'amélioration des conditions de travail, la suppression des tâches pénibles et répétitives et surtout une sécurité plus élevée, l'automatisation s'est montrée être la solution pour assurer tout cela.

Les exigences attendues de l'automatisation ne cessent de croître, ce qui pousse les automates programmables industriels (API) à être en constante évolution. Afin de répondre à la demande, l'API fait partie intégrante de la boucle de réglage des procédés industriels. Il a pour tâche principale de récolter des informations à partir des capteurs via ses interfaces d'entrées, de traiter ces informations pour prendre une décision en fonction du programme implémenté dans sa mémoire et ainsi commander les actionneurs avec des signaux via ses interfaces de sorties.

L'unité Margarinerie de CEVITAL est un exemple d'automatisation des systèmes de production en Algérie. Le processus de fabrication de la margarine est entièrement automatisé, l'intervention humaine est réduite à la supervision et la surveillance des différents paramètres des machines qui assurent le bon fonctionnement de la chaîne de production et à la réaction en un délai minimal en cas de défaillance signalée par le système de gestion des alarmes.

Cevital veille en permanence à rester à jour au niveau technologique et à mettre à la disposition de ses clients un produit compétitif en termes de qualité et de coût pour faire face à la concurrence du marché agro-alimentaire.

C'est pour cette raison que ses techniciens sont formés et orientés pour assurer une amélioration continue de tous les équipements faisant partie du processus de fabrication de margarine en suivant l'évolution technologique. C'est dans cette optique que s'inscrit le projet de migration S5 vers S7 qui a été lancé par Cevital afin d'accroître le rendement de ses équipements grâce à la supériorité en puissance des automates S7 par rapport aux automates S5 d'un côté, et d'éviter la non disponibilité de pièces de rechange d'un autre côté, étant donné que la firme SIEMENS a arrêté la fabrication des automates programmables S5 en fin d'année 2015.

Dans notre travail nous nous sommes intéressés au descenseur de pots (HAMBA BK8006-M) de la margarinerie équipé d'un automate programmable S5-100U qui cause actuellement des difficultés pour le service de maintenance, qui doit donc être remplacé par un automate S7-300.

A cet effet, le présent mémoire est structuré en quatre chapitres décrivant les volets principaux.

Le premier chapitre est dédié aux automates programmables ainsi qu'au logiciel d'ingénierie de SIEMENS qui est le TIA portal V15.

Introduction générale

Le deuxième chapitre est consacré à l'étude et la présentation du descenseur de pots HAMBА BK8006-M.

Le chapitre trois englobe une étude fonctionnelle en vue de l'automatisation.

Le dernier chapitre traite la partie conversion et migration du projet.

Description du complexe CEVITAL

Description du complexe CEVITAL

1. Présentation générale de l'entreprise

CEVITAL c'est un ensemble industriel intégré, concentré en première partie dans le secteur de l'agroalimentaire : raffinage d'huile et de sucre, produits dérivés, négoce de céréales, distribution de produits destinés à l'alimentation humaine et animale. Elle conçoit des produits de qualité supérieure à des prix compétitifs, grâce à ses installations performantes, son savoir-faire, son contrôle strict de qualité et son réseau de distribution. Elle couvre les besoins nationaux et a permis de faire passer l'Algérie du stade d'importateur à celui d'exportateur pour les huiles, les margarines et le sucre. Ses produits se vendent dans plusieurs pays, notamment en Europe, au Maghreb, au Moyen Orient et en Afrique de l'Ouest [1].

2. Historique

Fondé par Mr.Isaad Rebrab, CEVITAL est un groupe familial de plusieurs sociétés bâti sur une histoire, créé par des fonds privés en 1998 à Bejaïa, à l'entrée du pays dans l'économie de marché. Première entreprise privée algérienne à avoir investi dans des secteurs d'activités diversifiés, elle a traversé d'importantes étapes historiques pour atteindre sa taille et sa notoriété actuelle [1].

3. Situation géographique

CEVITAL Agro-Industrie est implanté au niveau du nouveau quai port de BEJAIA à 3 Km du sud-ouest de cette ville, à proximité de la RN 09.

Cette situation géographique de l'entreprise lui a beaucoup profité étant donné qu'elle lui donne l'avantage de proximité économique. En effet, elle se trouve proche du port et de l'aéroport [1].

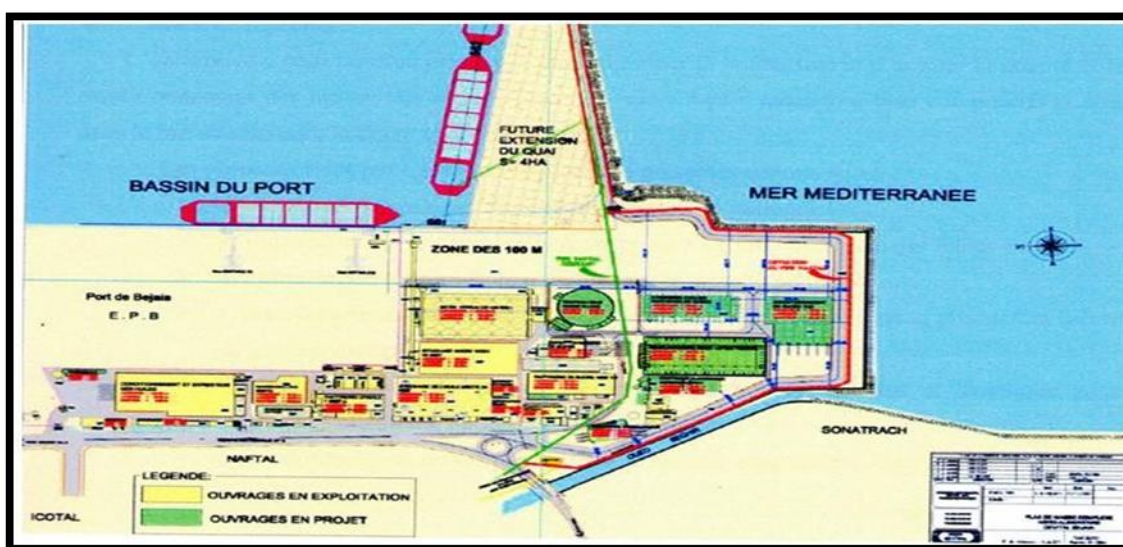


Figure 1 : Situation géographique du complexe Cevital.

4. Activités et unités du complexe CEVITAL

Lancé en mai 1998, le complexe CEVITAL a débuté son activité par le conditionnement d'huile en décembre 1998. En février 1999, les travaux de génie civil de la raffinerie ont débuté. Cette dernière est devenue fonctionnelle en août 1999.

L'ensemble des activités de CEVITAL est concentré sur la production et la commercialisation des huiles végétales, de margarine et de sucre et se présente comme suit :

- Raffinage des huiles (1800 tonnes/jour).
- Conditionnement d'huile (1400 tonnes/jour).
- Production de margarine (600 tonnes/jour).
- Fabrication d'emballage (PET) ; Poly-Ethylène-Téréphtalate (9600 unités/heure).
- Raffinage de sucre (1600 tonnes/jour).
- Stockage des céréales (120000 tonnes/jour).
- Minoterie et savonnerie en cours d'étude.

Elle possède également des silos portuaires ainsi qu'un terminal de déchargement portuaire d'une capacité de 2000 tonnes/heure ce qui en fait le premier terminal de déchargement portuaire en Méditerranée [1].

5. Missions et objectifs

L'entreprise a pour mission principale de développer la production et d'assurer la qualité et le conditionnement des huiles, des margarines et du sucre à des prix nettement plus compétitifs et cela dans le but de satisfaire le client et le fidéliser.

Les objectifs visés par CEVITAL peuvent se présenter comme suit :

- L'extension de ses produits sur le territoire national.
- L'importation de graines oléagineuse pour l'extraction directe des huiles brutes.
- L'optimisation de ses offres d'emploi sur le marché du travail.
- L'encouragement des agriculteurs par des aides financières pour la production locale des graines oléagineuses.
- La modernisation de ses installations en termes de machine et technique pour augmenter le volume de sa production.
- Le positionnement de ses produits sur le marché étranger par leurs exportations [1].

6. Différents organes constituant le complexe CEVITAL

L'organigramme suivant donne une vue générale sur les différents organes constituant le complexe CEVITAL :

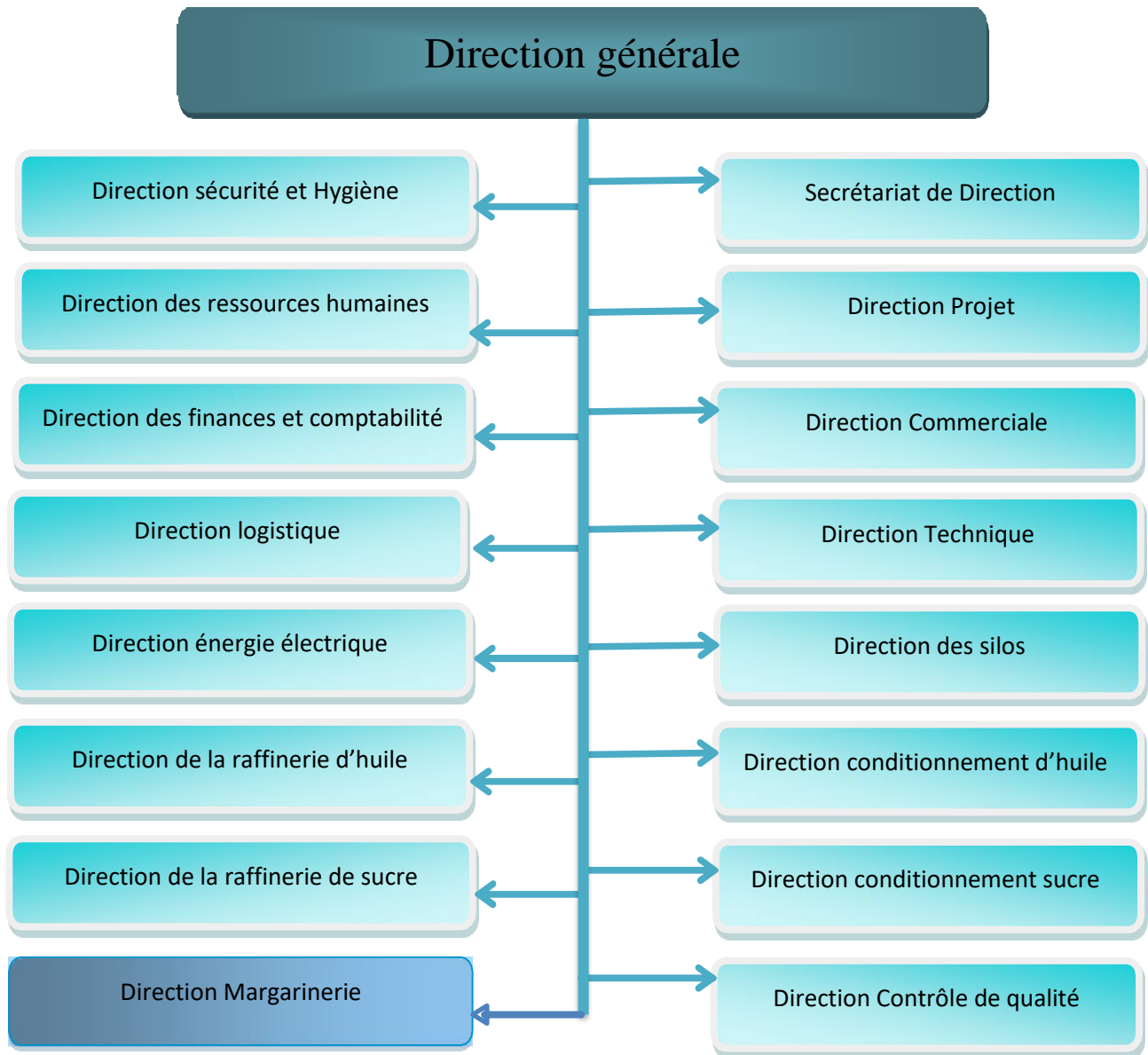


Figure 2 : Organigramme du complexe Cevital.

7. Unité de Production de margarine

Cette unité est spécialisée dans la production de la margarine et les graisses finies ou semi finies. Elle a une capacité de production de 600 tonnes/jour. Elle produit essentiellement MATINA, FLEURIAL, Elio2, Margarine de Feuilletage, SMEN et les Shortening.

L'atelier de production de l'unité margarinerie possède cinq (05) lignes de production. Chaque ligne est composée d'un certain ensemble de ressources connectées entre elles. Chacune est assignée à produire un (des) article (s) bien spécifique.

- La ligne (L1) pour la production des barquettes de 400 g de Mâtina, barquettes de 500 grammes Elio2, barquette de 250 g/500 g Fleurial.
- La ligne (L2) pour la production des barquettes de 500 g de Smen, plaquettes de 500 grammes de Feuilletage.
- La ligne (L3) pour la production des plaquettes de 500 g de Feuilletage.
- La ligne (L4) pour la production des barquettes de 1.8 kg de Smen.
- La ligne (L5) pour la production des cartons de 20 kg de Shortening des points de fusion (31- 32, 35-37 et 38-40).

CHAPITRE I

Automates programmables et logiciels associés

Chapitre I : Automates programmables et logiciels associés

I.1 Introduction

Un système est dit automatisé lorsque le processus qui permet de passer d'une situation initiale à une situation finale se fait sans intervention humaine, et que ce comportement est répétitif chaque fois que les conditions qui caractérisent la situation initiale sont remplies. L'automatisation conduit à une très grande rapidité, une meilleure régularité des résultats et évite à l'homme des tâches pénibles et répétitives.

Ce chapitre sera consacré à la description des automates programmables SIEMENS à structures modulaire et du logiciel associé TIA PORTAL V15.

I.2 L'automatisme

L'automatisme est l'art de donner vie à un ensemble de composants. Chaque composant de la machine sera sollicité au moment approprié, en fonction :

- Des ordres donnés par l'opérateur (consignes),
- Du système de traitement choisi pour l'automatisme (relayage, programme),
- Des informations que la machine renvoie au système de traitement.

I.2.1 Objectif de l'automatisation

Mise à part les objectifs à caractères financiers on trouve : [2]

- Eliminer les tâches répétitives de l'humain.
- Simplifier le travail de l'humain.
- Augmenter la sécurité.
- Accroître la productivité.
- Economiser les matières premières et l'énergie.
- S'adapter à des contextes particuliers.
- Maintenir la qualité.

I.2.2 Les systèmes automatisés

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

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

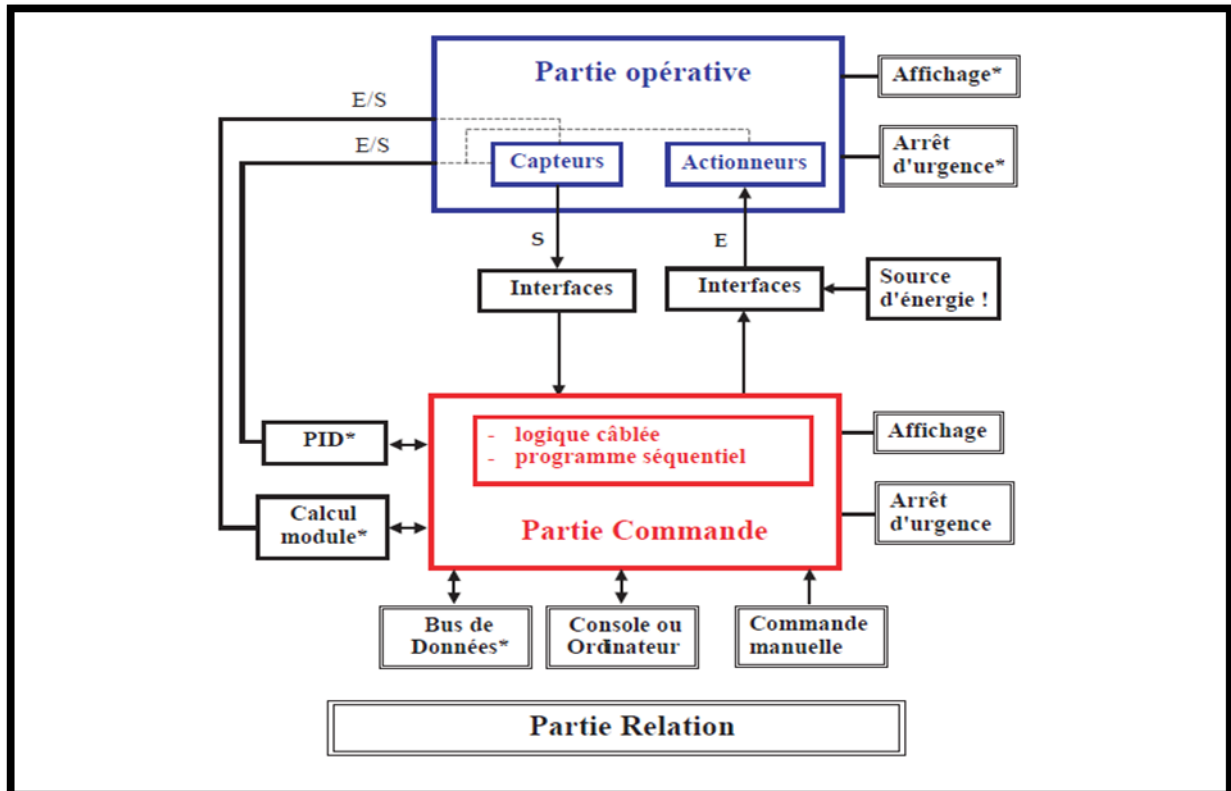


Figure I.1 : Structure d'un système automatisé. [3]

I.2.3.1 La partie commande (PC)

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

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

I.2.3.2 La partie opérative (PO)

Elle opère sur la matière d'œuvre et le produit. Elle regroupe :

- **Les effecteurs** : dispositifs terminaux qui agissent directement sur la matière d'œuvre pour lui donner sa valeur ajoutée (outils de coupe, pompes, têtes de soudure, etc.).
- **Les actionneurs** : éléments chargés de convertir l'énergie afin de l'adapter au besoin de la partie opérative ; cette énergie étant ensuite consommée par les effecteurs (moteur, vérin, électroaimant, résistance de chauffage, etc.).
- **Les préactionneurs** : éléments chargés :
 - D'adapter le faible niveau énergétique disponible en sortie de la P.C. au besoin de la P.O.
 - De distribuer ou de moduler l'énergie délivrée aux actionneurs (contacteur, distributeur, variateur de vitesse).
- **Les capteurs** : qui assument l'ensemble des fonctions de la chaîne d'acquisition de données (fin de course de vérin, détecteur de position, capteur de température, etc.).

I.2.3.3 La partie relation (PR)

Elle comporte le pupitre de dialogue homme-machine équipé des organes de commande permettant la mise en/hors énergie de l'installation, la sélection des modes de marche, la commande manuelle des actionneurs, la mise en référence, le départ des cycles, l'arrêt d'urgence... ainsi que des signalisations diverses telles que voyants lumineux, afficheurs, écrans vidéo, Klaxons, sonneries, etc.

I.3 Généralités sur les automates programmables

I.3.1 Historique

Les Automates Programmables Industriels (API) sont apparus aux Etats-Unis vers la fin des années soixante, à la demande de l'industrie automobile américaine (General Motors) qui réclamait plus d'adaptabilité de leurs systèmes de commande.

Les ingénieurs américains ont résolu le problème en créant un nouveau type de produit nommée automates programmables. Ils n'étaient rentables que pour des installations d'une certaine complexité, mais la situation a très vite changé, ce qui a rendu les systèmes câblés obsolètes. De nombreux modèles d'automates sont aujourd'hui disponibles ; depuis les nano automates bien adaptées aux machines et aux installations simples avec un petit nombre d'entrées/sorties, jusqu'aux automates multifonctions capables de gérer plusieurs milliers d'entrées/sorties et destinés au pilotage de processus complexes. [5]

I.3.2 Définition de l'automate programmable

Un automate programmable industriel (API) est une machine électronique spécialisée dans la conduite et la surveillance en temps réel de processus industriels et tertiaires. Il exécute une suite d'instructions introduites dans ses mémoires sous forme de programmes, et s'apparente par conséquent aux machines de traitement de l'information.

Trois caractéristiques fondamentales le distinguent des outils informatiques tels que les ordinateurs utilisés dans les entreprises et le tertiaire :

- Il peut être directement connecté aux capteurs et pré-actionneurs grâce à ses entrées/sorties industrielles.
- Il est conçu pour fonctionner dans des ambiances industrielles sévères (Température, vibrations, microcoupures de la tension d'alimentation, parasites, etc...).

Enfin, sa programmation à partir de langages spécialement développés pour le traitement de fonctions d'automatisme facilite son exploitation et sa mise en œuvre. [4]

I.3.3 Critères de choix d'un automate

Le choix d'un automate programmable est en premier lieu le choix d'une société, d'un groupe de contacts commerciaux et d'expériences vécues qui sont déjà un point de départ, les grandes sociétés privilégieront deux fabricants pour faire jouer la concurrence et pouvoir se retourner en cas de perte de vitesse de l'une d'entre elles. Le personnel de maintenance doit toutefois être formé sur ces matériels et une trop grande diversité des matériels peut avoir de graves répercussions.

Un automate utilisant des langages de programmation de type GRAFCET est également préférable pour assurer les mises au point et dépannages dans les meilleures conditions.

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

Il faut ensuite quantifier les besoins :

- **Nombre d'entrées / sorties** : le nombre de cartes peut avoir une incidence sur le nombre de racks dès que le nombre d'entrées / sorties nécessaires devient élevé.
- **Type de processeur** : la taille mémoire, la vitesse de traitement et les fonctions spéciales offertes par le processeur permettront le choix dans la gamme souvent très étendue.
- **Fonctions ou modules spéciaux** : certaines cartes (commande d'axe, pesage ...) permettront de soulager le processeur et devront offrir les caractéristiques souhaitées (résolution, ...).
- **Fonctions de communication** : l'automate doit pouvoir communiquer avec les autres systèmes de commande (API, supervision ...) et offrir des possibilités de communication avec des standards normalisés (Profibus ...). [4]

I.3.4 Avantages des automates programmables

- Evolutivité : très favorable à l'évolution. Très utilisé en reconstruction d'armoire.
- Fonctions : assure les fonctions conduites, dialogue, communication et sûreté.
- Taille des applications : gamme importante d'automate.
- Vitesse : temps de cycle de quelque ms.
- Modularité : haute modularité, présentation modularité en rack
- Architecture de commande : centralisée ou décentralisée avec l'apparition d'une offre importante en choix de réseaux, bus de terrain, blocs E/S déportées.
- Maintenance : échanges standard et aide au diagnostic intégré.

I.3.5 Traitement du programme automate

Tous les automates fonctionnent selon le même mode opératoire :

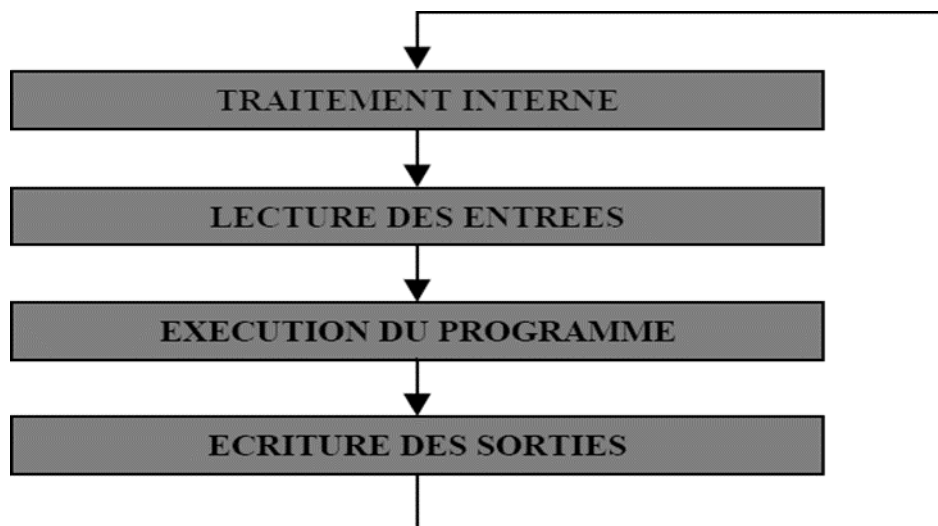


Figure I.2 : Fonctionnement cyclique d'un automate. [5]

- **Traitement interne** : L'automate effectue des opérations de contrôle et met à jour certains paramètres systèmes (détection des passages en RUN / STOP, mises à jour des valeurs de l'horodateur, ...).
- **Lecture des entrées** : L'automate lit les entrées (de façon synchrone) et les recopie dans la mémoire image des entrées.
- **Exécution du programme** : L'automate exécute le programme instruction par instruction et écrit les sorties dans la mémoire image des sorties.
- **Ecriture des sorties** : L'automate bascule les différentes sorties (de façon synchrone) aux positions définies dans la mémoire image des sorties.

Ces quatre opérations sont effectuées continuellement par l'automate (fonctionnement cyclique). [5]

I.3.6 Langage de programmation pour API

Les programmes utilisés avec les API peuvent être écrits dans différents formats.

Pour que les ingénieurs du domaine électrique ayant peu de connaissances en programmation puissent élaborer des programmes pour les API, le langage à contacts a été conçu. La plupart des fabricants d'automates ont adopté cette méthode d'écriture des programmes.

Toutefois, puisque chacun a eu tendance à développer ses propres versions, une norme internationale a été établie pour le langage à contacts et, par voie de conséquence, pour toutes les méthodes de programmation employées avec les API. [4]

La norme IEC 1131-3 définit cinq langages qui peuvent être utilisés pour la programmation des automates programmables industriels. Ces cinq langages sont :

I.3.6.1 Le langage LD (Ladder Diagram)

C'est une représentation graphique d'équations booléennes combinant des contacts (en entrée) et des relais (en sortie). Il permet la manipulation de données booléennes, à l'aide de symboles graphiques organisés dans un diagramme comme les éléments d'un schéma électrique à contacts. Les diagrammes LD sont limités à gauche et à droite par des barres d'alimentation.

[4]

I.3.6.2 Le langage IL (Instruction List)

Un langage textuel de bas niveau. Il est particulièrement adapté aux applications de petite taille. Les instructions opèrent toujours sur un résultat courant (ou registre IL). L'opérateur indique le type d'opération à effectuer entre le résultat courant et l'opérande. Le résultat de l'opération est stocké à son tour dans le résultat courant.

Un programme IL est une liste d'instructions. Chaque instruction doit commencer par une nouvelle ligne, et doit contenir un opérateur, complété éventuellement par des modificateurs et, si c'est nécessaire pour l'opération, un ou plusieurs opérandes, séparés par des virgules (','). Une étiquette suivie de deux points (': ') peut précéder l'instruction. Si un commentaire est attaché à l'instruction, il doit être le dernier élément de la ligne. Des lignes vides peuvent être insérées entre des instructions. Un commentaire peut être posé sur une ligne sans instruction [4].

I.3.6.3 Le langage FBD (Function Block Diagram)

C'est un langage graphique. Il permet la construction d'équations complexes à partir des opérateurs standards, de fonctions ou de blocs fonctionnels.

Les principales fonctions sont :

- L'énoncé RETURN (peut apparaître comme une sortie du diagramme, si liaison connectée prend l'état booléen TRUE, la fin du diagramme n'est pas interprétée.

- Les étiquettes et les sauts conditionnels sont utilisés pour contrôler l'exécution du diagramme. Aucune connexion ne peut être réalisée à droite d'un symbole d'étiquette ou de saut.
- Saut à une étiquette (le nom de l'étiquette est « LAB »).[4]

I.3.6.4 Le langage ST (Structured Text)

Un langage textuel de haut niveau dédié aux applications d'automatisation. Ce langage est principalement utilisé pour décrire les procédures complexes, difficilement modélisables avec les langages graphiques. C'est le langage par défaut pour la programmation des actions dans les étapes et des conditions associées aux transitions du langage SFC.

Un programme ST est une suite de lignes. Chaque ligne est terminée par un point-virgule (« ; »).

Les noms utilisés dans le code source (identificateurs de variables, constantes, mots clés du langage...) sont délimités par des séparateurs passifs ou des séparateurs actifs, qui ont un rôle d'opérateur. Des commentaires peuvent être librement insérés dans la programmation.[4]

I.3.6.5 Le langage SFC (Sequential Function Chart), ou GRAFCET

C'est un langage graphique utilisé pour décrire les opérations séquentielles.

Le procédé est représenté comme une suite connue d'étapes (états stables), reliées entre elles par des transitions, une condition booléenne est attachée à chaque transition. Les actions dans les étapes sont décrites avec les langages ST, IL, LD ou FBD.

Les principales règles graphiques sont :

- Un programme SFC doit contenir au moins une étape initiale.
- Une étape ne peut pas être suivie d'une autre étape.
- Une transition ne peut pas être suivie d'une autre transition. [4]

I.3.7 Architecture des automates programmables

De forme compacte ou modulaire, les automates sont organisés suivant l'architecture suivante :

- **Une unité centrale (CPU) :** assure le traitement de l'information et la gestion de l'ensemble des unités. Ce module comporte un microprocesseur, des circuits périphériques de gestion des entrées/sorties, des mémoires RAM et EEPROM nécessaires pour stocker les programmes, les données, et les paramètres de configuration du système.
- **Une alimentation (PS) :** à partir d'une tension 220V/50Hz ou dans certains cas de 24V fournit les tensions continues + /- 5V, +/-12V ou +/-15V.

➤ **Modules d'entrées et de sorties :**

- Un ou plusieurs modules d'entrées 'Tout Ou Rien' (TOR) ou analogiques pour l'acquisition des informations provenant de la partie opérative (procédé à conduire).
- Un ou plusieurs modules de sorties 'Tout Ou Rien' (TOR) ou analogiques pour transmettre à la partie opérative les signaux de commande. Il y a des modules qui intègrent en même temps des entrées et des sorties.

➤ **Module de fonction (FM) :** Ils réalisent les tâches de traitement des signaux de processus critique au niveau de temps et exigeant beaucoup de mémoire pour réduire la charge sur la CPU. Par exemple :

- Le positionnement.
- Régulation.
- Comptage.

➤ **Module de communication (CP) :** Il se charge des tâches de communication par transmission série. Et aussi l'établissement des liaisons point à point avec d'autres automates SIMATIC ou d'autres constructeurs.

- Interfaces série utilisant dans la plupart des cas comme support de communication, les liaisons RS-232 ou RS422/RS485.
- Interfaces pour assurer l'accès à un bus de terrain.
- Interface d'accès à un réseau Ethernet.

➤ **Modules de simulation (SM 374) :** c'est un module spécial qui offre à l'utilisateur la possibilité de tester son programme lors de la mise en service et en cours de fonctionnement, ce module fonctionne telles que :

- La simulation des signaux de capteurs aux moyens d'interrupteurs.
- La signalisation d'état des signaux de sortie par des LEDs.

➤ **Coupleurs :** Si l'application d'automatisation a besoins de plus de huit modules, il est possible de faire une extension de la configuration en utilisant un châssis de base et trois châssis d'extension au maximum. Chaque châssis peut recevoir huit modules, alors le coupleur relie les différents châssis. [5]

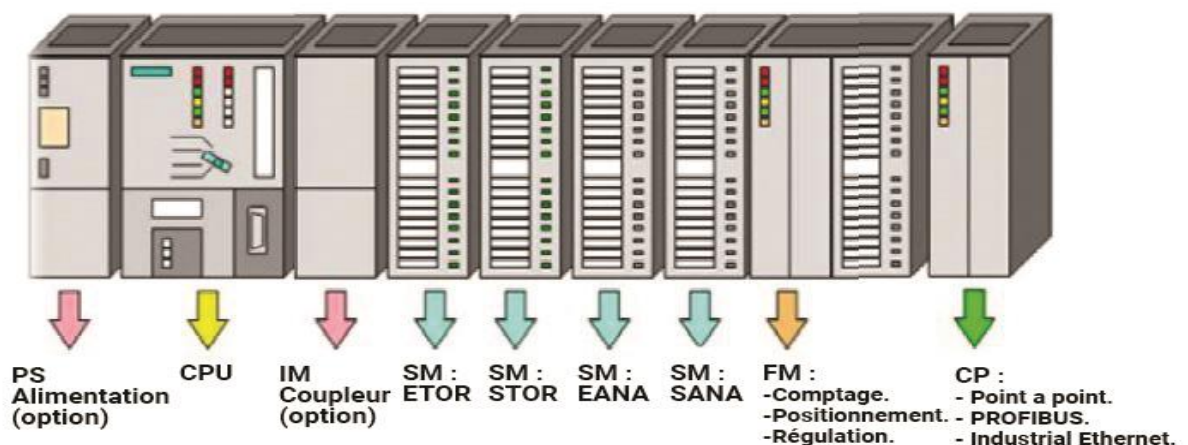


Figure I.3 : Automate Programmable Industriel SIEMENS [6].

I.3.8 Structure interne des automates programmables

La structure matérielle interne d'un API obéit au schéma ci-dessous :

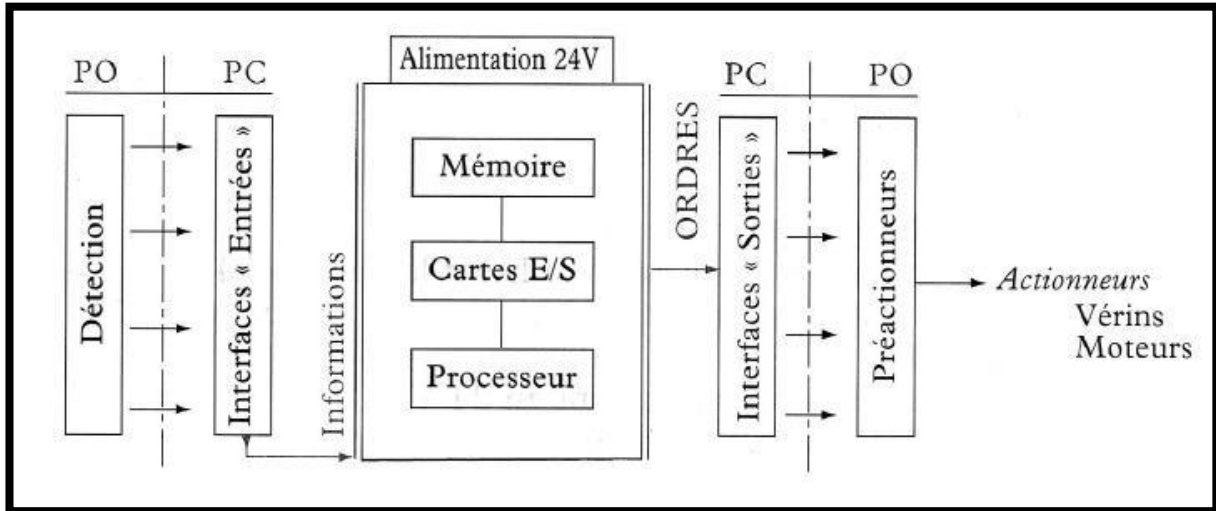


Figure I.4 : Structure interne d'un API. [4]

- Détaillons successivement chacun des composants qui apparaissent sur ce schéma : [4]

I.3.8.1 Le processeur :

Le processeur a pour rôle principal le traitement des instructions qui constituent le programme de fonctionnement de l'application. Mais en dehors de cette tâche de base, il réalise également d'autres fonctions :

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

Le processeur est organisé autour d'un certain nombre de registres, ce sont des mémoires rapides permettant la manipulation des informations qu'elles retiennent, ou leur combinaison avec des informations extérieures.

Les principaux registres existants dans un processeur sont :

- **L'accumulateur** : C'est le registre où s'effectuent les opérations du jeu d'instruction, les résultats sont contenus dans ce registre spécial.

- **Le registre d'instruction** : Il reçoit l'instruction à exécuter et décode le code opération. Cette instruction est désignée par le pointeur.
- **Le registre d'adresse** : Ce registre reçoit, parallèlement au registre d'instruction, la partie opérande de l'instruction. Il désigne le chemin par lequel circulera l'information lorsque le registre d'instruction validera le sens et ordonnera le transfert.
- **Le registre d'état** : C'est un ensemble de positions binaires décrivant, à chaque instant, la situation dans laquelle se trouve précisément la machine.
- **Les piles** : Une organisation spéciale de registres constitue une pile, ces mémoires sont utilisées pour contenir le résultat de chaque instruction après son exécution. Ce résultat sera utilisé ensuite par d'autres instructions, et cela pour faire place à la nouvelle information dans l'accumulateur.

