

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



Université A. Mira de Béjaïa
Faculté des Sciences Et Des Sciences De L'ingénieur
Département d'électrotechnique

Mémoire de Master

en

Electrotechnique

Option : Automatismes Industriels

Thème

*Migration d'un automate S5 vers S7 du
descenseur de la ligne 3 margarinerie Cevital*

présenté par :
Taouzinet Oualid

devant le jury composé de :

Président	M ^r A.Mokrani	U. A/Mira Béjaïa.
Rapporteurs	M ^r A/y.Achour	U. A/Mira Béjaïa.
Examineur	M ^r A. Mokrani	U. A/Mira Béjaïa.
Examineur	M ^r A/K. Laifaoui	U. A/Mira Béjaïa.

Béjaïa, juin 2013.

Remerciements

J'adresse mes vifs remerciements :

AM^r MOKRANI Ahmed, pour avoir accepté de présider ce jury.

A mon promoteur M^r ACHOUR ABD LYAZID, pour m'avoir assisté tout au long de se travail.

A M^r GOUDJIL BOUBKEUR INGENIEUR au sein de l'unité margarinerie CEVITAL.

Aux ingénieurs et techniciens de l'industrie de CEVITAL Mr BOUKRRARA KAMEL,
CHERRFI DJAMEL, AIMEN YACINE.

A Mr A. MOUKRANI,A/K. LIFAOUY pour avoir accepté de juger mon travail.

A nous amis pour leur aide et leurs encouragements.

A tous ceux qui ont contribué à ma formation de Master.

Dédicaces

Je dédie ce modeste travail à :

Ma chère mère CHERIFA qui ma vraiment aidé pour que je suis aujourd'hui devant vous.

Ma chère grande Mère Tassadit.

A la mémoire de mon grand père Debbou Seghir.

Mon frère Sofiane.

Ma sœur Asma.

Ma famille et tous mes proches.

Tous mes Amis kaci, Zina, Malek, Youba, Faicel, Toufik, Wassil, Takfa, Lyacine, Nadjim, Iyes,
Mehdi.

Table des matières

Table des Matières	i
Table des Figures	v
Liste des tableaux	vi
Introduction Générale	1
1 Description du descenseur	3
1.1 Introduction	3
1.2 Caractéristiques des matériaux du processus de fonctionnement	3
1.3 Branchement aux sources d'énergie, contrôles et vérification avant l'utilisation .	4
1.3.1 Connexion au réseau électrique[1]	4
1.3.2 Branchement au réseau pneumatique	4
1.4 Critères générales d'installation	5
1.5 Indication pour le transport, le positionnement et le fixage de la machine	5
1.6 La mise en marche	6
1.7 Système de commande	6
1.8 Alarmes et signalisations	7
1.9 Relèvement des anomalies et leur dépose	7
1.10 Contrôle et entretien sur l'automate du descenseur	8
1.10.1 remplacement de la batterie	8
1.10.2 remplacement EPROM	9
1.11 Les différentes parties de la machine	10
1.12 conclusion	13

2	Présentation des A.P.I	14
2.1	Introduction	14
2.2	Définition d'un automate programmable industriel	14
2.3	types d'automates étudiés	14
2.3.1	L'API S5 100U	14
2.3.2	L'API S7-300	15
2.3.2.1	Les caractéristiques du S7-300	16
2.3.2.2	Les modules du S7-300	17
2.4	Problématique	21
2.5	Cycle d'exécution d'un programme	21
2.5.1	Acquisition d'entrées	21
2.5.2	Traitement des données	21
2.5.3	Affectation des sorties	22
2.6	Principales de fonctions des automates programmables :	22
2.6.1	Carte d'entrées/sorties	22
2.6.2	Carte de comptage rapide	22
2.6.3	Carte de commande d'axe	22
2.6.4	Carte d'entrées/sorties analogique	22
2.6.5	Autre carte	22
2.7	Langages d'automates	23
2.7.1	Langage SFC (séquentiel function char) :	23
2.7.2	Langage FBD (function block diagram ou schéma par bloc)	23
2.7.3	Langage à contacte LD (ladder diagram ou schéma à relais)	23
2.7.4	Langage ST (structured text ou texte structuré)	24
2.7.5	Langage IL (instruction list ou liste d'instruction)	24
2.8	Technique de développement d'un projet sur un API	24
2.9	Choix d'un automate programmable industriel API	24
2.9.1	Amplitude des entrées/sorties	24
2.9.2	Type des entrées/sorties	24
2.9.3	Unité centrale	25
2.9.4	Alimentation	25
2.10	Domaines d'utilisation des automates programmables industriels (A.P.I)	25
2.11	conclusion	26

3	Présentation du SIMATIC manager	27
3.1	Introduction	27
3.2	présentation du STEP7	27
3.3	Création de projet avec STEP7	28
3.4	Programmation orientée objet	28
3.5	Mise en route du STEP7	28
3.6	GRAFCET	29
3.6.1	Les éléments de bases du GRAFCET	29
3.6.2	les règles d'évolution du GRAFCET	30
3.6.3	Cahier de charge du descenseur	30
3.6.4	GRAFCET du mode de fonctionnement	31
3.6.5	GRAFCET du système	34
3.6.6	GRAFCET de défaut	36
3.7	configuration des matériels sur HW Config	36
3.7.1	Introduction à la configuration matérielle	36
3.7.1.1	Configuration	37
3.7.1.2	Paramétrage	37
3.7.2	Manipulations de base pour la configuration matérielle	38
3.7.3	Marche à suivre pour la configuration d'une station	38
3.7.4	Organisation de la fenêtre de station	39
3.7.5	Table de configuration comme reproduction d'un profilé support ou châssis	39
3.7.6	Définition des propriétés des composants	40
3.7.7	Ouverture d'objets dans HW Config	41
3.7.8	Informations sur les règles d'enchâssement et autres règles	41
3.7.9	Configuration et paramétrage d'une installation centralisée	42
3.8	Présentation du simulateur S7-PLCSIM	43
3.8.1	Mise en route	43
3.8.2	Etat de fonctionnement de la CPU	44
3.8.3	Visualisation de la simulation du programme	45
3.9	conclusion	45

4 la migration de S5 vers S7	47
4.1 Introduction	47
4.2 Les différentes approches de résolution du problème	47
4.2.1 Conversion de S5 vers S7	47
4.2.1.1 La mise en route	47
4.2.1.2 Exécution du programme	48
4.2.1.3 Localisation des erreurs	49
4.2.1.4 Traitement des messages d'erreurs	49
4.2.1.5 Vérification de la cohérence	50
4.2.2 les phases de conversion	50
4.2.2.1 Messages d'erreur	51
4.2.2.2 Correction des erreurs	51
4.2.2.3 Avertissements	51
4.2.2.4 Les macro-instructions	56
4.2.2.5 Comparaison entre le programme en S5 et le programme en S7	57
4.3 conclusion	58
Conclusion	59
Bibliographie	60

Table des figures

1.1	les modules de S5-100U	9
1.2	chariot Elevateur	10
1.3	bande(petit tapis)	11
1.4	bandes(grand tapis)	12
1.5	Convoyeur à rouleaux aériens	13
2.1	la gamme SIMATIC S7	16
2.2	les modules de S7-300	17
2.3	Présentation de la CPU S7-300	18
3.1	Grafcet niveau 1 du mode de fonctionnement	31
3.2	Grafcet niveau 2 du mode de fonctionnement	32
3.3	Grafcet niveau 1 du fonctionnement général	34
3.4	Grafcet niveau 2 du fonctionnement général	35
3.5	Grafcet de default	36
3.6	Configuration des parties de l'automate dans HWconfig.	38
3.7	Représentation des modules dans la vue détaillée	39
3.8	tableau de configuration(profilé support/châssis)	40
3.9	les étapes essentielles pour configurer et paramétrer une installation	42
3.10	simulateur S7-PLCSIM.	43
4.1	phase 1 de la conversion	50

Liste des tableaux

2.1	Les fonctions des composantes constituant un A.P.I	20
4.1	tableau des erreurs signalisées 0	52
4.2	tableau des erreurs signalisées 1	53
4.3	tableau des erreurs signalisées 2	54
4.4	tableau des avertissements signalisées 0	54
4.5	tableau des avertissements signalisées	55
4.6	La comparaison entre les fonctions du STEP5 et celle de STEP7	57

Introduction Générale

L'automatisation correspond à une exécution totale ou partielle de tâches techniques par des machines fonctionnant sans intervention humaine. On pense naturellement à des systèmes mécaniques, hydrauliques, pneumatiques, électriques, électroniques ou électromécaniques, mais on peut trouver de nombreux exemples de fonctionnement sans intervention humaine en biologie, en médecine...

L'automatisation est un facteur essentiel de croissance de la productivité mais c'est aussi un élément indispensable de la qualité de la vie et des produits. Une des données principales de l'évolution de la production industrielle et le développement des systèmes automatisés, et par conséquent l'augmentation de la productivité. Pour fonctionner automatiquement, les systèmes industriels et les moyens de production font appel à des commandes programmées qui permettent une grande souplesse d'exploitation.

Quelques entreprises algériennes ont pris ces dernières années un peu d'avance en installant des systèmes automatisés très performants, mais n'arrivent souvent, pas à suivre l'évolution technologique, celle-ci est très coûteuse et évolue rapidement. Les constructeurs de machines sont toujours confrontés au défi de devoir proposer des machines plus flexibles et productives à des prix plus compétitifs, tout en assurant leur maintenance et le suivi de leur progrès industriel.

L'unité Margarinerie de CEVITAL est un exemple d'automatisation des systèmes de production en Algérie. Dans tout le processus de sa fabrication, les différentes étapes de l'élaboration de la margarine sont assurées par un matériel industriel automatisé ou l'intervention humaine est réduite à la surveillance des différents paramètres des machines qui assurent le bon fonctionnement de la chaîne de production.

Dans notre travail nous nous sommes intéressés au descenseur de la ligne 3 margarinerie dans lequel le système de commande actuel présente des difficultés pour le service de maintenance, nous avons essayé de concevoir un système de commande à base d'un automate programmable

industriel (API) S7-300 adéquat avec celui de S5-100U.

Ce mémoire comprend une introduction, quatre chapitres, 2 annexes et une conclusion.

Le chapitre un, est consacré à la présentation et la description du descenseur de la ligne 3 margarinerie Cevital, ainsi qu'à ces différentes parties : partie mécanique, système de commande, sa mise en marche.

Le deuxième chapitre, consiste en une étude comparative entre le S5-100U et le S7-300. Il introduit l'architecture de ces équipements, la problématique, le cycle d'exécution d'un programme et le choix de l'automate.

Le troisième chapitre traite de la résolution du problème du descenseur et l'automatisation du système qui se termine par le développement d'un GRAFCET de fonctionnement de la machine.

Le quatrième chapitre, quant à lui, abordera d'abord nous traitons l'ensemble des données pour élaborer un programme en STEP7, Pour ensuite passer à la simulation du déroulement du programme par le simulateur S7-PLCSIM.

Le chargement du programme dans la CPU de l'automate S7-300 sert à vérifier le bon fonctionnement de la machine.

Enfin, ce travail est clôturé par une conclusion générale.

Description du descenseur

1.1 Introduction

Le descenseur en fonctionnement automatique a comme fonction d'amener les cartons de la zone d'encartonnage à celle de la palettisation, pour les amener aux successifs usinages. La machine est composée de [1] :

- Châssis électrosoudé ;
- Installation électrique composée d'un tableau électrique avec panneau de commande des moteurs électriques ;
- Groupe de soulèvement ;
- Installation pneumatique pour les vérins ;
- Plusieurs photocellules pour le contrôle des phases de travail ;
- Capteur magnétique.

1.2 Caractéristiques des matériaux du processus de fonctionnement

La machine exécute automatiquement la descente verticale de cinq boites par cycle d'un transporteur. Le fonctionnement de la machine se vérifie par deux mouvements des rouleaux du transporteur permet l'entrée et la sortie des produits, par contre un deuxième motoréducteur à travers des chaines permet la descente du descenseur, des arrêts mécaniques actionnés par des vérins pneumatiques empêchent la chute pendant la descente [1].

1.3 Branchement aux sources d'énergie, contrôles et vérification avant l'utilisation

L'installation est alimentée soit par l'énergie électrique soit par l'énergie pneumatique. Les sources d'énergie dont la machine a besoin, doivent assurer pendant son fonctionnement normale les même valeurs indiquées dans le manuel de la machine. Le réseau électrique ou la machine est branchée, doit être adéquat aux normes en vigueur en matière de sécurité électrique et doit être pourvue d'un interrupteur différentiel. Il faut vérifier que la personne qui utilise la machine possède la déclaration de la conformité.

On suggère l'installation d'un interrupteur automatique amont de la machine pour la conformité de l'installation électrique aux bornes imposées de DPR547 en se qui concerne la connexion à la terre.

En particulier le branchement a partir du réseau électrique doit suivre la procédure suivante et effectuée par un personnel spécialisé[1].

1.3.1 Connexion au réseau électrique[1]

- S'assurer que l'interrupteur général soit en position " O ". insérer tous les connecteurs électriques dans leurs prises respectives. L'ordre d'accouplement de tous les connecteurs et de toutes les prises et indiqué opportunément avec des lettres et des nombres ;
- Porter un câble électrique de section 6mm² et cinq pôles à tableau électrique et brancher les connecteurs aux bornes correspondants. La boîte à borne située généralement à l'intérieur du tableau électrique dans la partie en bas à gauche est caractérisée par des sigles L1, L2, L3 et de la borne PE d'une couleur jaune/verte ou il faut brancher le câble au sol. Le câble du branchement du neutre doit être relié au borne correspondant N seulement si ce branchement est prévu ;
- Contrôler que la tension du réseau corresponde à la tension de travail de la machine et vérifier avant la mise en marche que le câble d'alimentation soit proportionnelle à l'absorption de la même ;
- Brancher tous les connecteurs disjoints pour le transport et contrôler absolument le sens de rotation des moteurs. Vérifier le correct sens de rotation des moteurs. Si le sens de rotation donne un résultat erroné il faut renverser la connexion des deux câbles de phase sur des bornes correspondants dans le tableau électrique.

1.3.2 Branchement au réseau pneumatique

Pour assurer son bon fonctionnement il est conseillé[1] :

- Le diamètre du réseau d'alimentation pneumatique de l'établissement ne soit pas inférieur à 1/2 cm avec une pression de 6-7 bar ;

- L'air comprimé doit être propre et séché, l'air humide et/ou sale endommage l'installation en usant tous les composants pneumatiques.
 1. Réunir tout les tuyaux dans leurs greffes respectives des dispositifs différents. L'ordre correct des liaisons est marqué avec des références numériques. Connecter l'installation pneumatique principale à la machine en utilisant l'attache rapide dont elle est équipée ;
 2. Tarer le régulateur de pression générale à 6/7atm. Tourner le galet vers le haut et après avoir tourné dans le sens inverse aux aiguilles d'une montre, la pression se baisse à zéro, toujours dans le sens des aiguilles d'une montre on augmente de plus en plus la pression ;
 3. Dans le cas où on trouve l'eau de condensation dans la cuve du filtre, il faut la décharger en appuyant sur un bouton spécial situé au bas de la cuve ;
 4. Décharger l'eau de condensation à l'intérieur du réservoir de blair en utilisant le robinet approprié ;
 5. Tarer la vitesse des cylindres en opérant sur les régulateurs de débit (ou régulateur de flux) sur les attaches d'entrées et de sorties d'air de chaque cylindre. ces opérations doivent être exécutées par des techniciens de la TMG IMPIANTI ou du personnel spécialisé seulement si les mouvements des vérins ne sont plus fluides . . .

1.4 Critères générales d'installation

La machine doit être installée dans un lieu sec et couvert, il faut éviter les milieux corrosifs, explosifs ou trop humides. Le lieu où est située la machine à besoin de sa propre illumination, car la machine est dépourvue. S'assurer que le plancher soit parfaitement horizontal et capable de supporter le poids de la machine. La zone d'installation (ou le lieu de travail) doit posséder toutes les qualités qui concernent la hauteur, le changement d'air et respecter les prescriptions imposées par les normes en vigueur, en particulier en référence au décret législative 626/994[1].

1.5 Indication pour le transport, le positionnement et le fixage de la machine

La machine a été étudiée pour être facilement transportable. La machine doit être située dans une position qui permet l'accès pour l'entretien et en particulier il faut s'assurer que les portes d'accès puissent être ouvertes sans difficulté [1]. il faudra réserver un espace pour permettre les manoeuvres d'un élévateur pour les opérations d'alimentation. la machine doit être positionnée de façon parfaitement horizontale par le réglage de la hauteur des supports filetés, en la contrôlant par une bulle. La stabilité de la machine est assuré par la fixation au sol[1].

1.6 La mise en marche

Avant de mettre en marche l'installation, il faut vérifier les points suivant [1] :

1. Le correct fonctionnement de la photocellule :
 - Couvrir la lecture de la photocellule ;
 - Contrôler que le témoin installé sur la photocellule s'allume ou s'éteint, si le témoin ne change pas son état cela signifie que :
 - a) La photocellule est déplacée de son axe de travail ;
 - b) Le câble de branchement est interrompu ;
 - c) La photocellule est cassée.
 - Vérifier l'exacte positionnement de la photocellule ou du catadioptré en les reportant dans la position correcte et répéter la procédure susmentionnée ;
 - Si la LED ne signale pas la couverture de la photocellule, il faut procéder en suivant les indications des anomalies et leur élimination.
2. vérifier que les barrières de prévention contre les accidents soit actives et que les tableaux de commandes soient fermés.

1.7 Système de commande

RESTART :

Après le branchement de l'alimentation électrique :

A ce point il faut vérifier que :[1]

1. Toutes les portes sont fermées ;
2. Les boutons d'urgence sur le tableau ne soient pas appuyés ;
3. La pression de l'air sur le circuit pneumatique soit présente et supérieur à 6 atm (max 7 atm) ;
4. Si toutes ces conditions sont vérifiées on appuyant sur le bouton RESTART clignotera pour 2, 3 secondes en signalisant l'état de l'attente chargée des circuits de puissance. Lorsque le témoin est éteint la machine sera désactivée. A ce point il est possible d'activer les fonctionnements MAN ou AUTO (manuel ou automatique).

AUTOMATIQUE :[1].

1. Réarmer la machine comme il est décrit ;
2. Mettre le sélecteur MAN/AUTO en AUTO ;

3. appuyer le bouton START pour 2, 3 secondes jusqu'à quand la lampe vertes du même reste allumé fixe pour indiquer que le mode automatique est actif;
4. Lorsque le bouton START reste allumé, le cycle automatique résulte démarré. Dans le cas contraire il faut enlever les causes et répéter la procédure .

MANUEL

1. Réarmer la machine comme est décrit ;
2. Mettre le sélecteur MAN /AUTO sur MAN ;
3. Appuyer sur l'un des boutons pour les mouvements en manuel :
 - Montée-descente élévateur ;
 - Convoyeur à rouleaux élévateur ;
 - Convoyeur à rouleaux en entrée .

RESET

1. Enlever les éventuels produits sur le convoyeur à rouleaux du descenseur ;
2. Tourner le sélecteur dans la modalité de fonctionnement MAN ;
3. Appuyer sur la touche RESET .

1.8 Alarmes et signalisations

Pendant l'utilisation de la machine on peut se présenter plusieurs causes d'alarmes qui entrent de différente façon sur le fonctionnement de la machine selon leur importance. L'activation d'un alarme est signalisée du clignotement de la lampe ALARMES. la signalisation continue jusqu'à quand l'alarme est reconnu par le bouton START[1].

1.9 Relèvement des anomalies et leur dépose

Machine désarmée [1] :

Cause : activation d'un alarme qui dans la suite est jugulée.

Conséquence : tous les mouvements sont désactivés.

Remède : procéder au réarme.

Urgence appuyée [1] :

Cause : l'un ou plusieurs boutons d'urgence appuyés.

Conséquence : désarme de la machine.

Remède : rétablir les boutons appuyés a leurs places.

Thermiques [1] :

Cause : intervention d'un interrupteur de protection des moteurs.

Conséquence : désarme de la machine.

Remède : enlever la cause du blocage avec la coupe circuit et rétablir le coupe circuit même.

TRIP [1] :intervention de la protection d'un dispositif d'actionnement des moteurs

Conséquence : la manutention et le désarme de la machine sont désactivés.

Remède : attendre le désarme de la machine (il est possible d'avancer le désarme de la machine en utilisant le bouton émergence, si l'alarme reste alors il faut choisir la première solution.

Portes ouvertes [1] :

Cause : ouverture d'une ou plusieurs des portes de sécurité de la machine.

Conséquence :désarme de la machine.

Remède : fermer toutes les portes de sécurité.

1.10 Contrôle et entretien sur l'automate du descenseur

Les opérations nécessaires pour l'entretien du module -simatic S5-100U-son[1] :

1. Logement porte-batterie ;
2. Connecteur 15 pin connexion pc... ;
3. Logement porte-eprom ;
4. Indicateur d'absence batterie ;
5. Interrupteur ON/OFF ;
6. Indicateur de mode de fonctionnement ;
7. Commutateur mode fonctionnement ;
8. Bornes connexion électrique.

1.10.1 remplacement de la batterie

La durée de fonctionnement d'une nouvelle batterie à lithium d'un module SIMATIC est au moins un an. L'absence ou la panne d'une batterie sont indiqués par un témoin jaune[1]. On ne peut pas recharger les batteries, car elles pourraient exploser, les batteries épuisées sont des refuses spécial. Exécuter la mise en place ou le change de batterie seulement si le module est

sous-tension c'est-à-dire si l'un des LED de l'indicateur de mode est allumé. Les caractéristiques des batteries sont : Batterie à lithium size 112mA 3.6V Pour le remplacement il faut suivre la procédure suivante :

- Ouvrir le conteneur ;
- Enlever la batterie existante ;
- Mettre la nouvelle batterie ;
- Fermer le conteneur.

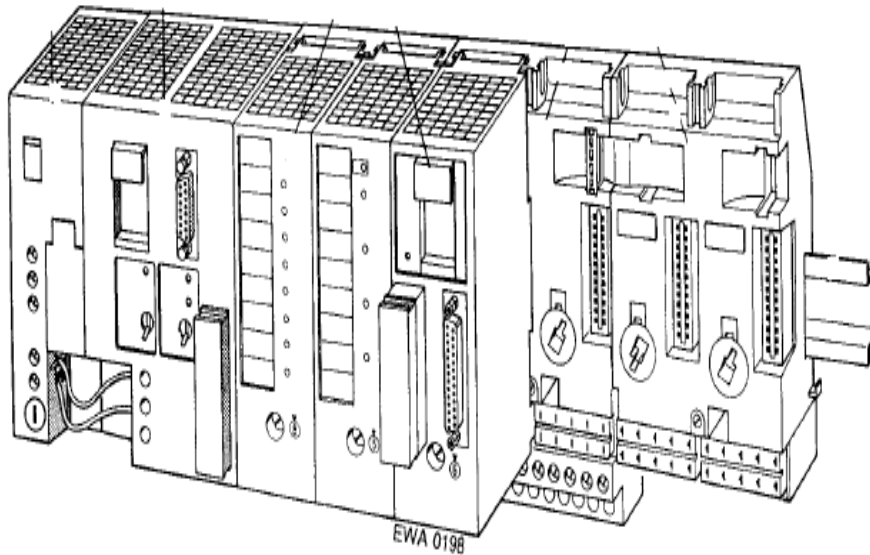


FIGURE 1.1 – les modules de S5-100U

1.10.2 remplacement EPROM

Pour le remplacement qui est nécessaire dans le cas d'une modification de la programmation ou dans le cas de panne il faut suivre la procédure suivante :

- Eteindre l'interrupteur général ;
- Enlever la batterie du module et remplacer l'eprom ;
- Allumé du nouveau l'interrupteur général ;
- Maintenant le LED rouge devrait clignoter rapidement pour quelques instants ;
- Quand le LED vert s'allume il faut mettre la batterie.

1.11 Les différentes parties de la machine

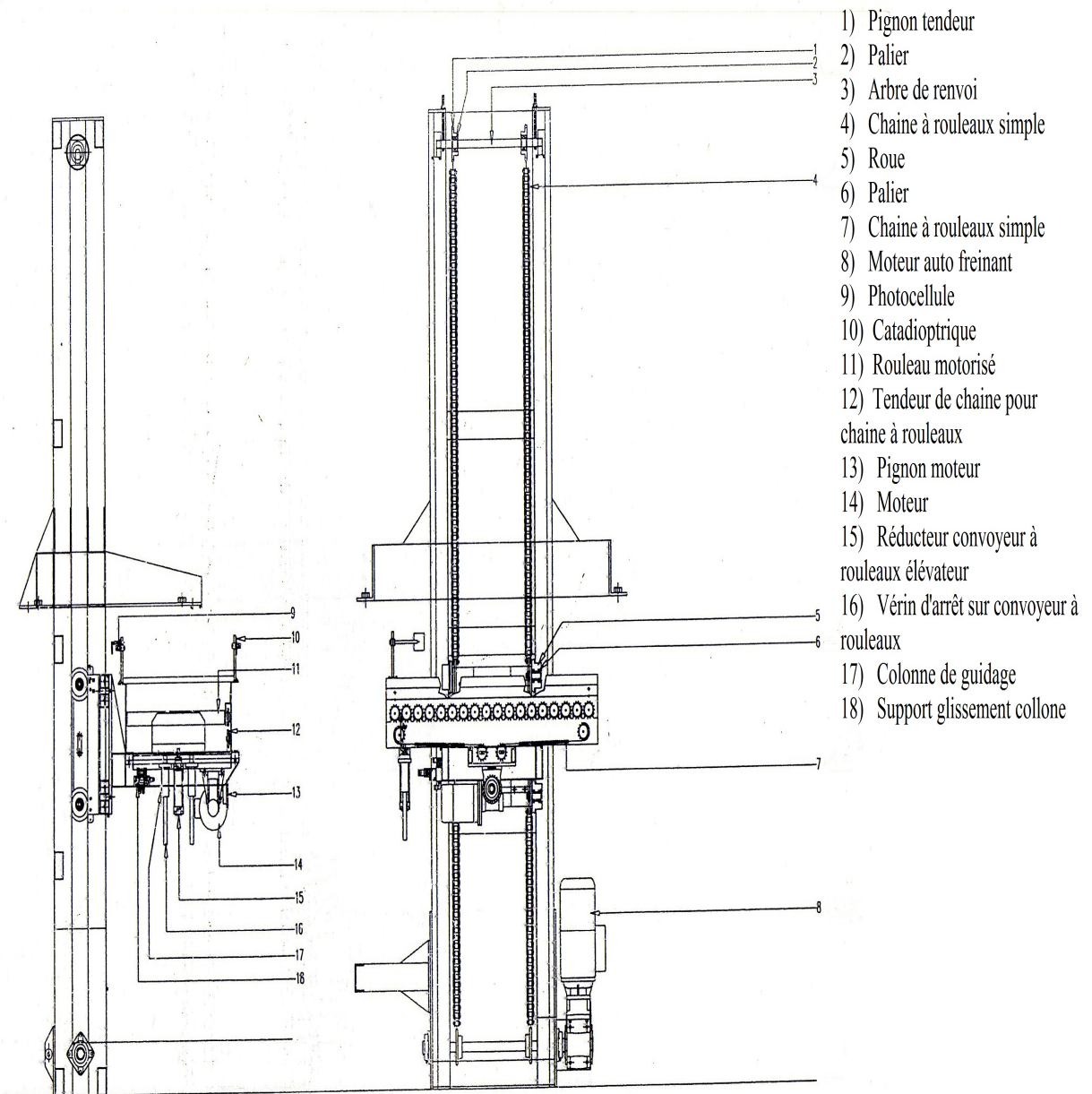


FIGURE 1.2 – chariot Elevateur

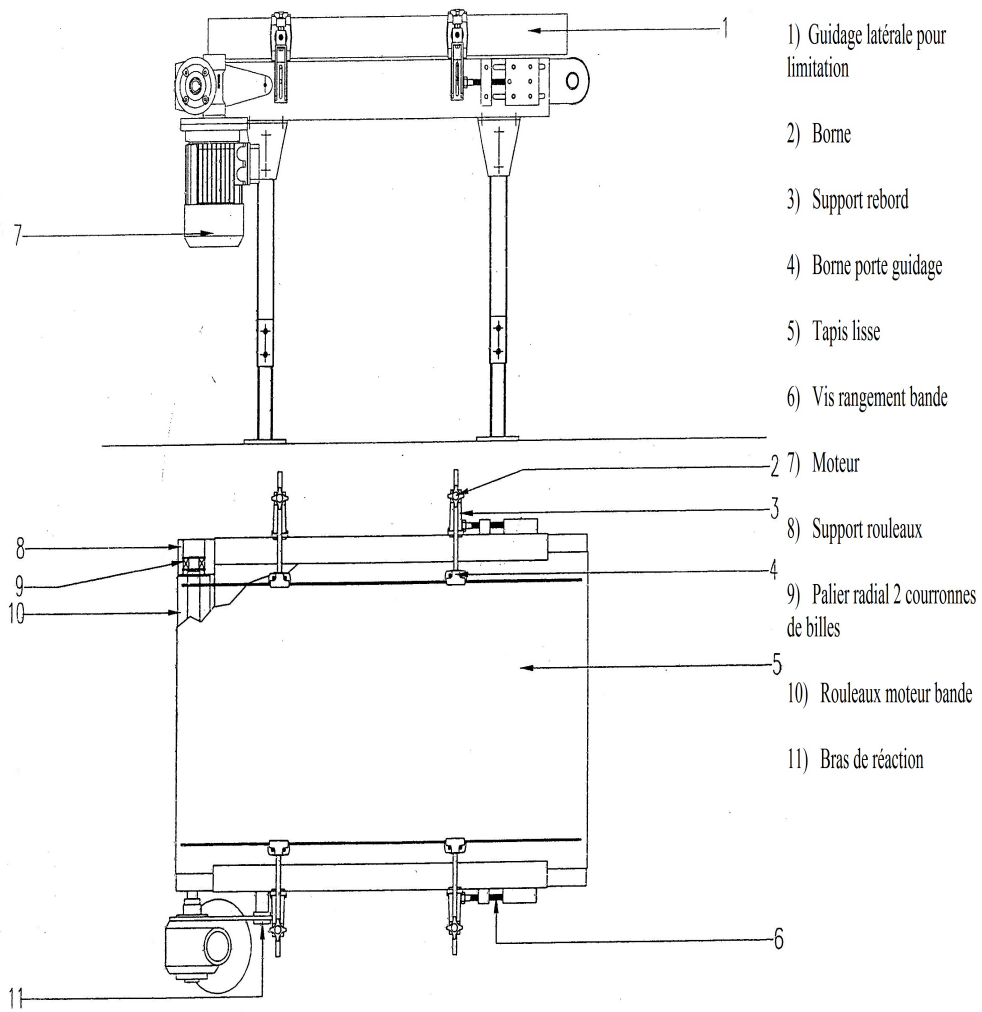


FIGURE 1.3 – bande(petit tapis)

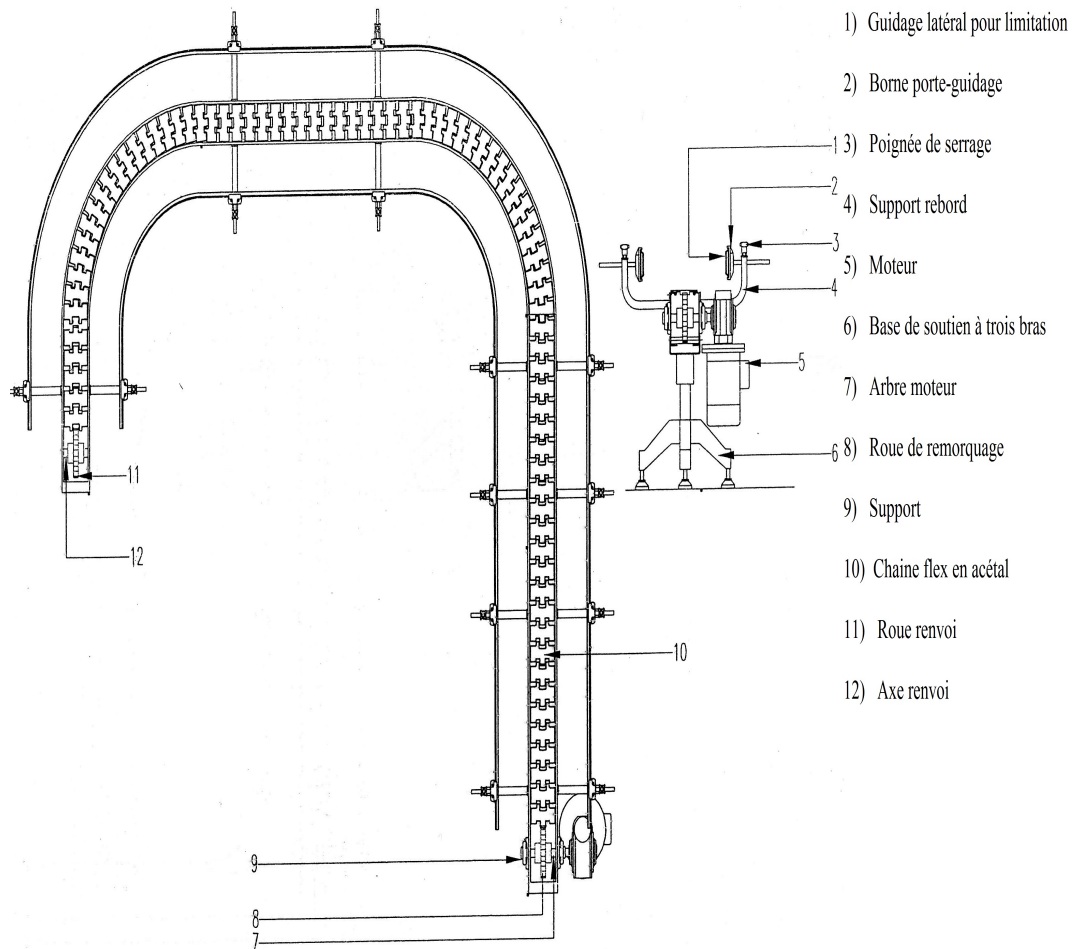


FIGURE 1.4 – bandes(grand tapis)