I.3.8.2 Les mémoires

Un système de processeur est accompagné par un ou plusieurs types de mémoires. Elles permettent :

- De stocker le système d'exploitation dans des ROM ou PROM,
- Le programme dans des EEPROM,
- Les données système lors du fonctionnement dans des RAM. Cette dernière est généralement secourue par pile ou batterie. On peut, en règle générale, augmenter la capacité mémoire par adjonction de barrettes mémoires type PCMCIA.

I.3.8.3 Les modules d'entrées/sorties

Ils assurent le rôle d'interface entre la CPU et le processus, en récupérant les informations sur l'état de ce dernier et en coordonnant les actions.

Plusieurs types de modules sont disponibles sur le marché selon l'utilisation souhaitée :

- Modules TOR (Tout Ou Rien) : l'information traitée ne peut prendre que deux états (vrai/faux, 0 ou 1 ...). C'est le type d'information délivrée par une cellule photoélectrique, un bouton poussoir ...etc.
- Modules analogiques : l'information traitée est continue et prend une valeur qui évolue dans une plage bien déterminée. C'est le type d'information délivrée par un capteur (débitmètre, capteur de niveau, thermomètre...etc.).
- Modules spécialisés : l'information traitée est contenue dans des mots codes sous forme binaire ou bien hexadécimale. C'est le type d'information délivrée par un ordinateur ou un module intelligent.

I.3.8.4 L'alimentation

Elle a pour rôle de fournir les tensions continues nécessaires aux composants avec de bonnes performances, notamment face aux microcoupures du réseau électrique qui constitue la source d'énergie principale. La tension d'alimentation peut être de 5V, 12V ou 24V.

D'autres alimentations peuvent être nécessaires pour les châssis d'extension et pour les modules entrées/sorties. Un onduleur est nécessaire pour éviter les risques de coupures non tolérées.

I.3.8.5 Liaisons de communication

Elles Permettent la communication de l'ensemble des blocs de l'automate et des éventuelles extensions. Les liaisons s'effectuent :

- Avec l'extérieur par des borniers sur lesquels arrivent des câbles transportant les signaux électriques.
- Avec l'intérieur par des bus reliant divers éléments, afin d'échanger des données, des états et des adresses.

I.4 TIA Portal (Totally Integrated Automation)

I.4.1 Introduction

En réponse à la pression internationale croissante de la concurrence, il est aujourd'hui plus que jamais important d'exploiter à fond tous les potentiels d'optimisation sur l'ensemble du cycle de vie d'une machine ou d'une installation.

Des processus optimisés permettent de réduire le coût total de possession, de réduire le temps entre la conception et la commercialisation et d'améliorer la qualité. Cet équilibre parfait entre qualité, temps et coûts et plus que jamais le facteur décisif de la réussite industrielle.

Totally Integrated Automation apporte une réponse optimale à toutes les exigences et offre un concept ouvert vis à vis des normes internationales et de systèmes tiers. Avec ses six principaux caractéristiques systèmes et robustesse, Le TIA Portal accompagne l'ensemble du cycle de vie d'une machine ou d'une installation. L'architecture système complète offre des solutions complètes pour chaque segment d'automatisation sur la base d'une gamme de produits complète [7].

I.4.2 Description du logiciel TIA Portal

La plateforme « Totally Intergrated Automation Portal » est le nouvel environnement de travail Siemens qui permet de mettre en œuvre des solutions d'automatisation avec un système d'ingénierie intégré comprenant les logiciels SIMATIC Step7 et SIMATIC WinnCC. [13]

I.4.3 Les avantages du logiciel TIA portal

- Programmation intuitive et rapide : avec des éditeurs de programmation nouvellement développés SCL, CONT, LOG, LIST et GRAPH.
- Efficacité accrue grâce aux innovations linguistiques de STEP 7 : programmation symbolique uniforme, Calculate Box, ajout de blocs durant le fonctionnement, et bien plus encore.
- Performance augmentée grâce à des fonctions intégrées : simulation avec PLCSIM, télémaintenance avec TeleService et diagnostic système cohérent.
- Technologie flexible : Fonctionnalité motion control évolutive et efficace pour les automates S7-1500 et S7-1200.
- Sécurité accrue avec Security Integrated : Protection du savoir-faire, protection contre la copie, protection d'accès et protection contre la falsification.
- Environnement de configuration commun avec pupitres IHM et entraînements dans l'environnement d'ingénierie TIA Portal. [13]

I.4.4 SIMATIC STEP 7

SIMATIC STEP 7, intégré à TIA Portal, est le logiciel de configuration, programmation, vérification et diagnostic de tous les automates SIMATIC. Doté d'un grand nombre de fonctions conviviales, SIMATIC STEP 7 garantit une efficacité nettement supérieure pour toutes les tâches d'automatisation, qu'il s'agisse de la programmation, de la simulation, de la mise en service ou de la maintenance. [9]

I.4.5 Vue du portail et vue du projet

Lorsqu'on lance TIA Portal, l'environnement de travail se décompose de deux types de vue :

a) Vue du portail

Elle est axée sur les tâches à exécuter et sa prise en main est très rapide.

Chaque portail permet de traiter une catégorie de tâche (action) la fenêtre affiche la liste la liste des actions peuvent être réalisées pour la tâche sélectionnée.

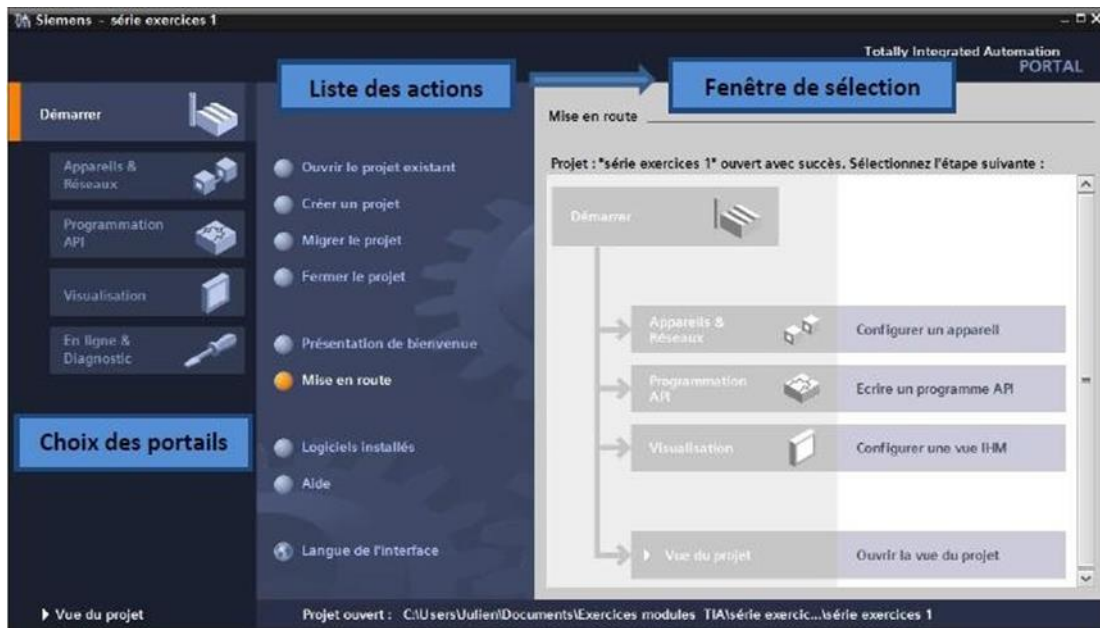


Figure I.5 : Vue du portail.

b) Vue du projet

Elle comporte une arborescence avec les différents éléments du projet, les éditeurs requis s'ouvrent en fonction des tâches à réaliser. Données, paramètres et éditeurs peuvent être visualisés dans une seule et même vue. [7]

L'élément « Projet » contient l'ensemble des éléments et des données nécessaires pour mettre en œuvre la solution d'automatisation souhaitée.

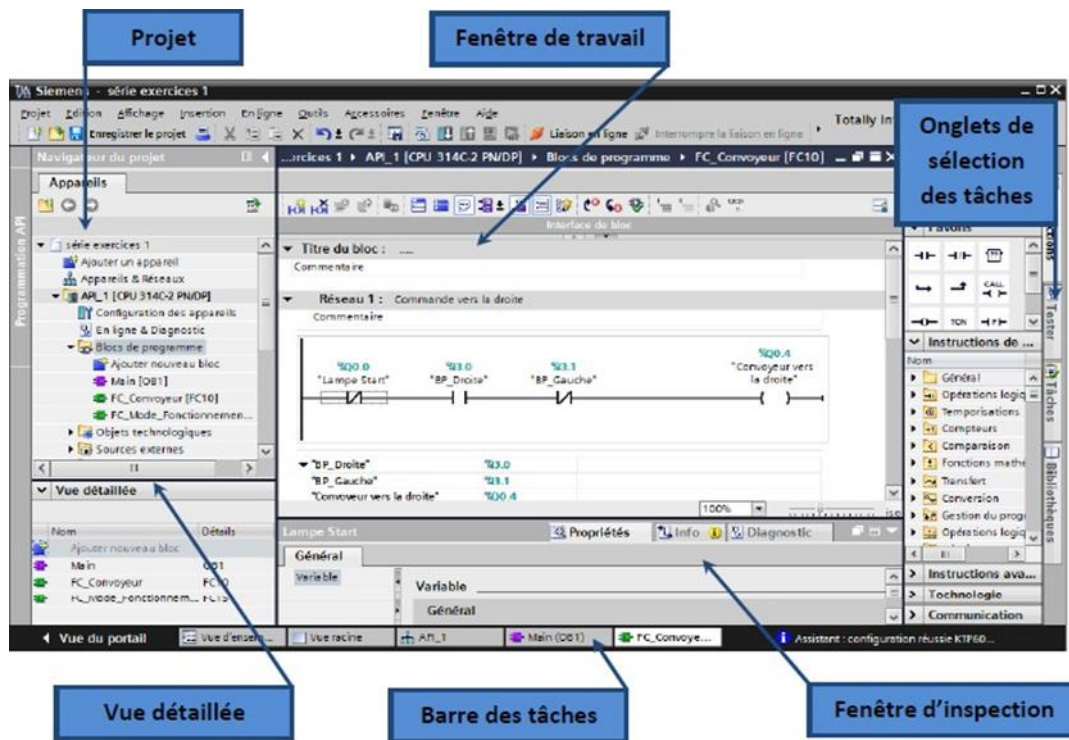


Figure I.6 : Vue du projet.

- **La fenêtre de travail** permet de visualiser les objets sélectionnés dans le projet pour être traités. Il peut s'agir des composants matériels, des blocs de programme, des tables des variables, des HMI, ...
- **La fenêtre d'inspection** permet de visualiser des informations complémentaires sur un objet sélectionné ou sur les actions en cours d'exécution (propriété du matériel sélectionné, messages d'erreurs lors de la compilation des blocs de programme, ...).
- **Les onglets de sélection de tâches** ont un contenu qui varie en fonction de l'objet sélectionné (configuration matérielle, bibliothèques des composants, bloc de programme, instructions de programmation).

Cet environnement de travail contient énormément de données. Il est possible de masquer ou réduire certaines de ces fenêtres lorsque l'on ne les utilise pas.

Il est également possible de redimensionner, réorganiser, désancrer les différentes fenêtres. [13]

I.4.6 Adressage des E/S

Pour connaître l'adressage des entrées et sorties présentes dans la configuration matérielle, il faut aller dans « Appareil et réseau » dans le navigateur du projet dans la fenêtre de travail, on doit s'assurer d'être dans l'onglet « Vue des appareils » et de sélectionner l'appareil voulu.

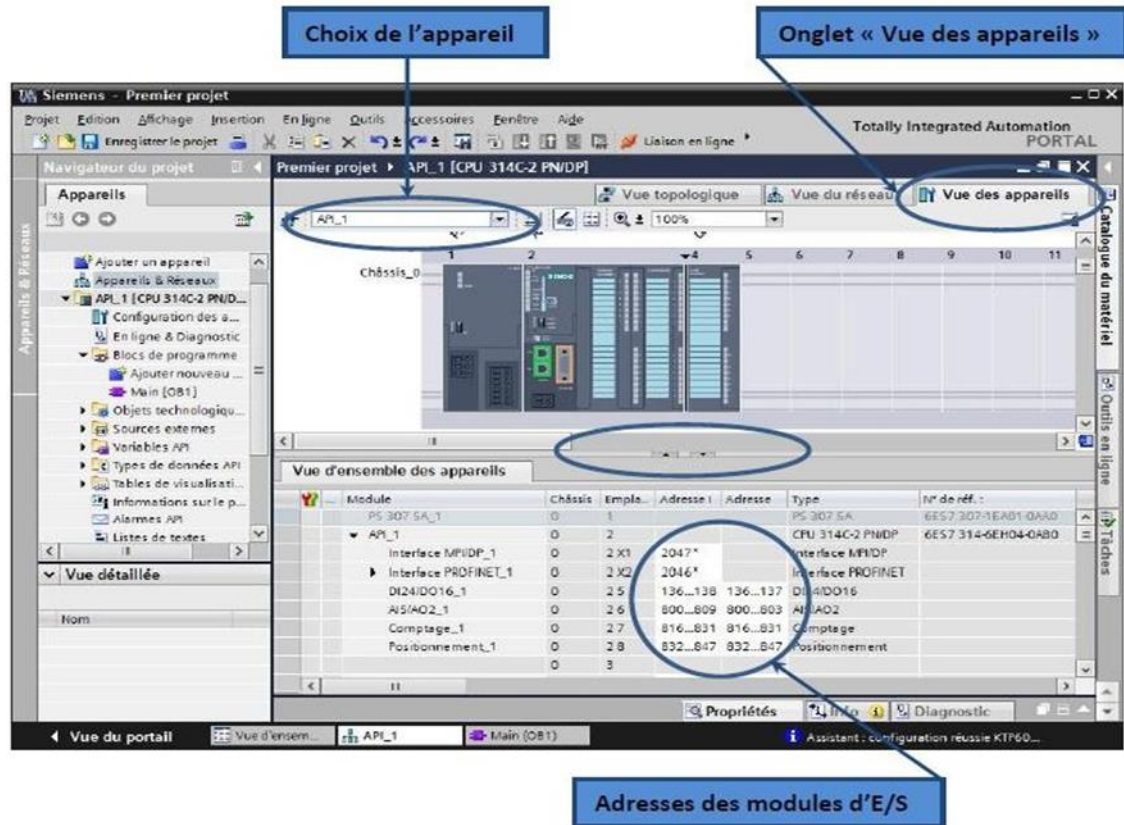


Figure I.7 : Adressage des E/S.

On sélectionne la CPU puis à l'aide des deux petites flèches (voir figure), on fait apparaître l'onglet « Vue d'ensembles des appareils ». Les adresses des entrées et sorties apparaissent. On peut les modifier en entrant une nouvelle valeur dans la case correspondante.

I.4.7 Les variables API

I.4.7.1 Adresses symboliques et absolue

Dans TIA portal, toutes les variables globales (entrées, sorties, mémentos, ...) possèdent une adresse symbolique et une adresse absolue.

L'adresse absolue représente l'identificateur d'opérande (I, Q, M, ...) et son adresse et numéro de bit.

Adresse symbolique correspond au nom que l'utilisateur a donné à la variable (ex : bouton marche). Le lien entre les adresses symbolique et absolue se fait dans la table des variables API.

Lors de la programmation, on peut choisir d'afficher les adresses absolues, symboliques ou encore les deux simultanément.

I.4.7.2 Table des variables API

C'est dans la table des variables API que l'on va pouvoir déclarer toutes les variables et les constantes utilisées dans le programme. Lorsque l'on définit une variable API, il faut définir :

- Un nom : c'est l'adressage symbolique de la variable.
- Le type de donnée : BOOL, INT, ...
- L'adresse absolue : par exemple Q 1.5

On peut également insérer un commentaire qui nous renseigne sur cette variable. Le commentaire peut être visible dans chaque réseau utilisant cette variable.

I.4.8 Liaison avec l'automate

Il faut maintenant charger la configuration de l'automate dans celui-ci. Pour cela, il faut tout d'abord connecter l'automate au PC en utilisant l'interface SIMATIC S7 PC USB adapté. Ensuite, après avoir sélectionné la vue « En ligne et diagnostique », sélectionnez les options suivantes :

- Mode : MPI
- Interface PG /PC : pc Adapter

I.4.9 Adresse Ethernet de la CPU

Toujours dans les propriétés de la CPU, il est possible de définir son adresse Ethernet. Un double clic sur l'icône Ethernet de la station fait apparaître la fenêtre d'inspection permettant de définir ses propriétés.

Pour établir une liaison entre la CPU et la console de programmation, il faut affecter aux deux appareils des adresses appartenant au même sous réseau

I.4.10 WinCC sur TIA portal

WinCC (TIA portal) est un logiciel d'ingénierie pour la configuration de pupitres SIMATIC, de PC industriel SIMATIC et de PC standard par le logiciel de visualisation. Le SIMATIC WinCC dans le TIA portal fait partie d'un nouveau concept d'ingénierie intégré qui offre un environnement d'ingénierie homogène pour la programmation et la configuration de solution de commande, de visualisation d'entraînement, c'est le logiciel pour toutes les applications IHM allant de solutions de commande simples avec basic panels aux applications SCADA pour système multipostes basé sur PC. [13]

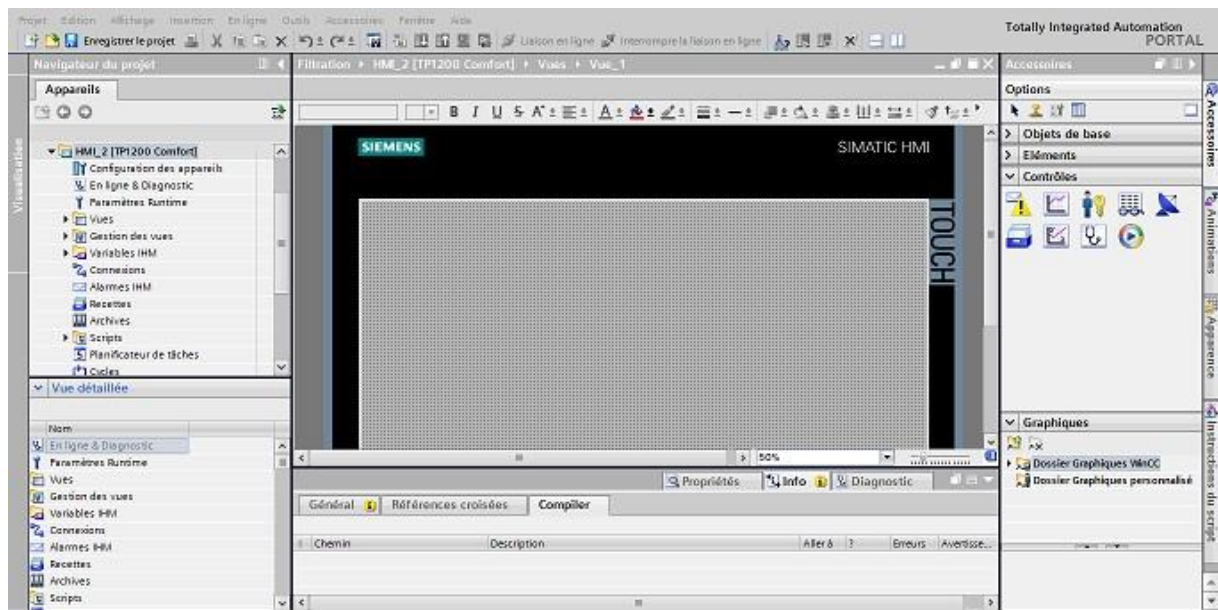


Figure I.8: Vue SIMATIC HMI.

I.5 Conclusion

L'automatisation d'une unité via le logiciel « TIA PORTAL V 15 », a pour but d'intégrer un nouveau programme sous l'automate programmable industriel, pour augmenter les performances, améliorer la sécurité de l'opérateur, éliminer l'effort physique, augmenter la précision et la rapidité de la tâche réalisée, et minimiser l'erreur.

CHAPITRE II

**Etude et présentation du descenseur de
pots HAMBА BK 8006-M**

Chapitre II : Etude et présentation du descenseur de pots HAMBА BK 8006-M

II.1 Introduction

Vu l'importance de l'industrie agroalimentaire, le groupe CEVITAL a toujours opté pour des technologies efficaces et fiables pour assurer le bon fonctionnement de leurs chaînes de production en toutes circonstances.

La ligne 1 de la margarinerie en est le parfait exemple équipé de la remplisseuse HAMBА BK 8006-M, une machine sophistiquée qui assure tout le processus de remplissage des barquettes de margarine d'une façon autonome, celle-ci est composée de plusieurs machines (stations) juxtaposés l'une à l'autre sur une longueur de 6,61 m et une hauteur de 8,16 m comme le montre la figure II.1.

Lors de notre travail nous nous intéresserons au descenseur (station de pots) qui s'occupe de l'approvisionnement en pots de la machine HAMBА.

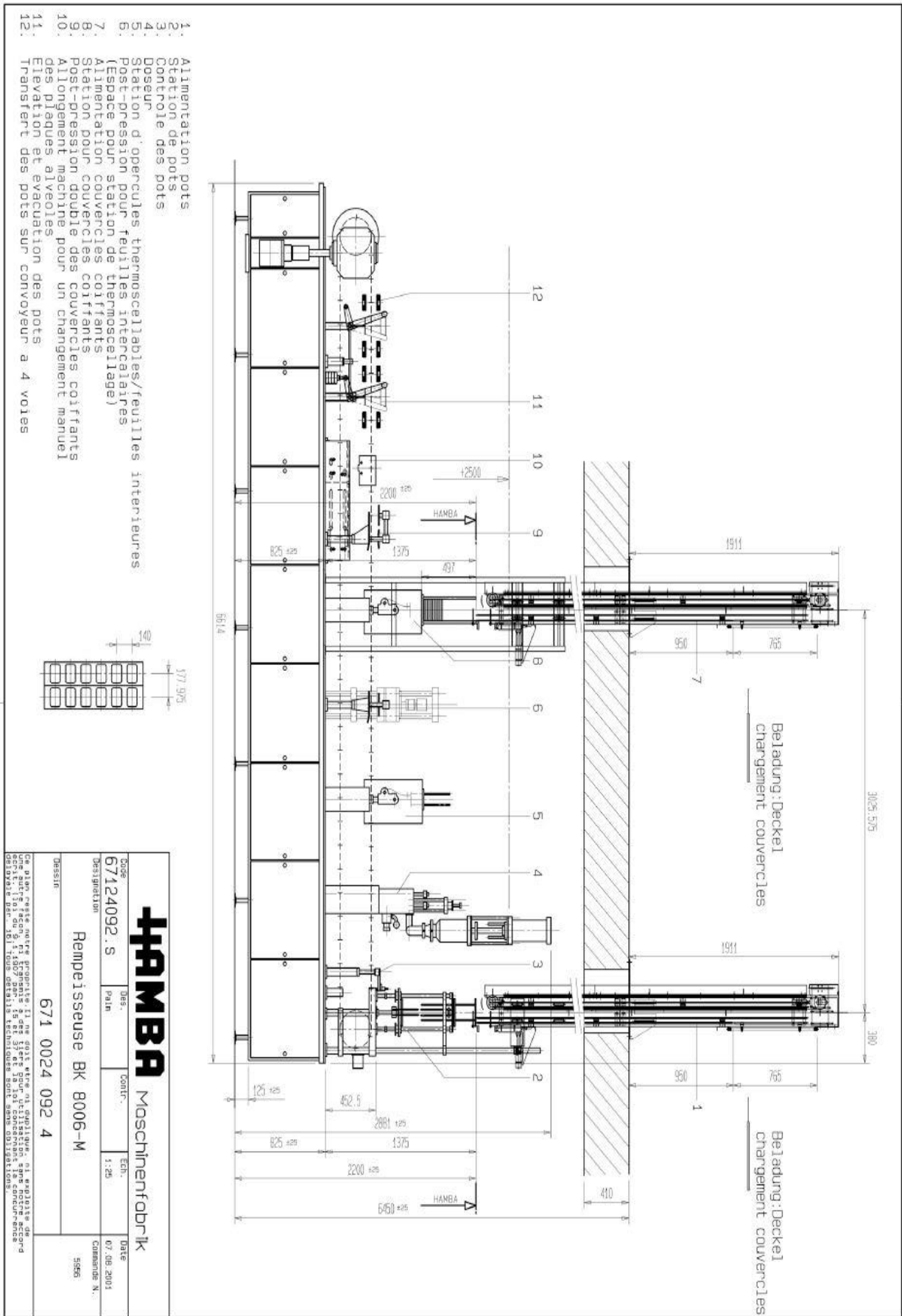


Figure II.1 : schéma remplisseuse BK 8006-M. [6]

II.2 Description de la machine

La machine remplisseuse HAMBА BK 8006-M est une remplisseuse lineaire à 6 voies avec 6 têtes de remplissage pour margarine, les coupelles (pots) sont emmagasinés dans la machine dans le sens verticale, le désempilage des pots est effectué avec système d'alimentation automatique avec détection des pots (avec ou sans pots), le doseur principal (250g ou 500g) pour produits (margarine) en connexion directe, elle possède une station d'opercules thermoscellable, l'insertion des couvercles se fait avec un système d'alimentation automatique et la sortie par quatre voies perpendiculaire à la machine.

La machine est dotée de protections en plexiglas et d'un système de graissage centralisé.



Figure II.2 : vue sur machine HAMBА.

II.3 Principe de fonctionnement

La machine HAMBА s'approvisionne en couvercles et en pots par deux descenseur (station de pots) (figure II.3 et figure II.4), qui fonction en autonome par rapport à la machine HAMBА, ceci est assuré par un automate S5 qui est en communication avec l'automate de la HAMBА par des signaux électriques.

Quand la machine HAMBА demande un approvisionnement en pots ou en couvercles, l'automate de la station de pots se prépare en vérifiant la réponse logique du programme (plain et vide) vis-à-vis du magasin de charge et de décharge, et si la chaîne de sécurité est assurée.

En bon fonctionnement la chaîne porteuse doit se positionner au point de référence sinon il y'aura un écrasement des pots dans le magasin de décharge, c'est Pour cela qu'on a des capteurs qui informe l'automate dans quel état se situe le descenseur (lieu d'affichage, lieu de transfert, point de livraison), une fois que toutes les conditions sont vérifiées l'automate du descenseur (station pots) attend le « OK » de l'automate (maitre) HAMBА.



Figure II.3 : vue sur station de pots.



Figure II.4 : vue sur station de pots (couvercles).

II.4 Branchement aux sources d'énergies

Notre Système est alimenté par une source Triphasée 400V/50Hz, elle alimente deux moteurs d'entrainements (moteur de la station de pots et la station de couvercles) d'une puissance de 1.1 kW chacun, ils possèdent une commande identique, en cas de défaut le réseau est protégé par un appareillage d'arrêt d'urgence. Pour l'adapter à l'alimentation de l'API (24V CC), une des phases est relié à un transformateur abaisseur puis a un redresseur (230V/24V 5A).

Sur l'armoire électrique on distingue 5 commandes et 8 voyants :

- Commende pour remplir/vider les pots avec 2 voyants d'état.
- Commende pour remplir/vider les couvercles avec 2 voyants d'état.
- Commende pour définir la taille des récipients (0= petit, 1=grand) avec 2 voyants d'état.
- Interrupteur principal (arrêt/mise en marche).
- Interrupteur d'arrêt d'urgence.
- Voyant d'indication (Système allumé / Prêt à fonctionner).
- Voyant d'indication (Système en Arrêt / Dysfonctionnement de l'installation).

[11]

II.5 Déroulements des menus sur l'OP

La figure II.6 montre le déroulement des menus sur l'opérateur qui permettent d'effectuer différentes opérations ou changement sur la machine.

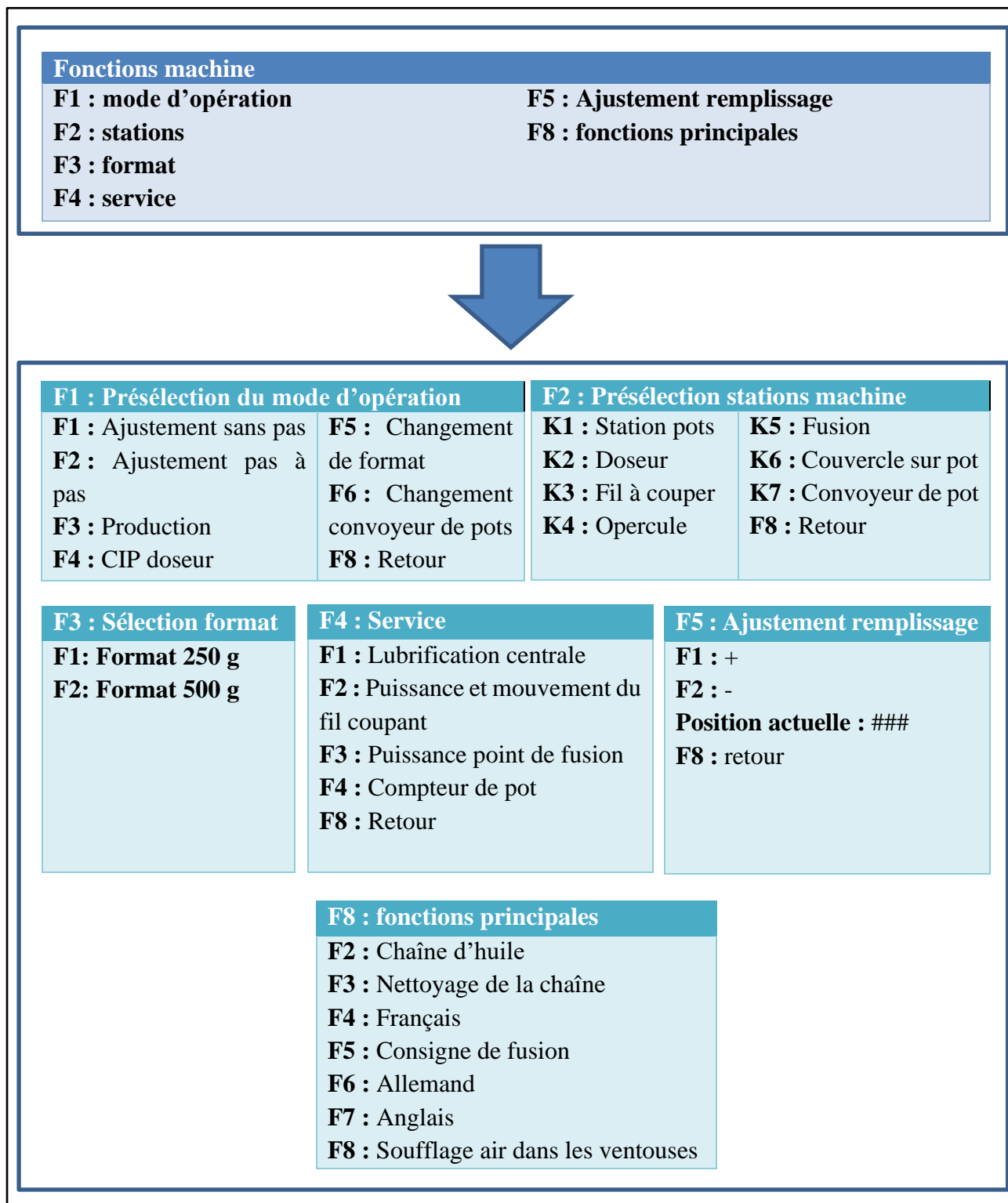


Figure II.5 : Déroulements des menus sur l'OP.[6]

II.6 Procédure de mise en marche

II.6.1 Vérification avant la mise en marche

- Mettre l'armoire de commande sous tension en mettant le sectionneur sur la position ON.
- Vérifier l'alimentation en air comprimé de la machine.
- Vérifier que la machine est réglée sur le format sélectionné.
- Vérifier que le doseur est prêt pour la production.
- Vérifier que les buses de dosage correspondent au format sélectionné.
- Mettre en marche la machine alimentation en pots et couvercles.
- Amener les pots au-dessus du magasin de pots et les couvercles au-dessus du magasin de couvercles.

II.6.2 Procédure de mise en marche

L'armoire de commande étant sous tension et l'afficheur allumé.

- Annuler les défauts en appuyant sur le bouton rouge clignotant.
- Appuyer sur ENTER.
- Choisir mode opération en appuyant sur F1 : MODE OPERATION.
- Sélectionner le format :
 - Appuyer sur F3 : FORMAT
 - Choisir le format F1 : 250g, F2 : 500g
- Appuyer sur ESC pour revenir à la première page.
- Procéder à l'ajustement sans pas ou avec pas en appuyant sur F1 ou F2.
- Appuyer sur F3 : production.
- Appuyer sur le bouton démarrage entraînement marche.
- Vérifier le poids des pots s'il est incorrect procéder à un ajustement.

II.6.3 Procédure de changement de format

L'armoire étant toujours sous tension.

- Appuyer sur ENTRE.
- Choisir mode opération en appuyant sur F1 : MODE OPERATION.
- Appuyer sur F5 : CHANGEMENT DE FORMAT.
- Appuyer sur ESC pour revenir à la première page.
- Sélectionner le format :
 - Appuyer sur F3 : FOMAT
 - Choisir le format F1 : 250g, F2 : 500g
- Appuyer sur le bouton démarrage entraînement marche.

II.6.4 Procédure de mise en marche du CIP

Mettre l'armoire sous tension.

- Appuyer sur ENTER.

- Appuyer sur F1 : mode opération.
- Appuyer sur F4 : CIP.
- Appuyer sur le bouton démarrage entraînement marche.

II.6.5 Procédure de mise à l'arrêt

- Appuyer sur le bouton blanc « éteindre mode entraînement toutes » les stations s'arrêtent automatiquement les unes après les autres jusqu'à ce que le dernier pot soit sortie par le convoyeur de sortie.
- Enlever le produit du doseur.
- Démontеr les buses de dosage et le fil coupant et nettoyer les manuellement.

II.6.6 Ajustement remplissage

Après démarrage de la production il est nécessaire de vérifier le poids du pot. On peut augmenter ou diminuer le poids en procédant comme suit :

- 1) Appuyer sur F5 : Ajustement emplissage.
- 2) Appuyer sur F1 : augmenter le poids.
- 3) Appuyer sur F2 : diminution du poids.

II.7 Problématique

Le système actuel présente certains inconvénients, à savoir :

- La non disponibilité de pièces de rechange car la firme SIEMENS a déjà arrêté la fabrication des automates programmables S5 depuis la fin 2015.
- La vitesse de transmission des données du programme est lente.
- Diagnostic difficile.
- Exigence de la part de fournisseur.

C'est pour cela, qu'on a opté de remplacer un automate programmable industriel S5-100U par S7-300 pour augmenter les performances et d'assurer une grande flexibilité

II.8 Conclusion

L'étude détaillée du processus industriel et l'identification des différentes parties de la machine nous a permis la compréhension de ses spécifications fonctionnelles et technologiques, ce qui constitue l'étape initiale et primordiale dans toute étude des systèmes automatisés. L'élaboration du programme est basée sur la connaissance de l'enchaînement des étapes du processus.

Comme nous avons opter pour l'API SIEMENS S7-300, nous procédons dans la suite de notre travail à la présentation des API S5-100U et S7-300.

CHAPITRE III

Etude fonctionnelle en vue d'une migration

Chapitre III : Etude fonctionnelle en vue d'une migration

III.1 Introduction

Dans la perspective d'implanter la migration, il y a un besoin considérable de clarification. Par conséquent, il est d'autant plus important de développer un concept de planification et de mise en œuvre la migration en attente. Chaque installation a des exigences différentes pour le processus de migration. Selon la complexité du système de contrôle de l'installation, les temps d'arrêt des machines acceptables et la flexibilité de production, la préparation, la procédure et la profondeur de migration requises peuvent différer. Il est toujours nécessaire de réfléchir et de planifier la migration des implantations entière, même si seule une migration partielle est envisagée. La question n'est pas "Comment migrer un contrôleur ?" mais "A quoi devrait ressembler l'implantation à la fin de la migration et quelles étapes de migration sont nécessaires ?".[14]

III.2 Considérations et problèmes à traiter avant la migration

Les parties implantées qui doivent être migrées :

- Même une migration partielle nécessite que l'installation entière soit prise en compte.

Les composants qui sont concernés :

- Solutions autonomes ou configuration d'installation complexe.
- Communication avec des systèmes tiers.
- Composants matériels et logiciels spéciaux existants.

Les considérations les plus importantes pour planifier le temps de migration :

- Planifier des temps de non-production.
- Production 24/7, Produire à l'avance pour amortir les temps d'arrêt.
- Décaler temporairement la production Stratégies de repli.
- Tampons de temps suffisants, tests complets jusqu'au "point de non-retour".
- Nouveau câblage de communication, même en cas d'utilisation potentielle continue des connexions de communication existantes Minimiser les risques.
- Capturer avec précision l'implantation réelle
- Planification détaillée de chaque métier individuel, identifier et considérer les dépendances.
- Migration progressive.
- Migration séparée.
- Conserver le câblage.
- Acceptations partielles.