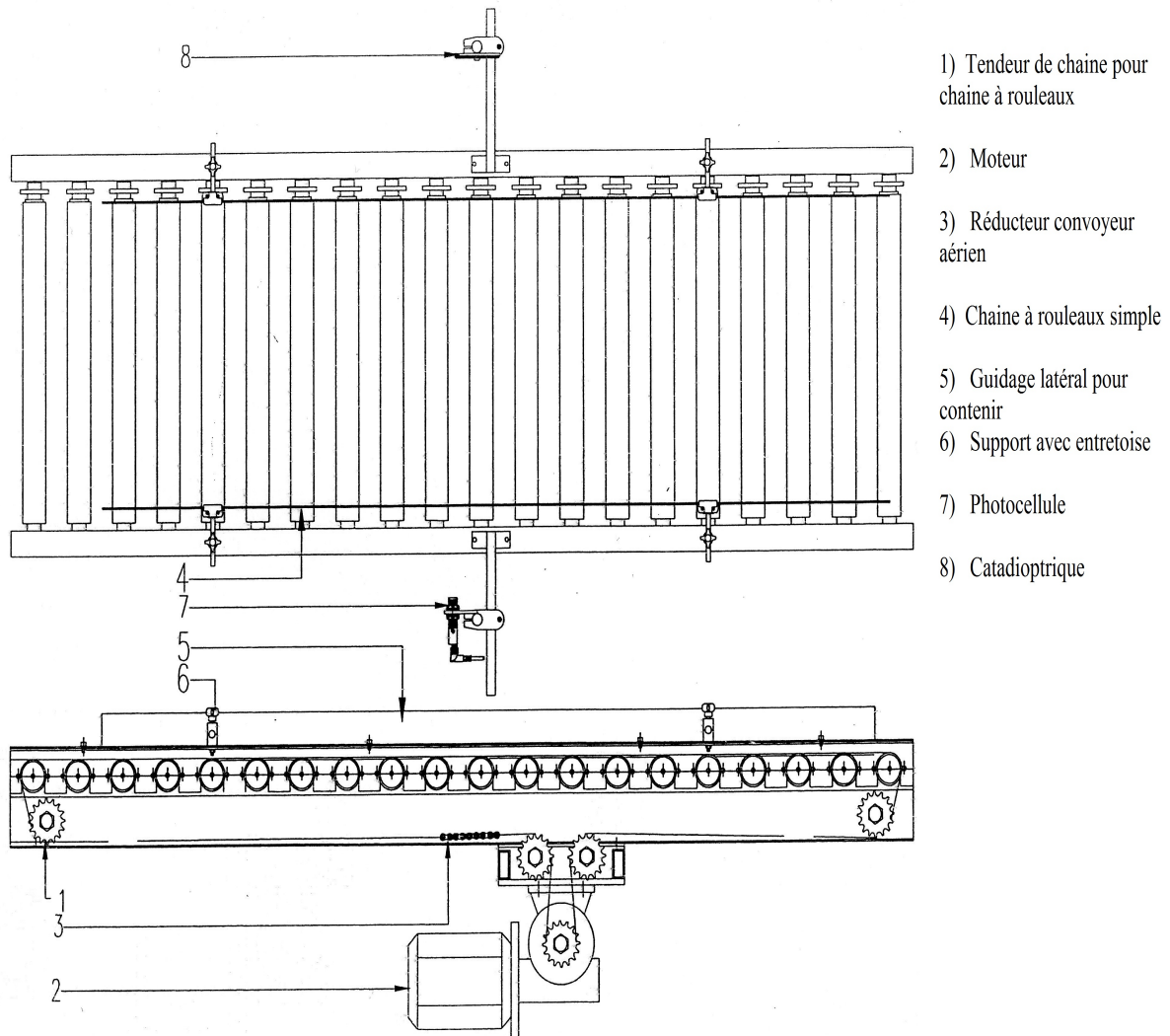


FIGURE 1.5 – Convoyeur à rouleaux aériens

1.12 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 choisi l'API SIEMENS S7-300, nous procédons dans la suite de notre travail à la présentation des API S5-100U et S7-300, et la méthode de programmation sous SIMATIC Manager. La machine fonctionne de façon complètement automatique.

Présentation des A.P.I

2.1 Introduction

Les premières API ont été introduit en 1969 aux Etats-Unis pour satisfaire les besoins de l'industrie automobile. Le but recherché était de remplacer les armoires à relais utilisées pour l'automatisation des chaines de fabrication par des équipements moins coûteux et surtout plus faciles à modifier. Depuis leur apparition, les automates programmables se sont réponsus très rapidement dans l'industrie, au point de représenter aujourd'hui plus de la moitié des équipements informatiques qui sont utilisés pour ce type d'application[2].

2.2 Définition d'un automate programmable industriel

Un automate programmable industriel (API) est un appareil électronique programmable, adapté à l'environnement industriel, qui réalise des fonctions d'automatisme pour assurer la commande de préactionneurs et d'actionneurs à partir d'information logique, analogique ou numérique[2].

2.3 types d'automates étudiés

2.3.1 L'API S5 100U

Structure interne du S5 100U :

a).Module d'alimentation : il assure la distribution d'énergie aux différents modules, il est de marque SITOP, sa référence est : 6EP1333-2AA00. [3]

b).Unité central(CPU) : à base de microprocesseur, elle réalise toutes les fonctions logiques, arithmétique et de traitement numérique (transfert, comptage, temporisation...)[3].

Sa référence est : 6ES5 100-8MA02.

c).Le bus interne : Il permet la communication de l'ensemble des blocs de l'automate et des éventuelles extensions[3].

d).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[3].

e).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 :

o Module d'entrées " 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.

o 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.

La référence de ces entrées est : 6ES5 421-8MA12. (4 entrées)[3].

- **Modules de sorties :** Ils permettent de commander les divers pré actionneurs et éléments de signalisation du S.A.P (système automatisés de production).

Il existe deux types modules de sorties :

o module de sortie < 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.

o Module de sortie analogique :

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

La référence de ces sorties est : 6ES5 441-8MA11 (2 sorties)[3].

2.3.2 L'API S7-300

Outre une alimentation en énergie, l'exploitation des machines, des équipements et des processus mis en œuvre sur un site de production requiert généralement des appareils de commande capable d'assurer le déclenchement, la commande, la surveillance et l'arrêt des installations.

De la logique câblée à l'automate programmable :

Le programme de commande des automates à logique câblée, couramment utilisée dans le passé, était déterminé par le câblage des contacteurs et des relais, spécifique à la tâche

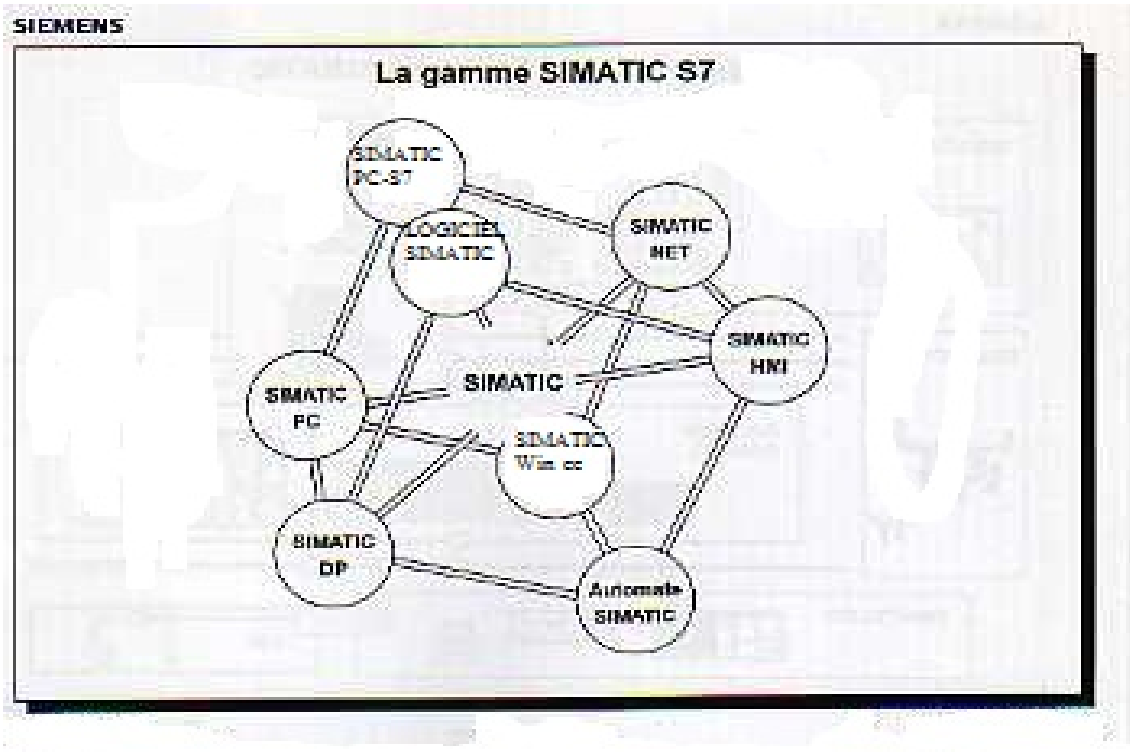


FIGURE 2.1 – la gamme SIMATIC S7

à exécuter.

Aujourd'hui, on utilise des automates programmables pour résoudre les tâches d'automatisation. La logique stockée dans la mémoire programme du système d'automatisation est indépendante de la configuration matériels et du câblage et peut donc être modifiée à tout moment à l'aide d'une console de programmation.

Automatisation entièrement intégrée :

Aujourd'hui les processus de production ne sont plus considérés isolément, mais intégrés dans une démarche globale. L'intégration complète de tout l'environnement d'automatisation est désormais réalisé grâce à :

- Une configuration et une programmation homogènes des différentes unités du système.
- Une gestion cohérente des données.
- Une communication globale entre tous les équipements d'automatisme mis en œuvre.

2.3.2.1 Les caractéristiques du S7-300

- . Mini-automate modulaire pour les applications d'entrée et de milieu gamme.
- . Gamme diversifiée de CPU.
- . Possibilité d'extension jusqu'à 32 modules.
- . Bus de fond de panier intégré aux modules.
- . Possibilité de mise en réseau avec :

- L'interface multipoint (MPI).
- PROFIBUS.
- Industriel Ethernet.
- . Raccordement central de la PG avec accès à tous les modules.
- . Liberté de montage aux différents emplacements.
- . Configuration et paramétrage à l'aide de l'outil "Configuration matérielle "[4].

2.3.2.2 Les modules du S7-300

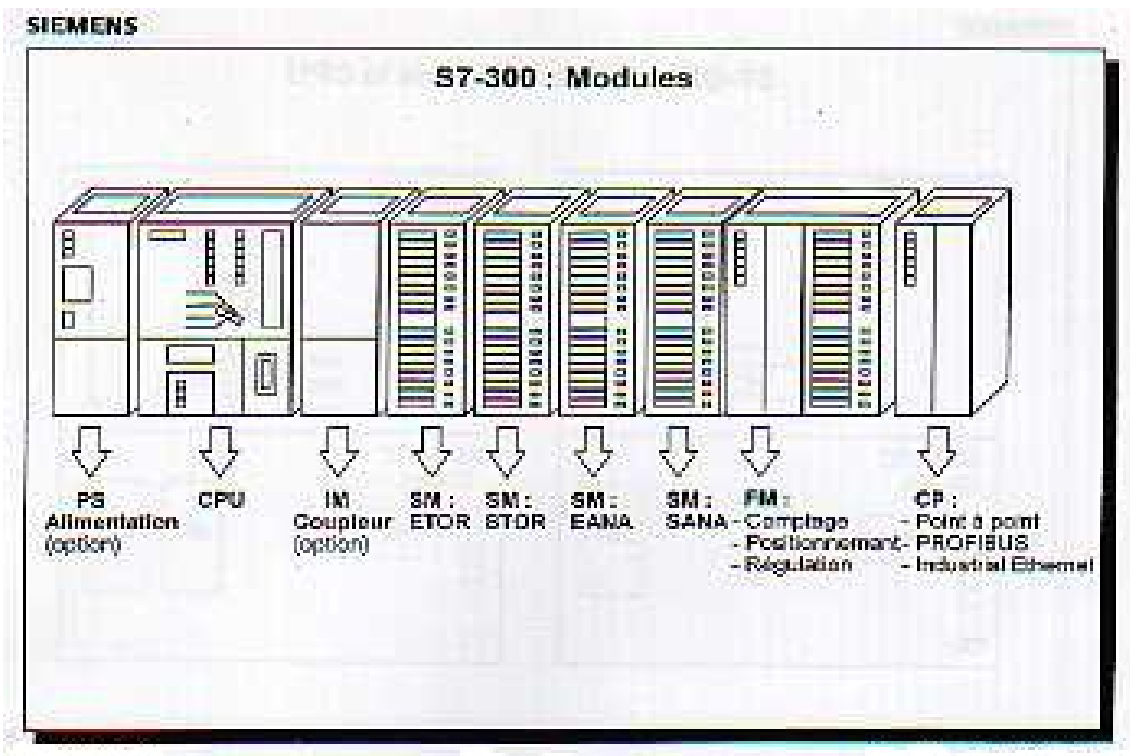


FIGURE 2.2 – les modules de S7-300

- Modules de signaux (SM) :

- . Modules ETOR : 24V=, 120/230V
- . Module STOR : 24V=, Relais
- . Module EANA : tension, courant,
- . Module SANA : tension, courant

- Coupleurs(IM) :

Les coupleurs IM360/IM365 ou IM365 permettent de réaliser des configurations à plusieurs châssis.

Le bus est relié en boucle entre les différents châssis.

– **Signalisation d'état(LED) :**

SF = Signalisation groupée de défauts, défaut interne de la CPU ou d'un module avec fonction de diagnostic.

BATF = Défaut de pile, pile à plat ou absente.

DC5V = Signalisation de la tension d'alimentation interne 5V.

Allumage fixe 5V OK, Clignote : surcharge courant.

FRCE = Forçage, signalisation qu'au moins une entrée ou une sortie est forcée de manière permanente.

RUN = Clignotement à la mise en route de la CPU, allumage continu en mode Run.

STOP = Allumage continu en mode STOP.

Clignotement lent lorsqu'un effacement général est requis.

Clignotement rapide lorsqu'un effacement général est en cours.

Clignotement lent lorsqu'un effacement général est requis par en fichage de la carte mémoire.

– **Carte mémoire :**

une carte mémoire peut être montée à cet emplacement. Elle conserve le contenu du programme en cas de coupure de courant, même en l'absence de pile.

– **Logement de la pile :**

un emplacement protégé par un cache est prévu pour une pile au lithium permettant de sauvegarder le contenu de la mémoire RAM en cas de coupure de courant.

– **Logement de la pile :**

pour raccorder la console de programmation ou un autre appareil sur l'interface MPI.

– **Logement de la pile :**

interface de raccordement d'une périphérie décentralisée directement à la CPU [5].

Introduction : un système d'automatisation S7-300 comprend les composantes indiquées dans le tableau ci-dessus.

Rail profilé : l'alimentation électrique, la CPU, la carte de couplage IM et 8 modules de signaux maxi. Sont montés sur le rail profilé (profil support).

Alimentation : le module d'alimentation délivre, sous une tension 24V courant de sortie assigné de 2A, 5A et 10A. La tension de sortie, à séparation galvanique, est Protégée contre les courts-circuits et la marche à vide. Une DEL indique le bon fonctionnement du module d'alimentation. En cas de surcharge de la tension sortie, le témoin se met à clignoter. Un sélecteur permet de sélectionner la valeur de la tension primaire (120 V ou 230 V).

Unité centrale : La CPU regroupe les éléments suivant en face avant :

– signalisation d'état de default.

– commutateur a clé amovible à 4 positions.

– raccordement a tension à 24 V DC.

– interface multipoint MPI pour console de programmation ou couplage à un autre système

Composantes	Fonction
Rail profilé Alimentation(PS)	... constitue le châssis du S7-300 ... convertit la tension secteur (120/130 V) en tension de service 24 a V DC pour alimenter le S7-300.
Unité centrale(CPU)	... exécute le programme utilisateur, Accessoires : carte mémoire, pile de sauvegarde.
Carte de couplage(IM)	... relie le bus entre les différents châssis.
Modules de signaux (SM) (TOR /analogiques)	... adaptent les différents niveaux de signaux de processus au S7-300. Accessoires : connecteurs de bus, connecteur frontal
Modules de fonction (FM)	... assurent des fonctions de positionnement, de régulation, etc ...
Processeurs de communication(CP)	... permettent le couplage entre plusieurs automates. Accessoires : Cartes, logiciel, carte d'interface

TABLE 2.1 – Les fonctions des composantes constituant un A.P.I

d'automatisation.

- compartiment pour pile de sauvegarde (non disponible dans la CPU 312IFM).
- logement pour carte mémoire (non disponible dans la CPU 312IFM, 314IFM).

Carte de couplage : Les coupleur permettent de disposer d'une configuration à plusieurs châssis.

Modules de signaux : Ces modules sont sélectionnés en fonction de la plage tension ou de la tension de sortie. Chaque module et dote d'un connecteur de bus permettant le bouclage du bus de fond de panier[5]. Les signaux du processus sont reliés au connecteur frontal enfichable.

Câblage : Il faut un câble PG pour raccorder directement une console de programmation. Pour connecter plusieurs systèmes d'automatisation, il faut en outre un câble PROFIBUS et des connecteurs.

FM : Les modules de fonction remplacent les cartes IP actuelles. **CP :** processeur de communication pour le système de bus PROFIBUS [5].

2.4 Problématique

- Plus de fourniture en termes de pièces de rechange.
- Difficulté de programmation " S5 sur base MS DOS ".
- Difficulté de diagnostic.
- Exigence de la part de fournisseur.

2.5 Cycle d'exécution d'un programme

Lorsque l'API est en fonctionnement, c'est-à-dire, lorsqu'il exécute son programme de contrôle sur le système extérieur, une série d'opération effectuée de façon séquentielle et répétitive [6].

2.5.1 Acquisition d'entrées

Au début de chaque cycle, l'automate programmable examine l'état de tous les signaux d'entrées et puis procède à leur écriture dans la mémoire image des entrées [6].

2.5.2 Traitement des données

Au début de l'exécution de programme, l'unité central lit successivement les instructions dans la mémoire interne. ensuite, elle procède au traitement des instructions et évaluation des grandeurs de sorties. Une fois ces valeurs sont calculées, elles sont stockées en mémoire image des sorties [6].

2.5.3 Affectation des sorties

Après l'exécution du programme, l'automate procède à la lecture des sorties dans la mémoire images de ces dernières, et puis les transferts vers les modules de sorties [6].

2.6 Principales de fonctions des automates programmables :

2.6.1 Carte d'entrées/sorties

Au nombre de 4, 8, 16, 32, elles peuvent aussi bien réaliser des fonctions d'entrées, de sorties ou les deux. Ce sont les plus utilisées et les tensions disponibles sont normalisées (24, 48, 110 ou 230V continu ou alternatif. . .).

Les cartes d'entrées permettent de recueillir l'information des capteurs, . . . qui lui sont raccordés et de la matérialiser par un bit image de l'état de capteur.

2.6.2 Carte de comptage rapide

Elles permettent d'acquérir des informations de fréquences élevées incompatibles avec le temps de traitement de l'automate.

2.6.3 Carte de commande d'axe

Elles permettent d'assurer le positionnement avec précision d'élément mécanique selon un ou plusieurs axes. La carte permet par exemple de piloter un servomoteur et de recevoir les informations de positionnement par un codeur. L'asservissement de position pouvant être réalisé en boucle fermée[6].

2.6.4 Carte d'entrées/sorties analogique

Elles permettent de réaliser l'acquisition d'un signal analogique et sa conversion numérique (Can) indispensable pour assurer un traitement par le microprocesseur .la fonction inverse (sortie analogique) est également réalisée.

2.6.5 Autre carte

- o Carte de régulation PID ;
- o Cartes de pesage ;

- o Cartes de communication (Ethernet. . .).

2.7 Langages d'automates

La norme IEC 1131-3 définit entre autre choses, cinq langages qui peuvent être utilisés pour la programmation d'application d'automatismes [7]. Les cinq langages sont :

2.7.1 Langage SFC (séquential function char) :

Issu du langage grafcet, ce langage de haut niveau permet la programmation aisée de tous les procédés séquentiels.

Un programme SFC est un réseau graphique d'étapes et de transitions, reliées par des liaisons orientées. Les liens de connexion multiple sont représentés par des divergences et des convergences. une partie du graphique, nommée macro étape peut être isolée, et représentée dans le graphique principal par un seul symbole [7].

Les principales règles graphiques sont :

- o Une étape ne peut pas être suivie d'une autre étape.
- o Une transition ne peut pas être suivie d'une autre transition.

2.7.2 Langage FBD (function block diagram ou schéma par bloc)

Ce langage permet de programmer graphiquement à l'aide de blocs, représentant des variables, des opérateurs ou des fonctions. Il permet de manipuler tous les types de variables.

Le diagramme FBD décrit une fonction entre des variables d'entrées et des variables de sorties. Une fonction est décrite comme un réseau de fonctions élémentaires. Les variables d'entrée et de sortie sont connectées aux boîtes fonctions par des arcs de liaison. Une sortie d'une boîte peut être connectée sur une entrée d'une autre boîte [7].

2.7.3 Langage à contacte LD (ladder diagram ou schéma à relais)

Ce langage 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 [7].

2.7.4 Langage ST (structured text ou texte structuré)

Ce langage est 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 et des conditions associées aux transitions du langage SFC [7].

2.7.5 Langage IL (instruction list ou liste d'instruction)

Ce langage est 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 les types 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 [7].

2.8 Technique de développement d'un projet sur un API

Avant d'entamer un projet sur un A.P.I il faut être méthodique pour développer une application complexe en technologie. La démarche suivie pour réaliser un projet sur un A.P.I. s'apparente d'avantage à la méthodologie pratiquée sur ordinateur qu'aux procédures utilisées en logique câblée [8].

2.9 Choix d'un automate programmable industriel API

Pour le choix d'un matériel et une configuration capable de résoudre le problème d'automatisation. Il revient à l'utilisateur d'établir le cahier de charge de son système. Cette phase mérite la plus grande attention.

2.9.1 Amplitude des entrées/sorties

Le premier paramètre à prendre en compte pour choisir un automate est le nombre d'entrées et de sorties nécessaires il pourra y avoir un bloc de base et des extensions, ou une unité centrale et des cartes d'entrées ou de sorties. On commencera donc par faire le bilan des entrées et des sorties [8].

2.9.2 Type des entrées/sorties

Les entrées et les sorties peuvent être :
Logique : entrées et sorties tout ou rien [8].

Analogique : liaison avec génératrice bathymétrique en entrée et variateur de vitesse en sortie par exemple[8].

Numérique : comptage rapide sur un codeur incrémental. Chaque entrée ou sortie devra être adaptée au capteur ou au pré actionneur. Les cartes assurent l'isolation galvanique entre l'unité centrale et le système. Les cartes de sortie peuvent être à relais, elles permettent de commander des contacteurs par exemple. Elles sont parfois à transistor et permettent alors de commuter des signaux à plusieurs centaines de hertz [8].

2.9.3 Unité centrale

C'est le cœur de l'automate. Elle comporte un microprocesseur et de la mémoire qui permettent de définir sa puissance. La capacité mémoire de l'automate est une donnée constructrice et dépend principalement de la gamme dans laquelle on se place. La capacité mémoire peut souvent être augmentée par rapport à la version de base [8].

2.9.4 Alimentation

Elle doit couvrir les besoins énergétiques de l'unité centrale et de toutes les extensions. Quand elle existe sur l'automate de base elle ne couvre pas les besoins d'un nombre important d'extension et il faudra rajouter une deuxième alimentation [8].

2.10 Domaines d'utilisation des automates programmables industriels (A.P.I)

Pour les raisons qui viennent d'être évoquées, les API s'adressent à des applications que l'on trouve la plupart des secteurs industriels. Ces machines fonctionnent dans les principaux secteurs et dans le domaine et dans le domaine de l'enseignement [8].

Parmi ses applications, on trouve :

- o Métallurgie et sidérurgie.
- o Mécanique et automobile.
- o Industriels chimiques.
- o Industries pétrolières.
- o Industries analogiques et alimentaires.
- o Transports et manutention.

2.11 conclusion

L'objectif de notre travail est de remplacer l'automate S5-100U par un automate S7-300 pour le pilotage du descenseur de la ligne 3 margarinerie CEVITAL. Le logiciel de programmation du S7 300 est STEP 7. On a préconisé deux solutions : soit 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, ou une deuxième solution qui consiste a concevoir le GRAFCET du processus puis d'établir les équations de ce dernier et enfin de les coder avec le STEP 7.

Présentation du SIMATIC manager

3.1 Introduction

Dans les mécanismes industriels automatisés, les tâches à exécuter par les différents éléments de la machine et le déroulement synchronisée de celle-ci sont définis par une séquence d'instruction appeler programme qu'on écrit par un langage de programmation et on sauvegarde dans une mémoire pour une exécution cyclique. Dans ce chapitre, nous présentons le logiciel de programmation STEP7 d'une façon générale ces principales applications comment créer un projet sur STEP7 et la présentation du simulateur S7-PLCIM.

3.2 présentation du STEP7

STEP 7 est le logiciel de base pour la programmation et la configuration dans SIMATIC. Il est formé d'un ensemble d'applications avec lesquelles on peut aisément réaliser des tâches partielles comme :

- La configuration et le paramétrage du matériel ;
- La création et le test de programmes utilisateur ;
- La configuration de réseaux et de liaisons.
- le convertisseur des fichiers S5.

S'y ajoute une vaste gamme de logiciels optionnels, dont entre autres ceux des langages de programmation SCL, GRAPH ou Hi Graph. Le gestionnaire de projets SIMATIC, encore appelé SIMATIC Manager, sert d'interface graphique à toutes ces applications. C'est lui qui organise la mise en commun dans un projet de toutes les données et de tous les paramètres requis pour réaliser une tâche d'automatisation. Les données y sont structurées thématiquement et représentées sous forme d'objets.

3.3 Création de projet avec STEP7

Un projet STEP 7 est composé de dossiers et d'objets. On pourrait le comparer à une structure arborescente comportant des dossiers et des fichiers, comme celle connue dans l'Explorateur Windows. Un dossier désigne un objet qui lui-même peut comporter d'autres dossiers et objets, comme par exemple un programme S7 contenant les dossiers Blocs et Sources ainsi que l'objet "Mnémoniques".

3.4 Programmation orientée objet

Le gestionnaire de projets SIMATIC associe directement les divers types d'objets avec l'application correspondante requise pour les éditer. Il n'est donc pas nécessaire qu'on sache quelle application qu'on doit démarrer pour éditer un objet donné. Il suffit simplement de savoir comment Procéder.

3.5 Mise en route du STEP7

Afin de faciliter la première utilisation du gestionnaire de projets SIMATIC, nous allons à présent commenter un exemple de marche à suivre. Il s'agit d'une description fondée sur la programmation pour SIMATIC S7.

On procède de la manière suivante :

1. Création d'un nouveau projet à l'aide de l'assistant, qu'on peut appeler avec la commande Fichier > Assistant Nouveau projet ;
2. Un double-clic sur le programme utilisateur (objet "Blocs") qui se trouve dans le dossier "Programme S7" ;
3. Un double-clic sur l'OB1, à présent affiché dans la partie droite de la fenêtre du projet. L'on peut éditer ce bloc ou en créer d'autres dans l'application qui démarre ;
4. Dans les propriétés de l'objet du programme S7, on indique l'adresse MPI du module dans lequel on peut tester les blocs créés. On sélectionne le programme S7 dans l'affichage en ligne, puis on choisit la commande Edition > Propriétés de l'objet ;
5. Dans l'affichage hors ligne, on sélectionne le programme utilisateur (dossier "Blocs") ou les blocs à charger ;
6. On Charge les objets sélectionnés dans le système cible, en choisissant la commande Système cible > Charger.

Pour tester le programme utilisateur chargé, on peut utiliser la visualisation et le forçage.

3.6 GRAFCET

GRAFCET (Graf de commande Etapes -transitions) est un outil graphique de description du cahier des charges d'un automatisme .accessible aussi bien a l'utilisateur qu'a l'automaticien, il facilite la communication et le dialogue entre les personnes concernées par l'automatisme[9]. Le GRAFCET est une représentation graphique qui décrit le comportement attendu de la partie commande d'un système automatisé.

Cette représentation graphique concise et facile à lire et aisément compréhensible par toute personne en relation avec le système automatisé, du concepteur à l'utilisateur.

De plus à chaque stade de son élaboration, se diagramme fonctionnel peut être affiné, corrigé ou modifié sans nécessiter la remise en cause des parties déjà étudiées.

Le modèle GRAFCET du système de commande doit s'intéresser à une description de la structure statique et à une description de son comportement dynamique. La description statique du GRAFCET est réalisée à l'aide d'éléments graphiques de base(les étapes, les transitions, les liaisons orientées) et d'une interprétation caractérisée par des actions associées aux étapes et des conditions de franchissement (réceptivités) associées aux transitions. Son comportement dynamique est défini par cinq règles d'évolution. [9].

3.6.1 Les éléments de bases du GRAFCET

Le GRAFCET utilise une succession alternée d'étapes, auxquelles sont associées des actions, des transitions avec leurs réceptivités et des liaisons orientées [9].

– **Etape :**

Une étape correspond à une situation dans laquelle le comportement de la partie commande est invariant vis-à-vis de ses sorties. Une étape est soit active, soit inactive. L'activité d'une étape est matérialisée par un jeton. Par convection, on associe à chaque étape une variable binaire X_i , l'indice " i " indique le numéro de l'étape [9] ;

– **Transition :**

Une transition correspond à une condition de passage d'une étape à une autre. A chaque transition est associée une condition logique appelée réceptivité, qui est une fonction booléenne écrite, de façon symbolique ou littérale, à droite du symbole de la transition [9] ;

– **Liaison orientée :**

Une liaison est un arc orienté reliant les étapes aux transitions et les transitions aux étapes [9].

3.6.2 les règles d'évolution du GRAFCET

Il existe cinq règles d'évolution dans le GRAFCET :

Règle 1 : L'étape initiale est représentée par un carrée double, l'initialisation précise les étapes actives au début de fonctionnement. Elles sont activées inconditionnellement [9].

Règle 2 : une transition est soit validée soit non validée. Elle est validée lorsque toutes les étapes immédiatement précédentes sont activées. Elle ne peut être franchie que lorsqu'elle est validée, et que la réceptivité associée à la transition est vraie. Une transition validée est obligatoirement franchie [9].

Règle 3 : le franchissement d'une transition provoque simultanément l'activation de toutes les étapes immédiatement suivantes et la désactivation de toutes les étapes immédiatement précédentes [9].

Règle 4 : plusieurs transitions simultanément franchissables, sont simultanément franchies.

Règle 5 : si au cours du fonctionnement, une même étape doit être désactivée et activée simultanément, elle reste activée. L'activation doit être prioritaire sur la désactivation au niveau d'une même étape [9].

3.6.3 Cahier de charge du descenseur

Pour pouvoir établir le GRAFCET du descenseur, on doit d'abord définir le cahier de charge du fonctionnement. Le descenseur possède 4 fins de course (A0, A1, B0, B1) et 2 capteurs de position (C0, C1) pour le vérin. Le descenseur possède aussi un dateur pour mentionner la date deux photocellules, l'une pour le comptage des cartons et l'autre pour vérifier que le palettiseur est libre. Dans le processus il y'a 4 moteurs M1 (moteur du grand tapis), M2 (moteur du petit tapis), M3 (moteur du chariot élévateur), M4 (moteur a deux sens de rotation des rouleaux) et un vérin pour empêcher les cartons de tomber. Le descenseur sert à ramener cinq boîtes de la margarine de la zone d'encartonnage à la zone de palettisation.