- Tests préliminaires en laboratoire.
- Tester les connexions au système de contrôle.

Fonctionnement de l'usine après la migration :

- Formation en temps opportun du personnel d'exploitation et de maintenance.
- Mettre en œuvre des processus modifiés / améliorés.
- Différents temps de cycle de l'installation.
- Planifier la planification des pièces de rechange pour l'expansion et les améliorations futures de l'usine.[9]

III.3 Présentation du S5-100U

III.3.1 Description de l'automate S5-100U

L'automate programmable S5-100U est conçu pour des applications de faible ou de moyenne envergure, il répond à toutes les exigences que l'on peut attendre d'un automate programmable. La figure ci-dessous montre l'API présente sur l'installation.



Figure III.1: API S5-100U. [12]

Structure interne du S5-100U :

III.3.1.1 Module d'alimentation :

Il assure la distribution d'énergie aux différents autres modules. [15]

III.3.1.2 Unité central (CPU) :

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

Dans notre système le CPU équipé est le CPU 102 : une configuration avec 7 modules de périphérie TOR (à 8 voies) et 4 modules de périphérie analogiques (à 4 voies), est autorisée.[15]

Sa référence est : 6ES5 102-8MA02.[11]

III.3.1.3 Le bus interne :

Il permet la communication de l'ensemble des blocs de l'automate et des éventuelles extensions, il en existe plusieurs modèles telle que : « module de bus (SIGUT), à languettes pour clips, interruptif (SIGUT) et interruptif (connexion par cosses à clips) ». [15]

III.3.1.4 Les mémoires :

Elles permettent de stocker le système d'exploitation (ROM ou PROM), le programme (EEPROM) et les données système lors du fonctionnement (RAM). Cette dernière est généralement secourue par pile ou batterie. On peut, en règle générale, augmenter la capacité mémoire par adjonction de barrettes mémoires [15].

III.3.1.5 Les modules d'entrées /sorties :

Modules d'entrées :

Ils permettent de recevoir les informations S.A.P (systèmes automatisés de production) ou du pupitre et mettre en forme (filtrage, ...) ce signal tout en l'isolant électriquement auto couplage, il existe deux types de modules d'entrées :

- Module d'entrées TOR « tout ou rien » :

Il permet à l'unité centrale de l'automate, d'effectuer une lecture de l'état logique des capteurs du système automatisé.

Les modules TOR peuvent être enfichés sur tous les emplacements (O à 31) et seulement deux états ("O" ou "1 ") par voie peuvent être reçus par celui-ci.

- Module d'entrées analogique :

Il reçoit un signal analogique qui représente l'état que peut ou doit prendre un capteur entre deux limites. Ce module est muni d'un convertisseur analogique-numérique et ne peut être utilisés que sur les emplacements O à 7 Contrairement un module TOR. [15]

Notre système possède un module d'entrées avec une référence : 6ES5 422-8MA11. [10]

Modules de sorties :

Ils permettent de commander les divers près actionneurs et éléments de signalisation du S.A.P (système automatisés de production), il existe deux types modules de sorties :

- Module de sortie TOR « tout ou rien » :

Il permet à l'automate programmable d'agir sur les actionneurs à travers les pré-actionneurs ou d'envoyer des messages à l'opérateur.

Les modules TOR peuvent être enfichés sur tous les emplacements (O à 31) et seulement deux états ("O" ou "1 ") par voie peuvent être émis par celui-ci.

- Module de sortie analogique :

Il s'émet un signal analogique qui représente l'état que peut ou doit prendre un actionneur entre deux limites. Ce module est muni d'un convertisseur numérique-analogique.

Les modules analogiques ne peuvent être utilisés que sur les emplacements O à 7. Contrairement un module TOR. [15]

Modules à entrées et sorties combinées :

Il est possible de transmettre à ces modules des données traitées par le programme utilisateur et de lire, pour traitement, les données en provenance de ces modules. Les adresses d'octet sont identiques dans la mémoire image des entrées et dans la mémoire image des sorties.

- Module d'entrées/sorties TOR :

Ce module ne peut être enfiché que sur les emplacements O à 7. Il occupe la même plage d'adresses qu'un module analogique. Cependant, seuls les deux premiers octets réservés sont utilisés et les informations d'entrées et de sorties occupent les mêmes adresses. [15]

Notre système possède deux modules d'entrées/sorties avec une référence : 6ES5 482-8MA11. [11].

III.3.2 Elaboration d'un programme S5

Les automates programmables accomplissent des tâches d'automatisation traduites sous forme de programme utilisateur, pour que l'automate puisse le comprendre, ce dernier doit être écrit dans un langage déterminé et suivant des règles bien définies. La firme SIEMENS a développé le STEP5 pour la famille SIMATIC5.

III.3.3 Les différents modes de représentation

Le langage de programmation STEP5 est unique pour tous les appareils de la gamme SIMATIC5. Le programme peut être introduit sous forme de :

- **Liste d'instruction (LIST) :** il est sous forme d'une suite de suite d'abréviations d'instruction.
- **Logigramme (LOG) :** le logigramme représente les fonctions logiques à l'aide de symboles graphiques.
- **Schéma à contact (CON) :** le schéma à contact CON représente les fonctions de commande à l'aide des symboles graphiques des schémas électriques.
- **GRAPH5 :** ce mode de représentation sert à décrire la structure des automatismes séquentiels.

III.3.4 Stratégie de la firme SIEMENS pour la série S5

En 2020 le S5 aura 41 ans, La figure suivante nous montre clairement la stratégie de la firme SIEMENS pour stopper définitivement l'API S5 du marché mondial. Au 01/10/2002 c'est

l'annonce du futur arrêt de commercialisation de S5. L'arrêt de fourniture de S5-90/95/100 sera le 01/10/2013 et 01/10/2014 pour le S5-115, le 01/10/2015 pour S5-135/155.[9]

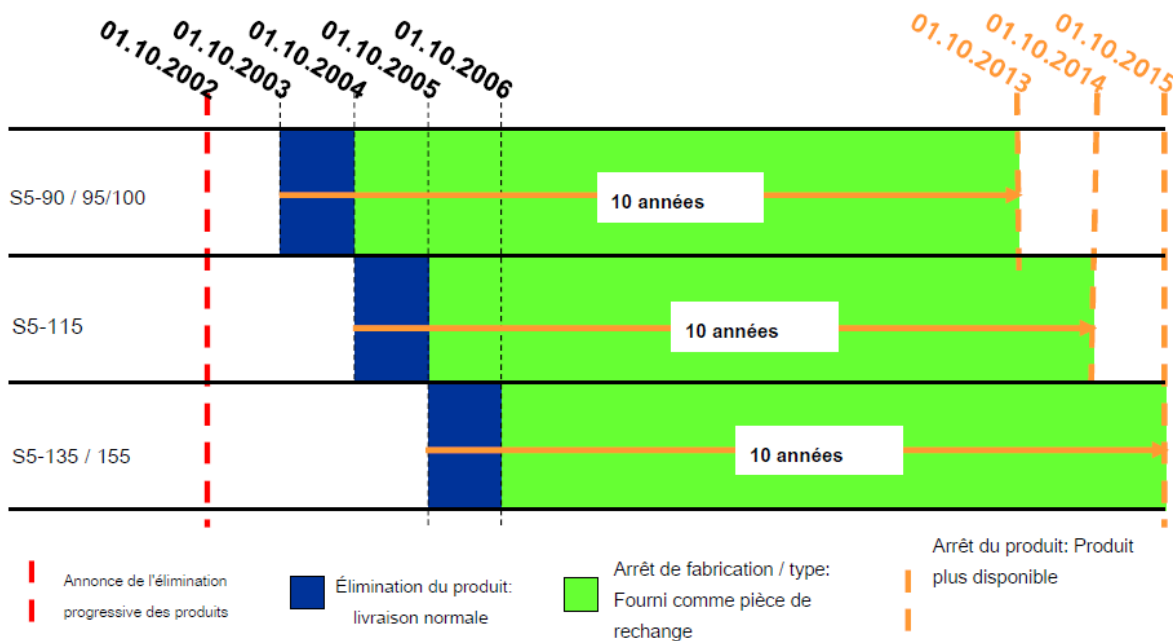


Figure III.2 : Cycle de vie de série S5 [9].

III.3.5 Solution pour remplacer SIMATIC S5

Après plusieurs années aux services des lignes de production de l'industrie, les automates SIMATIC S5 évoluent pour plus de performances. La nouvelle gamme S7, lancée en 1996, apporte une intégration idéale avec les interfaces homme-machine, ainsi qu'un atelier logiciel beaucoup plus convivial.

Dans la plupart des cas, la possibilité serait de changer le module d'entrée/sortie, tout en gardant l'unité centrale de traitement de S5, puisqu'habituellement l'entrée/sortie est le plus grand investissement sur une machine, mais cette stratégie est à écarter pour un système réduit (100U ou 95U), la meilleure approche serait de remplacer le système entier et réaliser une autre programmation en se basant sur le programme existant et un cahier des charges. [9]

III.4 Présentation du S7-300

Avec le portail d'automatisation totalement intégré (TIA Portal v15), SIMATIC S7 offre de nombreuses nouvelles options pour augmenter encore la productivité des machines et rendre le processus d'ingénierie plus efficace. Avec l'intégration de nombreuses nouvelles fonctionnalités de performances, le système d'automatisation S7 offre d'excellentes capacités de fonctionnement et des performances maximales. Les nouvelles fonctionnalités de performance sont :

- Performances système accrues.
- Fonctionnalité de contrôle de mouvement intégrée PROFINET IO IRT.
- Affichage intégré pour le contrôle et le diagnostic de l'opérateur local.
- Innovations linguistiques STEP 7. [13]



Figure III.3 : API S7-300.[12]

L'automate programmable S7-300, comme représenté dans les figures III.3, est un automate modulaire qui se compose des éléments suivants :

III.4.1 CPU

Par rapport aux automates programmables SIMATIC S5, les types de CPU disponibles de la génération d'automates S7 présentent des différences et des fonctions considérables.

On trouve les CPU 312 IFM, 313, 314, 314 IFM, 315, 315-2 DP et 316, 312ptp, 313C2DP...etc. [9]

Note : dans le cadre de notre migration nous utiliserons le CPU 315-2DP. [11]



Figure III.4 : CPU 315-2DP.[12]

III.4.1.1 LED de visualisation d'état et de défaut

- (Rouge)SF Défaut matériel ou logiciel.
- (Rouge)BATF Défaillance de la pile.
- (Vert) 5VDC L'alimentation 5V DC est correcte
- (Jaune)FRCE Le forçage permanent est actif.
- (Verte)RUN CPU en RUN.
- (Jaune)STOP CPU en STOP ou en ATTENTE ou en démarrage.

III.4.1.2 Commutateur de mode de fonctionnement

Le changement de mode se fait à l'aide d'une clé :

Position	Signalisation	Explication
RUN-P	Mode de fonctionnement RUN-PROGRAMME	Le CPU traite le programme utilisateur. Le programme peut être modifier. Dans cette position la clef ne peut être retirée.
RUN	Mode de fonctionnement RUN	Le CPU traite le programme. Le programme ne peut être modifié qu'avec légitimation par mot de passe. La clef peut être retirée
STOP	Mode de fonctionnement STOP	Le CPU ne traite aucun programme utilisateur. La clef peut être retirée.
MRES	Effacement général	Position instable du commutateur, pour effectuer l'effacement général il faut respecter un ordre particulier de commutation.

Tableau III.1 : Les positions du commutateur. [14]

III.4.1.3 Pile de sauvegarde ou accumulateur

L'utilisation de l'accumulateur ou de la pile de sauvegarde est nécessaire pour l'horloge temps réelle.

La pile de sauvegarde est aussi utilisée pour :

- La sauvegarde du programme utilisateur s'il n'est pas enregistré dans la mémoire morte.
- Pour étendre la zone rémanente de données.

L'accumulateur est rechargé à chaque mise sous tension de la CPU. Son autonomie est de quelques jours voire quelques semaines au maximum. La pile de sauvegarde n'est pas

rechargeable mais son autonomie peut aller jusqu'à une année.

III.4.1.4 Carte mémoire

La plupart des CPU possèdent une carte mémoire. Son rôle est de sauvegarder le programme utilisateur, le système d'exploitation et les paramètres qui déterminent le comportement de la CPU et des modules en cas de coupure du courant.

III.4.1.5 Interface MPI (interface multipoint)

L'interface MPI est l'interface de la CPU utilisée pour la console de programmation (PG), le pupitre operateur (OP) ou par la communication au sein d'un réseau MPI. La vitesse de transmission typique est de 187,5 KBauds.

III.4.2 Module d'alimentation

Divers modules d'alimentation sont mis à disposition pour l'alimentation du S7-300 et des capteurs/actionneurs en 24 V cc.

Pour notre système nous allons utiliser une alimentation PS-307 5A qui est une alimentation externe AC120/230V : DC24V/5A.

III.4.3 Module d'entrées/sorties

Plusieurs types de modules sont disponibles sur le marché selon l'utilisation souhaitée, pour notre système nous allons utiliser deux modules d'entrées/sorties TOR DI16/DO16 x DC24V/0,5A (6ES7 323-1BL00-0AA0) et un module d'entrées TOR DI16 x DC24V (6ES7 321-1BH02-0AA0).

III.4.3 Organisation du projet Step 7

Le projet regroupe la totalité des données et des programmes d'une solution d'automatisation, quels que soient le nombre d'unités centrales et leur mise en réseau. Un projet n'est donc pas limité à un programme utilisateur d'un module programmable, mais pourra réunir sous un nom de projet commun plusieurs programmes utilisateur pour plusieurs modules programmables. [13]

Les principaux objets du projet STEP 7 sont représentés sur la figure ci-dessous. Ils sont explicités ci-après.

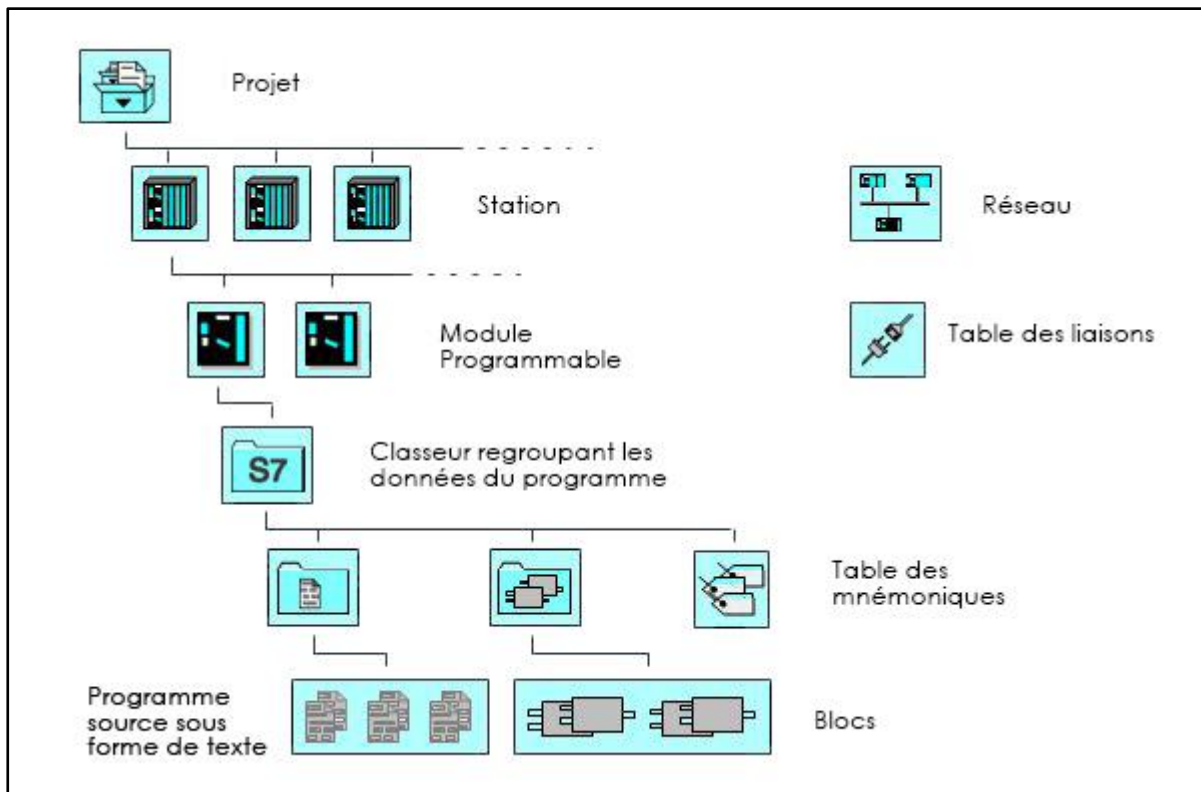


Figure III.5 : Structure hiérarchique des objets du projet STEP 7. [14]

Réseau : objet renfermant les paramètres de réseau (MPI ou PROFIBUS). Ceci permet la vérification des paramètres de communication par le programme dès qu'une station ou un module de celle-ci est connecté au réseau.

Station : la station représente la configuration d'un système d'automatisation avec ses châssis. Lorsque vous enfichez un module avec interface DP dans une station, le réseau maître DP qui part de la station fait partie avec tous ses esclaves de la station !

Une station peut comprendre un ou plusieurs modules programmables (CPU).

Matériel : objet renfermant les données de configuration et les paramètres d'une station. Les données de configuration et les paramètres de la station sont sauvegardés dans les blocs de données système (SDB).

Module programmable : les modules programmables sont à l'opposé des autres modules les modules porteurs des programmes utilisateur. Sous les modules programmables vous trouvez des dossiers appelés dans STEP 7 des "classeurs" contenant toutes les données du programme pour ce module :

- Programmes-sources sous forme de texte (créés à l'aide d'un éditeur de texte).
- Blocs (pouvant être chargés dans le module programmable).
- Table des mnémoniques.

Table des liaisons : la table des liaisons représente la totalité des liaisons d'un module

programmable (par exemple d'un CPU) au sein d'une station. Une liaison définit les propriétés de la communication entre deux partenaires de communication et est caractérisée par un ID de liaison. Celle-ci suffit pour programmer une communication déclenchée par événement à l'aide de blocs de communication standardisés qui peuvent être comparés aux blocs de dialogue de STEP 5.

Sources : les sources servent en programmation S7 à la génération des blocs. Elles ne peuvent pas être chargées dans une CPU S7.

Blocs : les blocs sont des parties du programme utilisateur délimités de par leur fonction, leur structure ou leur utilisation. Il est possible de charger les blocs dans des CPU S7.

Outre les blocs exécutables, vous trouvez dans le classeur des blocs les tables des variables.

Table des mnémoniques : vous affectez dans la table des mnémoniques des noms (c'est-à-dire des mnémoniques) aux entrées, sorties, mémentos et blocs.

III.5 Raisons de choix de SIEMENS S7-300

Nous avons opté pour cet API S7300, plus performant et plus puissant, facile à manier. Les principales raisons qui ont influées dans le choix de cet automate sont :

- Le nombre restreint des paramètres d'entrées et sorties logiques.
 - Le personnel technique de l'entreprise est qualifié dans l'utilisation des automates SIEMENS.
 - La performance des caractéristiques techniques de l'automate SIEMENS S7-300.
 - Le S7-300 dispose d'une gamme de modules complets pour une adaptation optimale aux tâches les plus diverses.
 - Le S7-300 se caractérise par la facilité de réalisation d'architecture décentralisées et la simplicité d'emploi.
 - Souplesse d'utilisation grâce à des architectures décentralisées simples et aux multiples possibilités de mise en réseau.
 - Facilité et confort d'utilisation grâce à une configuration simple.
 - Evolutivité permettant l'intégration de nouvelles tâches.
 - Haut niveau de performance procuré par les nombreuses fonctions intégrées
- L'API S7-300 est de conception modulaire, nous pouvons le composer en fonction de nos besoins à partir d'un vaste éventail de modules qui comprend :
- Des CPU de différents niveaux de performance.
 - Des modules de signaux pour entrées et sorties TOR et des modules analogiques.
 - Des modules d'alimentation pour le raccordement du S7-300 sur secteur 120/230 V.
 - Des coupleurs pour configurer un automate sur plusieurs profilés-support.

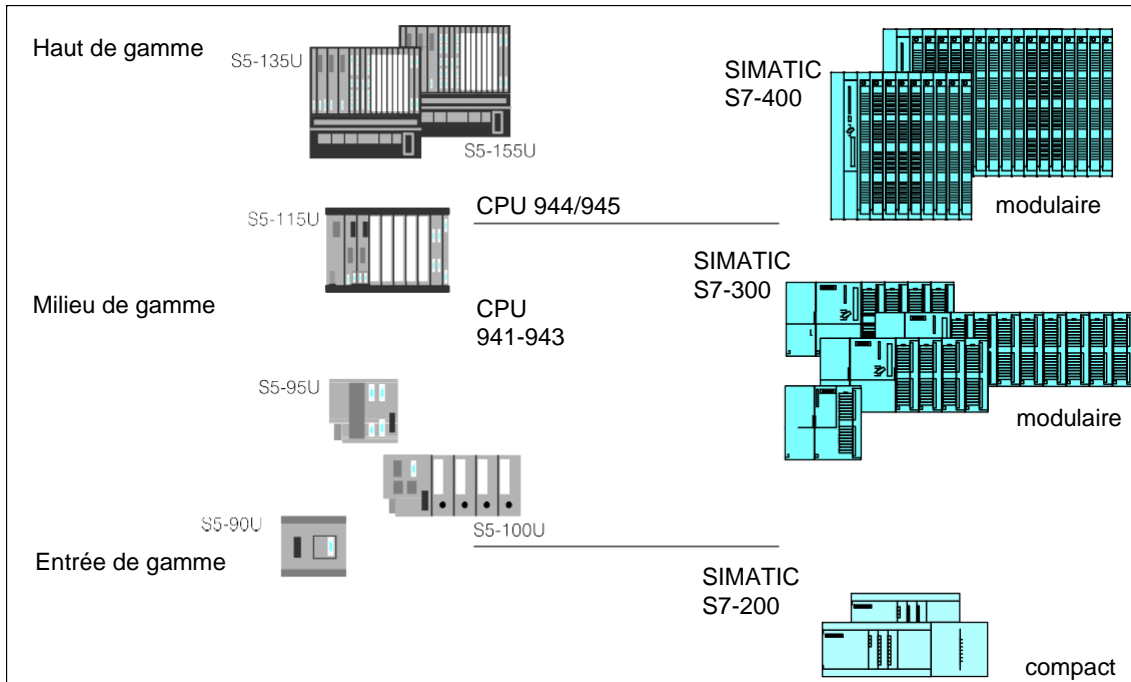


Figure III.6 : Différentes gammes des SIMATIC S5 et des nouveaux SIMATIC S7. [14]

III.6 Comparatif

III.6.1 Mise en parallèle

Le tableau suivant énumère les blocs STEP 5 et les blocs STEP 7 en parallèle montrant à chaque fois la correspondance entre ceux-ci et indique quand un bloc S5 doit être remplacé par un bloc ou une fonction STEP 7 équivalente.

Bloc STEP 5	Bloc STEP 7	Explication
Bloc d'organisation (OB)	Bloc d'organisation (OB)	Interface au système d'exploitation
OB spéciaux intégrés	Fonctions système (SFC) Blocs fonctionnels système (SFB)	Les fonctions système de STEP 7 remplacent les blocs d'organisation spéciaux de STEP 5 pouvant être appelés dans le programme utilisateur.
Bloc fonctionnel (FB, FX)	Fonction (FC)	Les fonctions (FC) de STEP 7 ont les mêmes propriétés que les blocs fonctionnels de STEP 5.
Bloc de programme (PB)	Bloc fonctionnel (FB)	Les blocs de programme ont leur équivalent dans STEP 7 : on les appelle des blocs fonctionnels. A la différence de leurs homologues dans STEP 5, ils possèdent des propriétés nouvelles et ouvrent de nouvelles perspectives en matière de programmation. Attention : les blocs de programme sont convertis en fonctions STEP 7 (FC).
Bloc séquentiel (SB)	-	Les blocs séquentiels n'existent plus dans STEP 7.
Bloc de données (DB, DX)	Bloc de données (DB)	Les blocs de données sont plus longs dans STEP 7 que leurs équivalents dans STEP 5 (jusqu'à 8 kilo-octets pour le S7-300, et jusqu'à 64 kilo-octets pour le S7-400)

Blocs de données DX0, DB1 dans leur fonction spéciale	Blocs de données système (SDB) (Paramétrage CPU)	Les nouveaux blocs de données système renferment toutes les données de configuration matérielle ainsi que les paramétrages de CPU nécessaires à l'exécution du programme.
Blocs de commentaire DK, DKX, FK, FKX, PK	-	Les blocs de commentaire n'existent plus dans STEP 7. Le commentaire est contenu dans les blocs correspondants dans la base de données hors ligne.

Tableau III.2 : Mise en parallèle des blocs STEP 5 et blocs STEP 7. [14]

III.6.1.1 Fonctions et blocs fonctionnels

Fonctions (FC) : Une fonction (FC) est un bloc de code "sans mémoire" dont les paramètres de sortie affichent en fin d'exécution les valeurs qu'elle a calculées. Le traitement ultérieur et la sauvegarde de ces résultats doivent donc être considérés par l'utilisateur lorsqu'il programme l'appel de la fonction.

Blocs fonctionnels (FB) : Un bloc fonctionnel (FB) est un bloc de code "avec mémoire". Lui sert de mémoire un bloc de données d'instance qui lui est associé et dans lequel les paramètres effectifs et les données statiques du bloc fonctionnel sont stockés.

Les blocs fonctionnels sont par exemple utilisés lorsqu'il s'agit de programmer des régulateurs.

III.6.1.2 Blocs de données

Les blocs de données servent à stocker les données du programme utilisateur. On fait la distinction entre les blocs de données globaux et les blocs de données d'instance :

- Les blocs de données globaux ne sont pas affectés à un bloc précis (comme dans STEP 5).
- Les blocs de données d'instance sont associés à un bloc fonctionnel et peuvent contenir en plus des données de ce FB les données de multi-instances que l'on aura éventuellement définies.

Un bloc de données est soit un bloc de données global, soit un bloc de données d'instance.

III.6.1.3 Blocs système

Fonctions et blocs fonctionnels système (SFC et SFB) : Il n'est pas nécessaire de programmer chaque fonction. Vous pouvez recourir à des blocs préprogrammés intégrés au système d'exploitation des unités centrales, par exemple pour programmer les fonctions de communication. Il s'agit des blocs suivants :

- Fonctions système (SFC), possédant les mêmes propriétés que les fonctions (FC),
- Blocs fonctionnels système (SFB), possédant les mêmes propriétés que les blocs fonctionnels (FB).

Blocs de données système (SDB) : Nous avons jusqu'ici parlé de blocs renfermant le code ou les données du programme utilisateur. Il existe à côté de ces blocs des blocs qui servent à stocker les adresses ou les paramètres des modules. On les appelle des blocs de données

système (SDB). Les blocs de données système sont générés par des applications spécifiques de STEP 7, par exemple lors de la configuration matérielle ou à la création des tables de liaisons.

III.6.1.4 Blocs d'organisation

Les blocs d'organisation (OB) servent d'interface entre le système d'exploitation et le programme utilisateur. Ces blocs d'organisation remplissent des tâches différentes bien précises.

Classification des OB : Vous écrivez votre programme utilisateur LIST pour la CPU S7 en sélectionnant les blocs d'organisation (OB) dont vous avez besoin pour votre automatisme.

OB d'erreur : Les OB d'erreur sont appelés quand une erreur survient dans le déroulement du programme. Ils vous permettent de programmer des réactions à l'erreur. Lorsqu'il n'existe pas d'OB pour un type d'erreur, la CPU passe à l'arrêt.

III.6.2 Types de données

STEP 7 utilise de nouveaux types de données. Vous voyez les types de données S5 et S7 en parallèle dans le tableau ci-après.

Types de données dans S5	Types de données dans S7	Classe
BOOL, BYTE, WORD, DWORD, Nombre entier 16 bits, nombre entier 32 bits, virgule flottante, valeur temporelle, - (Caractères ASCII)	BOOL, BYTE, WORD, DWORD, INT, DINT, REAL, S5TIME, TIME, DATE; TIME_OF_DAY, CHAR	Types de données simples
-	DATE_AND_TIME, STRING, ARRAY, STRUCT	Types de données complexes
Temporisations, compteurs, blocs - -	TIMER, COUNTER, BLOCK_FC, BLOCK_FB, BLOCK_DB, BLOCK_SDB, POINTER, ANY	Types de paramètre

Tableau III.3 : Types de données dans S5 et S7. [14]

III.6.3 Transposition des blocs à la conversion

Correspondance des blocs : La structure des blocs a été modifiée dans S7. Vous voyez sur la figure ci-dessous la correspondance des blocs de STEP 5 et STEP 7. A la conversion, les blocs de STEP 5 sont remplacés de façon analogue par les blocs de STEP 7.

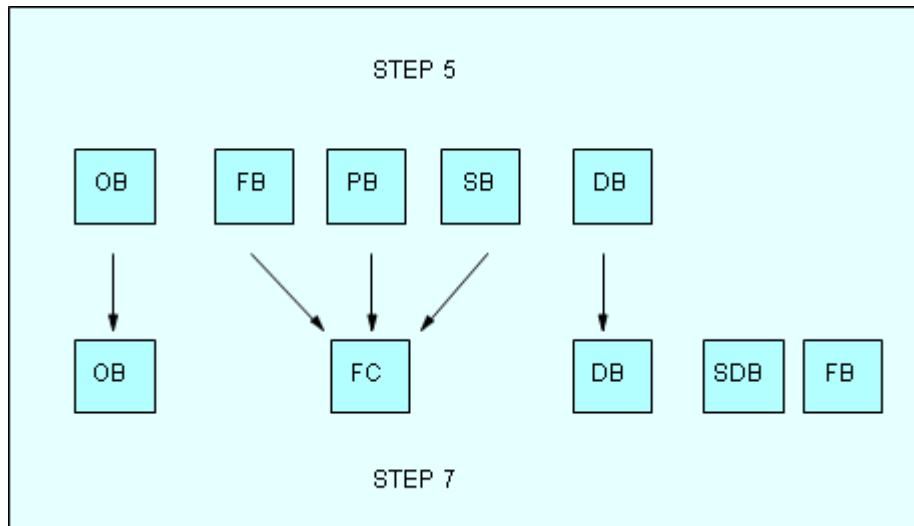


Figure III.7 : Blocs S5 et S7 remplissant des fonctions similaires.[14]

Dans STEP 5, il était possible de stocker des données dans des blocs de données, DB et DX et dans des drapeaux. Il n'était pas possible de structurer les données ou de définir des types de données définis par l'utilisateur dans STEP 5. STEP 7 et S7-300 offrent cependant diverses options pour structurer les données et accéder facilement à ces structures.

III.7 Avantages de la modernisation :

Depuis leur lancement, les contrôleurs SIMATIC S5 ont bien fonctionné. Avec l'élimination de cette gamme de produits, la maintenance de ces systèmes de contrôle deviendra de plus en plus complexe et coûteuse. Pendant ce temps, les mécanismes et les technologies ont changé. Un système d'automatisation SIMATIC moderne tel que le S7-300 peut offrir les avantages techniques et financiers suivants :

- Productivité accrue.
- Coûts de production réduits.
- Utilisation des machines accrue.
- Conformité aux nouvelles réglementations, par exemple : La sécurité.
- Meilleure qualité des produits et contrôle des processus.
- Plus grande flexibilité dans la production et la planification de la production.
- Prise en charge de l'intégration et de l'expansion futures de vos usines.
- Prise en charge de la technologie de fabrication de pointe.
- Le risque pour les anciennes usines augmente continuellement en raison de la situation difficile d'approvisionnement en pièces détachées et de la disponibilité

croissante d'ingénieurs de service qualifiés.

III.8 Conclusion :

L'objectif de notre travail est de remplacer l'automate S5-100U par un automate S7-300 pour le pilotage du descenseur HAMBА de la ligne 1 margarinerie CEVITAL. Le logiciel de programmation du S7-300 est STEP 7. On a préconisé d'utiliser des commandes qui permettent de translater le programme en STEP 5 vers le STEP 7 et de corriger les erreurs générées par cette procédure puis d'effectuer une migration du programme en STEP 7 vers TIA PORTAL.

CHAPITRE IV

Migration du S5 vers S7

Chapitre IV : Migration du S5 vers S7

IV.1 Introduction

L'intégration totale est un concept révolutionnaire visant à réunir l'univers de la fabrication manufacturière et l'univers des procédés. Toutes les briques matérielles et logicielles nécessaires à la réalisation d'un projet portent désormais un seul nom : SIMATIC.

L'intégration totale est rendue possible par l'homogénéité parfaite des données, tant bien :

- Au niveau de la base de données : Les données ne sont plus saisies qu'une seule fois mais sont disponibles dans toute l'usine. Les erreurs dues à la transposition des données et les incohérences appartiennent désormais au passé.
- Au niveau de la conception et de la programmation : Toutes les briques servant à la réalisation du projet sont conçues, configurées, programmées, mises en service, testées et surveillées sous une seule interface utilisateur avec l'outil qui leur est dédié.
- Qu'au niveau de la communication : nous pouvons voir dans la table des liaisons de qui communique avec qui, les liaisons pouvant à tout moment et en tout lieu être modifiées. Il est désormais possible de configurer différents réseaux avec un seul outil simple à utiliser.[9]

Dans ce chapitre nous avons opté pour une approche en deux étapes. La première est basée sur la conversion du S5 vers S7 en utilisant le convertisseur S5/S7 et la deuxième est de faire une migration du S7 vers TIA PORTAL grâce à l'outil de migration de TIA PORTAL V15.

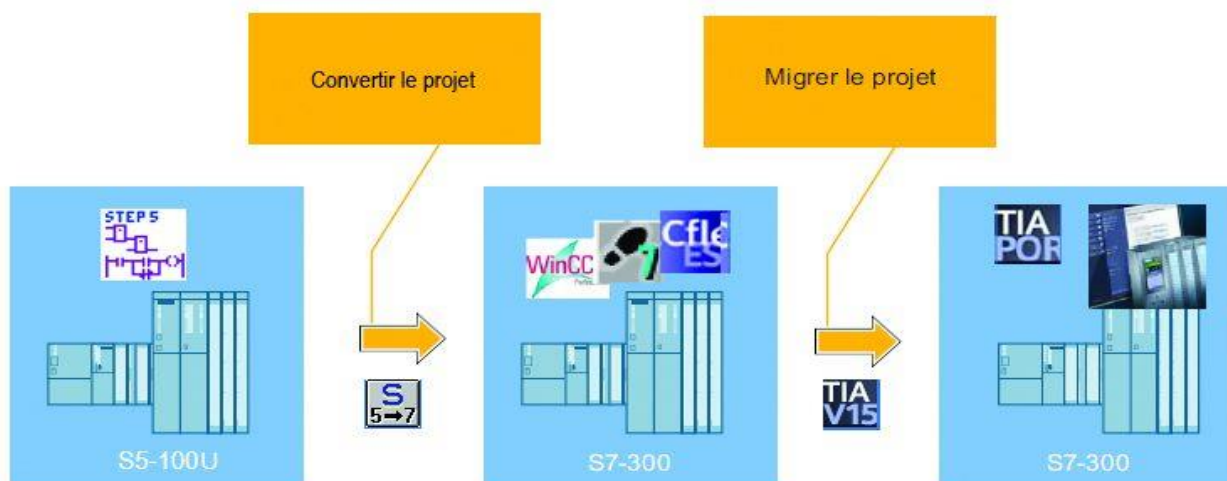


Figure IV.1 : Vue d'ensemble des possibilités de migration.[13]

IV.2 Conversion du S5 vers S7

Pour une grande part, la programmation S7 en LIST, CONT et LOG est compatible respectivement avec LIST, CONT et LOG de S5. Par conséquent, la conversion en sera très facile. Il suffit de les convertir en programmes S7.[9]

Le "Convertisseur S5/S7" permet de convertir, si possible de façon complète, le jeu d'instructions des programmes S5 existants en programmes S7, la conversion des programmes S5 existants se fait toujours en LIST et les programmes sont toujours convertis entièrement et nécessitent des corrections des erreurs générées par la conversion.