On se basant sur le cahier de charge et le fonctionnement du descenseur on peut déduire le GRAFCET de niveau 1 et 2 adapté pour le descenseur. Et déduire le GRAFCET des défauts et celui du mode de fonctionnement à partir du programme convertis.

3.6.4 GRAFCET du mode de fonctionnement

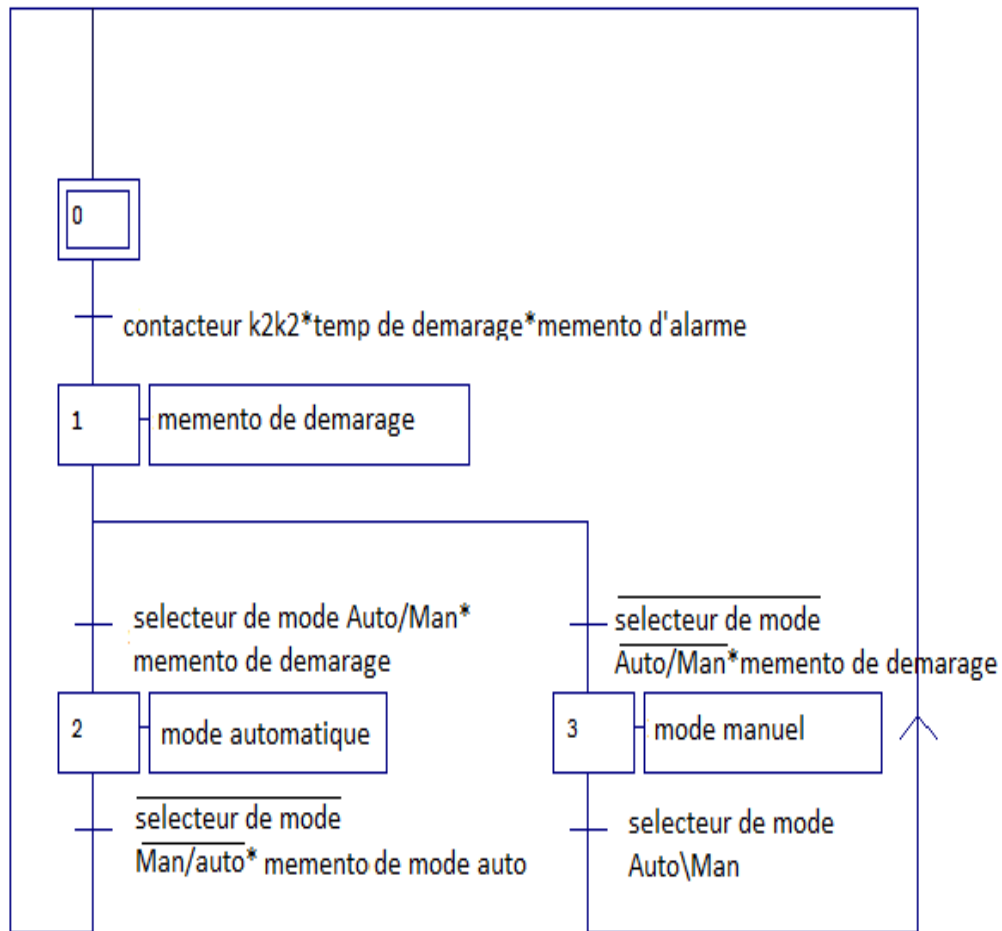


FIGURE 3.1 – Grafcet niveau 1 du mode de fonctionnement

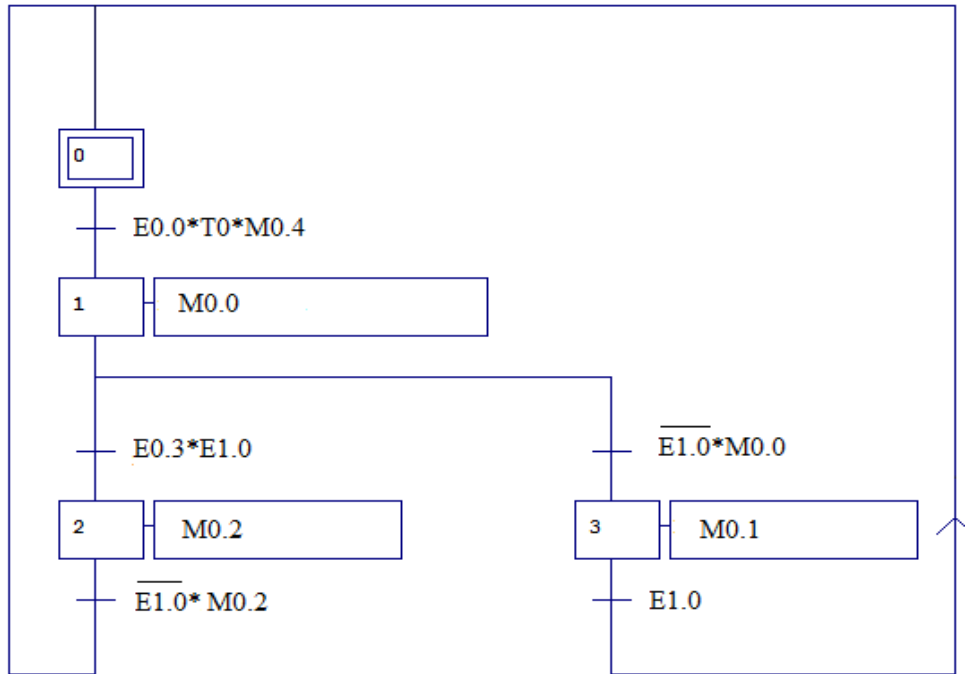


FIGURE 3.2 – Grafcet niveau 2 du mode de fonctionnement

Explication du GRAFCET du mode de fonctionnement

On appui sur le contacteur K2K2, lorsqu'on élimine tout le défaut (ALARM M0.4) en active le memento de démarrage après T0.

On met le sélecteur de mode Man/Auto sur la position " 0 ", et lorsque le memento de démarrage et active, on et sur le mode manuel.

On appui sur le bouton poussoir E0.3 (START), en met le sélecteur de mode Man/Auto à 1, on active le memento de démarrage M0.0, le memento pour le mode Auto s'active (excitation de la bobine M0.2 maintient le mode Auto en fonctionnement).

3.6.5 GRAFCET du système

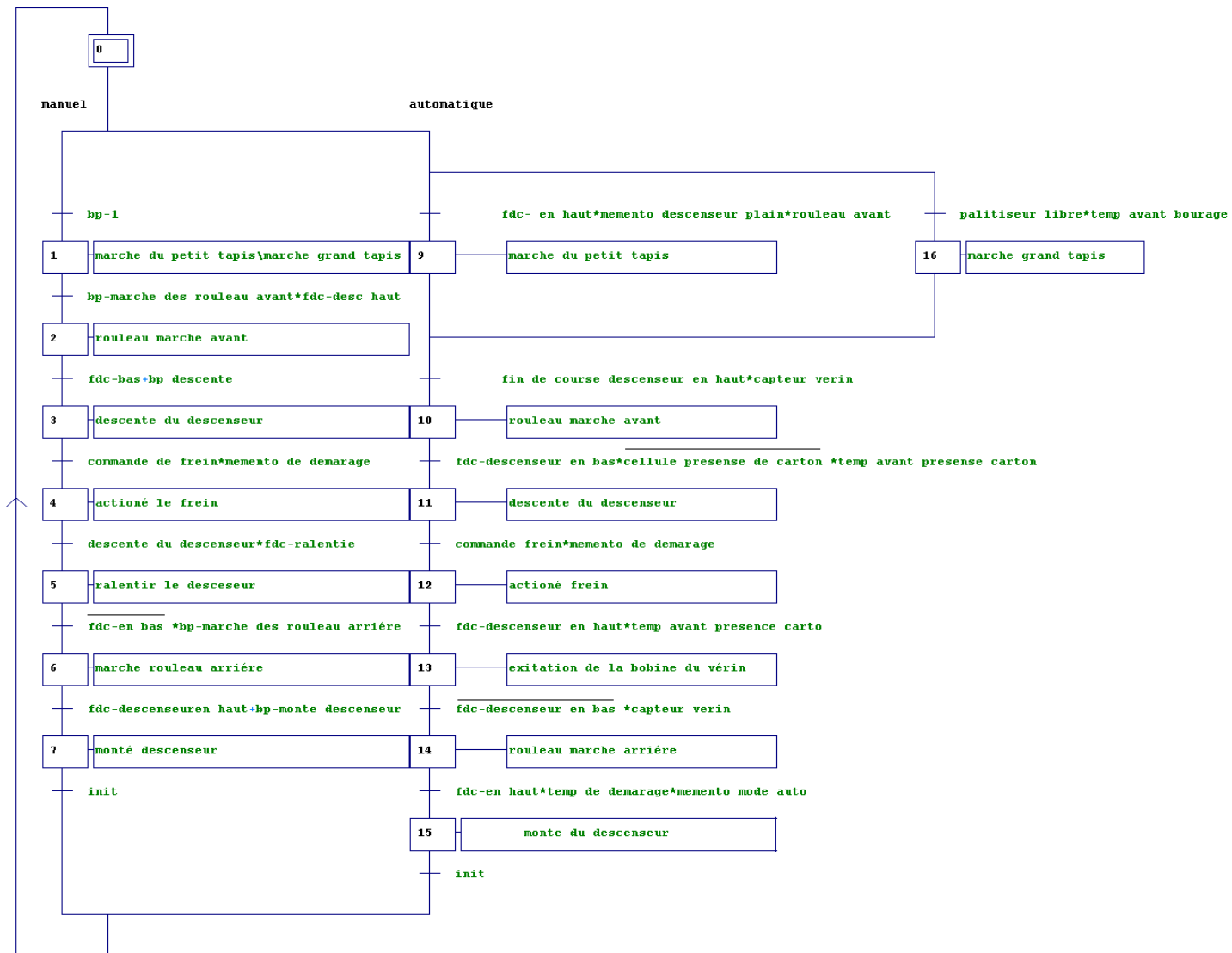


FIGURE 3.3 – Grafcet niveau 1 du fonctionnement général

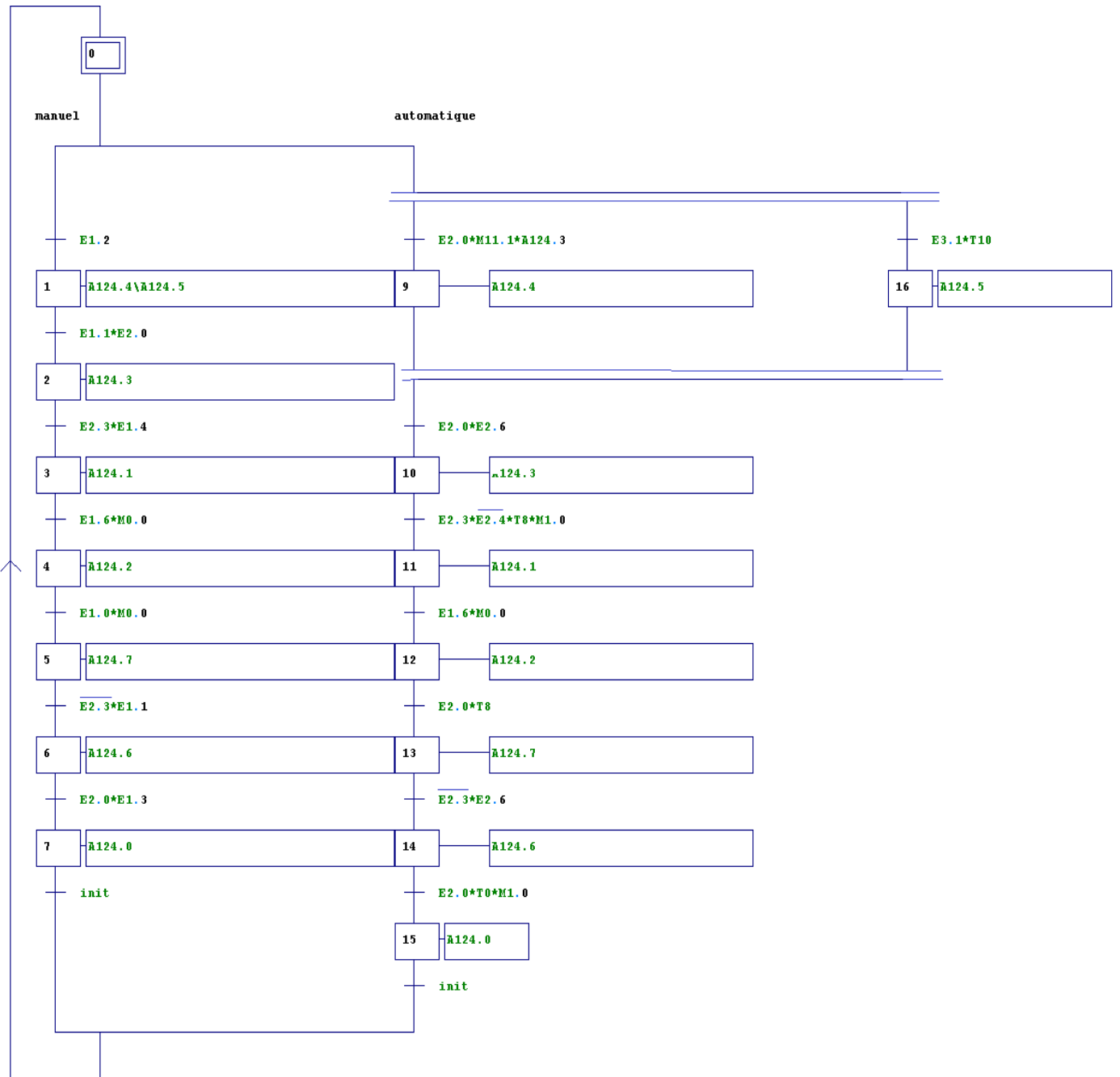


FIGURE 3.4 – Grafcet niveau 2 du fonctionnement général

3.6.6 GRAFCET de défaut

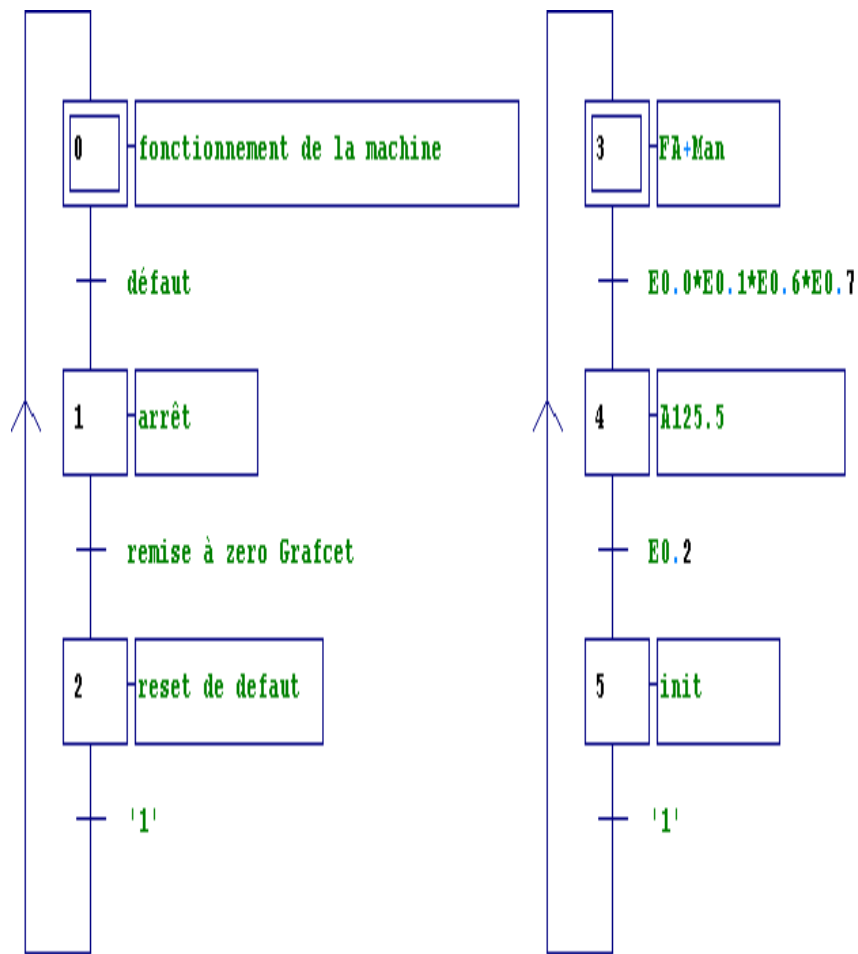


FIGURE 3.5 – Grafcet de défaut

3.7 configuration des matériels sur HW Config

3.7.1 Introduction à la configuration matérielle

On réalise la configuration matérielle dans HW Config. Pour ouvrir HW Config, on sélectionne une station dans SIMATIC Manager, puis on effectue un double clic sur l'icône "Matériel".