IV.2.1 Préparation de la conversion

IV.2.1.1 Fichier requis

Les fichiers suivants sont nécessaires à la conversion de votre programme S5 :

- Fichier programme <nom>**ST.S5D** : le programme S5 à convertir.
- Liste des références croisées <nom>**XR.INI** : elle contient la structure du programme qui est nécessaire afin de conserver la structure et la hiérarchie d'appel du programme S5.
- Liste d'assignation S5 <nom>**Z0.SEQ** : Si nous désirons convertir également le fichier des mnémoniques (noms symboliques), nous avons besoin en outre, de ce fichier.[9]

Donc pour préparer la conversion nous procédons comme suit :

- Créer à l'aide du logiciel S5 une liste croisée actuelle pour le programme S5.
- Copier dans un répertoire DOS le fichier programme STEP 5, la liste croisée correspondante et, le cas échéant, la liste d'assignation.[10]

IV.2.1.2 Vérification des opérandes

Il s'avérera peut-être nécessaire d'adapter le programme à convertir à la CPU S7 que nous comptons utiliser. Pour cela, il faut procéder comme suit :

- Déterminer la CPU S7 que vous désirez utiliser.
- Rechercher ses caractéristiques et comparer :
 - Le nombre d'opérandes.
 - Le nombre de blocs.

Aux opérandes et blocs utilisés.

IV.2.1.3 Les phase de conversion

En cliquant sur le bouton "Convertir", nous mettons en route la procédure de conversion, elle se compose de deux phases de conversion et de transposition de la liste d'assignation.

Au cours de la première phase de conversion, le programme S5 est converti en un fichier source S5 avec tous les blocs et tous les commentaires.

Au cours de la deuxième phase, le fichier source S5 est converti en un fichier source LIST avec les nouveaux numéros de bloc et la syntaxe de S7.

VI.2.1.4 Fichiers générés

Le convertisseur S5/S7 génère les fichiers suivants :

- Fichier <nom>**A0.SEQ** : ce fichier est créé pendant la première phase de conversion, il contient le fichier <nom>**ST.S5D** sous forme ASCII.
- Fichier <nom>**AC.AWL** : ce fichier est créé pendant la seconde phase de conversion, il contient le programme LIST. De cette seconde phase peuvent également provenir des messages résultant de définitions incorrectes de macro-instructions.
- Fichier <nom>**S7.SEQ** : ce fichier est créé lors de la transposition de la liste d'assignation. Il contient la liste d'assignation convertie en un format que l'éditeur de mnémoniques peut importer.
- Fichier d'erreurs <nom>**AF.SEQ** : ce fichier est affiché dans la partie supérieure de la fenêtre "Conversion de fichiers S5", contient les erreurs et les avertissements figurant dans le programme converti. Ces messages sont générés pendant la première et seconde phase de la conversion et pendant la transposition de la liste d'assignation.

Une fois la conversion terminée, une boîte de dialogue indiquant le nombre d'erreurs et d'avertissements s'affiche.

IV.2.1.4 Interprétation des messages

Parmi tous les messages du convertisseur, on distingue les messages d'erreur et les avertissements, qui sont visualisable dans la zone inférieure de la fenêtre "Messages" le fichier dans lequel l'erreur s'est produite y est cité aussi, afin d'y remédier nous pouvons consulter l'aide en ligne pour comprendre la signification du message et corriger l'erreur comme il est proposé.

A. Messages d'erreur :

Un message d'erreur est émis quand une partie du programme S5 n'est pas convertible et ne peut figurer qu'en tant que commentaire dans le programme S7. Le tableau suivant dresse la liste de tous les messages d'erreur avec leurs significations et les mesures à prendre pour remédier à l'erreur.

Message d'erreur	Origine	Signification	Remède
Paramètre absolu diverge de l'identificateur d'opérande	Phase 1	L'identificateur d'opérande n'est pas correct.	Vérifiez l'instruction.

Migration du S5 vers S7

Bloc introuvable	Phase 1	Le bloc appelé (FB, FX) manque ou il figure dans la liste des blocs mais n'existe pas dans le fichier programme.	Vérifiez la structure du programme.
	Phase 2	Un bloc est appelé qui n'existe pas dans le fichier de programme.	Vérifiez que la liste de références croisées a bien été indiquée lors de la conversion ou contrôlez la structure du programme.
La commande n'est pas autorisée dans ce bloc.	Phase 1	Saut à l'intérieur d'un bloc de programme, par exemple.	Vérifiez l'instruction.
Commande non définie.	Phase 1	L'instruction MC5/LIST n'est pas valable.	Corrigez le fichier programme S5.
	Phase 2	L'instruction n'existe pas dans S7.	Editez une macro-instruction ou remplacez l'instruction par la séquence d'instructions de S7 appropriée.
L'accès par bits au compteur/à la temporisation n'est pas possible. Veuillez vérifier.	Phase 2	Le programme S5 contient des accès par bit à des temporisations et à des compteurs.	Vérifiez le programme LIST.
CALL OB n'est pas autorisé.	Phase 2	L'appel de blocs d'organisation n'est pas autorisé dans S7.	Le cas échéant, utilisez l'instruction CALL SFC.
CALL SFCxy a été généré, veuillez compléter la liste des paramètres.	Phase 2	Des paramètres SFC manquent.	Complétez la liste des paramètres SFC.
Fichier introuvable	Globale	Le fichier sélectionné n'existe pas.	Vérifiez le fichier de programme.
Profondeur d'imbrication incorrecte.	Phase 1	Toutes les parenthèses ne sont pas correctement fermées	Respectez les niveaux de parenthèse, éliminez l'erreur de programmation.
Opérande incorrect	Phase 1	L'opérande ne convient pas à l'opération.	Vérifiez la source S5.
	Phase 2	L'opérande ne convient pas à l'opération.	Modifiez le fichier LIST.
Erreur de conversion	Phase 2	Opération BI sans constante	Complétez l'opération de chargement par une constante.
Erreur dans le fichier macro, xy non pris en compte	Phase 2	Erreur de macro-instruction	Vérifiez la macro-instruction.
Paramètre formel non défini	Phase 1	Il y a plus de paramètres que dans le bloc appelant.	Vérifiez le fichier programme S5.
Fichier ou répertoire introuvable	Phase 1	Le fichier programme ne contient aucun bloc.	Vérifiez le fichier programme.
Longueur de commentaire incorrecte	Phase 1	Erreur dans le fichier S5	Vérifiez le fichier programme.
Commentaire trop long	Phase 1	Erreur dans le fichier S5	Vérifiez le fichier programme.
Aucun nom de bloc spécifié	Phase 1	Le nom de bloc ne comporte que des espaces.	Entrez un nom de bloc.
Droits d'accès manquants	Globale	Le fichier est protégé en écriture.	Supprimez la protection en écriture.
Marque non définie	Phase 1	Le repère de saut n'est pas défini dans l'étiquette.	Vérifiez le fichier S5.
Marque incorrecte	Phase 1	Le repère de saut contient des caractères non valables.	Vérifiez le fichier S5.
Opérateur incorrect	Phase 1	L'opérateur dans le fichier S5 est inconnu ou impossible à convertir.	Remplacez l'opérateur par l'opération S7 appropriée.

Migration du S5 vers S7

Opérateur incorrect, peut éventuellement être remplacé par l'instruction : \\''L P# paramètre formel\\''.	Phase 2	L'opérateur ne peut pas être chargé sous cette forme dans S7.	Utilisez éventuellement l'instruction indiquée.
Nombre de paramètres incorrect	Phase 1	Erreur dans le programme S5	Vérifiez le fichier programme.
Paramètre erroné	Phase 1	Erreur dans le programme S5	Vérifiez le fichier programme.
Type de paramètre incorrect.	Phase 1	Erreur dans le programme S5	Vérifiez le fichier programme.
Erreur d'écriture disquette	Globale	Le fichier est protégé en écriture ou il n'y a plus de place sur la disquette.	Supprimez la protection en écriture ou effacez les données dont vous n'avez pas besoin.
Débordement de la mémoire dans la PG (problèmes de place)	Phase 1	La mémoire centrale est insuffisante.	Effacez de la mémoire centrale les fichiers dont vous n'avez plus besoin.
Le repère de saut ne peut être généré.	Phase 2	L'opération SPR dépasse la limite du bloc.	Éliminez l'erreur dans le programme S5.
Un code MC5 incorrect a été converti.	Phase 1	Conversion d'une ancienne opération de S5.	Aucun

Tableau IV.1 : Liste des messages d'erreur, leurs significations et les mesures à prendre pour y remédier.[9]

B. Avertissements :

Un avertissement est émis quand une partie du programme S5 est certes convertie, mais qu'il faudrait en vérifier la validité. Le tableau suivant dresse la liste de tous les avertissements, leurs significations et les réactions conseillées.

Avertissement	Origine	Signification	Réaction
Version incorrecte (no de produit)	Phase 1	Un bloc fonctionnel standard de S5 doit être remplacé par une FC de S7.	Aucune
Version incorrecte (bloc GRAPH 5)	Phase 1	Les blocs GRAPH 5 ne sont pas convertibles.	Utilisez éventuellement un bloc créé avec GRAPH pour S7.
Vérifiez la base de temps choisie.	Phase 2	Dans S7, la base de temps peut être plus serrée que dans S5.	Paramétrez la base de temps à l'aide de l'application « Configuration matérielle ».
I/D n'influence que l'accu 1 - L, qui est maintenant l'accu 1.	Phase 2	Les accumulateurs de S7 sont étendus à 32 bits.	Examinez les conséquences d'une opération indirecte d'incrément ou de décrémentation dans le programme LIST.
Tenez compte de la nouvelle numérotation de blocs.	Phase 2	L'appel indirect de bloc ne tient pas compte des nouveaux numéros de bloc (le numéro est prélevé dans le mot de memento ou de données approprié).	Modifiez la logique dans S5 ou utilisez des appels de bloc fixes.
OB23 et OB24 sont convertis en OB 122.	Phase 2	Les OB23 et OB24 sont remplacés tous deux par l'OB122 dans S7.	Regroupez le contenu des OB23 et 24 dans un OB122 et effacez l'autre OB122.
L'OB a été interprété par l'AG115U comme un OB34.	Phase 2	Selon la CPU employée, l'OB34 peut avoir des significations différentes.	Vérifiez que cet OB convient à votre programme.

Migration du S5 vers S7

'Masques DB S5' n'est plus utilisé pour le paramétrage de S7.	Phase 1	Il y a MASK dans DW0 et dans DW1.	Paramétrez l'AP avec STEP 7.
L'opération de saut suivant l'opération 'B' ne peut être convertie (utilisez SPL).	Phase 2	Une opération de substitution suivie d'une opération SPA ne peut être convertie automatiquement.	Remplacez l'opération par SPL dans le fichier LIST et revoyez le saut.
Le convertisseur ne définit pas les paramètres système	Phase 2	DB1 et DX0 sont convertis mais ils n'ont plus la même signification que dans S5.	Effectuez le paramétrage du système dans la table de configuration.
Tenez compte des différentes opérations d'arrêt.	Phase 2	Il n'est pas fait de différence entre STP, STS et STW.	Vérifiez le fichier programme.
RLG est mis à 1.	Phase 2	Pour les opérations SU et RU de S5, le RLG est mis à 1 dans S7.	Ajoutez au besoin l'opération CLEAR.
Pré-en-tête manque.	Phase 1	Pour les blocs FB et FX, les désignations des repères de saut manquent ; pour les blocs DB et DX, ce sont les formats de données qui manquent.	Examinez si les étiquettes se trouvent dans un autre fichier.
Dans le cas d'un AG115U, changez-le en OB 100.	Phase 2	L'OB21 de mise en route de S5 est converti automatiquement en OB101.	Si le programme S5 était exécuté sur un automate S5-115U, il faut transformer l'OB101 en OB100.

Tableau IV.2 : Avertissements, signification et réaction conseillée.[9]

IV.2.2 Convention

Nous allons effectuer la conversion à l'aide de l'application « Conversion de fichiers S5 ».

1. Nous allons faire appel à l'outil de conversion « **Conversion de fichiers S5** »,
(→Démarrer → Siemens Automation → Conversion de fichiers S5)

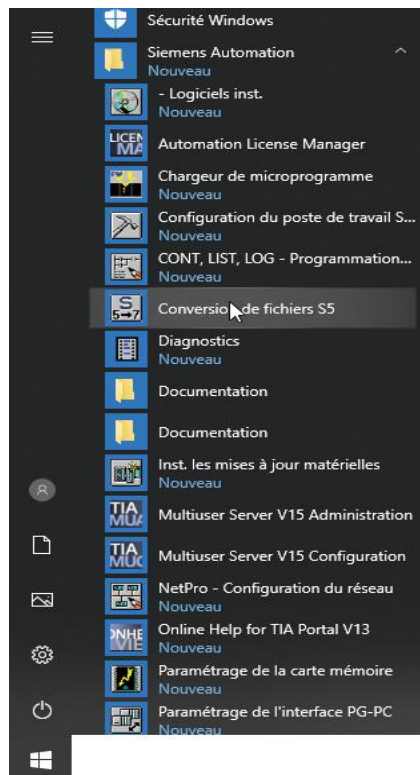


Figure IV.2 : Ouverture du convertisseur S5.

- Ouvrir le fichier de programme S5 dans l'outil « Conversion de fichiers S5 ». (→Fichier → Ouvrir)

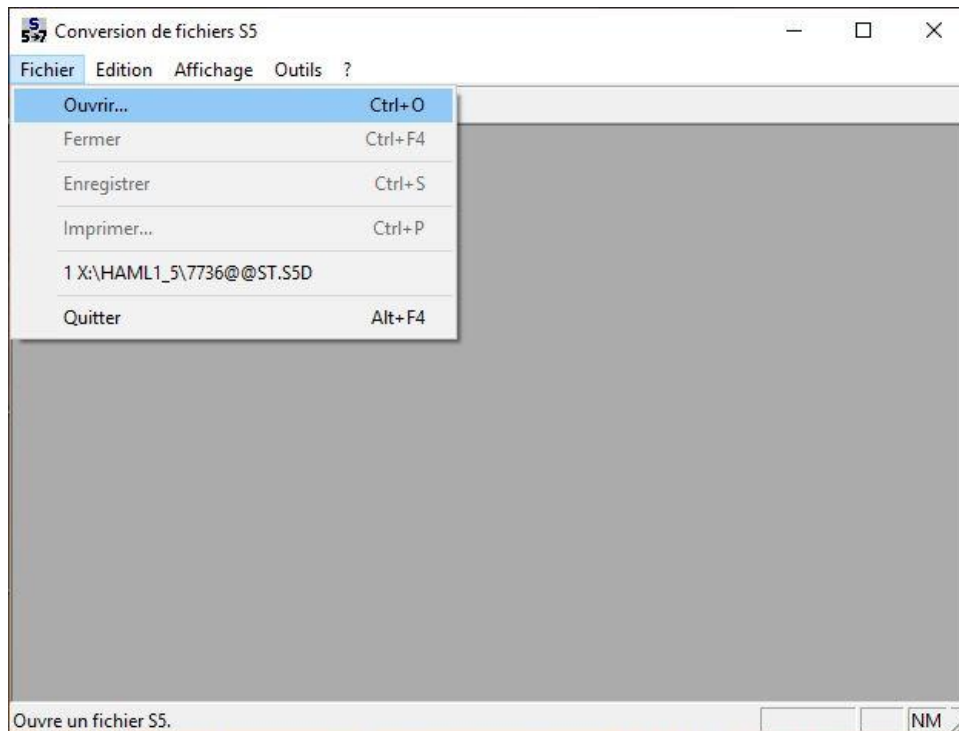


Figure IV.3 : Ouverture du fichier S5 à convertir.

- Sélectionner le fichier de programme S5 qui se termine par, « **ST.S5D** ». (→ segmenST.S5D → OK)

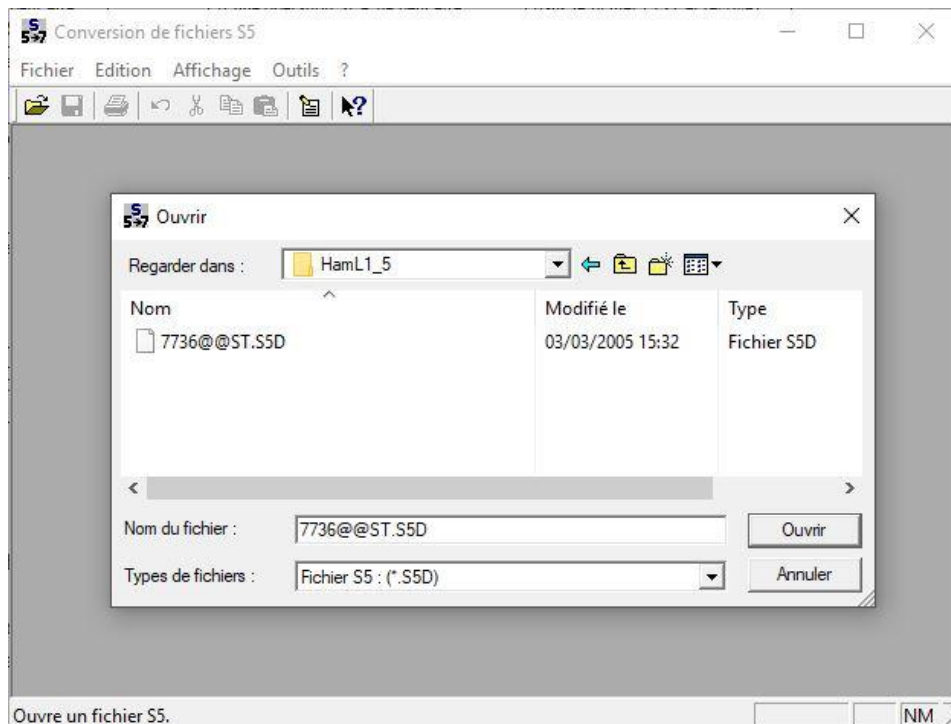


Figure IV.4 : Choix du fichier à convertir.

4. Dans les champs d'édition qui apparaissent alors, la totalité des fichiers et des blocs concernés par la conversion est affichée.
En cliquant sur un nom de fichier ou de bloc, nous pouvons changer les entrées par défaut des noms des nouveaux fichiers et des numéros des blocs.
Valider les noms et démarrer la conversion avec le bouton , « Convertir ». (→ Convertir).

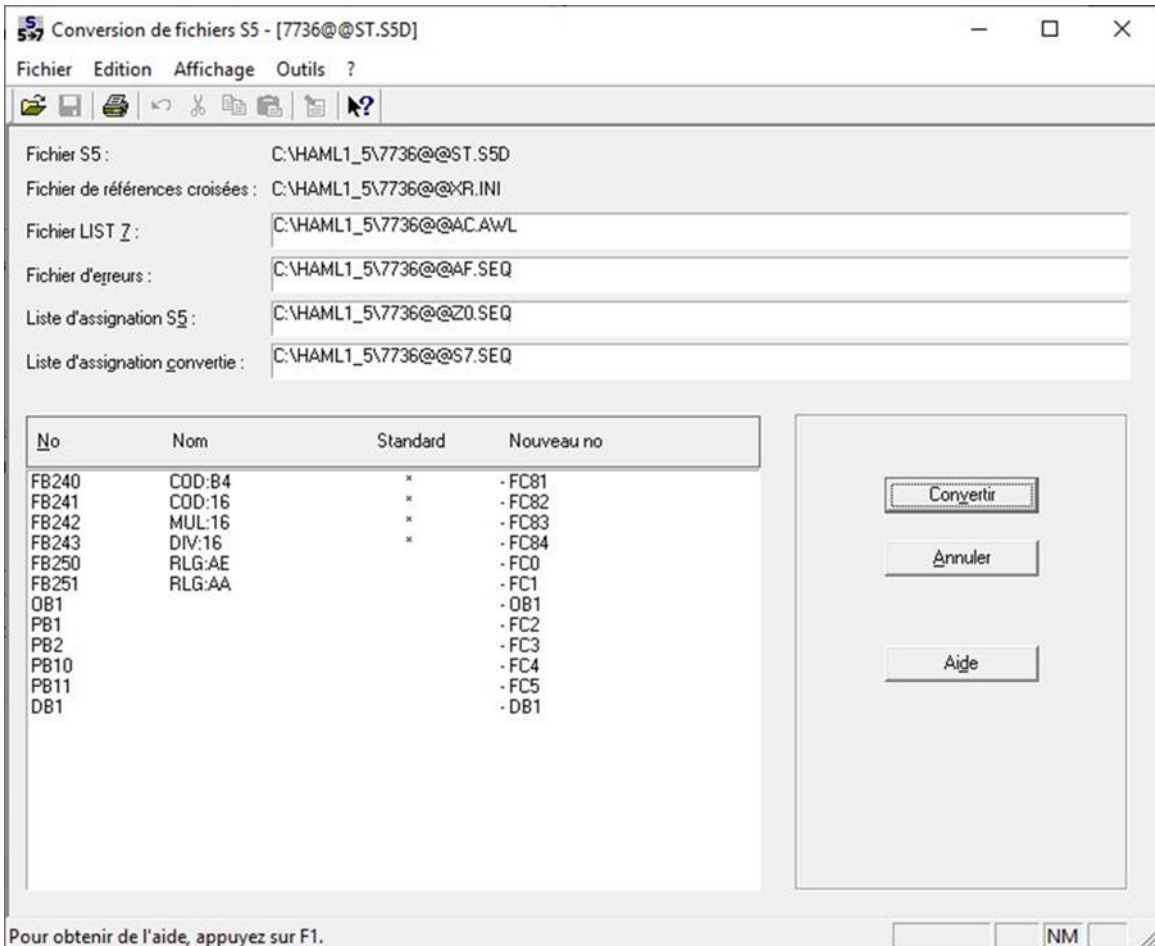


Figure IV.5 : Menu de conversion.

5. Pendant la conversion, une fenêtre (fenêtre de statut) montre le statut actuel de la conversion. En cliquant sur le bouton « Annuler » nous pouvons arrêter prématurément la conversion.
6. Confirmer ensuite la fin de la conversion en cliquant sur le bouton « OK » de la fenêtre qui apparaît alors, (→ OK).



Figure IV.6 : Menu fin de conversion.

7. Dans le cas où des problèmes sont apparus lors de la conversion, une fenêtre séparée affiche les messages d'erreurs et d'avertissement. On y trouvera également des indications à la résolution des erreurs. Et nous allons effectuer maintenant les mesures d'aide indiquées pour l'élimination de ces erreurs.

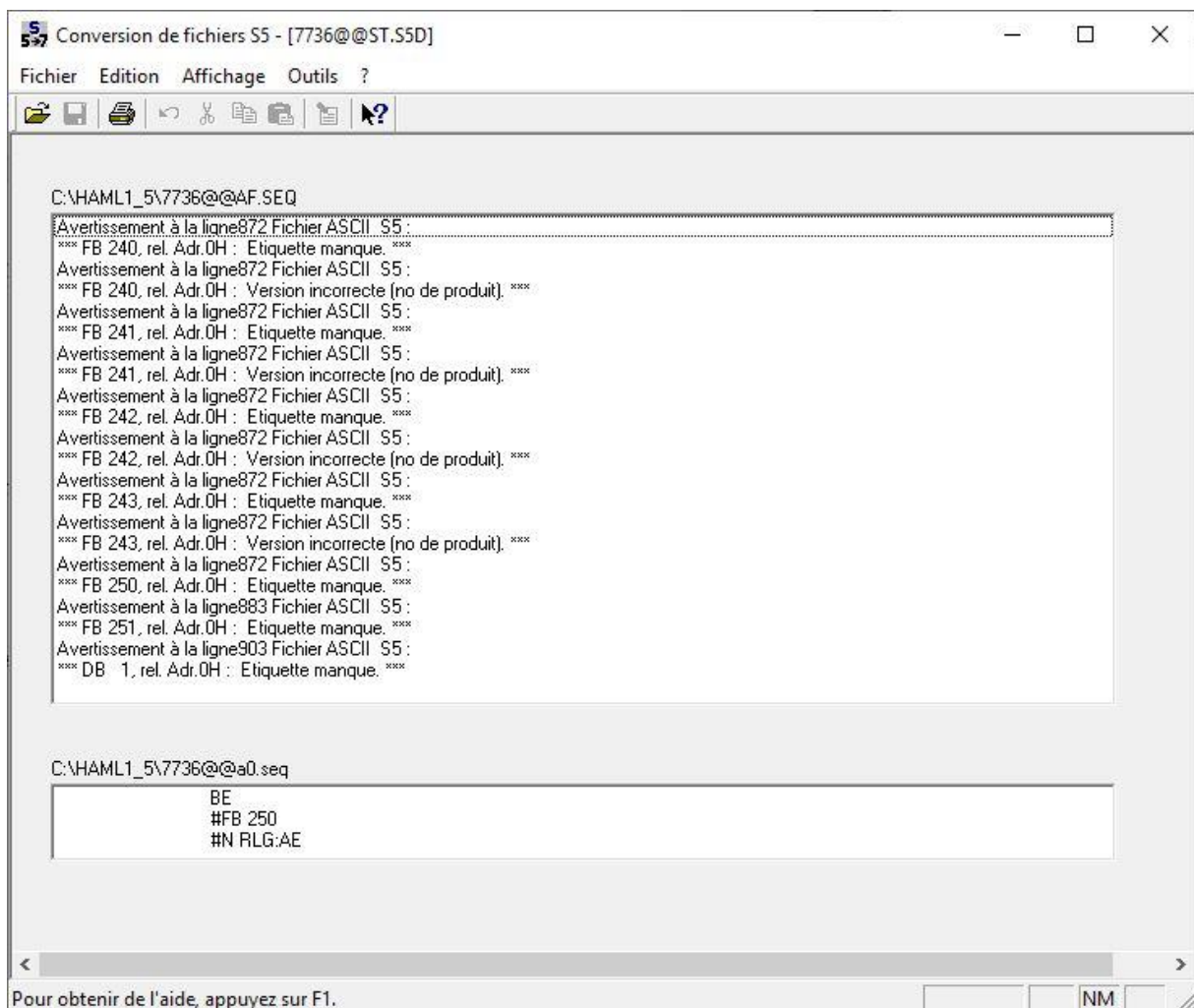


Figure IV.7 : Messages d'erreurs et avertissements.

IV.2.3 Vérification et retouche du programme converti

Maintenant que la conversion est complétée nous allons procéder à la correction des erreurs et avertissements signaler par le convertisseur.

IV.2.3.1 Préparation

Les étapes de préparation suivantes sont nécessaires avant de retoucher le fichier source LIST généré :

- Imprimer les messages.
- Créer un programme S7 dans un projet via le gestionnaire de projets SIMATIC si cela n'est pas déjà fait.
- Importer, à l'aide de la commande **Insertion** → **Source externe**, le programme source LIST généré dans le classeur "Sources" du programme S7 que nous avons créé.
- Ouvrir le fichier converti.

IV.2.3.2 Compilation

Nous devons compiler le programme converti et éventuellement le retoucher avec le compilateur LIST afin de le rendre exécutable. Pour ce faire nous devons procéder exactement comme pour le fichier texte que nous venons de créer.

IV.2.3.3 Vérification de la cohérence :

La commande **Fichier** → **Vérifier la cohérence** nous permet de vérifier à tout moment la syntaxe et la cohérence du fichier source sans toutefois déclencher la génération des blocs.

La vérification porte sur :

- La syntaxe
- Les mnémoniques
- Et l'existence des blocs appelés dans le programme.

Nous obtenons ensuite un protocole indiquant le nom du fichier compilé, le nombre de lignes compilées ainsi que le nombre d'erreurs et d'avertissements.

IV.2.3.4 Compilation du fichier source :

Un protocole s'affiche après la compilation. Il indique les erreurs éventuelles comme après la vérification de cohérence. Si vous avez programmé plusieurs blocs dans un fichier source, seuls ceux sans erreur sont compilés et sauvegardés.

IV.2.3.5 Correction des erreurs :

Si notre programme converti contient des erreurs ou des avertissements, ceux-ci sont énumérés après la vérification de cohérence ou la compilation dans une sous-fenêtre en dessous du fichier source.

La cause de l'erreur est également précisée. Lorsque nous sélectionnons un message d'erreur, l'emplacement correspondant du fichier source s'affiche dans la fenêtre supérieure, cela nous permet de remédier rapidement aux erreurs éventuelles.

Nous pouvons procéder aux corrections et aux modifications en mode de substitution que nous activons à l'aide de la touche d'insertion.

IV.2.3.6 Réalisation des vérifications et retouches

Après une conversion réussie, le fichier créé en LIST doit être intégré dans un projet STEP7. Ce projet va être créé avec « **SIMATIC Manager** », (→SIMATIC Manager) pour cela :

1. Nous allons créer un nouveau projet (→ Fichier → Nouveau).

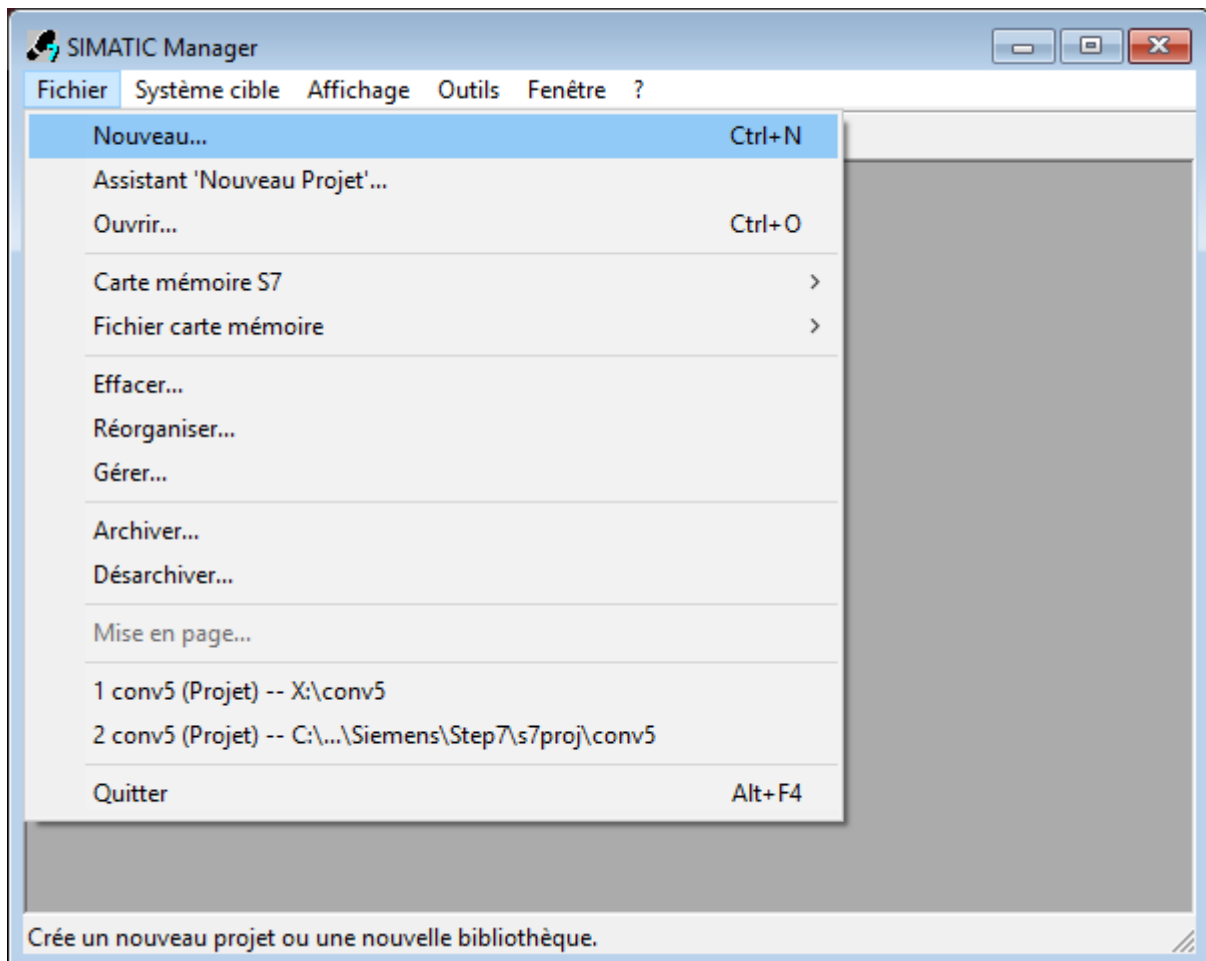


Figure IV.8 : SIMATIC Manager création d'un nouveau projet.

2. Créer au niveau du chemin par défaut un projet avec le nom « conversion_S7 », (→OK)

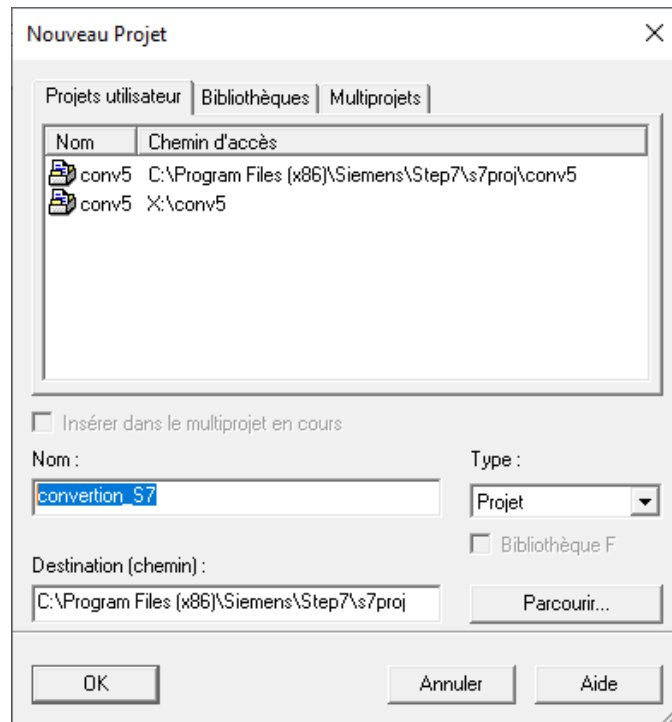


Figure IV.9 : SIMATIC Manager création du projet.

3. Insérer un nouveau programme S7 (→Insertion →Programme →Programme S7).

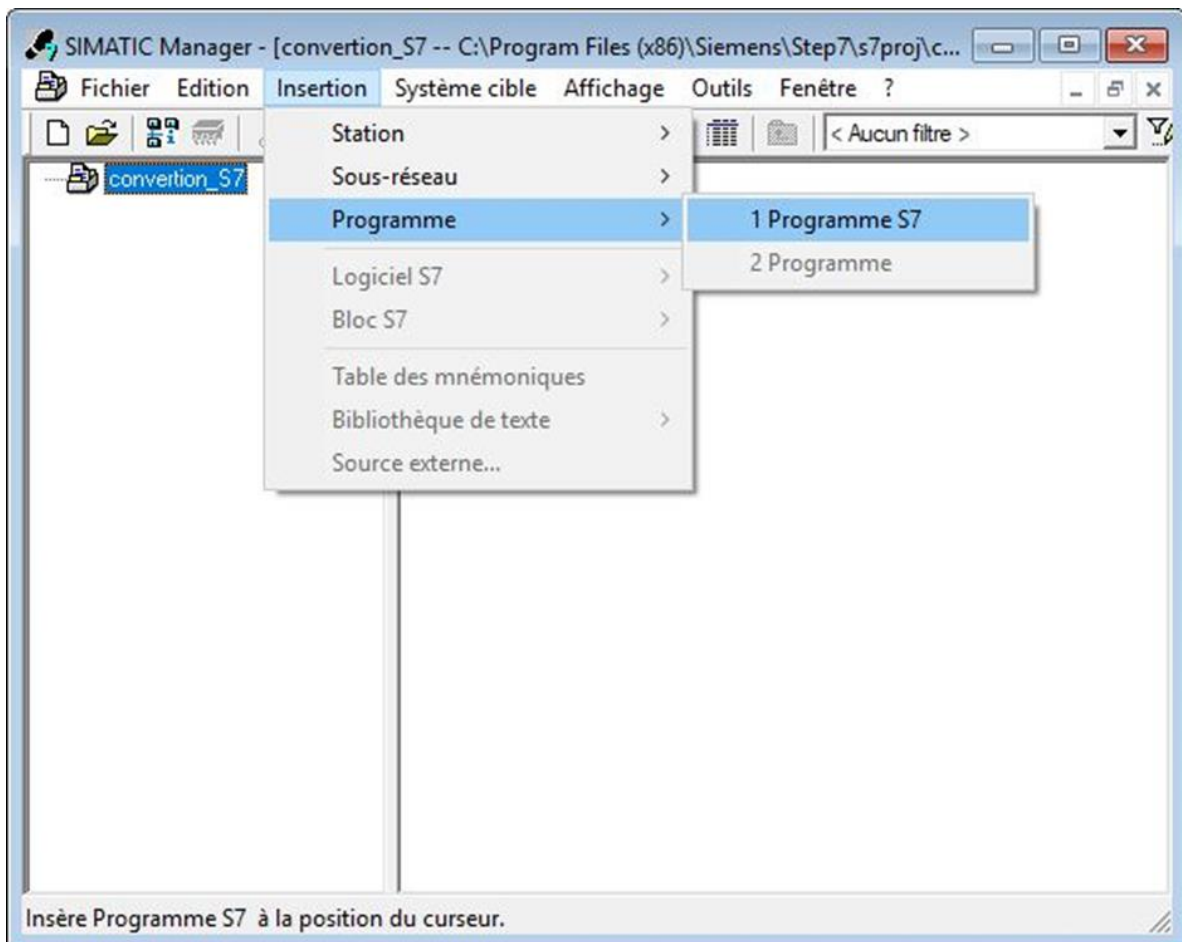


Figure IV.10 : Insertion d'un nouveau programme S7.

4. Sélectionner le dossier « Sources », (→ Sources). Ensuite nous allons insérer les fichiers *.AWL stockés dans le répertoire défini lors de la conversion comme « Source externe ». (→ Insertion → Source externe).

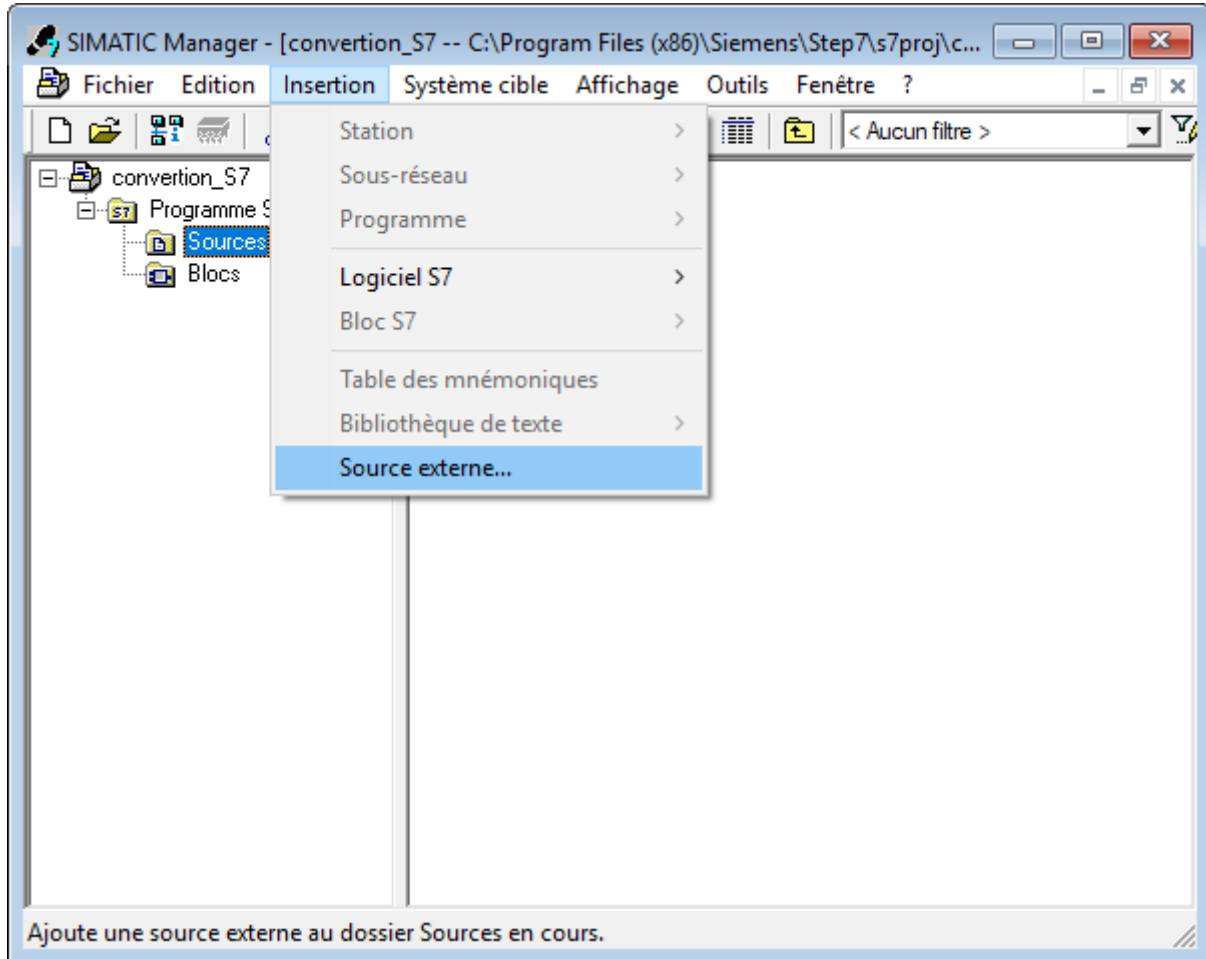


Figure IV.11 : Insertion d'une source externe.

5. On choisit le fichier « **7736@@AC.AWL** », (→ **7736@@AC.AWL** → Ouvrir).

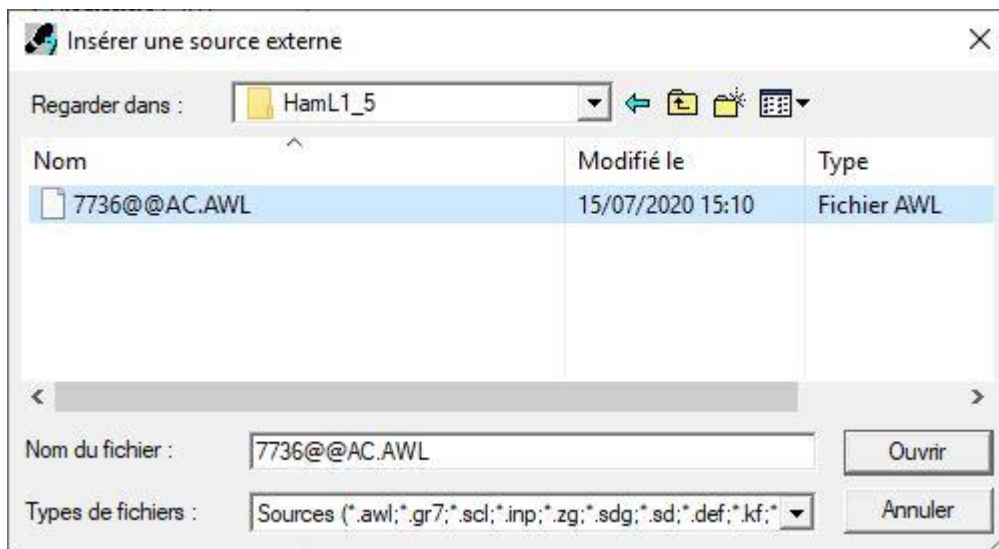


Figure IV.12 : Choix du fichier .AWL.

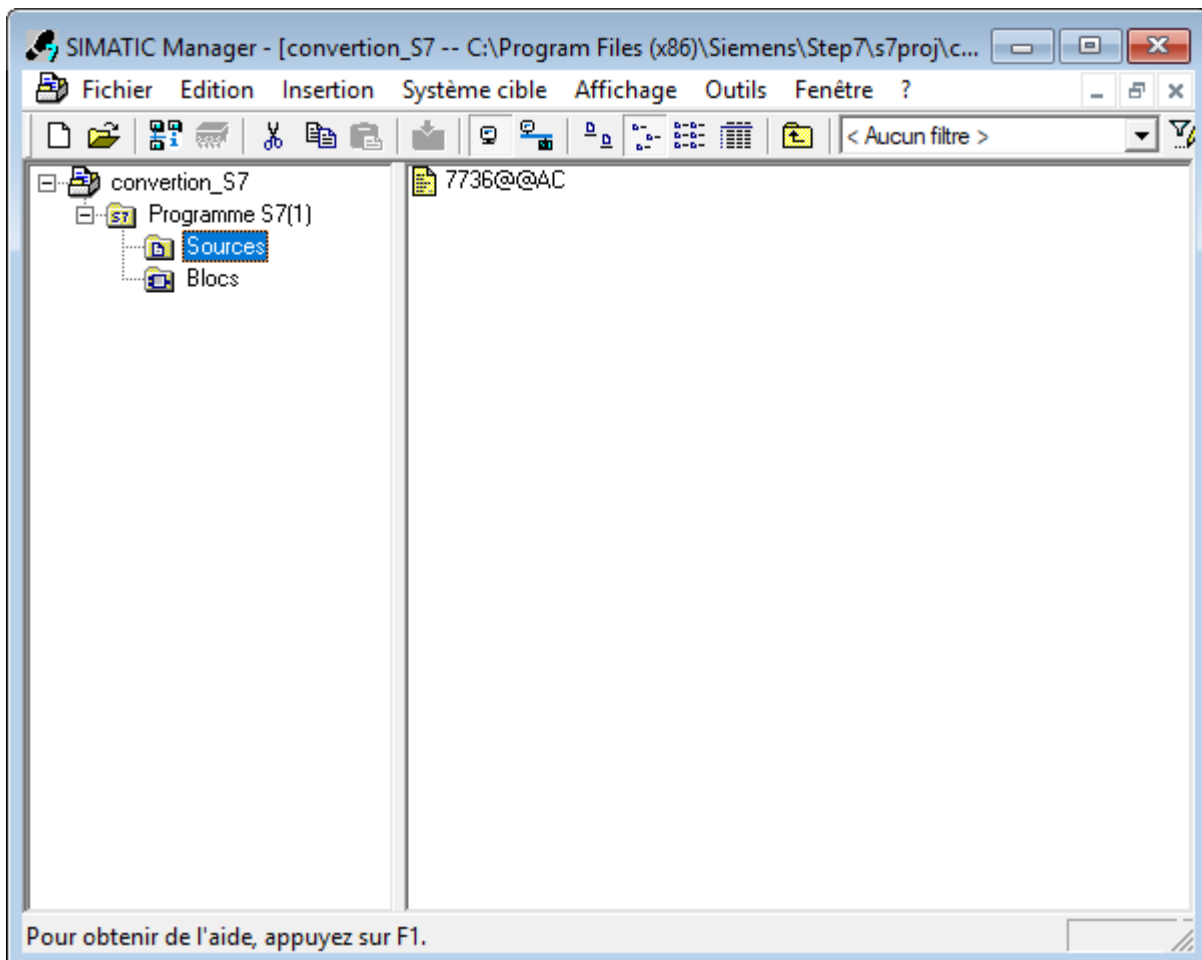


Figure IV.13 : Vue sur fenêtre SIMATIQUE Manager une fois le fichier source insérer.

6. A présent nous pouvons ouvrir le fichier sources LIST et le compiler :

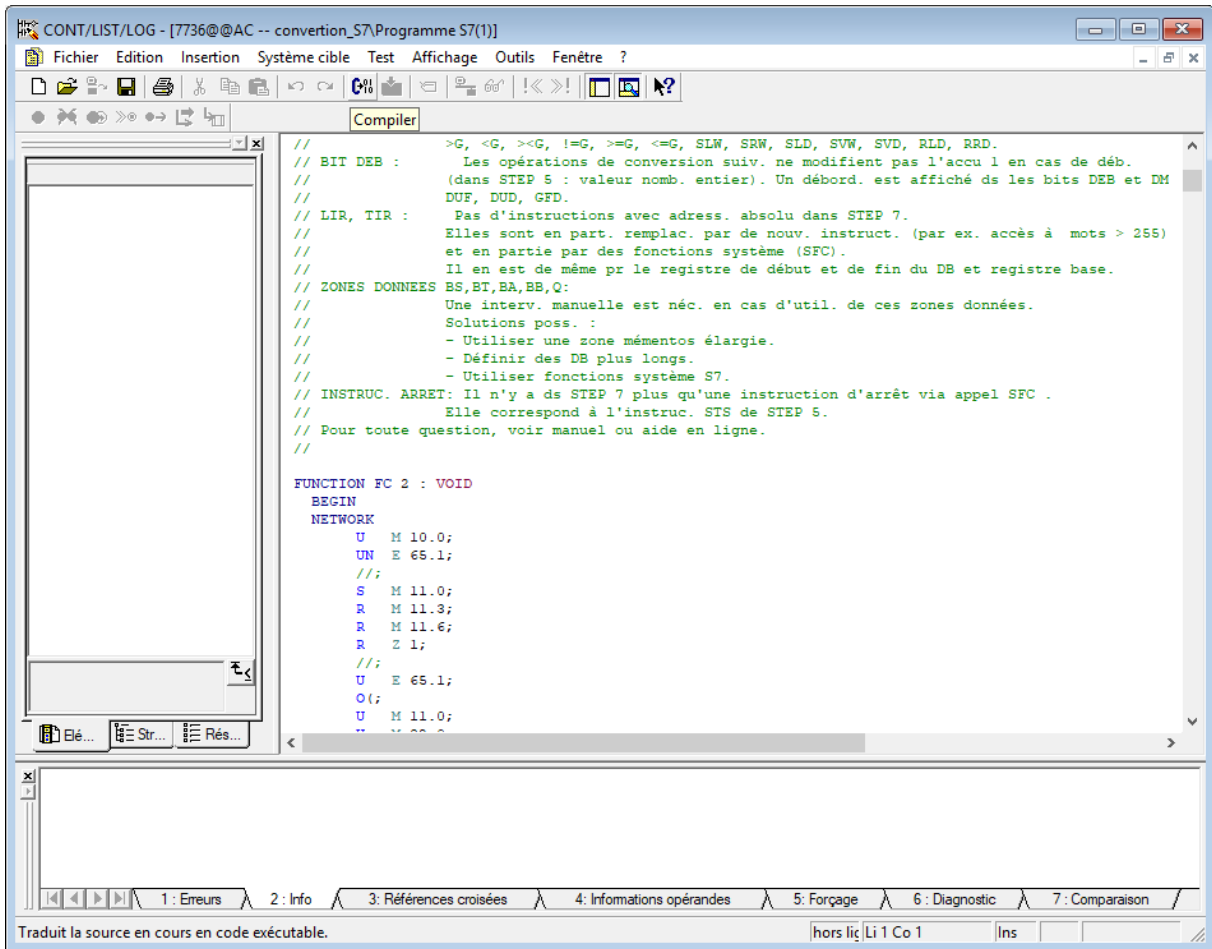


Figure IV.14 : Vue sur sources LIST.

Sur cette fenêtre nous pouvons visualiser toutes les lignes du programme, effectuer des vérifications et des modifications sur le programme afin de corriger les éventuelles erreurs et d'obtenir un programme fonctionnel.

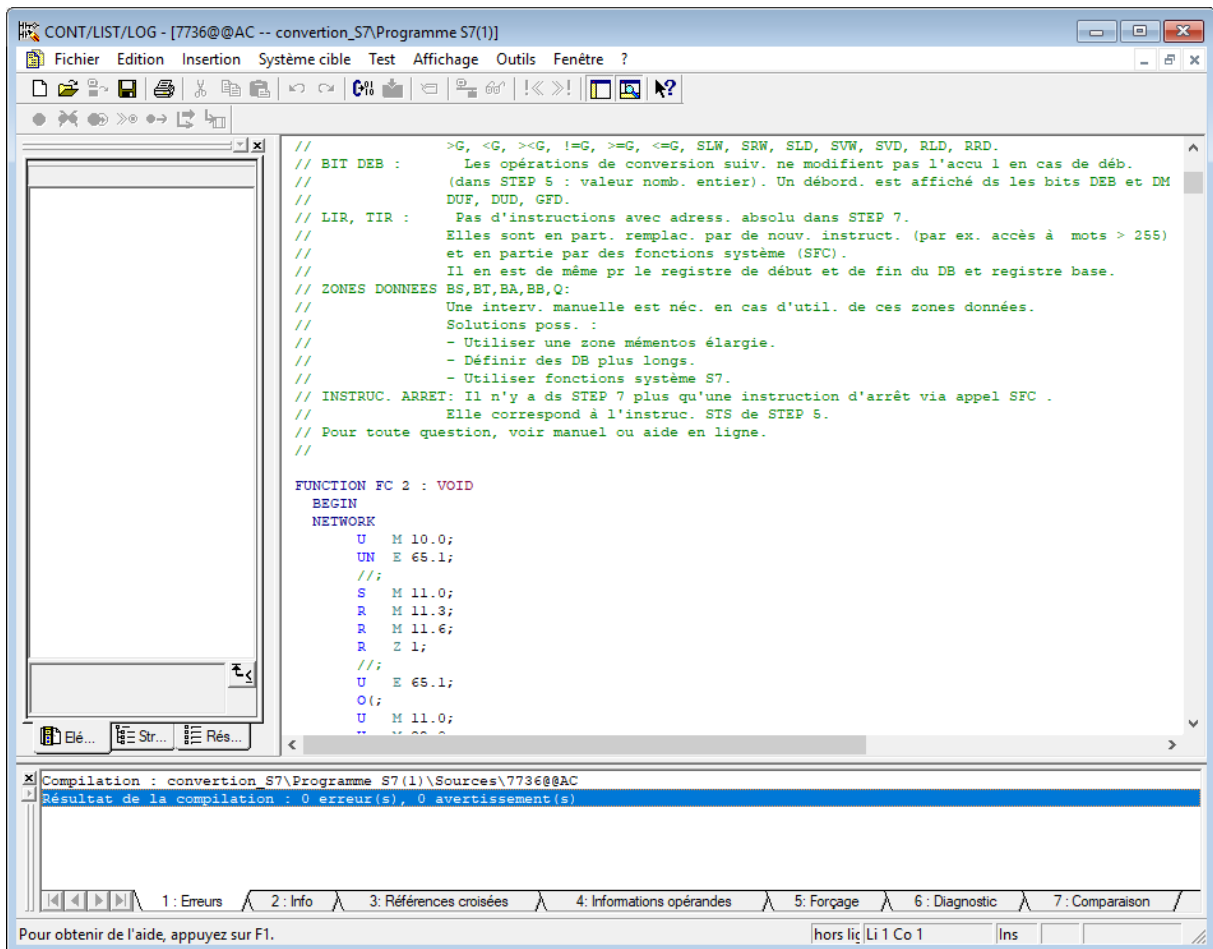


Figure IV.15 : Résultats de compilation.

Ici nous pouvons voir que le résultat de la compilation donne 0 erreurs et 0 avertissements, donc le programme a été correctement converti vers le STEP7 et qu'il est fonctionnel.

IV.2.4 Configuration des matériels sur HW Config

Par « configuration », on entend dans ce qui suit la disposition de profilés support ou chassais, de modules, d'appareils de la périphérie décentralisée de cartouche interface dans une fenêtre de station. Les profils support ou châssis sont représentés par une table de configuration, dans laquelle on peut enficher un nombre défini de module, tout comme dans les profilés support ou châssis « réels ».

STEP7 effectue automatiquement une adresse à chaque module dans la table de configuration. On peut modifier les adresses des modules d'une station, à condition que la CPU permette l'adresse libre.

On peut copier la configuration aussi souvent que nous désirons dans d'autre STEP7, la modifier si besoin et la charger dans une ou plusieurs installations existantes. A la mise en route de l'automate programmable, la CPU compare la configuration prévue créée avec STEP 7 à la configuration sur site de l'installation. Aussi, les erreurs éventuelles sont immédiatement détectées et signalées.

IV.2.4.1 Paramétrage

Par « paramétrage », on entend ce qui suit :

Le réglage des paramètres module paramétrables pour la configuration centralisée et pour un réseau. Exemple : un CPU est un module paramétrable, la surveillance du temps de cycle est un paramètre que on peut définir.

La définition des paramètres de bus, des maitres d'esclaves pour un réseau maitre (PROFIBUS) ou d'autres définitions pour l'échange de données entre des composantes. Ces paramètres sont chargés dans la CPU qui, lors de son démarrage, les transmet aux modules correspondants. Il est très facile de remplacer des modules, car les paramètres définis avec STEP 7 sont automatiquement chargés dans les nouveaux modules à la mise en route.

La configuration est obligatoire :

- Lorsque on souhaite modifier les paramètres prédéfinis d'un module (par exemple : valider l'alarme de processus pour un module).
- Lorsqu'on souhaite configurer des liaisons de communication.
- Pour les stations avec une périphérie décentralisée (PROFIBUS DP ou PROFINET IO).
- Pour les stations S7-300 avec plusieurs CPU (multiprocesseur) ou châssis d'extension.
- Pour les automates à hautes disponibilité.[13]

IV.2.4.2 Manipulations de base pour la configuration matérielle

Pour réaliser la configuration d'un automate programmable on va utiliser deux fenêtres :

La fenêtre de station dans laquelle on va disposer le profil support/châssis pour la configuration de la station.

La fenêtre « catalogue du matériel » dans laquelle on va sélectionner les composants matériels requis, comme par exemple les profilés support ou châssis, les modules et cartouches interface.

Affichage du catalogue du matériel si la fenêtre « catalogue du matériel » ne s'affiche pas, choisissons la commande affichage catalogue. Elle permet d'afficher ou de masquer le catalogue du matériel.

IV.2.4.3 Organisation de projet créé

Dans STEP 7, le projet regroupe la totalité des données et des programmes d'une solution d'automatisation, quels que soient le nombre d'unités centrales et leur mise en réseau. Un projet n'est donc pas limité à un programme utilisateur d'un module programmable, mais pourra réunir sous un nom de projet commun plusieurs programmes utilisateur pour plusieurs modules programmables.

Pour notre projet, il contient un bloc d'organisation (OB1), 5 fonctions (FC0, FC1, FC2, FC3, FC4, FC5) et un bloc de données (DB1).

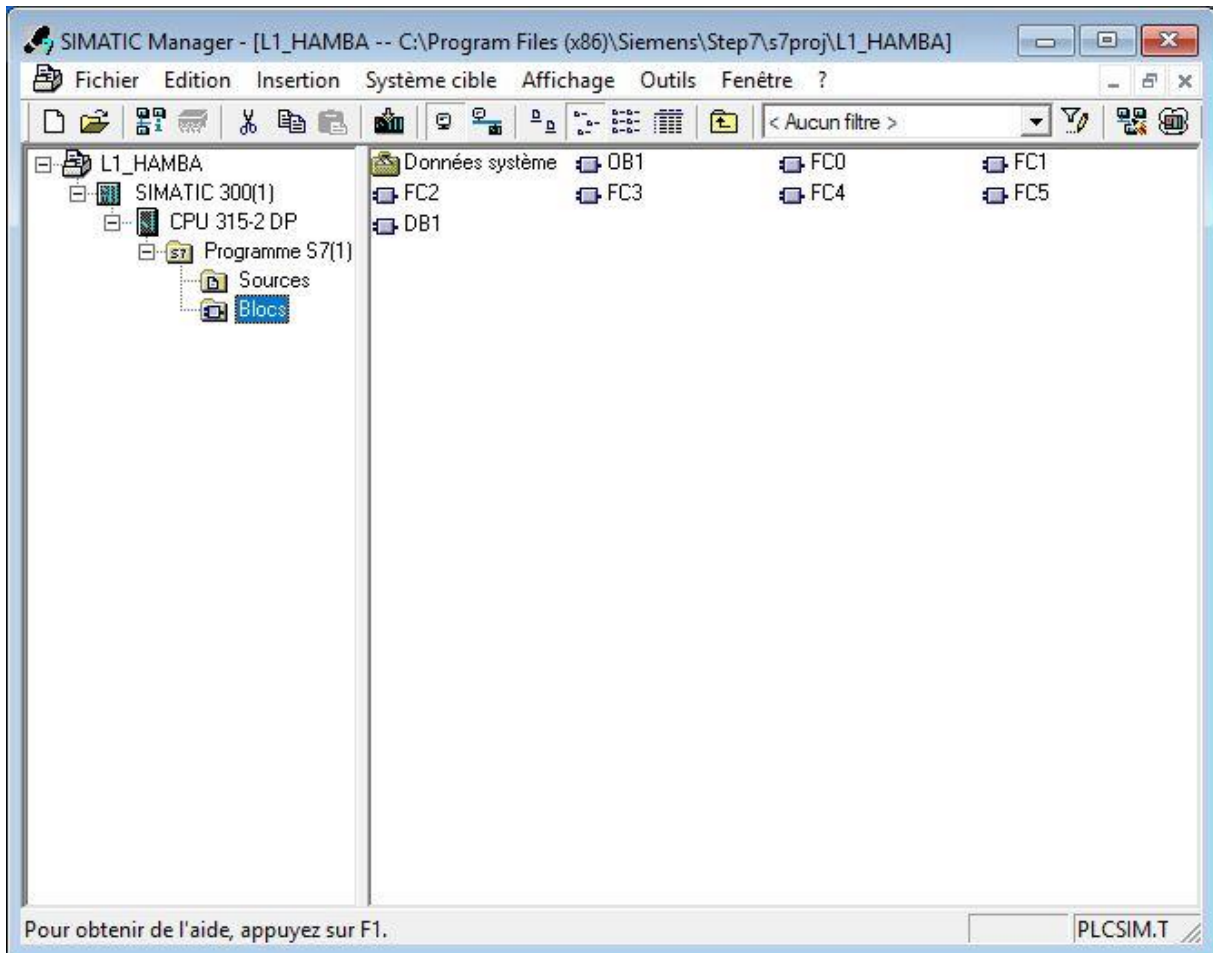


Figure IV.16 : projet créé sur STEP7.

- Vérification de cohérences :

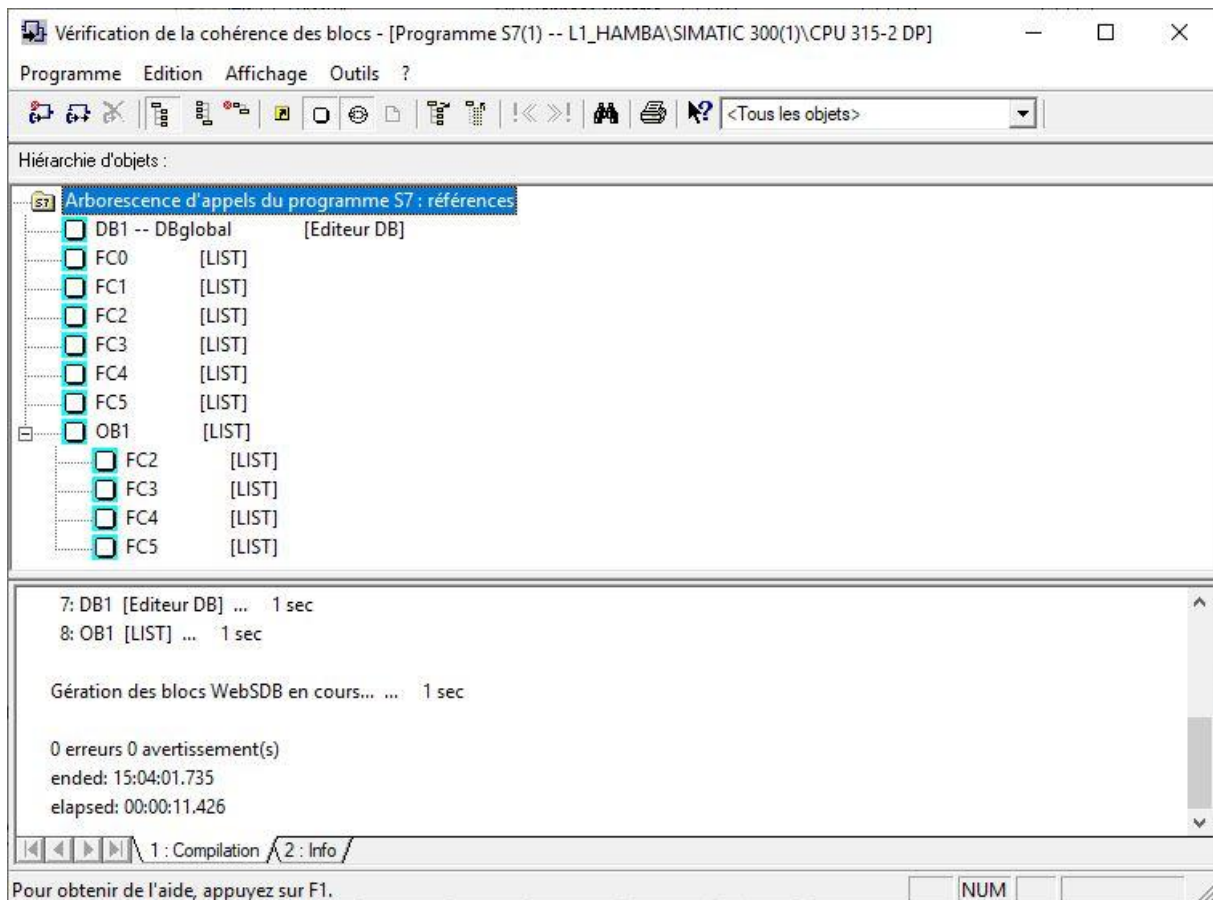


Figure IV.17 : Résultat de la vérification de cohérence des blocs.

IV.3 Migration S7 vers TIA PORTAL

Le TIA Portal est le nouvel environnement de développement et d'ingénierie de SIEMENS pour les applications SIMATIC STEP7 et SIMATIC WinCC. TIA Portal a notamment donné naissance à de nouvelles CPU S7-1200 et S7-1500, accompagnées de leurs firmwares et propose une multitude de nouvelles fonctions [9]. Notre choix est porté sur cet environnement pour effectuer la migration dans le but de donner la possibilité de remplacer l'automate S7-300 par un automate S7-1500.

IV.3.1 Condition de Migration :

- Il existe un fichier déjà converti en format AM11 ou le logiciel d'origine, compatible avec toutes les configurations utilisées dans le projet source et installé avec une licence invalide.
- Le projet source ne possède pas de protection d'accès.
- Pour que la migration puisse s'effectuer correctement, le projet source doit être cohérent.

Remarque : durant la migration, le système ne doit pas se retrouver en mode veille ou de veille prolongée, si non la migration interrompue.

IV.3.2. Marche à suivre pour faire la migration :

1. Choisir la commande « Migrer le projet » dans le menu « Projet ». La boîte de dialogue « Migrer le projet » s'ouvre.
2. Entrer, dans le « champ Chemin » source, le chemin d'accès et le nom de fichier du projet à migrer, sélectionnez un projet converti au format de migration AMA11 ou existant au format du projet source.
3. Cocher la case « Exclure la configuration matérielle » pour faire migrer uniquement le logiciel. Si nous avons sélectionné un fichier de migration réalisé avec l'outil de migration, la case à cocher n'est pas accessible. Dans ce cas, il faut indiquer dès la conversion avec l'outil de migration si nous voulons exclure la configuration matérielle de la migration.
4. Dans le champ « Nom du projet », sélectionner un nom pour le nouveau projet.
5. Dans le champ « Chemin cible », sélectionner un répertoire dans lequel le nouveau projet doit être enregistré.
6. Dans le champ « Auteur », entrer le nom du responsable projet.
7. Au besoin ajouter un commentaire dans le champ « Commentaire ».
8. Cliquer sur « Migrer ».[14]

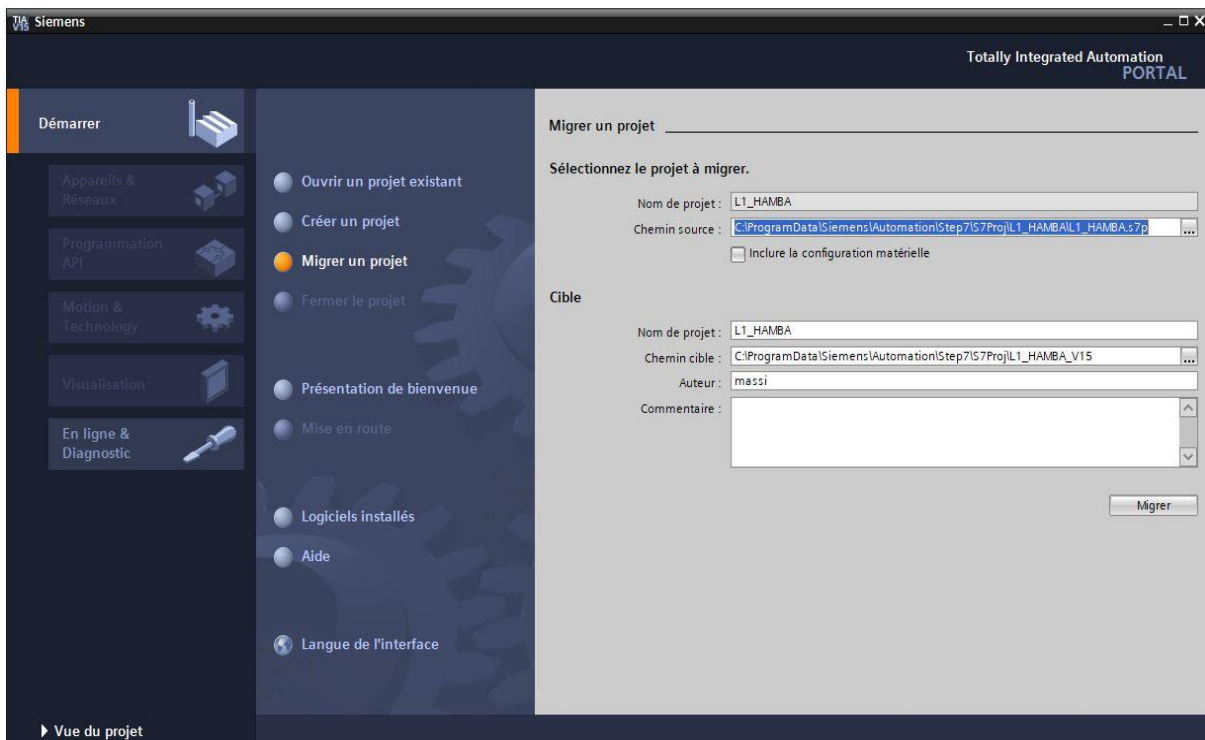


Figure IV.18 : Migration vers TIA Portal.

Migration du S5 vers S7

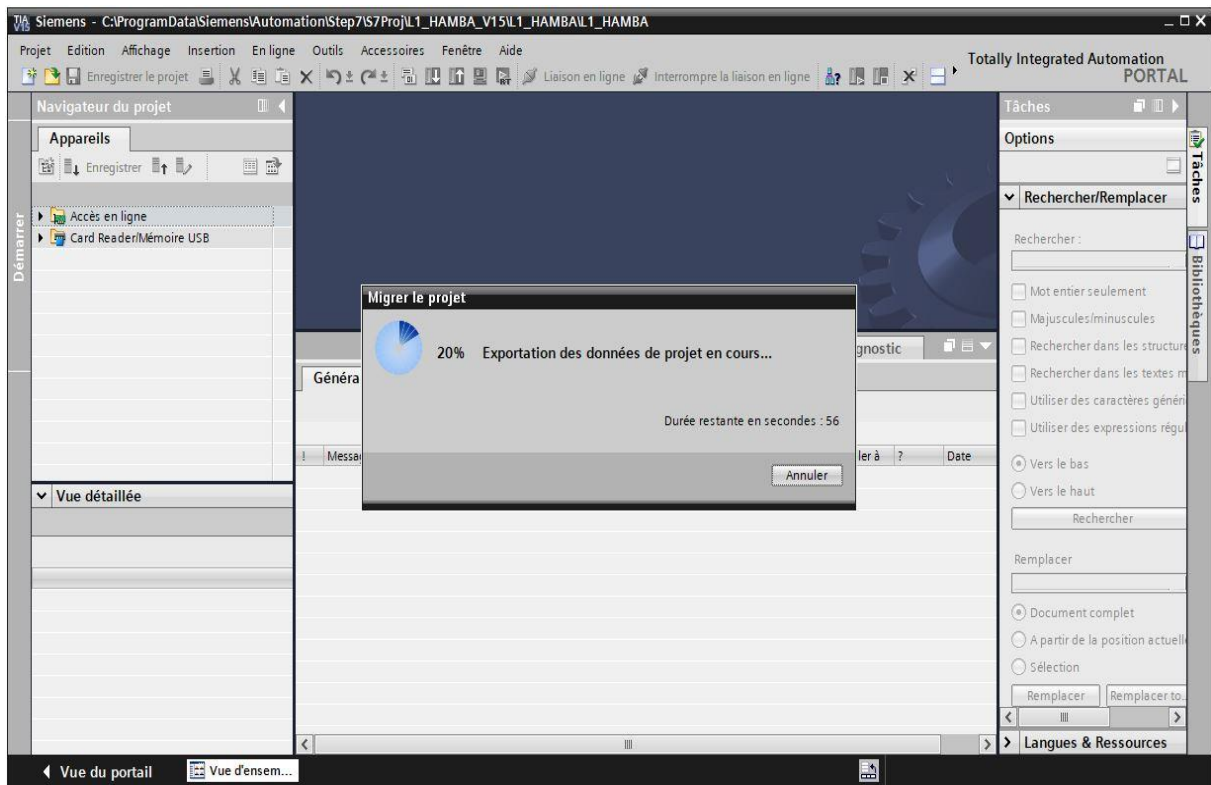


Figure IV.19 : Migration en cours.