3.7.1.1 Configuration

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

STEP 7 affecte 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'adressage libre.

on peut copier la configuration aussi souvent que nous désirons dans d'autres projets STEP 7, 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 STEP7 à la configuration sur site de l'installation. Aussi, les erreurs éventuelles sont elles immédiatement détectées et signalées.

3.7.1.2 Paramétrage

Par "paramétrage", on entend dans ce qui suit :

- " le réglage des paramètres des modules paramétrables pour la configuration centralisée et pour un réseau. Exemple : une 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 maîtres et d'esclaves pour un réseau maître (PROFIBUS) ou d'autres définitions pour l'échange de données entre des composants.

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 STEP7 sont automatiquement chargés dans le nouveau module à la mise en route.

Quand la "Configuration matérielle" est-elle requise ?

Les paramètres des automates programmables S7 et des modules ont été prédéfinis de sorte à ne pas nécessiter de configuration dans bien des cas.

La configuration est obligatoire :

- lorsque on souhaite modifier les paramètres prédéfinis d'un module (par exemple, valider l'alarme de processus pour un module) ;
- lorsque on souhaite configurer des liaisons de communication ;
- pour les stations avec une périphérie décentralisée (PROFIBUS DP ou PROFINET IO) ;
- pour les stations S7-400 avec plusieurs CPU (multiprocesseur) ou châssis d'extension ;
- pour les automates à hautes disponibilité.

3.7.2 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 les profils 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 profils 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.

3.7.3 Marche à suivre pour la configuration d'une station

Quelle que soit la technique de configuration d'une station, il faut toujours nous en tenir aux étapes suivantes pour la configurer :

1. On sélectionne un composant matériel dans la fenêtre "Catalogue du matériel" ;
2. On amène le composant sélectionné dans la fenêtre de station en utilisant la fonction glisser lâcher.

La figure suivante illustre les manipulations de base :

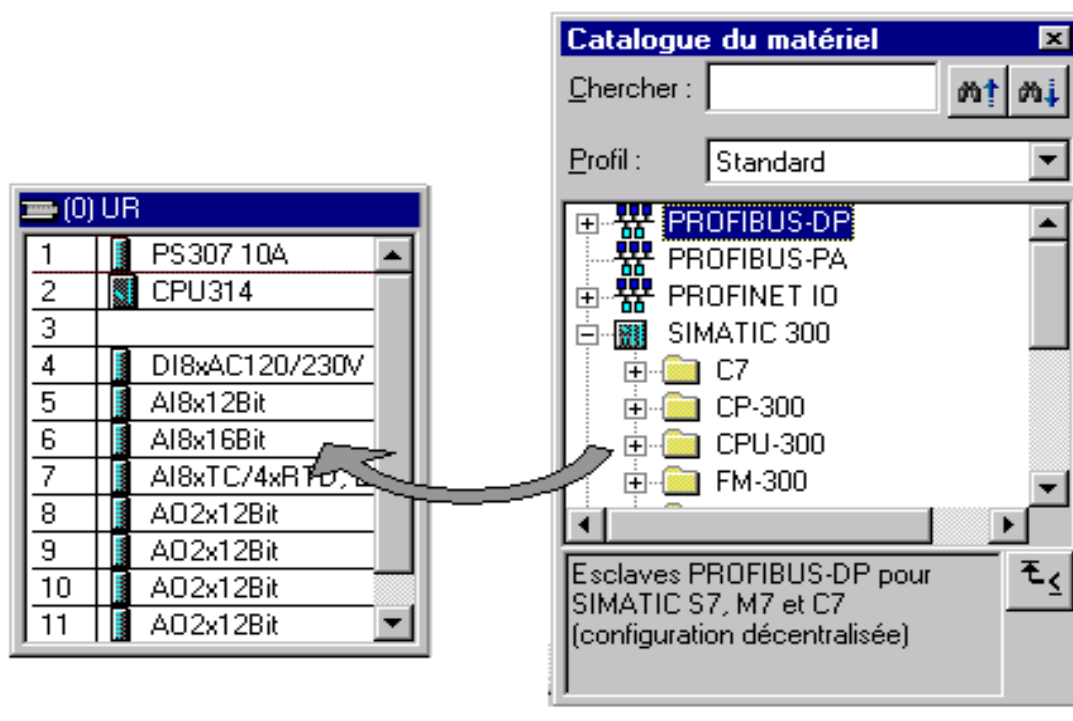
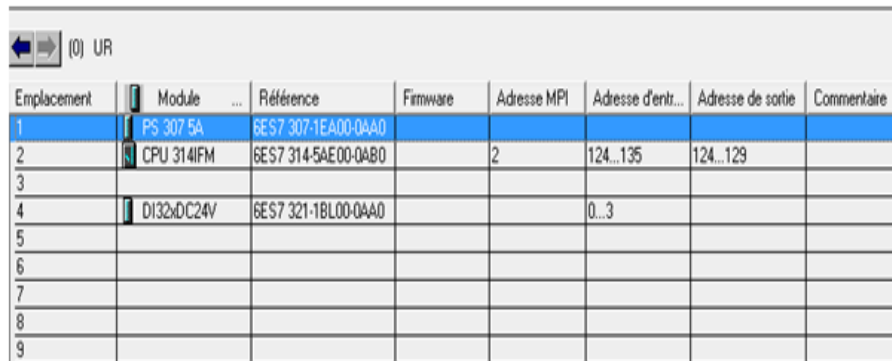


FIGURE 3.6 – Configuration des parties de l'automate dans HWconfig.

3.7.4 Organisation de la fenêtre de station

La partie inférieure de la fenêtre de station donne une vue détaillée du profilé support ou châssis sélectionné ou inséré. Les numéros de référence et les adresses des modules y sont énumérés sous forme de tableau.

Pour un profilé support ou châssis de base équipé de modules, ce tableau se présente comme suit (vue détaillée) :



Emplacement	Module	Référence	Firmware	Adresse MPI	Adresse d'entr...	Adresse de sortie	Commentaire
1	PS 307 5A	6ES7 307-1EA00-0AA0					
2	CPU 314FM	6ES7 314-5AE00-0AB0		2	124...135	124...129	
3							
4	DI32xDC24V	6ES7 321-1BL00-0AA0			0...3		
5							
6							
7							
8							
9							

FIGURE 3.7 – Représentation des modules dans la vue détaillée

3.7.5 Table de configuration comme reproduction d'un profilé support ou châssis

En configuration centralisée, nous montons les modules à côté de la CPU sur un profilé support ou un châssis, puis sur d'autres profilés support ou châssis. Le nombre des profilés support ou châssis autorisés dépend de la CPU utilisée.

Tout comme dans notre installation réelle, nous montons les modules dans des profilés support ou des châssis avec STEP 7. En fait, dans STEP 7, les profilés support ou châssis sont représentés par des "tables de configuration" dont le nombre de lignes correspond au nombre de modules enfichables sur le profilé support ou châssis réel.

La figure suivante montre, à l'aide d'un exemple, comment transposer une configuration réelle dans une table de configuration. La table de configuration correspond au profilé support ou châssis utilisé; STEP 7 fait automatiquement précéder, entre parenthèses, le nom du profilé support ou châssis par son numéro.

Exemple : (0) UR correspond au châssis de base (Universal Rack) numéro 0.

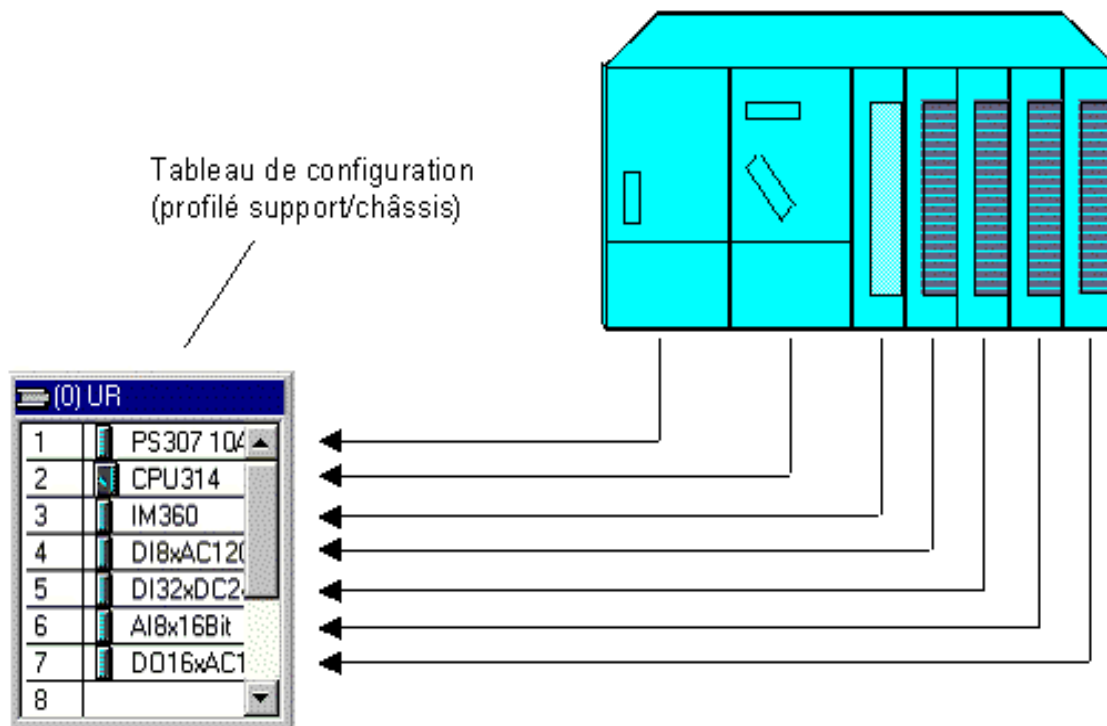


FIGURE 3.8 – tableau de configuration (profilé support/châssis)

3.7.6 Définition des propriétés des composants

Lorsque nous allons disposer des composants dans la fenêtre de station, l'on peut modifier les propriétés prédéfinies (paramètres ou adresses) dans une boîte de dialogue que nous appelons toujours de la manière suivante :

- On Effectue un double clic sur le composant ou on choisit la commande **Edition > Propriétés de l'objet.** ;
- Avec le bouton droit de la souris : on déplace le curseur sur le composant, on clique sur le bouton droit de la souris et on choisit la commande Propriétés de l'objet dans le menu contextuel.

Propriétés des unités centrales Les propriétés des CPU sont de première importance pour le comportement du système. Dans les pages d'onglet d'une CPU, l'on peut, par exemple, définir : le comportement à la mise en route, les zones de données locales et les priorités des alarmes, les zones de mémoire, le comportement pour la rémanence, les mémentos d'horloge, le niveau de protection et le mot de passe - pour ne citer que quelques propriétés. STEP 7 "sait" ce qu'on peut définir et dans quelles plages de valeurs.

l'on peut paramétrer les interfaces (par exemple, l'interface MPI ou l'interface PROFIBUS DP intégrée) dans la page d'onglet "Général" de la CPU ou dans les propriétés de l'interface de la CPU. Ces boîtes de dialogue vont nous permettre également d'accéder à celles des propriétés du sous-réseau correspondant auquel la CPU doit être connectée.

Autres méthodes de paramétrage :

Pour les automates programmables S7300/400, il est possible de définir les paramètres de certains modules dans le programme utilisateur (par exemple pour les modules analogiques). Pour cela, on appelle les fonctions système (SFC) WR-PARM, WR-DPARM et PARM-MOD dans le programme utilisateur. Ces paramètres seront toutefois perdus à la mise en route (démarrage à chaud).

Pour les automates programmables S7300/400, il est possible de définir les paramètres des modules de signaux dans le programme C. Pour cela, vous appelez dans le programme C la fonction M7 API "M7StoreRecord". Cette fonction transmet les paramètres à un module de signaux. nous trouverons des informations détaillées sur les fonctions M7 API dans les manuels relatifs au logiciel système pour M7-300/400.

3.7.7 Ouverture d'objets dans HW Config

Dans STEP 7, l'on ouvre des objets afin de les éditer [11].

Dans SIMATIC Manager, p. ex., on sélectionne l'objet "Matériel" et choisissez la commande de menu Edition > Ouvrir l'objet afin de démarrer HW Config. Avec HW Config, l'on édite la configuration de la station.

Un double clic sur l'objet a ici la même signification que "Editer l'objet" [11]. **Particularités dans HW Config** Lorsque on édite un objet dans HW Config, un double clic a le même effet que la commande de menu **Edition > Propriétés de l'objet**. En règle générale, l'on attribue des adresses, définissez des paramètres ou saisissez d'autres informations. Si l'on doit démarrer une application pour éditer un objet, un bouton est proposé dans la boîte de dialogue des propriétés. A partir de la version V5.4 de STEP 7, nous disposons de la commande de menu Edition > Ouvrir l'objet avec. Cette commande de menu nous permettons de démarrer une application pour éditer l'objet, si ce dernier le requiert. Cette commande de menu peut uniquement être activée si une application d'édition est requise et disponible [11].

3.7.8 Informations sur les règles d'enfichage et autres règles

STEP 7 nous assiste lors de la configuration d'une station, si bien qu'en général, nous obtenons immédiatement un message en retour si, par exemple, un module ne peut pas être enfiché à l'emplacement souhaité [11].

A partir de STEP 7 V5.2, l'on peut reconnaître d'emblée les emplacements possibles des modules, à condition que l'écran soit paramétré pour afficher plus de 256 couleurs. Lorsque nous sélectionnons un module dans le catalogue du matériel et que ce module est enfichable dans le châssis configuré, les emplacements possibles pour ce module sont mis en évidence par une couleur. La configuration devient alors plus simple et plus rapide [11].

De plus, les plages d'adresses sont automatiquement vérifiées, si bien qu'une affectation double d'adresses est impossible.

Tenez compte des informations sur les causes et conséquences d'une manipulation qui s'affichent

dans la barre d'état au bord inférieur de la fenêtre ainsi que dans des boîtes de message. nous avons en outre la possibilité d'obtenir des informations plus détaillées grâce à l'aide sur les messages.

Il n'est pas tenu compte des règles temporaires supplémentaires (valables pour une version donnée), comme par exemple des restrictions relatives aux emplacements d'enfichage disponibles qui dépendent d'une restriction fonctionnelle pour des modules individuels. l'on doit donc toujours tenir compte de la documentation ou de l'information produit actuelle relatives aux modules [11].

3.7.9 Configuration et paramétrage d'une installation centralisée

nous avons ouvert ou créé un projet dans SIMATIC Manager.

Marche à suivre pour configurer et paramétrer une installation, l'on doit procéder aux étapes suivantes :

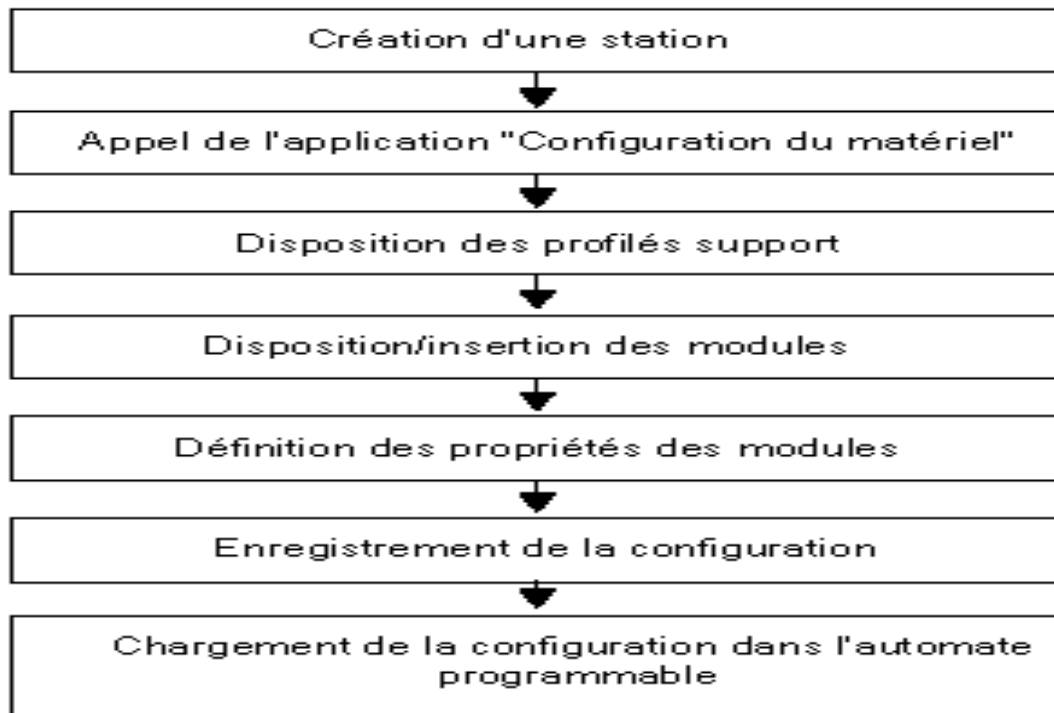


FIGURE 3.9 – les étapes essentielles pour configurer et paramétrer une installation

3.8 Présentation du simulateur S7-PLCSIM

S7-PLCSIM est une application qui nous permet d'exécuter et de tester notre programme dans un automate programmable. La simulation étant complètement réalisée au sein du logiciel STEP7.