IV.3.3 Résultats de la migration

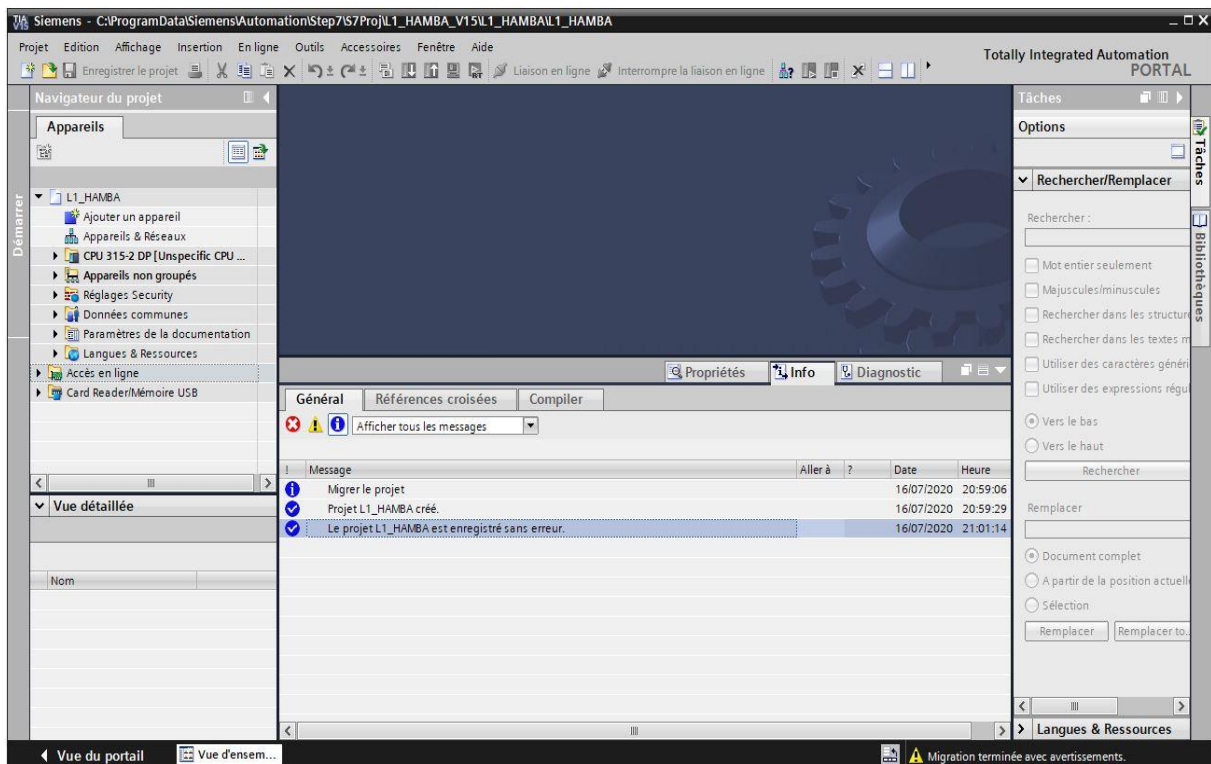


Figure IV.20 : Fin de la migration sans erreurs.

Migration du S5 vers S7

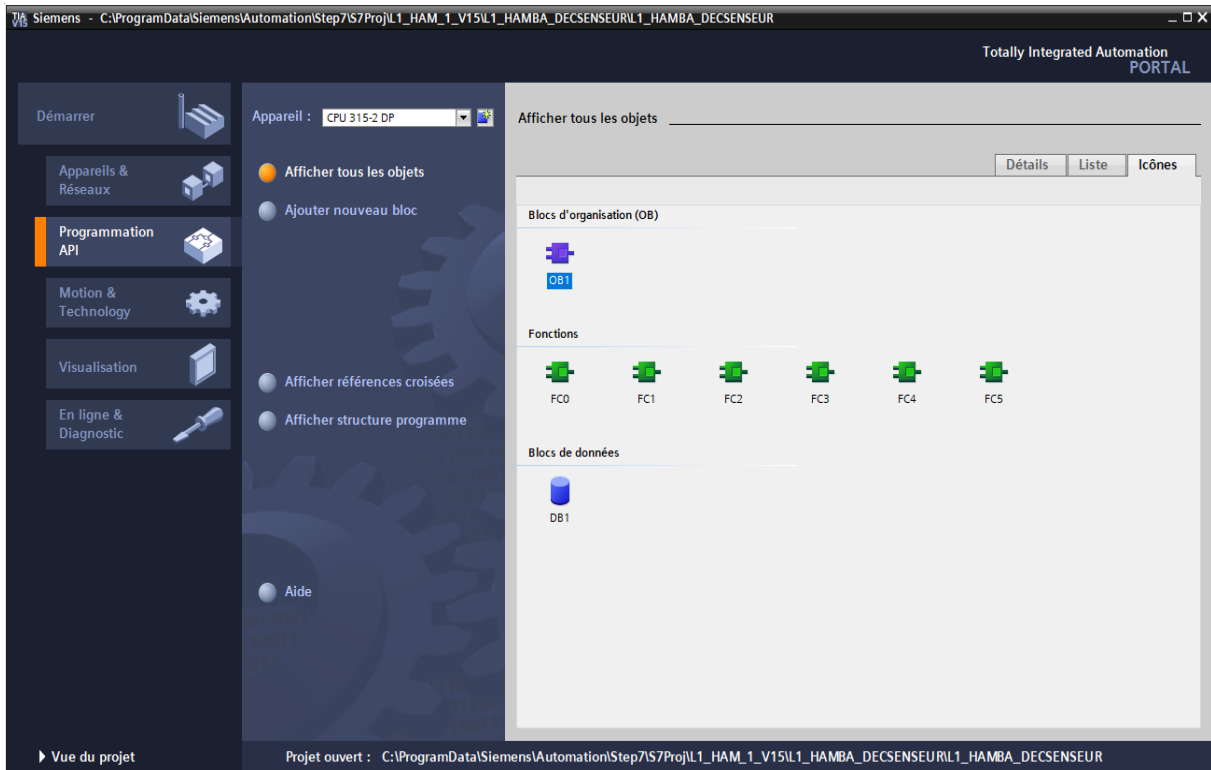


Figure IV.21 : Résultat de la migration.

Nom	Description	Type de données	Adresse	Réma.	Acces.	Visibl.	Commentaire
22	Remplir (couvertures)	Table de variables s. Bool	%I65.0				
23	pires inferieur (valve couverture)	Table de variables s. Bool	%Q65.2				
24	Cylindre pour piles au dessus (c.)	Table de variables s. Bool	%I73.2				Station de couvertures
25	Bouchon fermé (valve couverture)	Table de variables s. Bool	%Q65.3				
26	Bouchons pour piles fermé (co.)	Table de variables s. Bool	%I73.5				Station de couvertures
27	Lieu de transfert est libre LS (co.)	Table de variables s. Bool	%I72.2				Capteur station de couvertures
28	Tag_28	Table de variables s. Bool	%M13.4				
29	III en position (couverture)	Table de variables s. Bool	%I73.3				Capteur station de couvertures
30	Temporisation 30	Table de variables s. Timer	%T30				
31	Temporisation 31	Table de variables s. Timer	%T31				
32	Tag_32	Table de variables s. Bool	%M8.4				
33	Tag_33	Table de variables s. Bool	%M8.3				
34	Tag_34	Table de variables s. Bool	%M13.0				
35	Tag_35	Table de variables s. Bool	%M13.1				
36	Pas d'arrêt HAMB	Table de variables s. Bool	%I64.5				
37	Tag_37	Table de variables s. Bool	%M1.7				
38	Temporisation 1	Table de variables s. Timer	%T1				
39	Vigant	Table de variables s. Bool	%I64.1				
40	Temporisation 4	Table de variables s. Timer	%T4				
41	Remplir couvertures	Table de variables s. Bool	%Q64.2				
42	Protection moteurs (problemes)	Table de variables s. Bool	%I65.6				Problemes
43	vider couverture	Table de variables s. Bool	%Q64.3				
44	Bouchon ouvert (valve couv...	Table de variables s. Bool	%Q65.4				
45	Bouchons pour piles ouvert (co.)	Table de variables s. Bool	%I73.4				Station de couvertures
46	Cylindre pour piles en dessous	Table de variables s. Bool	%I73.3				Station de couvertures
47	Demande de couvertures	Table de variables s. Bool	%I64.7				Extérieur de la machine de remplissage
48	Préélectionner les conteneurs	Table de variables s. Bool	%I64.3				Possibilité de préélectionner les contenu
49	Option garde-corps ouvert (val...	Table de variables s. Bool	%Q65.1				
50	Option garde-corps fermé (valu...	Table de variables s. Bool	%Q65.0				
51	Remplissage (pots)	Table de variables s. Bool	%I65.4				
52	Tag_52	Table de variables s. Bool	%M12.0				
53	Tag_53	Table de variables s. Bool	%M12.3				
54	Tag_54	Table de variables s. Bool	%M12.6				
55	Compteur de pots	Table de variables s. Counter	%C2				Compteur 2
56	Tag_56	Table de variables s. Bool	%M82.4				
57	Espace de travail LS 1 (pots)	Table de variables s. Bool	%I72.4				Capteur station pots

Figure IV.22 : Table des variables sur TIA Portal.

Migration du S5 vers S7

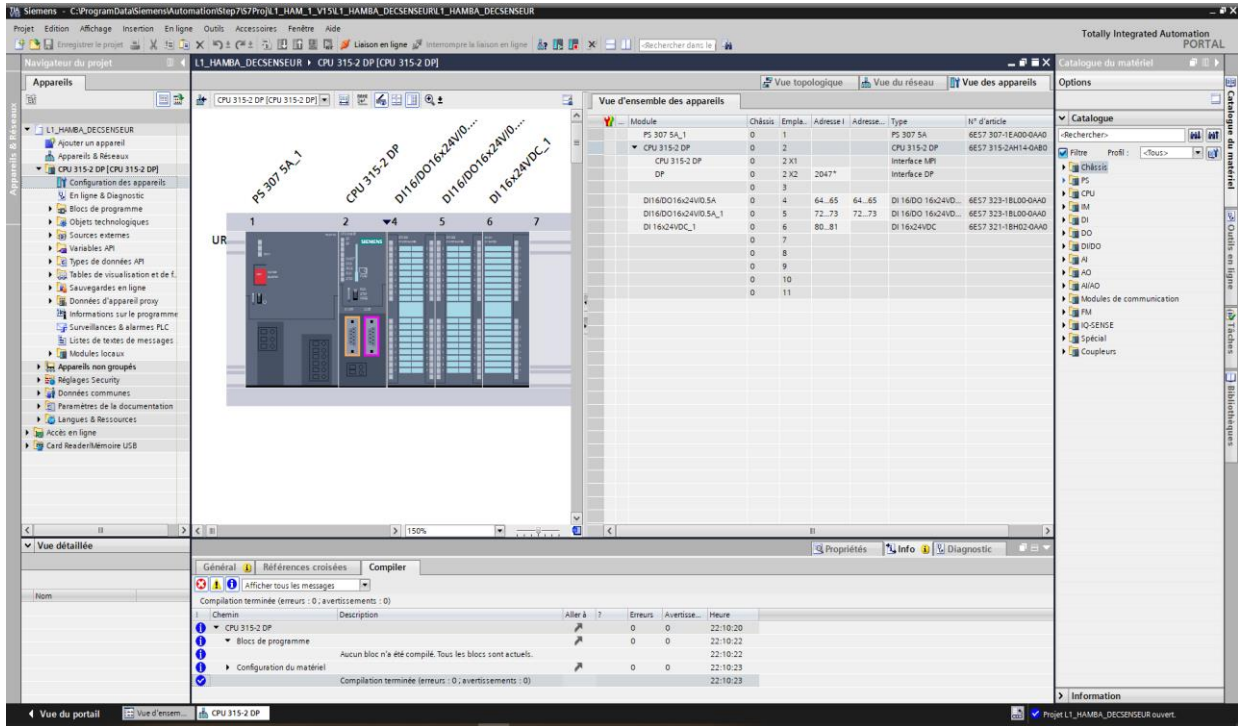


Figure IV.23 : Configuration matériel sur TIA Portal.

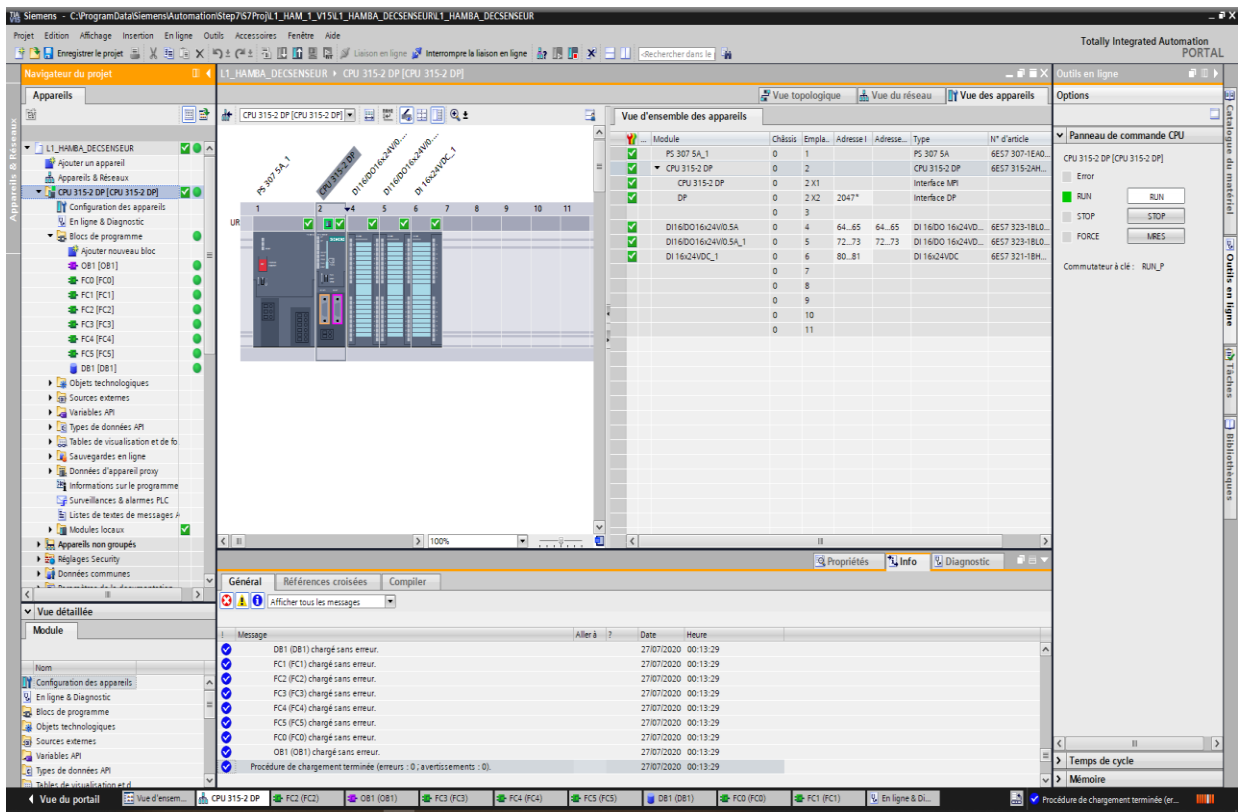


Figure IV.24 : Chargement des blocs, mise en ligne et simulation.

IV.4 Conclusion

La veille technologique est primordiale pour une entreprise qui vise à maintenir et développer ses activités. La migration vers la technologie SIMATIC S7 peut procurer des avantages décisifs vis-à-vis de la concurrence, grâce à l'amélioration de la fonctionnalité, de la disponibilité et de l'efficacité de notre automatisme.

Notre projet a consisté en l'étude de la migration de SIMATIC S5 vers SIMATIC S7 puis S7 vers TIA PORTAL de notre système « Descenseur de pots HAMBBA » avec une API S7-315 2DP.

Conclusion générale

Conclusion générale

La migration de SIMATIC S5 vers SIMATIC S7 se traduit tout d'abord par un accroissement de la productivité et de l'efficacité globale en réduisant les arrêts des machines et des installations et permet de bénéficier des possibilités de futures d'extension et de modernisation.

Du point de vu méthodologie, la collecte des données techniques nécessaires pour notre travail nous a permis de toucher plusieurs domaines : l'électrotechnique, l'électronique, la mécanique et l'informatique industrielle. . .

Nous avons entamé ce projet par une étude préliminaire de l'entreprise CEVITAL, son historique, ses produits et ses différentes activités. Nous nous sommes intéressés, en particulier, à la margarinerie.

Ensuite, nous avons abordé notre travail par l'étude des caractéristiques des automates S5-100U et le S7-300 et les logiciels Step5, Step7, et TIA Portal V15 qui font l'objet de notre étude de migration. Le descenseur est équipé d'un automate S5-100U, installé depuis 2001. Ce dernier est efficace et robuste et assure un bon fonctionnement. Mais, de nos jours, il pose deux problèmes majeurs :

- La non disponibilité de pièces de rechange.
- Diagnostic difficile.

Pour cela, le changement de l'automate S5-100U par un automate S7 -300, plus récent, est devenu une nécessité. Nous avons opté pour une approche en deux étapes. La première est basée sur la conversion avec un convertisseur S5/S7 et la deuxième et de faire la conversion S7/TIA PORTAL.

La réalisation de ce projet de fin d'études a été le point départ d'un nouveau tournant dans notre formation au sein de l'entreprise. Nous avons été amenés à faire des recherches, à échanger et à partager des points de vue, tout en mettant en pratique nos acquis théoriques.

Enfin, cette expérience restera pour nous une référence et un point de départ pour apprendre, développer et améliorer notre savoir-faire.

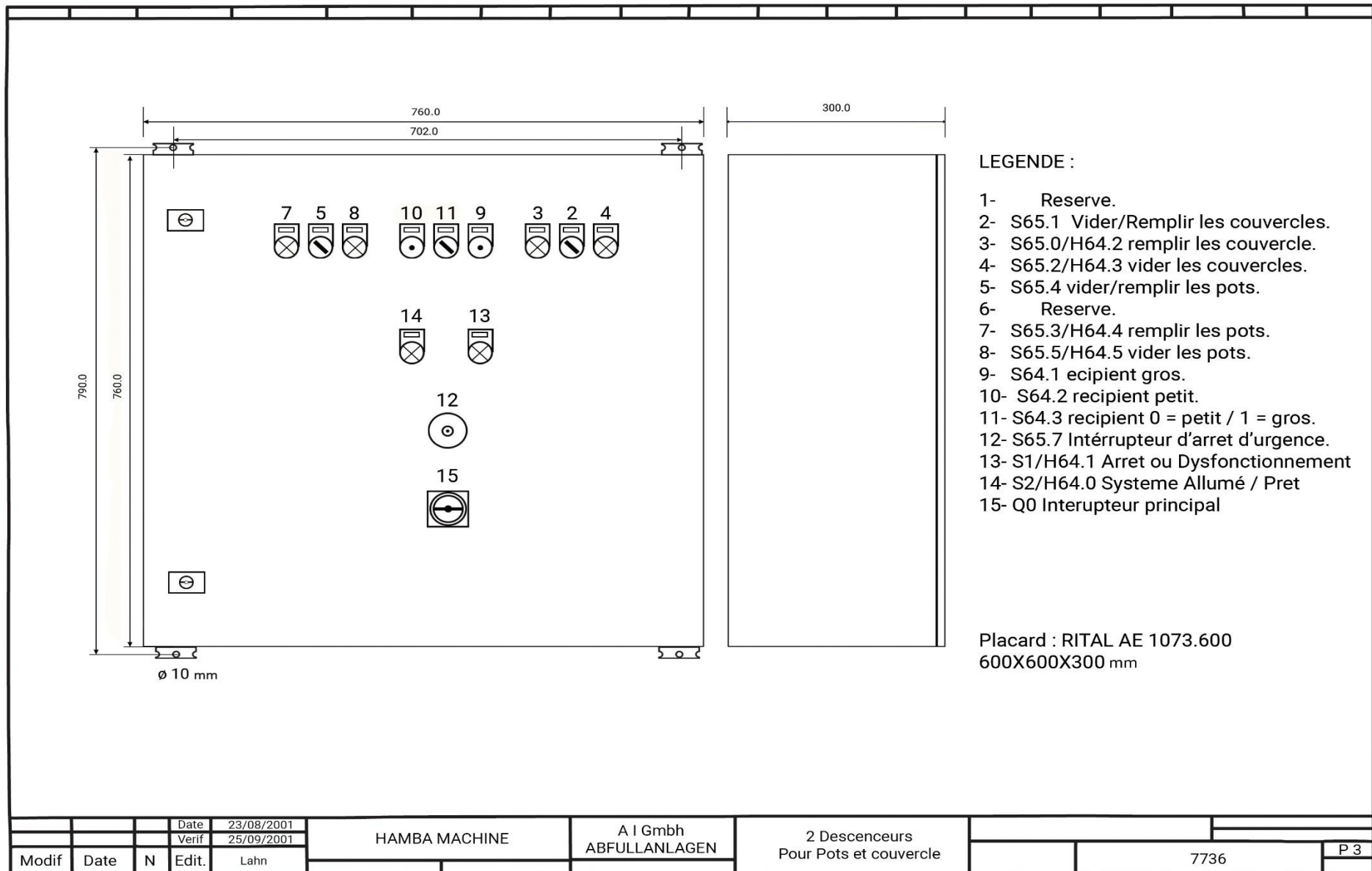
Références bibliographiques

BIBLIOGRAPHIE

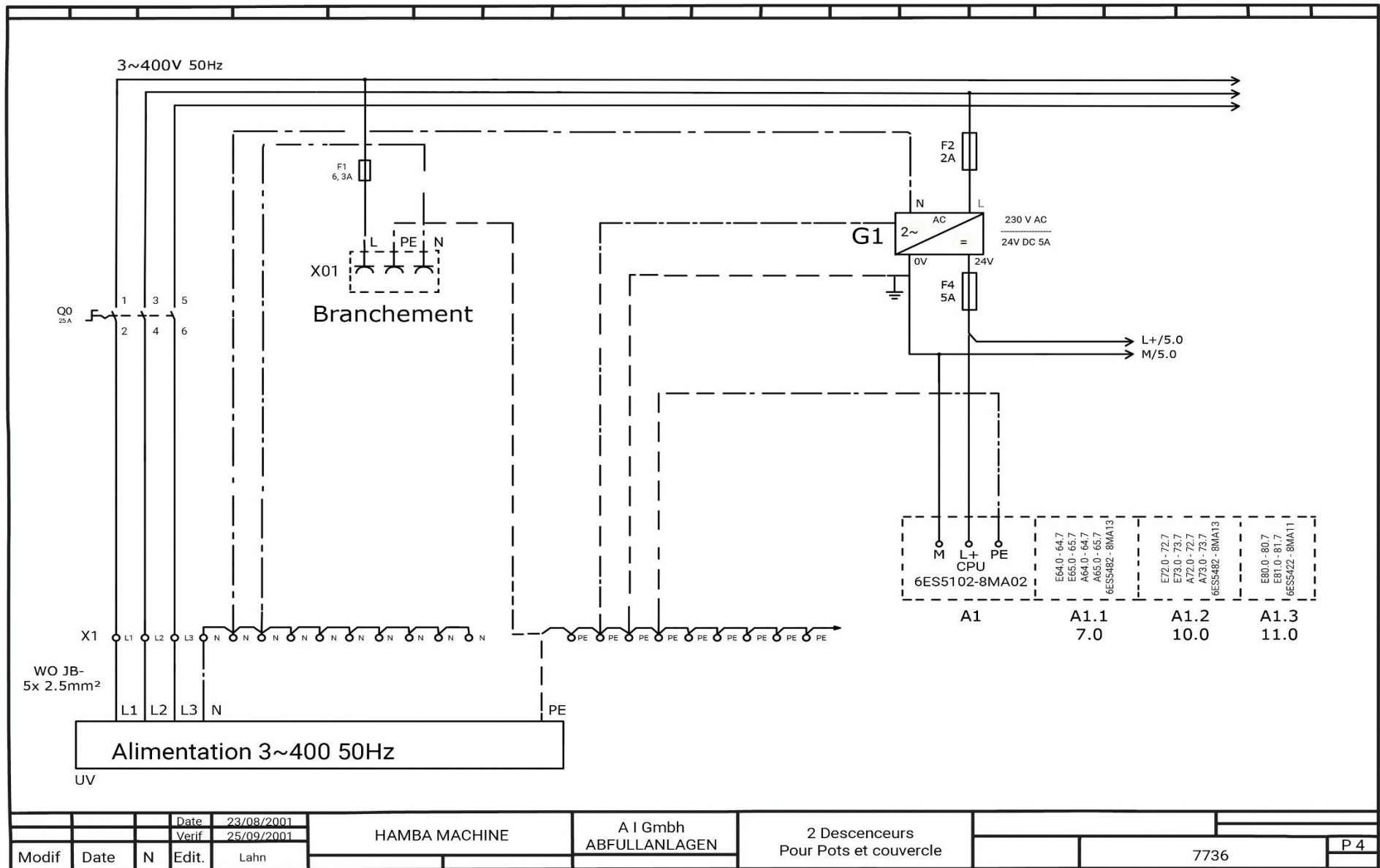
- [1] Site officiel de CEVITAL « www.cevital.com ».
- [2] C. VRIGNON et M. THENAISIE, « l'automatisation », ISTIA, 17 octobre 2005.
- [3] Alain GONZAGA, « Les automates programmables industriels », 2004.
- [4] André SIMON, « Automates programmables, programmation, et logique programmé », Edition L'ELANE, 1983.
- [5] Michel G, « Architecture et application des automates programmables industriels », DUNOD, Paris, 1987.
- [6] Catalogue Pièces de Rechange-HAMBA 8006M L1 (Document CEVITAL).
- [7] Siemens AG, « SIMATIC STEP 7 dans le portail Totally Integrated Automation Portal », 2013.
- [8] Hans Berger, "Automating with SIMATIC", second Edition, 2003.
- [9] Manuel SIEMENS STEP 7 « Pour une transition facile de S5 à S7 », Edition mai 2010.
- [10] Manuel Siemens Simatic STEP5, Edition 1992.
- [11] S Elec- Station-B & C-HAMBA 8006M L1 (Document CEVITAL).
- [12] Site official Siemens « www.siemens.com ».
- [13] Manuel Siemens Step 7 et WinCC engineering, Edition octobre 2018.
- [14] Guide for Migrating SIMATIC S5 Projects to SIMATIC S7-1500 and TIA Portal, Edition Mai 2015.
- [15] SIMATIC S5 Automate programmable S5-100U, Reference 6ES5998-OUB33 Edition 2006.

Annexes

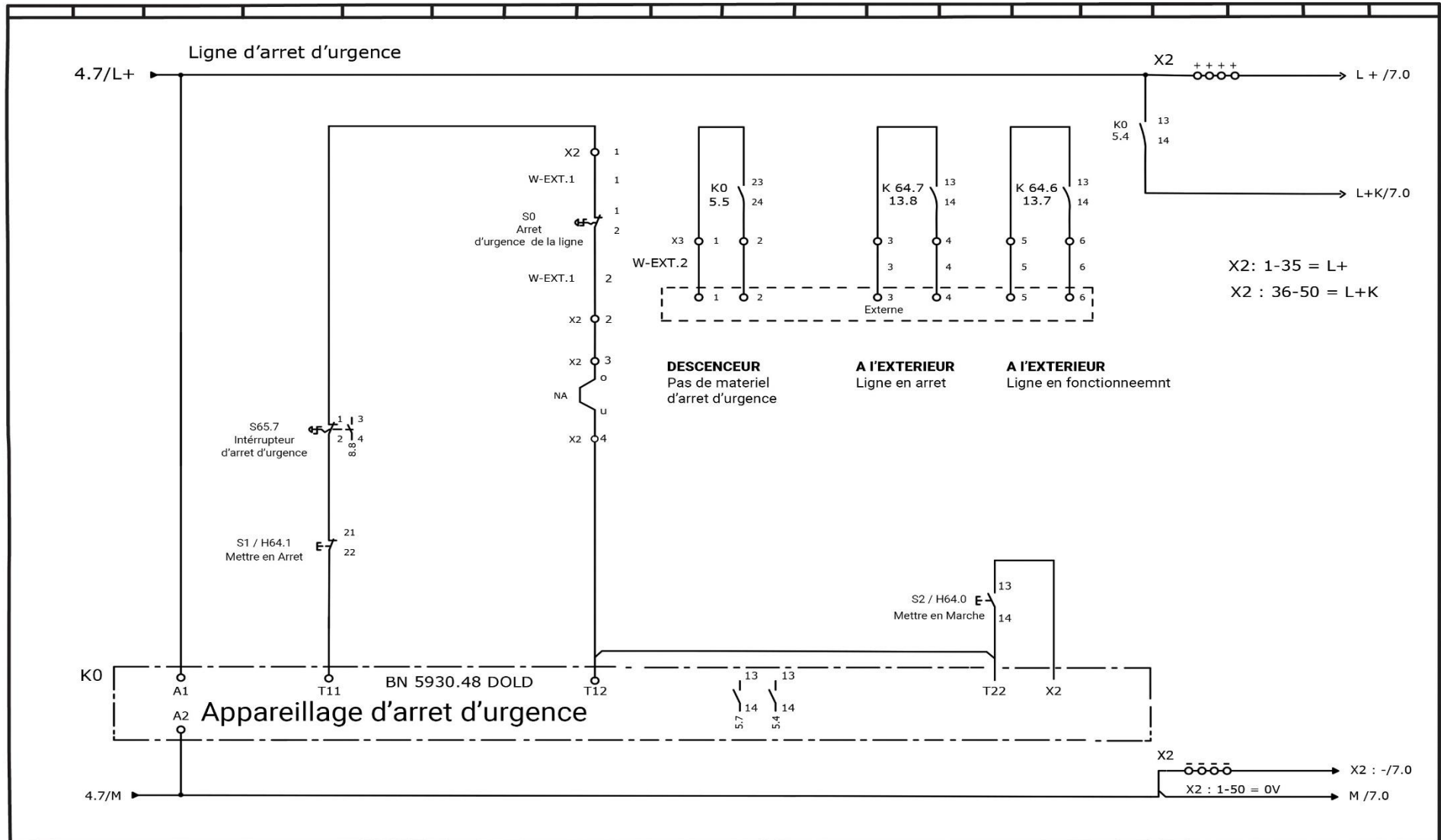
Armoire électrique :



Alimentation :

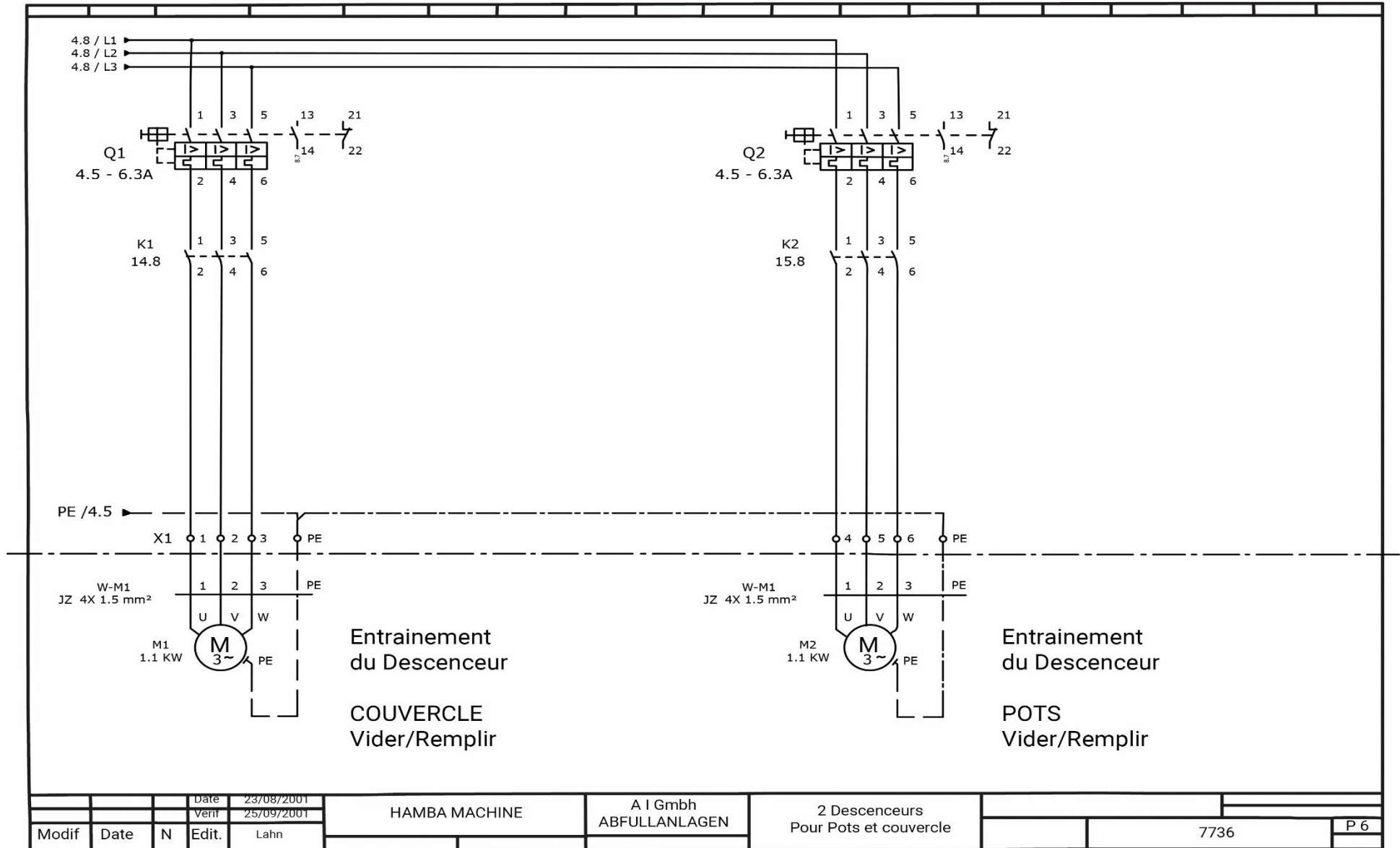


Circuit d'arrêt d'urgence :