S7-PLCSIM dispose d'une interface simple nous permettant de visualiser et de forcer les différents paramètres utilisés par le programme, par exemple, activer ou désactiver des entrées.

Tout en exécutant notre programme dans la CPU simulée, nous avons également la possibilité de mettre en œuvre les diverses applications du logiciel STEP7, comme par exemple la table des variables (VAT) afin d'y visualiser et d'y forcer des variables [10,12].

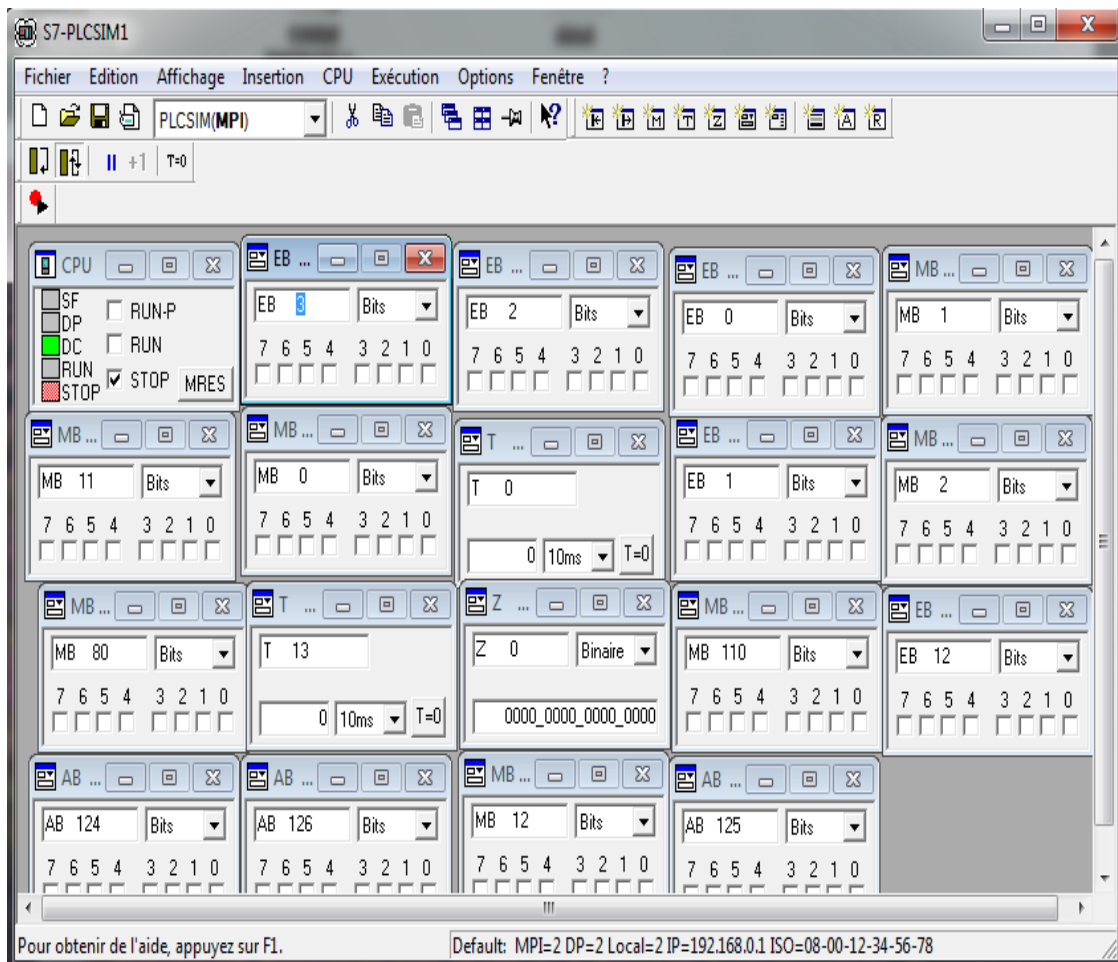


FIGURE 3.10 – simulateur S7-PLCSIM.

3.8.1 Mise en route

Pour l'utilisation du simulateur S7-PLCSIM, on suit les procédures suivantes pour sa mise en route. Le monde de simulation est disponible à partir du gestionnaire de projets SIMATIC à

condition qu'aucune liaison à des A.P.I réels ne soit établie. Procédons comme suit pour utiliser S7-PLCSIM [10,12].

1. Ouvrir le gestionnaire de projets SIMATIC ;
2. Sélectionner la commande outils> simulation de modules Cela lance l'application S7-PLCSIM et ouvre une fenêtre CPU (ayant l'adresse MPI par défaut, c'est-à-dire 2) ;
3. dans le gestionnaire de projet SIMATIC, on cherche le projet à simuler ;
4. dans le projet, on cherche le dossier blocs (on reporte à l'aide en ligne du gestionnaire de projets SIMATIC pour une présentation des objets de STEP7) ;
5. dans le gestionnaire de projets SIMATIC, choisir la commande système cible>charger pour charger le dossier blocs dans l'automate ;
6. lorsque le message " -vous charger les données système ? " s'affiche, on choisit Non si on ne veut pas charger de configuration matérielle dans l'AP de simulation ou Oui si on désire en charger une ;
7. dans l'application S7-PLCSIM, on crée de nouvelles fenêtres pour visualiser les informations provenant de l'automate programmable de simulation :
 - On sélectionne la commande Insertion >Entrée. La fenêtre affiche EB0 (octet d'entrée 0) ;
 - On sélectionne la commande Insertion >Sortie pour afficher une seconde fenêtre, AB0 (octet de sortie 0) ;
 - On sélectionne la commande Insertion >temporisation pour afficher trois fenêtres de temporisation. On tape 2, 3 et 4 (pour les temporisations T2, T3 et T4) dans les fenêtres respectives, en appuyant sur la touche Entrée après chaque valeur.
8. Choisissons le menu CPU dans S7-PLCSIM et vérifions que la commande Mettre sous tension est activée ;
9. On choisit la commande Exécution > Mode d'exécution et vérifions que la commande cycle continu est activée ;
10. On met la CPU de simulation en marche en cliquant sur l'une des cases à coucher RUN ou RUN-P ;
11. Un clique sur bit 0 de EB0 pour simuler la mise à 1 de l'entrée 0.0 et on observe la réaction des temporisations et de AB0 ;
12. On sélectionne la commande fichier > enregistre CPU sous. . . pour enregistrer l'état actuel de l'AP de simulation dans un nouveau fichier. . .

3.8.2 Etat de fonctionnement de la CPU

- **Etat de marche (RUN-P) :**

La CPU exécute le programme tout en nous permettant de le modifier, de même que ses paramètres. Afin de pouvoir Utiliser les applications de STEP7 pour forcer un paramètre

quelconque du programme durant son exécution, nous devons mettre la CPU à l'état RUN-P [10,12].

– **Etat de marche (RUN) :**

La CPU exécute le programme en lisant les entrées, exécutant le programme, puis en actualisant les sorties. Lorsque la CPU se trouve à l'état de marche (RUN), on ne peut ni charger aucun programme, ni utiliser les applications de STEP7 pour forcer un paramètre quelconque (comme les valeurs d'entrées) [10,12].

– **Etat d'arrêt (stop) :**

La CPU n'exécute pas le programme. Contrairement à l'état d'arrêt (STOP) des CPU réelles, les sorties ne prennent pas de valeurs (de sécurité) prédéfinies, mais conservent l'état auquel elles étaient lorsque la CPU est passée à l'état d'arrêt (STOP). Nous pouvons charger des programmes dans la CPU lorsqu'elle est à l'arrêt [10,12].

3.8.3 Visualisation de la simulation du programme

On a également la possibilité d'utiliser les applications de STEP7 pour visualiser la simulation d'un programme [10] :

1. Après avoir créé les fenêtres (étape 7 ci dessus), on active le gestionnaire de projets SIMATIC ;
2. Choisir le mode en ligne (affichage>en ligne) ;
3. Parcourir la boîte de recherche jusqu'au dossier blocs dans le projet et ouvrir OB1, se qui entraîne alors l'ouverture de l'application " CONT/LIST/LOG " ;
4. Lorsque la CPU de simulation se trouve à l'état de marche qu'on activé le bit 0 de EB0, activer la fenêtre d'application " CONT/LIST/LOG " et choisir la commande test >visualisation pour observer les effets de ces modifications sur le programme[12].

3.9 conclusion

Dans ce chapitre nous avons élaboré un programme avec le logiciel d'automatisation SIMATIC S7.

La fonction " traitement de données " est assurée par un automate programmable modulaire S7-300.

Le programme de l'automate est écrit dans le langage CON (contact) du logiciel STEP7, il est établi a partir des équations logiques qu'on a déterminé a partir du GRAFCET de la machine. Pour mieux comprendre le programme, on introduit une table de mnémoniques qui

remplace les adresses d'entrées et sorties.

Le simulateur S7-PLCSIM nous permis de simuler le programme sans CPU.

la migration de S5 vers S7

4.1 Introduction

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 LIST7. Les programmes sont toujours convertis entièrement et nécessitent des corrections des erreurs générées par la conversion.

4.2 Les différentes approches de résolution du problème

Nous avons deux choix pour résoudre ce problème, le premier consiste à faire de la conversion directe avec le convertisseur S5/S7, et la deuxième c'est de reprogrammer le descenseur en se basant sur le cahier des charges.

4.2.1 Conversion de S5 vers S7

Pour une grande part, la programmation S7 en LIST, CONT et LOG est compatible avec - respectivement - LIST, CONT et LOG de S5. Par conséquent, si nous sommes des utilisateurs de S5 et que nous souhaitons mettre en œuvre dans S7 des programmes existants, la conversion sera très facile. Nous pouvons rester fidèles à nos programmes S5 éprouvés; il suffit de les convertir en programmes S7 [13].

4.2.1.1 La mise en route

La procédure complète de conversion comprend 4 étapes :[14]

1. on prépare les fichiers S5 à convertir ;

Pour préparer la conversion, il faut copier les fichiers suivants dans un répertoire DOS :

- <Nom>ST.S5D Fichier programme (le programme S5 à convertir) ;
 - <Nom>XR.INI Liste des références croisées (contient la structure du programme) ;
Si nous désirons convertir également le fichier des mnémoniques, nous avons besoin, en outre, du fichier :
 - <Nom>Z0.SEQ Liste d’assignation.
Tous les fichiers générés lors de la conversion sont archivés dans ce répertoire.
2. nous effectuons la conversion à l’aide de l’application ”Convertisseur S5/S7” ;
Durant cette conversion :
 - Les instructions S5 sont remplacées par des instructions S7 équivalentes ou par des séquences d’instructions (macro-instructions) ;
 - Les appels de blocs fonctionnels standards S5 sont remplacés par les fonctions S7 équivalentes ;
 - Les commentaires STEP 5 sont transposés dans le programme S7 ;
 - La liste d’assignation STEP 5 est modifiée en fonction de la conversion des blocs.
 3. nous retouchons le programme converti. A la fin de la conversion, une fenêtre de messages séparée affiche la localisation des erreurs de conversion dans le programme. l’on doit alors prendre les mesures nécessaires pour remédier à ces erreurs ;
 4. on compile le fichier S7 terminé dans l’éditeur de programme ”Program Editor” (LIST).

4.2.1.2 Exécution du programme

Pour convertir un programme S5, On procède comme suit :[14]

1. on doit assurer que les fichiers de programme S5 ont bien été préparés ;
2. On choisit un fichier programme S5 en sélectionnant la commande Ouvrir du menu Fichier ;
3. Dans la boîte de dialogue ”Convertisseur-[...]” sont affichés tous les fichiers et les blocs concernés par la conversion. En cliquant sur le nom d’un fichier ou d’un bloc, on a la possibilité de modifier la désignation standard des nouveaux fichiers ainsi que le numéro des nouveaux blocs ;
4. On Valide la nouvelle désignation en cliquant sur le bouton ”OK” ;
5. Pendant la conversion, une fenêtre d’état affiche l’état actuel de la conversion. En cliquant sur le bouton ”Annuler”, l’on peut quitter la conversion avant qu’elle soit achevée ;
6. Pour confirmer qu’on veut quitter la conversion, On clique sur le bouton ”OK” de la boîte de dialogue qui apparaît alors ;
7. A la fin de la conversion, les erreurs et les avertissements constatés au cours de la conversion sont affichés dans une fenêtre séparée (fenêtre de messages). On Prend maintenant les mesures nécessaires pour remédier à ces erreurs ;

8. Répétez les étapes 1 à 6. Au cours de la deuxième phase de conversion, l'on doit confirmer qu'on désire substituer les nouveaux fichiers aux fichiers existants <Nom>AC.AWL (fichier source LIST), <Nom>AF.SEQ (fichier d'erreurs) et <Nom>S7.SEQ (liste d'assignation convertie). En cliquant sur "OK", les fichiers de la première phase de conversion seront effacés;
9. Lorsque la conversion est terminée, l'on doit intégrer le fichier LIST à un projet S7, en sélectionnant la commande Source externe du menu Insertion dans l'application "SIMATIC Manager", avant de pouvoir le retoucher et le compiler[13].

4.2.1.3 Localisation des erreurs

A la fin de la conversion apparaît une fenêtre affichant, dans 2 zones de liste, les erreurs et les avertissements qui ont été détectés au cours de la conversion (fenêtre de messages)[13]. Dans la zone de liste inférieure, On peut visualiser la localisation de l'erreur dans le fichier correspondant[13].

Pour localiser une erreur dans le programme :

- cliquez deux fois sur le message d'erreur dans la zone de liste supérieure (fichier d'erreurs). La localisation de l'erreur dans le programme est alors visualisée dans la zone de liste inférieure. Dans le cas où le message d'erreur signale l'inexistence d'un bloc, c'est la ligne suivante qui est visualisée[13].

4.2.1.4 Traitement des messages d'erreurs

Zones de liste de la fenêtre de messages[13].

A la fin de la conversion apparaît une fenêtre affichant, dans 2 zones de liste, les erreurs et les avertissements qui ont été détectés au cours de la conversion (fenêtre de messages).

Zones de liste de la fenêtre de messages

La zone de liste supérieure affiche le fichier d'erreurs <Nom>AF.SEQ. Elle contient les messages suivants :

- Issus de la première phase de conversion dans le fichier <Nom>A0.SEQ (fichier source ASCII du programme S5) ;
- Issus de la deuxième phase de conversion dans le fichier <Nom>AC.AWL (fichier source LIST) ;
- Issus de la transposition de la liste d'assignation dans le fichier <Nom>S7.SEQ (liste d'assignation convertie). Ces fichiers peuvent contenir les erreurs indiquées dans le fichier d'erreurs. Dans le fichier source LIST, les erreurs qui se sont produites sont indiquées sous forme de commentaire de ligne, à l'endroit correspondant dans le programme.

Dans la zone de liste inférieure, l'on peut visualiser la localisation de l'erreur dans le fichier correspondant[13].

4.2.1.5 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[13].