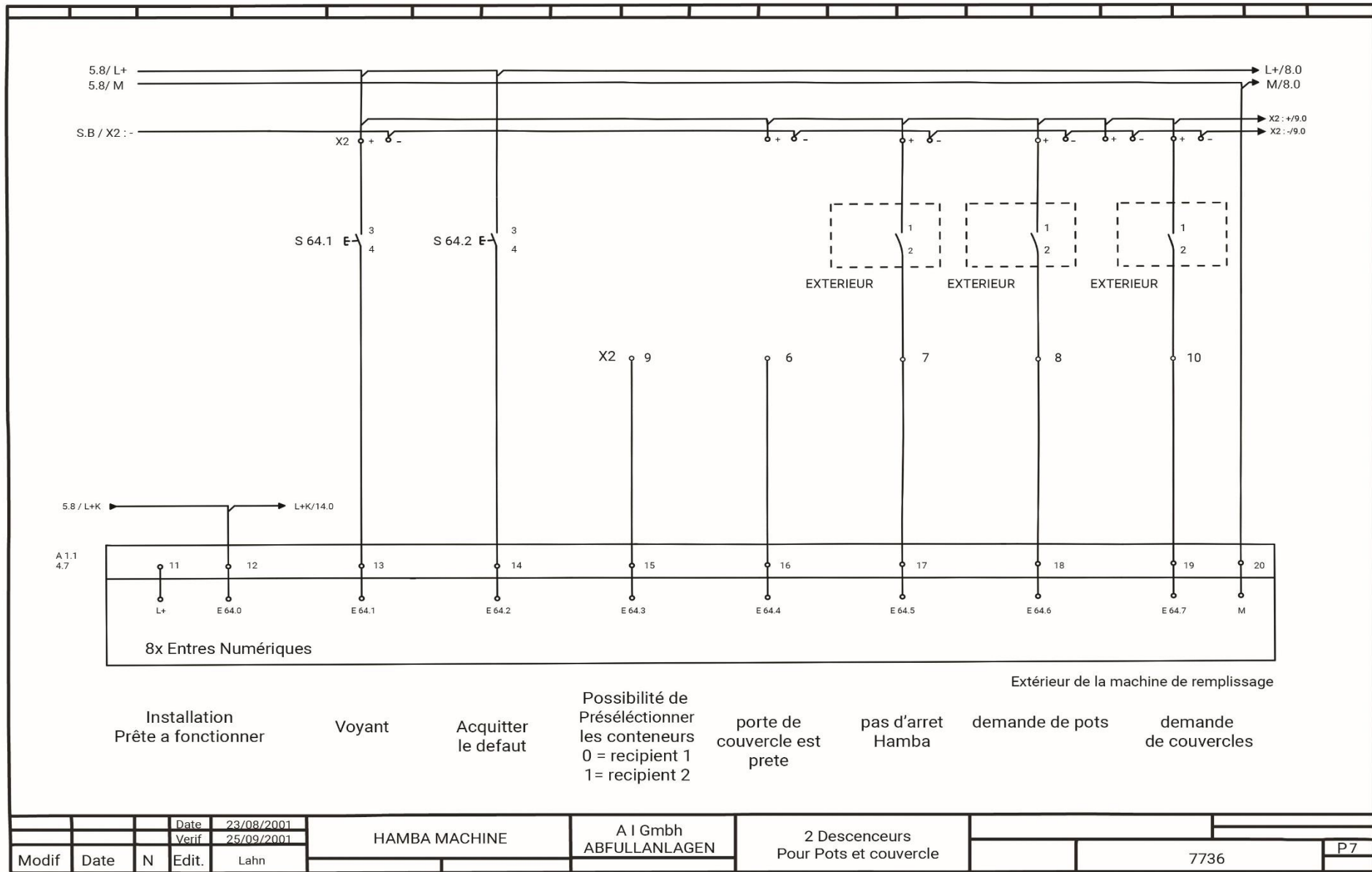


			Date	23/08/2001	HAMBA MACHINE	A I GmbH ABFULLANLAGEN	2 Descenceurs Pour Pots et couvercle	7736	P5
			Verif	25/09/2001					
Modif	Date	N	Edit.	Lahn					

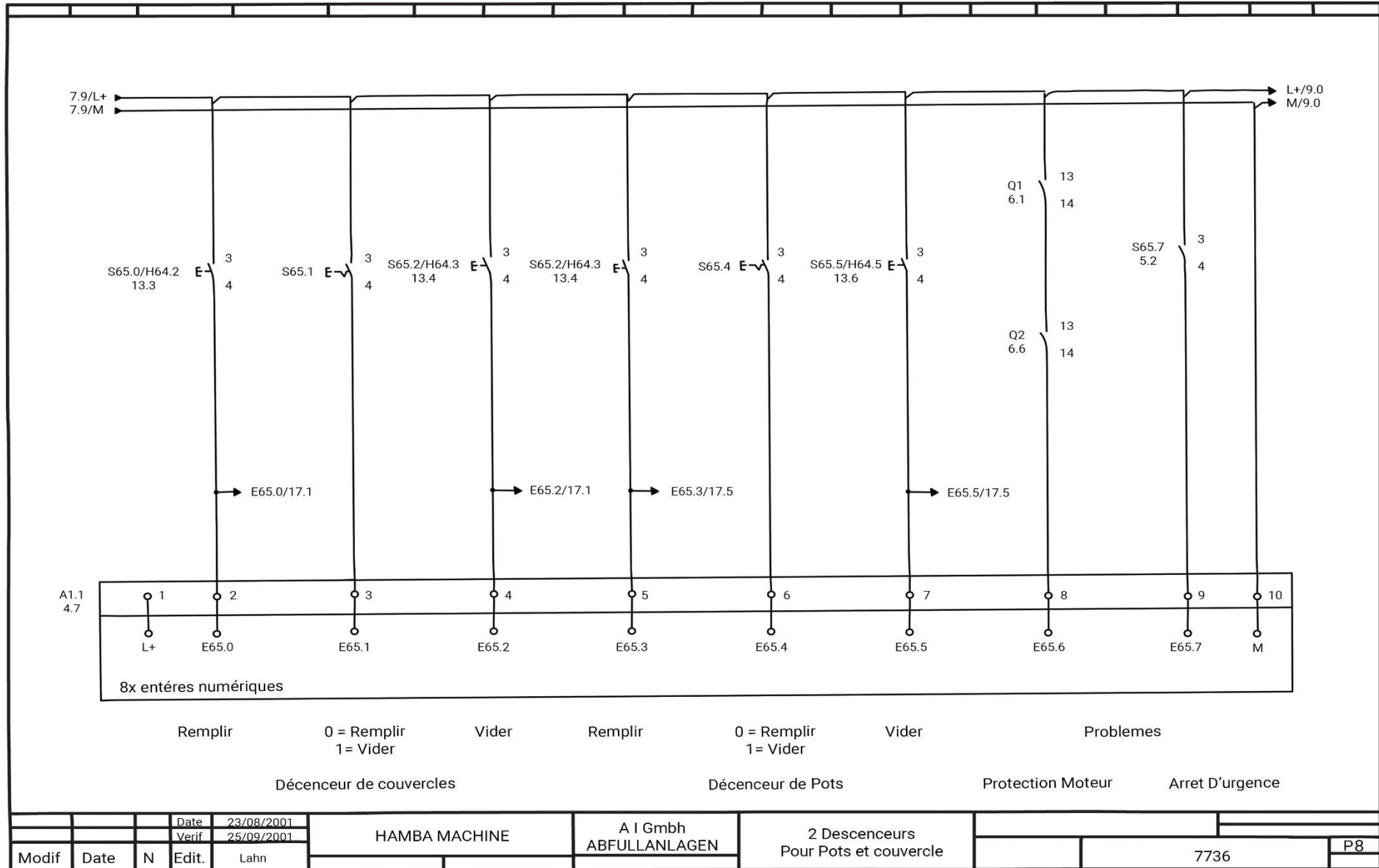
Alimentation principale :



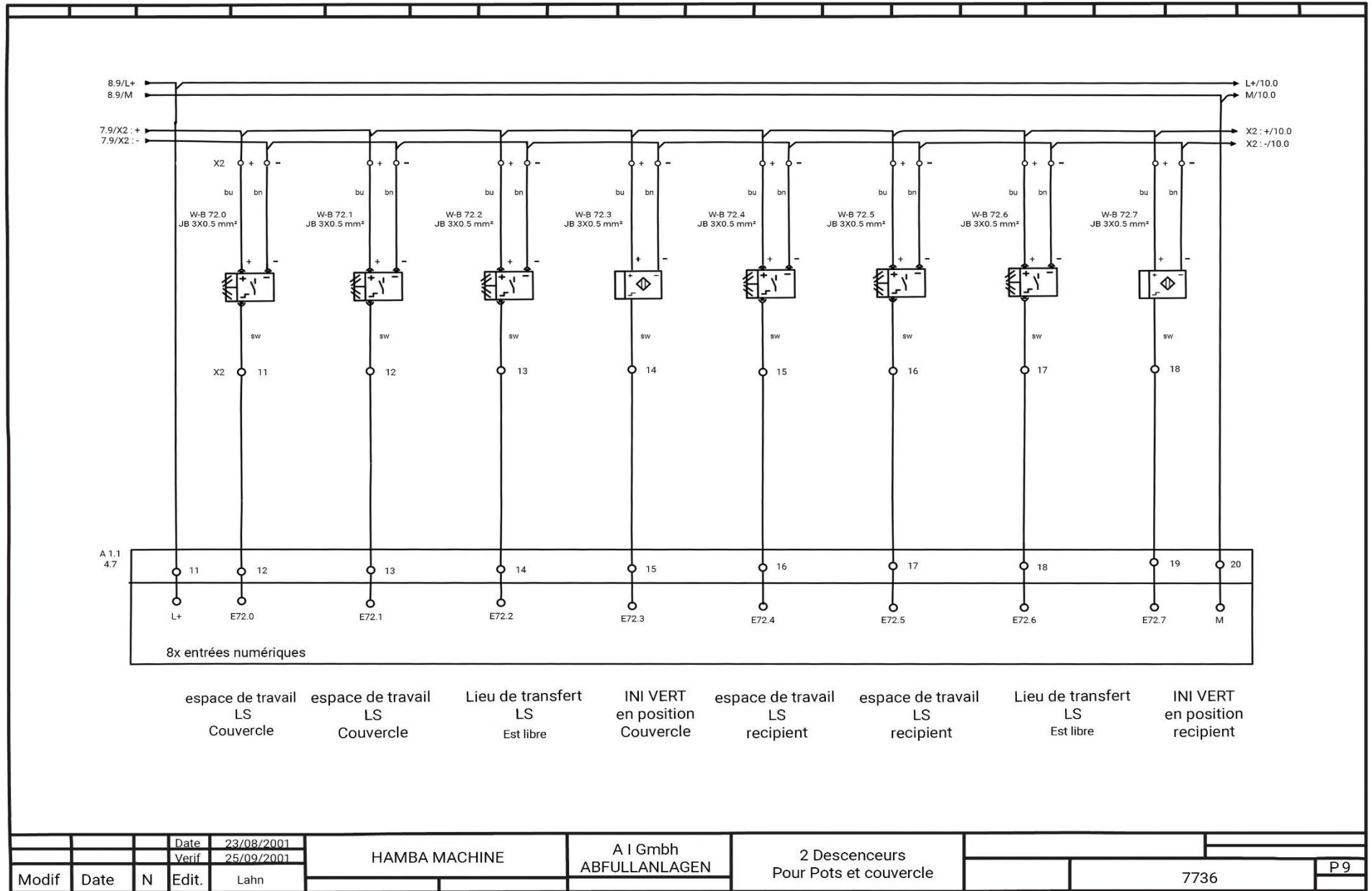
Entrées numériques :



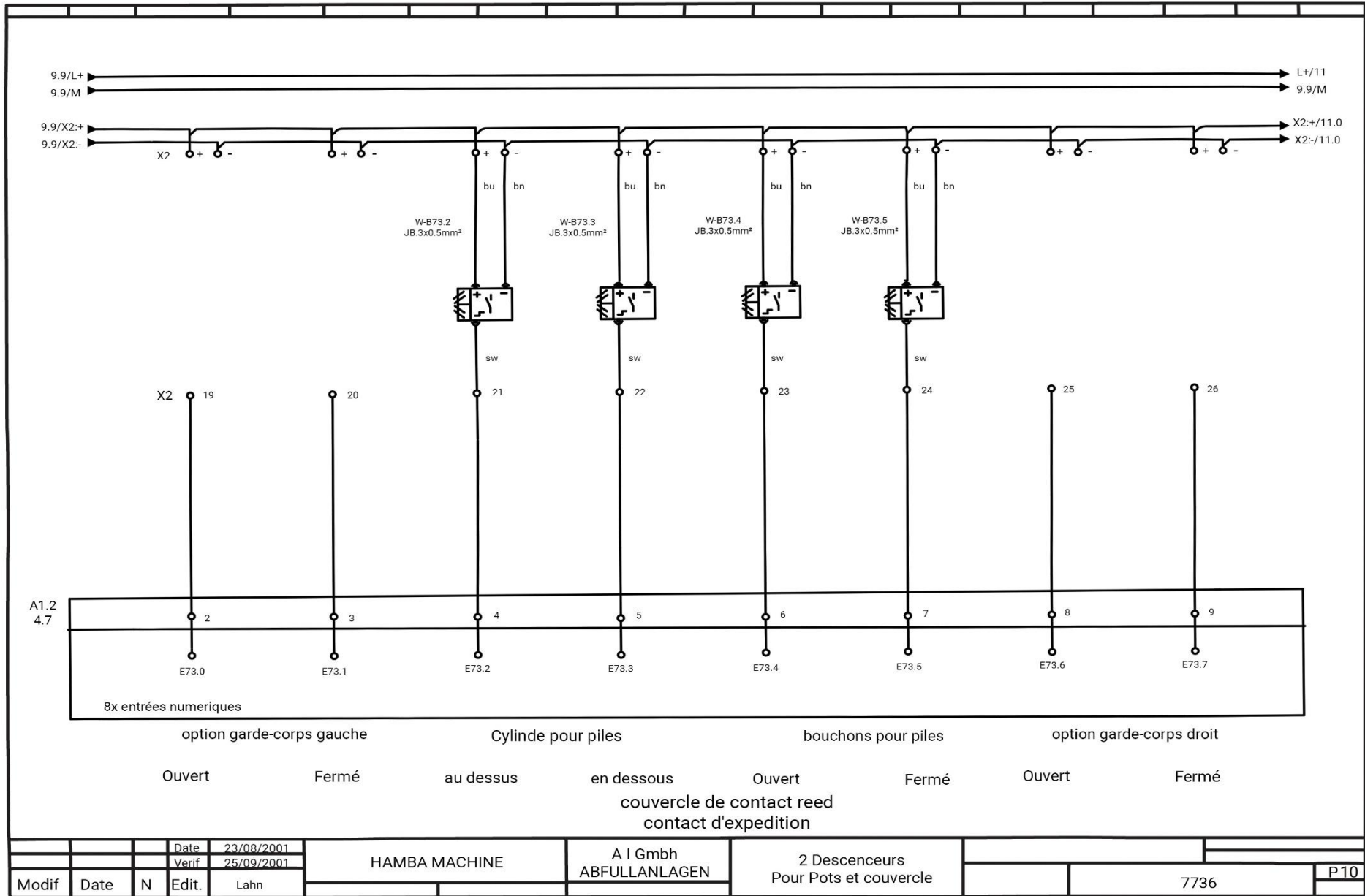
Annexes



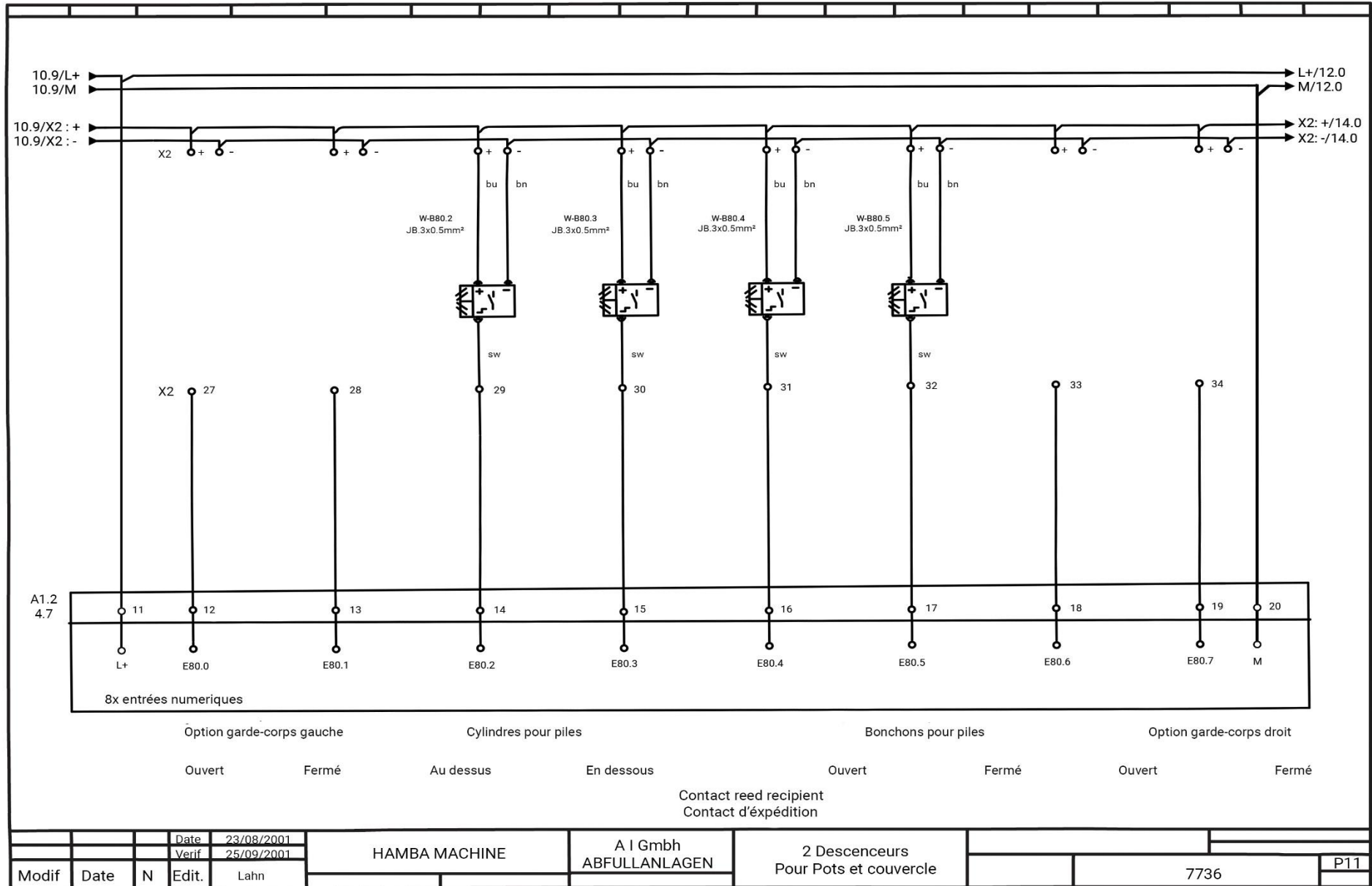
Annexes



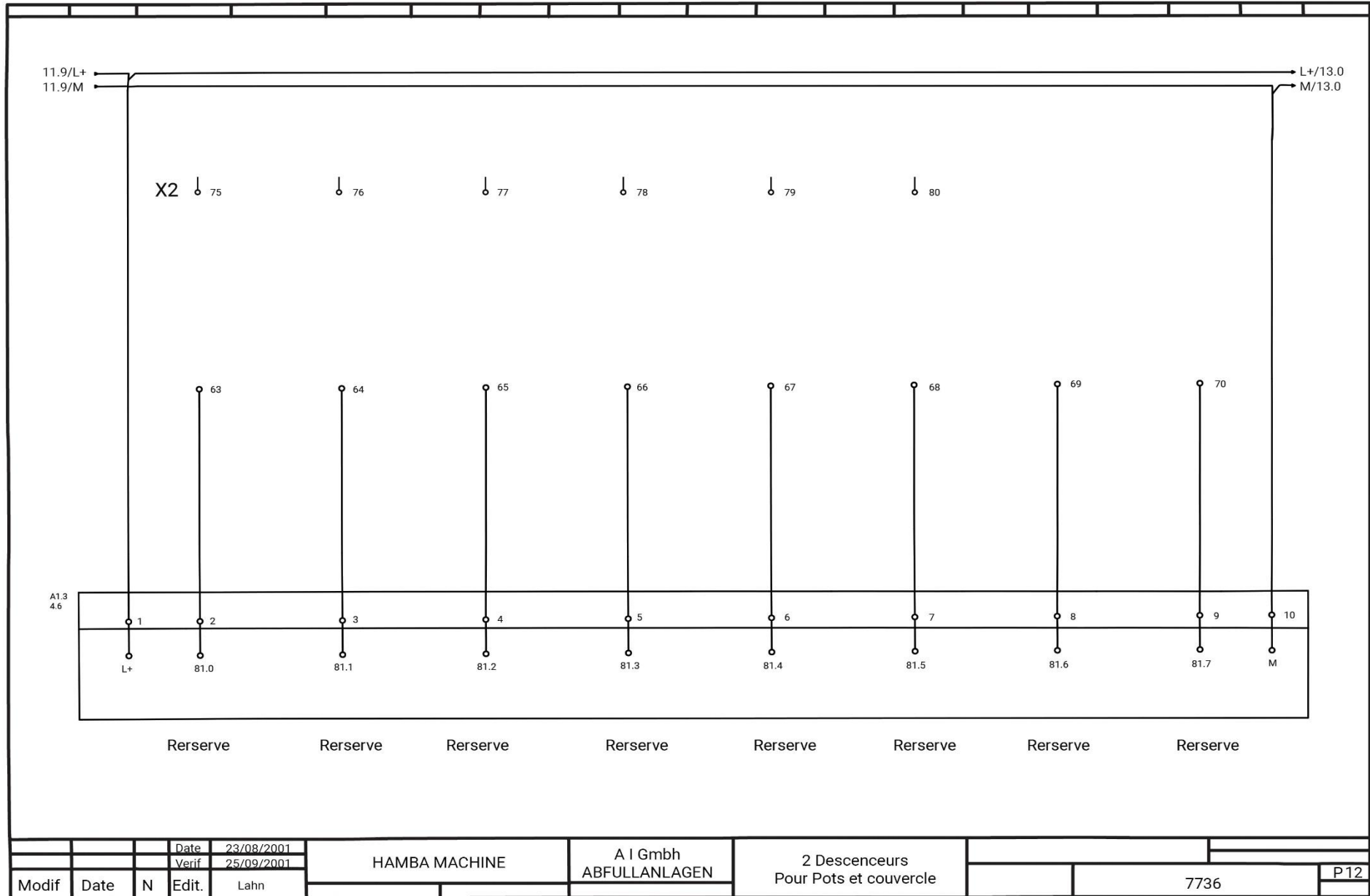
Annexes



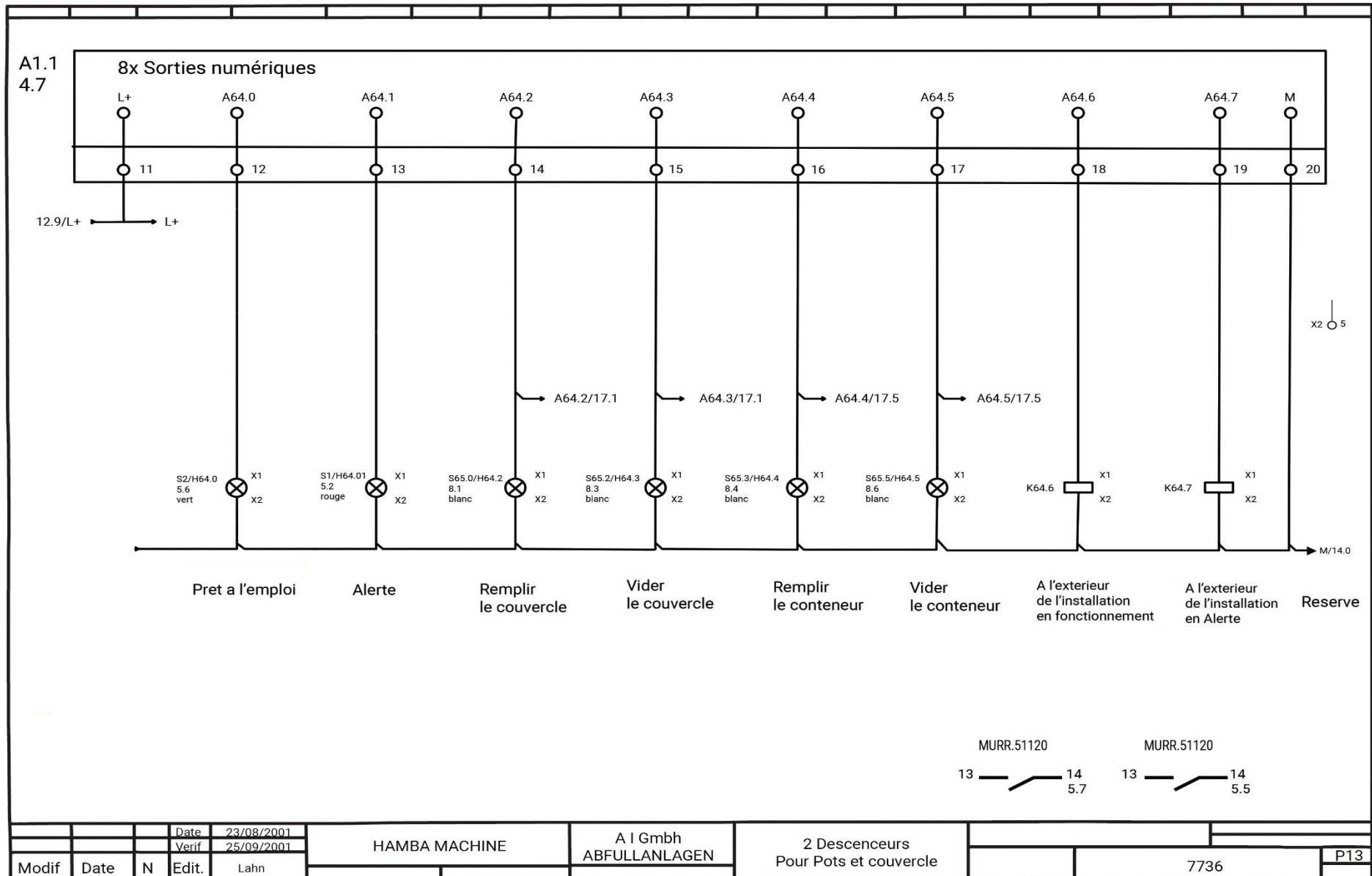
Annexes



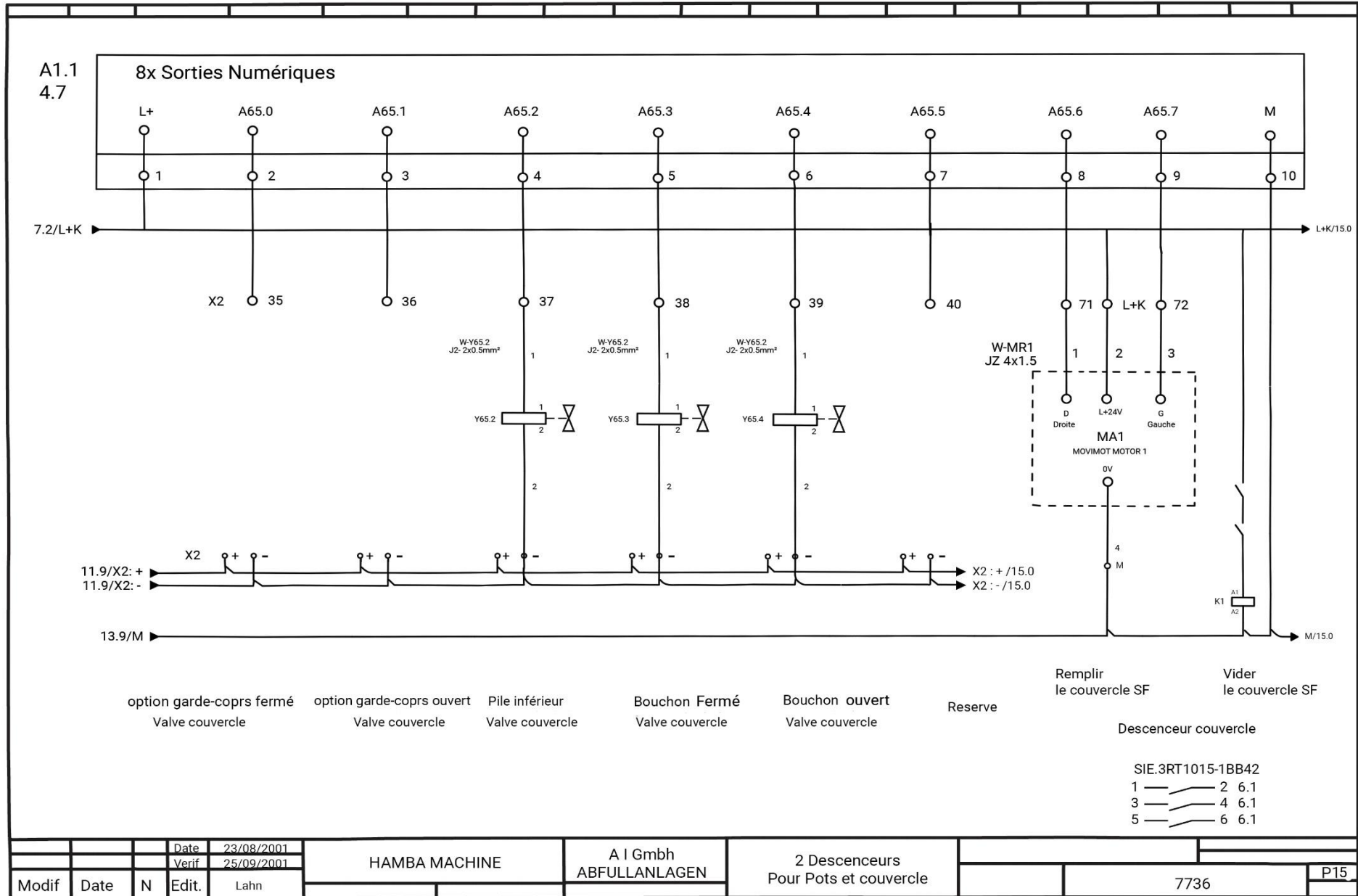
Annexes



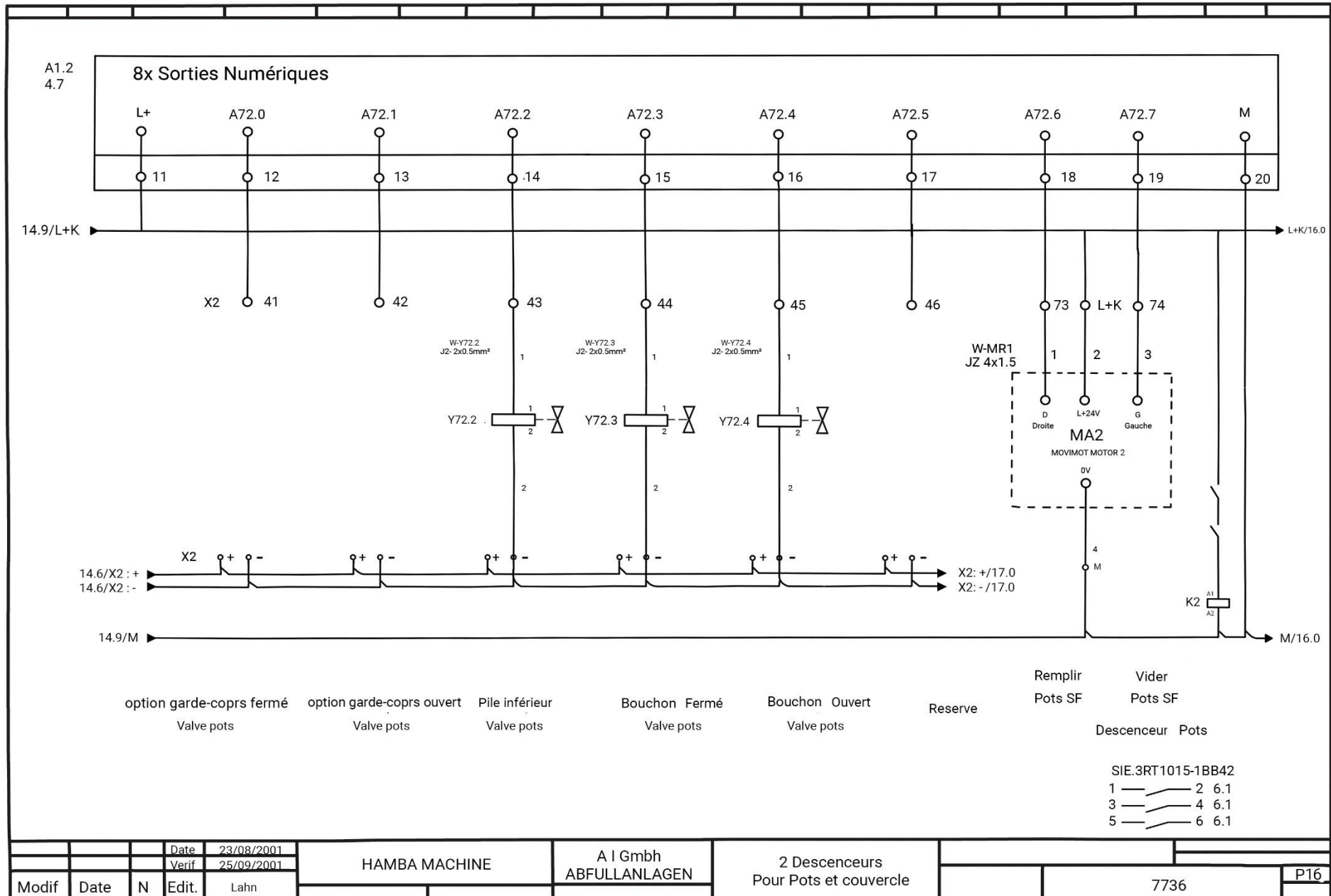
Sorties numériques :



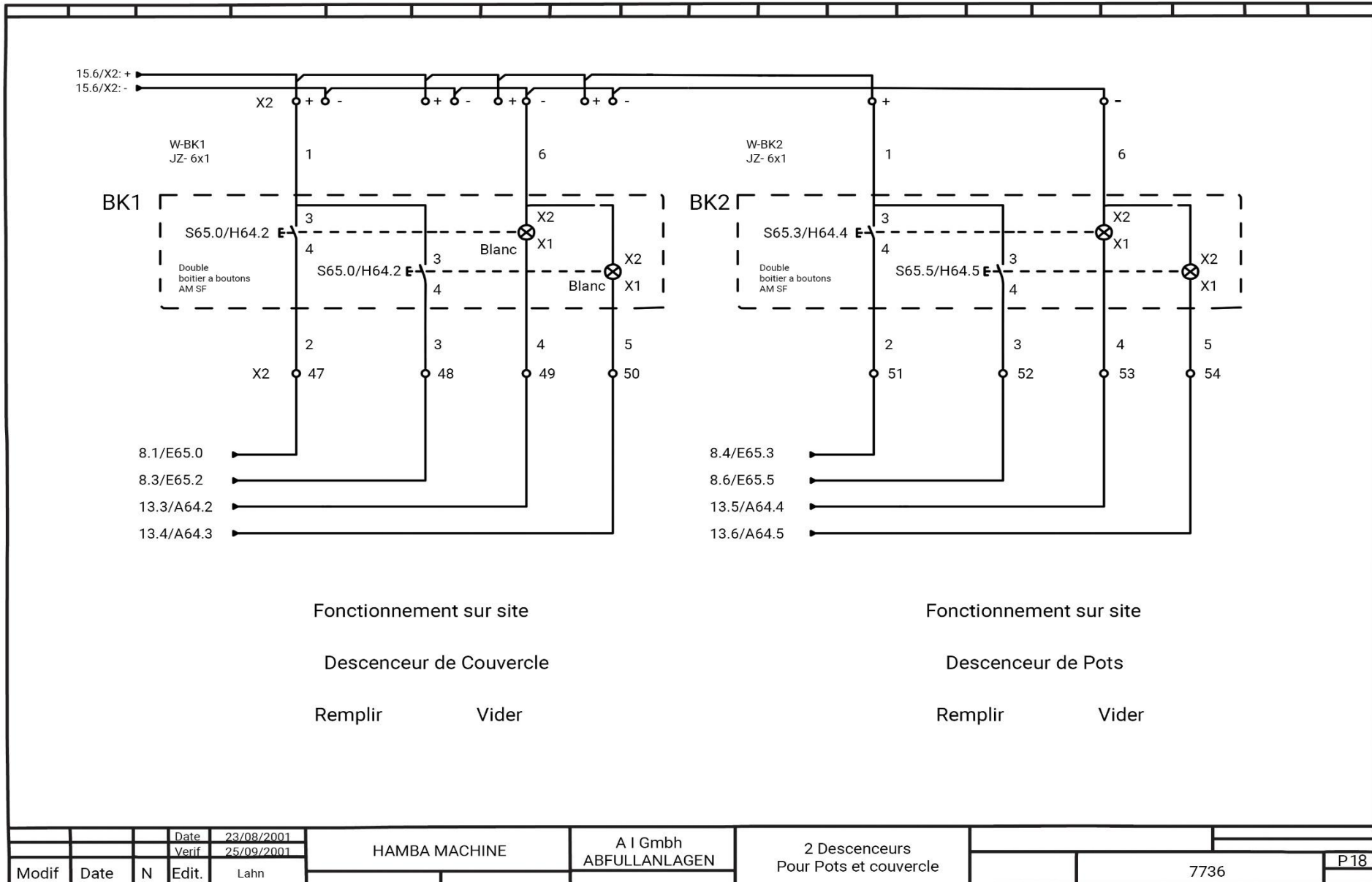
Annexes



Annexes



Commande du descenseur :



Totally Integrated Automation Portal			
L1_HAMBA_DECSenseur / CPU 315-2 DP [CPU 315-2 DP] / Blocs de programme			
OB1 [OB1]			
OB1 Propriétés			
Général			
Nom	OB1	Numéro	1
Langage	LIST	Numérotation	Manuel
Information			
Titre		Auteur	
Famille		Version	0.0
		Commentaire	
		ID utilisateur	
OB1			
Nom	Type de données	Décalage	Valeur par déf.
			Commentaire
▼ Temp			
OB1_EV_CLASS	Byte	0.0	Bits 0-3 = 1 (événement à venir), bits 4-7 = 1 (classe d'événements 1)
OB1_SCAN_1	Byte	1.0	1 (scan de redémarrage à froid 1 de l'OB 1), 3 (scan 2-n de l'OB 1)
OB1_PRIORITY	Byte	2.0	1 (la priorité de 1 est la plus basse)
OB1_OB_NUMBR	Byte	3.0	1 (bloc d'organisation 1, OB1)
OB1_RESERVED_1	Byte	4.0	Réservé au système
OB1_RESERVED_2	Byte	5.0	Réservé au système
OB1_PREV_CYCLE	Int	6.0	Temps de cycle de l'analyse précédent OB1 (millisecondes)
OB1_MIN_CYCLE	Int	8.0	Temps de cycle minimum de OB1 (millisecondes)
OB1_MAX_CYCLE	Int	10.0	Temps de cycle maximum de OB1 (millisecondes)
OB1_DATE_TIME	Date_And_Time	12.0	Date et heure de démarrage de OB1
Constant			
Réseau 1 :			
0001	CALL	"FC2"	
0002	CALL	"FC3"	
0003	CALL	"FC4"	
0004	CALL	"FC5"	
Réseau 2 : Réinitialiser toujours a 0			
0001	A	"Tag_102"	
0002	R	"Tag_102"	

Totally Integrated Automation Portal					
L1_HAMBA_DECSENSEUR / CPU 315-2 DP [CPU 315-2 DP] / Blocs de programme					
FC3 [FC3]					
FC3 Propriétés					
Général					
Nom	FC3	Numéro	3	Type	FC
Langage	LIST	Numérotation	Manuel		
Information					
Titre	station de pots	Auteur		Commentaire	
Famille		Version	0.0	ID utilisateur	
FC3					
Nom	Type de données	Décalage	Valeur par déf.	Commentaire	
Input					
Output					
InOut					
Temp					
Constant					
▼ Return					
FC3	Void				
Réseau 1 :					
0001	A		"Tag_1"		
0002	AN		"Remplir/vider (pots)"		
0003	;;				
0004	S		"Tag_52"		
0005	R		"Tag_53"		
0006	R		"Tag_54"		
0007	R		"Compteur de pots"		
0008	;;				
0009	A		"Remplir/vider (pots)"		
0010	O(
0011	A		"Tag_52"		
0012	A		"Tag_56"		
0013	A		"Espace de travail LS 1 (pots)"		
0014	A		"Tag_58"		
0015)				
0016	ON		"Installation prete a fonctionner"		
0017	R		"Tag_52"		
Réseau 2 :					
0001	A		"Tag_1"		
0002	A		"Remplir/vider (pots)"		
0003	A		"Tag_59"		
0004	S		"Tag_53"		
0005	R		"Tag_52"		
0006	R		"Tag_54"		
0007	R		"Compteur de pots"		
0008	;;				

Totally Integrated Automation Portal			
0009	AN	"Remplir/vider (pots)"	
0010	O(
0011	A	"Tag_53"	
0012	A	"Tag_59"	
0013	A	"Tag_58"	
0014	A	"Espace de travail LS 1 (pots)"	
0015)		
0016	ON	"Installation prete a fonctionner"	
0017	R	"Tag_53"	
Réseau 3 :			
0001	A	"Tag_60"	
0002	A(
0003	A	"Remplir pots SF"	
0004	A	"Espace de travail LS 2 (pots)"	
0005	A	"Espace de travail LS 1 (pots)"	
0006	O(
0007	A	"Vider pots SF"	
0008	A	"Espace de travail LS 1 (pots)"	
0009)		
0010)		
0011	CU	"Compteur de pots"	
0012	//;		
0013	L	"Compteur de pots"	
0014	L	C#5	
0015	>=I		
0016	S	"Tag_54"	
0017	R	"Tag_52"	
0018	R	"Tag_53"	
0019	R	"Compteur de pots"	
Réseau 4 :			
0001	AN	"Remplir/vider (pots)"	
0002	A	"Remplir pots SF"	
0003	O(
0004	A	"Tag_16"	
0005	AN	"Tag_64"	
0006	AN	"Tag_65"	
0007	A	"Tag_19"	
0008)		
0009	S	"Tag_64"	
0010	//;		
0011	AN	"Remplir pots SF"	
0012	A	"Tag_58"	
0013	R	"Tag_64"	
Réseau 5 :			
0001	A	"Remplir/vider (pots)"	
0002	A	"Vider pots SF"	
0003	S	"Tag_65"	
0004	//;		
0005	AN	"Vider pots SF"	
0006	A	"Tag_58"	

Totally Integrated Automation Portal			
0007	R	"Tag_65"	
Réseau 6 : Conteneurs de stockage en position			
0001	A	"Tag_58"	
0002	A	"Espace de travail LS 1 (pots)"	
0003	A(
0004	A	"Tag_52"	
0005	ON	"Remplir pots SF"	
0006)		
0007	S	"Tag_66"	
0008	R	"Tag_67"	
0009	//:		
0010	AN	"Espace de travail LS 1 (pots)"	
0011	A	"Remplir (pots)"	
0012	R	"Tag_66"	
0013	//:		
0014	A	"Tag_58"	
0015	AN	"Remplir/vider (pots)"	
0016	AN	"Espace de travail LS 1 (pots)"	
0017	AN	"Tag_66"	
0018	A	"Tag_69"	
0019	S	"Tag_67"	
0020	R	"Tag_66"	
0021	//:		
0022	AN	"Installation prete a fonctionner"	
0023	R	"Tag_67"	
0024	R	"Tag_66"	
Réseau 7 :			
0001	A	"Tag_58"	
0002	AN	"Piles inferieur (valve pots)"	
0003	A	"Cylindre pour piles au dessus (pots)"	
0004	A	"Bouchon fermé (valve pots)"	
0005	A	"Bouchons pour piles fermé (pots)"	
0006	A	"Espace de travail LS 2 (pots)"	
0007	=	"Tag_74"	
Réseau 8 :			
0001	AN	"Tag_58"	
0002	A	"Remplir (pots)"	
0003	O		
0004	A	"INI en position (pots)"	
0005	L	S5T#1S_500MS	
0006	SE	"Temporisation 20"	
0007	A	"Temporisation 20"	
0008	S	"Tag_58"	
0009	//:		
0010	O	"Remplir pots SF"	
0011	O	"Vider pots SF"	
0012	L	S5T#2S	
0013	SD	"Temporisation 21"	
0014	A	"Temporisation 21"	
0015	R	"Tag_58"	

Totally Integrated Automation Portal		
0016 //;		
Réseau 9 :		
0001	A	"Tag_58"
0002	AN	"Remplir/vider (pots)"
0003	AN	"Espace de travail LS 1 (pots)"
0004	AN	"Tag_69"
0005	=	"Tag_78"
Réseau 10 :		
0001	A	"Tag_56"
0002	AN	"Tag_69"
0003	A	"Tag_78"
0004	O	
0005	A	"Tag_69"
0006	AN	"Tag_56"
0007	AN	"Tag_79"
0008	=	"Tag_69"
Réseau 11 :		
0001	AN	"Tag_80"
0002	A	"Tag_58"
0003	A	"Pas d'arret HAMBA"
0004	A	"Tag_69"
0005	A	"Tag_74"
0006	A	"Espace de travail LS 2 (pots)"
0007	AN	"Remplir/vider (pots)"
0008	A(
0009	AN	"Espace de travail LS 1 (pots)"
0010	//;	
0011	O(
0012	AN	"Tag_52"
0013	A	"Espace de travail LS 1 (pots)"
0014	AN	"Tag_54"
0015)	
0016	O(
0017	A	"Tag_67"
0018)	
0019)	
0020	S	"Tag_79"
0021	//;	
0022	A	"Tag_60"
0023	R	"Tag_79"
0024	R	"Tag_67"
0025	NOF 0	
Réseau 12 :		
0001	AN	"Remplir/vider (pots)"
0002	A	"Tag_52"
0003	A(

Totally Integrated Automation Portal			
0004	AN	"Tag_58"	
0005	A	"Tag_37"	
0006	O		
0007	A	"Tag_58"	
0008	AN	"Espace de travail LS 1 (pots)"	
0009	A	"Tag_69"	
0010	O		
0011	A	"Tag_58"	
0012	AN	"Espace de travail LS 1 (pots)"	
0013	A	"Temporisation 1"	
0014	A	"Tag_78"	
0015)		
0016	O	"Voyant"	
0017	O		
0018	AN	"Tag_58"	
0019	A	"Installation prete a fonctionner"	
0020	A	"Temporisation 4"	
0021	AN	"Remplir pots SF"	
0022	AN	"Vider pots SF"	
0023	=	"remplir pots"	
Réseau 13 :			
0001	A(
0002	A	"Tag_1"	
0003	A	"Tag_79"	
0004	O		
0005	A	"Tag_16"	
0006	AN	"Tag_58"	
0007	A	"Tag_64"	
0008)		
0009	A	"Pas d'arret HAMBAs"	
0010	A	"Protection moteurs (problemes)"	
0011	AN	"Vider pots SF"	
0012	=	"Remplir pots SF"	
Réseau 14 :			
0001	AN	"Tag_79"	
0002	A	"Tag_58"	
0003	A	"Tag_74"	
0004	A	"Remplir/vider (pots)"	
0005	A	"Espace de travail LS 1 (pots)"	
0006	A	"Tag_59"	
0007	S	"Tag_80"	
0008	A	"Tag_60"	
0009	AN	"Espace de travail LS 1 (pots)"	
0010	O	"Tag_54"	
0011	R	"Tag_80"	
0012	NOP 0		
Réseau 15 :			
0001	A	"Remplir/vider (pots)"	
0002	A(
0003	ON	"Tag_58"	

Totally Integrated Automation Portal			
0004	O		
0005	A	"Tag_58"	
0006	AN	"Espace de travail LS 1 (pots)"	
0007	O		
0008	A	"Tag_58"	
0009	A	"Espace de travail LS 1 (pots)"	
0010	A	"Temporisation 1"	
0011)		
0012	O	"Voyant"	
0013	=	"vider pots"	
Réseau 16 :			
0001	A(
0002	A	"Tag_1"	
0003	A	"Tag_80"	
0004	O		
0005	A	"Tag_16"	
0006	AN	"Tag_58"	
0007	A	"Tag_65"	
0008)		
0009	A	"Protection moteurs (problemes)"	
0010	AN	"Remplir pots SF"	
0011	A	"Espace de travail LS 2 (pots)"	
0012	A	"Lieu de transfert est libre LS (post)"	
0013	=	"Vider pots SF"	
Réseau 17 :			
0001	A	"Tag_1"	
0002	A	"Pas d'arret HAMB A"	
0003	A	"Tag_58"	
0004	AN	"Remplir/vider (pots)"	
0005	A	"Cylindre pour piles au dessus (pots)"	
0006	AN	"Espace de travail LS 2 (pots)"	
0007	A	"Lieu de transfert est libre LS (post)"	
0008	A	"Bouchon fermé (valve pots)"	
0009	A	"Bouchons pour piles fermé (pots)"	
0010	S	"Piles inferieur (valve pots)"	
0011	A	"Tag_1"	
0012	A	"Pas d'arret HAMB A"	
0013	A	"Bouchon ouvert (valve pots)"	
0014	A	"Bouchons pour piles ouvert (pots)"	
0015	R	"Piles inferieur (valve pots)"	
0016	NO P 0		
Réseau 18 :			
0001	A	"Tag_1"	
0002	A	"Pas d'arret HAMB A"	
0003	A	"Piles inferieur (valve pots)"	
0004	A	"Cylindre pour piles en dessous (pots)"	
0005	A	"Demande de pots"	
0006	S	"Bouchon ouvert (valve pots)"	
0007	R	"Bouchon fermé (valve pots)"	
0008	//;		

Totally Integrated Automation Portal			
0009	A	"Installation prete a fonctionner"	
0010	A	"Pas d'arret HAMBA"	
0011	AN	"Piles inferieur (valve pots)"	
0012	A	"Cylindre pour piles au dessus (pots)"	
0013	A	"Espace de travail LS 2 (pots)"	
0014	A	"Lieu de transfert est libre LS (post)"	
0015	S	"Bouchon fermé (valve pots)"	
0016	R	"Bouchon ouvert (valve pots)"	
Réseau 19 :			
0001	A	"Installation prete a fonctionner"	
0002	A	"Pas d'arret HAMBA"	
0003	A	"Préselectionner les conteneurs"	
0004	S	"Option garde-corps ouvert (valve pots)"	
0005	R	"Option garde-corps fermé (valve pots)"	
0006	//;		
0007	A	"Installation prete a fonctionner"	
0008	AN	"Préselectionner les conteneurs"	
0009	A	"Pas d'arret HAMBA"	
0010	S	"Option garde-corps fermé (valve pots)"	
0011	R	"Option garde-corps ouvert (valve pots)"	

Totally Integrated Automation Portal			
L1_HAMBA_DECSENSEUR / CPU 315-2 DP [CPU 315-2 DP] / Blocs de programme			
FC4 [FC4]			
FC4 Propriétés			
Général			
Nom	FC4	Numéro	4
Langage	LIST	Type	FC
		Numérotation	Manuel
Information			
Titre		Auteur	
Famille		Version	0.0
		Commentaire	
		ID utilisateur	
FC4			
Nom	Type de données	Décalage	Valeur par déf.
			Commentaire
Input			
Output			
InOut			
Temp			
Constant			
▼ Return			
FC4	Void		
Réseau 1 : M 10.2 indicateur initial			
0001	A	"Installation prete a fonctionner"	
0002	AN	"Tag_17"	
0003	AN	"Tag_18"	
0004	A	"Tag_9"	
0005	AN	"Tag_64"	
0006	AN	"Tag_65"	
0007	A	"Tag_58"	
0008	AN	"Vider couvercle SF"	
0009	AN	"Remplir couvercle SF"	
0010	AN	"Remplir pots SF"	
0011	AN	"Vider pots SF"	
0012	S	"Tag_90"	
0013	R	"Tag_16"	
0014	//;		
0015	AN	"Installation prete a fonctionner"	
0016	ON	"Protection moteurs (problemes)"	
0017	R	"Tag_90"	
Réseau 2 : M 10.3 Système pas en position			
0001	A	"Installation prete a fonctionner"	
0002	AN	"Tag_16"	
0003	AN	"Tag_90"	
0004	A(
0005	O	"Tag_17"	
0006	O	"Tag_18"	
0007	ON	"Tag_9"	
0008	O	"Tag_64"	

Totally Integrated Automation Portal			
0009	O	"Tag_65"	
0010	ON	"Tag_58"	
0011)		
0012	S	"Tag_91"	
0013	ON	"Installation prete a fonctionner"	
0014	ON	"Protection moteurs (problemes)"	
0015	O	"Tag_16"	
0016	R	"Tag_91"	
0017	NOP	0	
Réseau 3 : M10.4 Convoyeur vertical en pos			
0001	A	"Tag_91"	
0002	A	"Acquitter le defaut"	
0003	S	"Tag_16"	
0004	//;		
0005	AN	"Installation prete a fonctionner"	
0006	R	"Tag_16"	
Réseau 4 : M10.0 Fonctionnement automatique A64.6 prêt			
0001	A	"Installation prete a fonctionner"	
0002	A	"Tag_90"	
0003	S	"Tag_1"	
0004	//;		
0005	ON	"Installation prete a fonctionner"	
0006	ON	"Protection moteurs (problemes)"	
0007	R	"Tag_1"	
0008	//;		
0009	A	"Tag_1"	
0010	AN	"Exterieur de l'intallation en alerte"	
0011	=	"Exterieur de l'intallation en foctionnement"	
Réseau 5 :			
0001	A	"Installation prete a fonctionner"	
0002	AN	"Temporisation 2"	
0003	L	S5T#1S	
0004	SP	"Temporisation 1"	
0005	//;		
0006	A	"Installation prete a fonctionner"	
0007	AN	"Temporisation 1"	
0008	L	S5T#500MS	
0009	SP	"Temporisation 2"	
0010	//;		
Réseau 6 :			
0001	A	"Installation prete a fonctionner"	
0002	AN	"Temporisation 4"	
0003	L	S5T#400MS	
0004	SP	"Temporisation 3"	
0005	//;		
0006	A	"Installation prete a fonctionner"	

Totally Integrated Automation Portal			
0007	AN	"Temporisation 3"	
0008	L	S5T#400MS	
0009	SP	"Temporisation 4"	
0010	//;		
Réseau 7 : Défaut de protection du moteur			
0001	AN	"Protection moteurs (problemes)"	
0002	=	"Tag_97"	
Réseau 8 : voyant panne			
0001	A(
0002	A	"Installation prete a fonctionner"	
0003	AN	"Tag_90"	
0004	A	"Temporisation 2"	
0005)		
0006	O(
0007	AN	"Protection moteurs (problemes)"	
0008	A	"Temporisation 3"	
0009	O	"Arret d'urgence"	
0010)		
0011	O(
0012	A	"Installation prete a fonctionner"	
0013	A	"Tag_99"	
0014	A	"Temporisation 3"	
0015)		
0016	O	"Voyant"	
0017	=	"Alerte"	
Réseau 9 : voyant prête à l'emploi			
0001	A	"Installation prete a fonctionner"	
0002	O	"Voyant"	
0003	=	"Pret a l'emploi"	
Réseau 10 : Option			
0001	A	"Tag_102"	
0002	A	"Option garde-corps fermé (valve couvercle)"	
0003	AN	"Option garde-corp gauche fermé (couvercles)"	
0004	AN	"Option garde-corps droit fermé (couvercles)"	
0005	L	S5T#5S	
0006	SD	"Temporisation 5"	
Réseau 11 : Option			
0001	A	"Tag_102"	
0002	A	"Option garde-corps ouvert (valve couvercle)"	
0003	AN	"Option garde-corp gauche ouvert (couvercles)"	
0004	AN	"Option garde-corps droit ouvert (couvercles)"	
0005	L	S5T#5S	
0006	SD	"Temporisation 6"	

Totally Integrated Automation Portal		
Réseau 12 : Option		
0001	A	"piles inferieur (valve couvercle)"
0002	AN	"Cylindre pour piles en dessous (couvercles)"
0003	L	SST#5S
0004	SD	"Temporisation 7"
Réseau 13 :		
0001	AN	"piles inferieur (valve couvercle)"
0002	AN	"Cylindre pour piles au dessus (couvercles)"
0003	L	SST#5S
0004	SD	"Temporisation 8"
Réseau 14 :		
0001	A	"Bouchon fermé (valve couvercle)"
0002	AN	"Bouchons pour piles fermé (couvercles)"
0003	L	SST#5S
0004	SD	"Temporisation 9"
Réseau 15 :		
0001	A	"Bouchon ouvert (valve couvercle)"
0002	AN	"Bouchons pour piles ouvert (couvercles)"
0003	L	SST#5S
0004	SD	"Temporisation 10"
Réseau 16 : Option		
0001	A	"Tag_102"
0002	A	"Option garde-corps fermé (valve pots)"
0003	AN	"Option garde-corps gauche fermé (pots)"
0004	AN	"option garde-corps droit fermé (pots)"
0005	L	SST#5S
0006	SD	"Temporisation 11"
Réseau 17 :		
0001	A	"Tag_102"
0002	A	"Option garde-corps ouvert (valve pots)"
0003	AN	"Option garde-corps gauche ouvert (pots)"
0004	AN	"Option garde-corps droit ouvert (pots)"
0005	L	SST#5S
0006	SD	"Temporisation 12"
Réseau 18 :		
0001	A	"Piles inferieur (valve pots)"
0002	AN	"Cylindre pour piles en dessous (pots)"

Totally Integrated Automation Portal		
0003	L	SST#5S
0004	SD	"Temporisation 13"
Réseau 19 :		
0001	AN	"Piles inferieur (valve pots)"
0002	AN	"Cylindre pour piles au dessus (pots)"
0003	L	SST#5S
0004	SD	"Temporisation 14"
Réseau 20 :		
0001	A	"Bouchon fermé (valve pots)"
0002	AN	"Bouchons pour piles fermé (pots)"
0003	L	SST#5S
0004	SD	"Temporisation 15"
Réseau 21 :		
0001	A	"Bouchon ouvert (valve pots)"
0002	AN	"Bouchons pour piles ouvert (pots)"
0003	L	SST#5S
0004	SD	"Temporisation 16"
Réseau 22 : Défaut de collecte		
0001	A	"Temporisation 5"
0002	O	"Temporisation 6"
0003	O	"Temporisation 7"
0004	O	"Temporisation 8"
0005	O	"Temporisation 9"
0006	O	"Temporisation 10"
0007	O	"Temporisation 11"
0008	O	"Temporisation 12"
0009	O	"Temporisation 13"
0010	O	"Temporisation 14"
0011	O	"Temporisation 15"
0012	O	"Temporisation 16"
0013	=	"Tag_99"
Réseau 23 : Installation de sortie en panne		
0001	O	"Tag_97"
0002	O	"Tag_99"
0003	O	"Arrêt d'urgence"
0004	=	"Exterieur de l'intallation en alerte"

Totally Integrated Automation Portal			
<p>L1_HAMBA_DECSenseur / CPU 315-2 DP [CPU 315-2 DP] / Blocs de programme</p> <p>FC5 [FC5]</p>			
FC5 Propriétés			
Général			
Nom	FC5	Numéro	5
Langage	LIST	Numérotation	Manuel
Type	FC		
Information			
Titre		Auteur	
Famille		Version	0.0
Commentaire			
ID utilisateur			
FC5			
Nom	Type de données	Décalage	Valeur par déf.
Commentaire			
Input			
Output			
InOut			
Temp			
Constant			
▼ Return			
FC5	Void		
Réseau 1 :			
0001	A	"Installation prete a fonctionner"	
0002	A	"Tag_123"	
0003	=	"Tag_124"	
0004	A	"Tag_124"	
0005	R	"Tag_123"	
0006	AN	"Installation prete a fonctionner"	
0007	S	"Tag_123"	
0008	NOP	0	
Réseau 2 :			
0001	A	"INI en positon (couvercle) "	
0002	A	"Tag_125"	
0003	=	"Tag_12"	
0004	A	"Tag_12"	
0005	R	"Tag_125"	
0006	AN	"INI en positon (couvercle) "	
0007	S	"Tag_125"	
0008	NOP	0	
Réseau 3 :			
0001	A	"INI en posistion (pots) "	
0002	A	"Tag_126"	
0003	=	"Tag_60"	
0004	A	"Tag_60"	
0005	R	"Tag_126"	

Totally Integrated Automation Portal			
0006	AN	"INI en position (pots) "	
0007	S	"Tag_126"	
0008	NOP	0	
Réseau 4 :			
0001	A	"Acquitter le défaut"	
0002	A	"Tag_127"	
0003	=	"Tag_19"	
0004	A	"Tag_19"	
0005	R	"Tag_127"	
0006	AN	"Acquitter le défaut"	
0007	S	"Tag_127"	
0008	NOP	0	
Réseau 5 :			
0001	A	"Remplir (couverts) "	
0002	A	"Tag_128"	
0003	=	"Tag_7"	
0004	A	"Tag_7"	
0005	R	"Tag_128"	
0006	AN	"Remplir (couverts) "	
0007	S	"Tag_128"	
0008	NOP	0	
Réseau 6 :			
0001	A	"Vider (couverts) "	
0002	A	"Tag_130"	
0003	=	"Tag_11"	
0004	A	"Tag_11"	
0005	R	"Tag_130"	
0006	AN	"Vider (couverts) "	
0007	S	"Tag_130"	
0008	NOP	0	
Réseau 7 :			
0001	A	"Remplir (pots) "	
0002	A	"Tag_131"	
0003	=	"Tag_56"	
0004	A	"Tag_56"	
0005	R	"Tag_131"	
0006	AN	"Remplir (pots) "	
0007	S	"Tag_131"	
0008	NOP	0	
Réseau 8 :			
0001	A	"vider (pots) "	
0002	A	"Tag_133"	
0003	=	"Tag_59"	

Annexes

Totally Integrated Automation Portal		
0004	A	"Tag_59"
0005	R	"Tag_133"
0006	AN	"vider (pots)"
0007	S	"Tag_133"
0008	NOP	0

Totally Integrated Automation Portal																			
L1_HAMBA_DECSenseur / CPU 315-2 DP [CPU 315-2 DP] / Variables API / Table de variables standard [133] Entrée, sortie																			
Entrée, sortie																			
W	DW OR D	LW OR D	Adresse	7	6	5	4	3	2	1	0	B							
			IB64	◆	◆	◆		◆	◆	◆	◆								
			IB65	◆	◆	◆	◆	◆	◆	◆	◆								
			IB72	◆	◆	◆	◆	◆	◆	◆	◆								
			IB73	◆	◆	◆	◆	◆	◆	◆	◆								
			IB80	◆	◆	◆	◆	◆	◆	◆	◆								
			IB81																
			QB64	◆	◆	◆	◆	◆	◆	◆	◆								
			QB65	◆	◆		◆	◆	◆	◆	◆								
			QB72	◆	◆		◆	◆	◆	◆	◆								
			QB73																

Totally Integrated Automation Portal		
L1_HAMBA_DECSenseur / CPU 315-2 DP [CPU 315-2 DP] / Variables API / Table de variables standard [133]		
Temporisations		
Temporisations		
Adresse		
T1	-1	◆
T2	-1	◆
T3	-1	◆
T4	-1	◆
T5	-1	◆
T6	-1	◆
T7	-1	◆
T8	-1	◆
T9	-1	◆
T10	-1	◆
T11	-1	◆
T12	-1	◆
T13	-1	◆
T14	-1	◆
T15	-1	◆
T16	-1	◆
T20	-1	◆
T21	-1	◆
T30	-1	◆
T31	-1	◆

Totally Integrated Automation Portal								
L1_HAMBA_DECSENSEUR / CPU 315-2 DP [CPU 315-2 DP] / Variables API								
Table de variables standard [133]								
Variables API								
Nom	Type de données	Adresse	Rémanence	Acces- sible de- puis IHM/O PC UA	Ecri- ture autor- isée à partir de IHM/O PC UA	Visi- ble dans l'ingé- nierie IHM	Surveillance	Commentaire
Tag_1	Bool	%M10.0		True	True	True		
Remplir/vider (cou- vercles)	Bool	%I65.1		True	True	True		0=remplir , 1=vider
Tag_3	Bool	%M11.0		True	True	True		
Tag_4	Bool	%M11.3		True	True	True		
Tag_5	Bool	%M11.6		True	True	True		
Compteur de cou- verles	Counter	%C1		True	True	True		compteur 1
Tag_7	Bool	%M82.0		True	True	True		
Espace de travail LS 1 (couvercles)	Bool	%I72.0		True	True	True		Capteur station de cou- vercles
Tag_9	Bool	%M8.0		True	True	True		
Installation prete a fonctionner	Bool	%I64.0		True	True	True		
Tag_11	Bool	%M82.2		True	True	True		
Tag_12	Bool	%M80.2		True	True	True		
Vider couvercle SF	Bool	%Q65.6		True	True	True		
Espace de travail LS 2 (couvercles)	Bool	%I72.1		True	True	True		Capteur station de cou- vercles
Remplir couvercle SF	Bool	%Q65.7		True	True	True		
Tag_16	Bool	%M10.4		True	True	True		
Tag_17	Bool	%M11.1		True	True	True		
Tag_18	Bool	%M11.4		True	True	True		
Tag_19	Bool	%M80.6		True	True	True		
Tag_20	Bool	%M13.3		True	True	True		
Tag_21	Bool	%M13.2		True	True	True		
Remplir (couvercles)	Bool	%I65.0		True	True	True		
pires inferieur (valve couvercle)	Bool	%Q65.2		True	True	True		
Cylindre pour piles au dessus (couvercles)	Bool	%I73.2		True	True	True		Station de couvercles
Bouchon fermé (valve couvercle)	Bool	%Q65.3		True	True	True		
Bouchons pour piles fermé (couvercles)	Bool	%I73.5		True	True	True		Station de couvercles
Lieu de transfert est libre LS (couvercles)	Bool	%I72.2		True	True	True		Capteur station de cou- vercles
Tag_28	Bool	%M13.4		True	True	True		
INI en positon (cou- vercle)	Bool	%I72.3		True	True	True		Capteur station de cou- vercles

































Annexes

Totally Integrated Automation Portal										
Nom	Type de données	Adresse	Rémanence	Acces- sible de- puis IHM/O PC UA	Ecri- ture au- tor- isée à partir de IHM/O PC UA	Visi- ble dans l'ingé- nierie IHM	Surveillance	Commentaire		
Temporisation 30	Timer	%T30		True	True	True				
Temporisation 31	Timer	%T31		True	True	True				
Tag_32	Bool	%M8.4		True	True	True				
Tag_33	Bool	%M8.3		True	True	True				
Tag_34	Bool	%M13.0		True	True	True				
Tag_35	Bool	%M13.1		True	True	True				
Pas d'arrêt HAMB	Bool	%I64.5		True	True	True				
Tag_37	Bool	%M1.7		True	True	True				
Temporisation 1	Timer	%T1		True	True	True				
Voyant	Bool	%I64.1		True	True	True				
Temporisation 4	Timer	%T4		True	True	True				
Remplir couvercles	Bool	%Q64.2		True	True	True				
Protection moteurs (problemes)	Bool	%I65.6		True	True	True		Problemes		
vider couvercle	Bool	%Q64.3		True	True	True				
Bouchon ouvert (valve couvercle)	Bool	%Q65.4		True	True	True				
Bouchons pour piles ouvert (couvercles)	Bool	%I73.4		True	True	True		Station de couvercles		
Cylindre pour piles en dessous (couvercles)	Bool	%I73.3		True	True	True		Station de couvercles		
Demande de couvercles	Bool	%I64.7		True	True	True		Extérieur de la machine de remplissage		
Préselectionner les conteneurs	Bool	%I64.3		True	True	True		Possibilité de préselectionner les conteneurs 0=recipient 1, 1=recipient 2		
Option garde-corps ouvert (valve couvercle)	Bool	%Q65.1		True	True	True				
Option garde-corps fermé (valve couvercle)	Bool	%Q65.0		True	True	True				
Remplir/vider (pots)	Bool	%I65.4		True	True	True				
Tag_52	Bool	%M12.0		True	True	True				
Tag_53	Bool	%M12.3		True	True	True				
Tag_54	Bool	%M12.6		True	True	True				
Compteur de pots	Counter	%C2		True	True	True		Compteur 2		
Tag_56	Bool	%M82.4		True	True	True				
Espace de travail LS 1 (pots)	Bool	%I72.4		True	True	True		Capteur station pots		
Tag_58	Bool	%M9.0		True	True	True				
Tag_59	Bool	%M82.6		True	True	True				
Tag_60	Bool	%M80.4		True	True	True				
Remplir pots SF	Bool	%Q72.6		True	True	True				









Annexes

Totally Integrated Automation Portal								
Nom	Type de données	Adresse	Rémanence	Accessible depuis IHM/O PC UA	Ecriture autorisée à partir de IHM/O PC UA	Visible dans l'ingénierie IHM	Surveillance	Commentaire
Espace de travail LS 2 (pots)	Bool	%I72.5		True	True	True		Capteur station pots
Vider pots SF	Bool	%Q72.7		True	True	True		
Tag_64	Bool	%M12.1		True	True	True		
Tag_65	Bool	%M12.4		True	True	True		
Tag_66	Bool	%M14.3		True	True	True		
Tag_67	Bool	%M14.2		True	True	True		
Remplir (pots)	Bool	%I65.3		True	True	True		
Tag_69	Bool	%M8.2		True	True	True		
Piles inferieur (valve pots)	Bool	%Q72.2		True	True	True		
Cylindre pour piles au dessus (pots)	Bool	%I80.2		True	True	True		Station de pots
Bouchon fermé (valve pots)	Bool	%Q72.3		True	True	True		
Bouchons pour piles fermé (pots)	Bool	%I80.5		True	True	True		Station de pots
Tag_74	Bool	%M14.4		True	True	True		
INI en position (pots)	Bool	%I72.7		True	True	True		Capteur station pots
Temporisation 20	Timer	%T20		True	True	True		
Temporisation 21	Timer	%T21		True	True	True		
Tag_78	Bool	%M8.1		True	True	True		
Tag_79	Bool	%M14.0		True	True	True		
Tag_80	Bool	%M14.1		True	True	True		
remplir pots	Bool	%Q64.4		True	True	True		
vider pots	Bool	%Q64.5		True	True	True		
Lieu de transfert est libre LS (post)	Bool	%I72.6		True	True	True		Capteur station pots
Bouchon ouvert (valve pots)	Bool	%Q72.4		True	True	True		
Bouchons pour piles ouvert (pots)	Bool	%I80.4		True	True	True		Station de pots
Cylindre pour piles en dessous (pots)	Bool	%I80.3		True	True	True		Station de pots
Demande de pots	Bool	%I64.6		True	True	True		Extérieur de la machine de remplissage
Option garde-corps ouvert (valve pots)	Bool	%Q72.1		True	True	True		
Option garde-corps fermé (valve pots)	Bool	%Q72.0		True	True	True		
Tag_90	Bool	%M10.2		True	True	True		
Tag_91	Bool	%M10.3		True	True	True		
Acquitter le défaut	Bool	%I64.2		True	True	True		
Exterieur de l'installation en alerte	Bool	%Q64.7		True	True	True		

Annexes

Totally Integrated Automation Portal								
Nom	Type de données	Adresse	Rémanence	Acces-sible depuis IHM/O PC UA	Ecri-ture autor-isée à partir de IHM/O PC UA	Visi-ble dans l'ingé-nierie IHM	Surveillance	Commentaire
 Exterieur de l'intal-lation en foctionne-ment	Bool	%Q64.6		True	True	True		
 Temporisation 2	Timer	%T2		True	True	True		
 Temporisation 3	Timer	%T3		True	True	True		
 Tag_97	Bool	%M10.1		True	True	True		
 Arret d'urgence	Bool	%I65.7		True	True	True		Problemes
 Tag_99	Bool	%M10.5		True	True	True		
 Alerte	Bool	%Q64.1		True	True	True		
 Pret a l'emploi	Bool	%Q64.0		True	True	True		
 Tag_102	Bool	%M0.0		True	True	True		
 Option garde-corp gauche fermé (cou-vertcles)	Bool	%I73.1		True	True	True		Station de couvercles
 Option garde-corps droit fermé (couver-cles)	Bool	%I73.7		True	True	True		Station de couvercles
 Temporisation 5	Timer	%T5		True	True	True		
 Option garde-corp gauche ouvert (cou-vertcles)	Bool	%I73.0		True	True	True		Station de couvercles
 Option garde-corps droit ouvert (couver-cles)	Bool	%I73.6		True	True	True		Station de couvercles
 Temporisation 6	Timer	%T6		True	True	True		
 Temporisation 7	Timer	%T7		True	True	True		
 Temporisation 8	Timer	%T8		True	True	True		
 Temporisation 9	Timer	%T9		True	True	True		
 Temporisation 10	Timer	%T10		True	True	True		
 Option garde-corps gauche fermé (pots)	Bool	%I80.1		True	True	True		Station de pots
 option garde-corps droit fermé (pots)	Bool	%I80.7		True	True	True		Station de pots
 Temporisation 11	Timer	%T11		True	True	True		
 Option garde-corps gauche ouvert (pots)	Bool	%I80.0		True	True	True		Station de pots
 Option garde-corps droit ouvert (pots)	Bool	%I80.6		True	True	True		Station de pots
 Temporisation 12	Timer	%T12		True	True	True		
 Temporisation 13	Timer	%T13		True	True	True		
 Temporisation 14	Timer	%T14		True	True	True		
 Temporisation 15	Timer	%T15		True	True	True		
 Temporisation 16	Timer	%T16		True	True	True		
 Tag_123	Bool	%M80.1		True	True	True		
 Tag_124	Bool	%M80.0		True	True	True		
 Tag_125	Bool	%M80.3		True	True	True		

Annexes

Totally Integrated Automation Portal									
Nom	Type de données	Adresse	Rémanence	Acces- sible depuis IHM/O PC UA	Ecri- ture autor- isée à partir de IHM/O PC UA	Visi- ble dans l'ingé- nierie IHM	Surveillance	Commentaire	
 Tag_126	Bool	%M80.5		True	True	True			
 Tag_127	Bool	%M80.7		True	True	True			
 Tag_128	Bool	%M82.1		True	True	True			
 Vider (couvercles)	Bool	%I65.2		True	True	True			
 Tag_130	Bool	%M82.3		True	True	True			
 Tag_131	Bool	%M82.5		True	True	True			
 vider (pots)	Bool	%I65.5		True	True	True			
 Tag_133	Bool	%M82.7		True	True	True			