4.2.2 les phases 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 la 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[13]. Au cours de la deuxième phase, le fichier

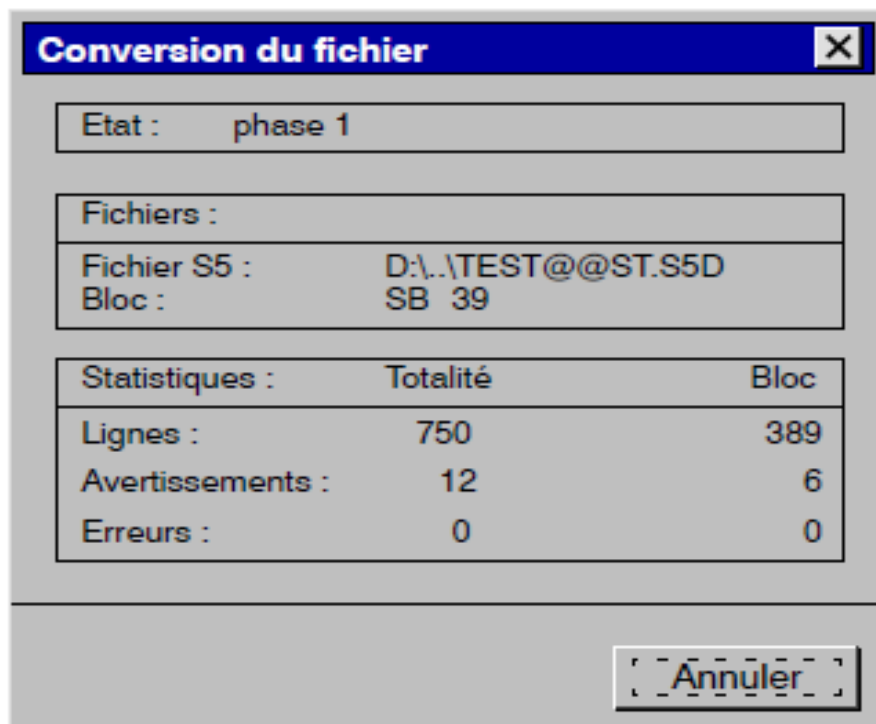


FIGURE 4.1 – phase 1 de la conversion

source S5 est converti en un fichier source LIST avec les nouveaux numéros de bloc et la syntaxe de S7.

La conversion une fois terminée, une boîte de dialogue indiquant le nombre d’erreurs et d’avertissements s’affiche[13].

4.2.2.1 Messages d'erreur

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[14].

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

remarque : Dans S5, les blocs d'organisation n'ont pas les mêmes fonctions que dans S7. Lors de la retouche du programme converti, nous devons remplacer les OB qui ne sont pas convertis automatiquement par :

- des blocs d'organisation avec d'autres fonctions ;
- de nouvelles opérations S7 ;
- les paramètres du système que nous définissons lors du paramétrage du matériel.

Les messages d'erreur qu'on peut rencontrer sont les suivants [14] :

4.2.2.2 Correction des erreurs

A la fin de la conversion apparaît une fenêtre affichant, dans 2 zones de liste, les erreurs et les avertissements qui ont été détectés au cours de la conversion (fenêtre de messages)[13].

Pour corriger une erreur [14] :

1. On imprime le fichier d'erreurs ainsi que le fichier source LIST à l'aide de la commande Imprimer du menu Fichier ;
2. On interprète les messages d'erreur ;
3. Le cas échéant, on définit des macro-instructions pour les instructions S5 non convertibles ;
4. Exécutez une nouvelle conversion ;
5. on Corrige le fichier source LIST dans l'éditeur de programme "Program Editor" (LIST).

4.2.2.3 Avertissements

En peut rencontrer les avertissements suivants :

Message d'erreur	Origine	Signification	Remède
Paramètre absolu incompatible avec l'ID d'opérande	Phase 1	L'identificateur d'opérande n'est pas correct	Contrôlez l'instruction.
Bloc inexistant	Phase 1	Le bloc appelé (FB, FX) manque ou il figure dans la liste des blocs mais n'existe pas dans le fichier programme.	Contrôler la structure du programme
	Phase 2	Un bloc est appelé qui n'existe pas dans le fichier programme.	Vérifiez que la liste de références croisées a bien été indiquée lors de la conversion bien été ou revoyez la structure du programme.
Instruction interdite dans le bloc.	Phase 1	P.ex. saut dans un bloc de programme	Contrôlez l'instruction.
Instruction non définie	Phase 1	L'instruction LIST MC5 n'est pas valable.	Corrigez le fichier programme S5 défectueux.
	Phase 2	L'instruction n'existe pas dans S7.	Editez une macro-instruction ou remplacez l'instruction par conversion la séquence d'instructions S7 appropriée.
L'accès par bit à une temporisation ou à un compteur n'est plus possible (contrôlez SVP).	Phase 2	Le programme S5 contient des accès par bit à des temporisations ou à des compteurs.	Contrôlez le programme LIST.
CALL SFC xy a été généré. Veuillez compléter la liste des paramètres.	Phase 2	Des paramètres manquent pour SFC.	Complétez la liste des paramètres pour SFC.
Fichier inexistant	Globale	Le fichier sélectionné n'existe pas.	Contrôlez le fichier programme.

TABLE 4.1 – tableau des erreurs signalisées 0

Niveau de parenthèse incorrect	Phase 1	La clôture de parenthèse n'est pas équilibrée.	Respectez les niveaux de parenthèse, éliminez l'erreur de programmation.
Opérande incorrect	Phase 1	L'opérande ne convient pas à l'instruction.	Contrôlez la source S5.
	Phase 2	L'opérande ne convient	Modifiez le fichier LIST.
Erreur de conversion	Phase 2	Opération BI sans constante	Complétez l'instruction de chargement par une
			constante.
Paramètre formel non défini	Phase 1	Il y a plus de paramètres que dans le bloc appelant.	Contrôlez le fichier programme S5.
Index inexistant	Phase 1	Le fichier programme ne contient aucun bloc.	Contrôlez le fichier programme.
Longueur incorrecte du commentaire	Phase 1	Erreur dans le fichier S5	Contrôlez le fichier programme.
Commentaire trop long.	Phase 1	Erreur dans le fichier S5	Contrôlez le fichier programme.
Nom de bloc manquant	Phase 1	Le nom de bloc ne comporte que des blancs.	Entrez un nom de bloc.
Droits d'accès manquants	Globale	Le fichier est protégé en écriture.	Supprimez la protection contre l'écriture.
Repère non défini	Phase 1	Le repère de saut n'est pas défini dans l'étiquette.	Contrôlez le fichier S5.
Repère non valable	Phase 1	Le repère de saut contient des caractères non valables.	Contrôlez le fichier S5.
Opérateur non valable	Phase 1	L'opérateur dans le fichier S5 est inconnu ou impossible à convertir.	Remplacez l'opérateur par l'instruction S7 appropriée.
Opérateur non valable, à remplacer éventuellement par l'instruction "L P paramètre formel".	Phase 2	L'opérateur ne peut être chargé sous cette forme . dans S7	Utilisez éventuellement l'instruction indiquée.
Nombre de paramètres incorrect	Phase 1	Erreur dans le programme S5	Contrôlez le fichier programme.
Paramètre erroné	Phase 1	Erreur dans le programme S5	Contrôlez le fichier

TABLE 4.2 – tableau des erreurs signalisées 1

Format de paramètre incorrect	Phase 1	Erreur dans le programme S5	programme. Contrôlez le fichier programme.
Erreur d'écriture disquette	Globale	Le fichier est protégé contre l'écriture, ou il n'y a plus de place sur la disquette.	Supprimez la protection contre l'écriture ou effacez les données dont vous n'avez pas besoin.
Dépassement de la capacité de mémoire dans la PG (mémoire insuffisante)	Phase 1	La mémoire centrale est insuffisante.	Effacez de la mémoire centrale les fichiers dont vous n'avez plus
Impossible de générer repère de saut	Phase 2	L'instruction SPR dépasse la limite du bloc.	besoin. Eliminez l'erreur dans le programme S5.
Un code MC5 non valable a été converti	Phase 1	Conversion d'une ancienne instruction S5	Aucun

TABLE 4.3 – tableau des erreurs signalisées 2

Avertissement	Origine	Signification	Remède
Sortie interdite (n° du produit)	Phase 1	Un bloc fonctionnel standard S5 doit être remplacé par une FC de S7	Aucun.
Sortie interdite (bloc GRAPH 5)	Phase 1	Les blocs GRAPH 5 ne sont pas convertibles.	Utilisez éventuellement un bloc S7 GRAPH.
Contrôlez les valeurs de la base de temps SVP.	Phase 2	Dans S7, la base de temps peut être plus serrée que dans S5.	temps à l'aide de la fonction "Paramétrage de cartes".

TABLE 4.4 – tableau des avertissements signalisées 0

Tenir compte de la nouvelle numérotation des blocs	Phase 2	L'appel indirect de bloc ne tient pas compte des nouveaux numéros de bloc (le n° est prélevé dans le mot de memento ou de données approprié).	Modifiez la logique dans le programme S5 ou utilisez des appels de bloc fixes.
OB 23 et OB 24 sont transposés en OB 122	Phase 2	OB 23 et OB 24 sont remplacés tous deux par OB 122 en S7.	Regroupez le contenu des OB 23 et 24 dans un OB 122 et effacez l'autre OB 122.
L'OB a été interprété comme OB 34 de S5-100U.	Phase 2	Selon la CPU employée, l'OB 34 peut avoir des significations différentes.	Vérifiez que cet OB convient à votre programme.
Le masque DB S5 n'est pas employé pour le paramétrage de S7.	Phase 1	Il y a MASK dans DW0 et DW1.	Paramétrez l'AP avec STEP 7.
Opération de saut non convertible après l'opération de substitution	Phase 2	Une opération de substitution suivie d'une SPA opération 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 certes convertis, mais ils n'ont plus la même signification que dans S5.	Effectuez le paramétrage du système avec STEP7.
Tenir compte des opérations d'arrêts différentes	Phase 2	Il n'est pas fait de différence entre STP, STS et STW.	Contrôlez le fichier programme.
Etiquette manquante	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.	Vérifiez si les étiquettes se trouvent dans un autre fichier.
Si S5-100U, transformer en OB 100	Phase 2	L'OB 21 de mise en route de S5 est converti automatiquement en OB 101.	Si le programme S5 était exploité sur un automate S5-100U, il faut transformer l'OB 101 en OB 100.

TABLE 4.5 – tableau des avertissements signalisés

4.2.2.4 Les macro-instructions

Avec l'application "Convertisseur S5/S7", l'on dispose de 2 fichiers (S7S5CAPA.MAC pour le jeu d'instructions SIMATIC et S7S5CAPB.MAC pour le jeu d'instructions CEI) nous permettant de définir facilement la transposition d'instructions S5 à l'aide de macro-instructions. Si nous utilisons les deux jeux d'instructions, l'on doit entrer les macro-instructions dans chaque fichier séparément. Si le convertisseur trouve une instruction S5 pour laquelle il existe déjà une macro-instruction de transposition, la transposition standard est désactivée et l'instruction sera transposée selon notre propre définition.

On distingue les macro-instructions pour instruction S5 des macro-instructions pour OB (bloc d'organisation).

On peut définir au maximum 256 macro-instructions de chaque type.

Si nous avons définis plus d'une macro-instruction pour une instruction S5, seule la première définition sera prise en compte[13].

– **Structure d'une macro-instruction pour instruction S5 :**

```
MAKRO :< InstructionS5 >
< Sequenced'instructionsS7 > .
ENDMAKRO
```

– **Structure d'une macro-instruction pour OB :**

```
OBCALL :< Numrodel'OB >
CALL < FonctionS7 >;
ENDMAKRO.
```

Programmation de macro-instructions

On procède de la manière suivante :

1. On va sélectionner la commande Macro de remplacement du menu Edition. Le fichier de macro-instructions est alors appelé automatiquement ;
2. On va définir maintenant notre macro-instruction. Pour ce faire, l'on doit utiliser les commandes du menu Edition ;
3. On va enregistrer la macro-instruction en sélectionnant la commande Enregistrer du menu Fichier ;
4. On va fermer la fenêtre de macro-instructions en sélectionnant la commande Quitter du menu Fichier.

Le nom des fonctions	Le programme en STEP5	Le programme en STEP7
fonction généralité	FB1	FC0
Fonction d'alarme	FB6	FC1
La fonction de gestion	FB10	FC2
Le bloc d'organisation	OB1	OB1
La fonction des sorties	PB1	FC3
La fonction d'allumage des lampes start et alarme.	PB10	FC4

TABLE 4.6 – La comparaison entre les fonctions du STEP5 et celle de STEP7

4.2.2.5 Comparaison entre le programme en S5 et le programme en S7

Blocs : il existe dans STEP7 les types de blocs suivants :

OB : organisation blocs comment dans STEP5

FB : fonction blocs .

FC : fonction call comparable au FB dans STEP5

Types blocs PB, FX, DX, et SB n'existent plus dans STEP7.

Les OB ne peuvent pas être appelés via le programme. Les OB contiennent des fonctions spéciales en S5, ces fonctions sont comparées aux instructions spéciales, il sont remplacés par de nouvelles instructions dans S7

Adressage : dans STEP7 toutes les zones de mémoire (E,A,M,D, etc) sont par octets. L DW 2 Dans STEP5 devient ainsi D BW4 dans STEP7[14].

Nombre d'accumulateurs : la (CPU STEP7) possède comme la (CPU STEP5) 2 ou 4 accumulateurs.

Si un programme STEP5 d'une CPU S5 avec 2 accumulateurs est converti pour une CPU S7 avec 4 accumulateurs, il faut inclure l'instruction ENT (+, -, *, /).

Chaque opération arithmétique, si le contenu de l'accumulateur 2 doit être ultérieurement traité.

Largeur d'accumulateurs : les accumulateurs dans le S7 ont une largeur de 32 bits. Opération arithmétique dans STEP 7 :

- Un mot de poids fort reste inchangé en cas de débordement ;
- Une conversion implicite au format de 32 bits n'est pas effectuée.

BIT DM : le bit DM (débordement mémoire) reste dans step7 inchangé dans les opérations suivantes :

T, >F, <F, ><F, !=F, >=F, >=F, >D, <D, ><D, !=D, >=D, <=D, >G, <G, ><G, !=G, >=G, <=G, SLW, SRW, SLD, SVW, SVD, RLD, RRD .

BIT DEB : les opérations de conversion suivantes ne modifient pas l'accumulateur 1 en cas de débordement. Dans STEP5 un débordement est affiché dans les bits DEB et DM. **LIR**, **TIR** : se sont des instructions qui n'ont pas d'adresse absolu dans STEP7. Elles sont remplacées par de nouvelles instructions et en partie par des fonctions système(SFC). **Zones de donnée BS, BT, BA, BB, Q** : une intervention manuelle est nécessaire en cas d'utilité de ces zones données.

On vous propose les solutions suivantes :

- Utilisée une zone mémentos élargie ;
- Définir des DB plus longs ;
- Utilisée fonctions système S7.

Instruction d'arrêt : il n'y a pas dans STEP7 plus qu'une instruction d'arrêt via appel SFC. Elle correspond à l'instruction STS de STEP5.

4.3 conclusion

La nécessité de suivre l'évolution technologique oblige tôt ou tard d'envisager une modernisation afin de garantir la compétitivité. La migration vers la technologie SIMATIC S7 peut nous 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. On prépare dès aujourd'hui notre succès de demain - en migrant de SIMATIC S5 vers SIMATIC S7.

Conclusion Générale

La migration de SIMATIC 5 vers SIMATIC 7 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 nous permet de bénéficier des possibilités futures d'extension et de modernisation.

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

L'objectif de notre travail est de faire fonctionner le descenseur de la ligne 3 de l'unité margarinerie par les nouveaux automates S7-300.

Nous avons pris connaissance de la machine existante, les éléments constituant la machine étudiée ainsi que leurs rôles.

Nous avons ensuite étudié les automates programmables industriels de la gamme SIEMENS, leurs caractéristiques et leur domaine d'utilisation.

Le modèle du GRAFCET a été traduit à partir du gestionnaire de projet SIMATIC manager et S7PLCSIM pour la simulation, le langage de programmation des automates programmables, les réseaux réalisés permettent de simuler le programme qui commande l'installation.

Ce travail nous a permis aussi de développer un sens de communication entre les différents ingénieurs et techniciens de l'industrie et d'acquiescer le sens de travail en équipe.

Avec ce modeste travail, les techniciens et les ingénieurs peuvent s'en servir pour bien comprendre le fonctionnement de la machine et d'optimiser le temps des opérations de maintenance.

Nous espérons que l'entreprise CEVITAL prendra notre travail comme référence pour réaliser cette migration dans les brefs délais.

Bibliographie

- [1] manuel de l'instruction pour l'usage (document CEVITAL).
- [2] HENRI Nussbaumer, " informatique industrielle " presses polytechniques de Romandes. 1987.
- [3] Manuel Siemens Simatic STEP5 (CEVITAL).
- [4] Manuel Siemens Simatic STEP7 (MAINTENANCE 1) .
- [5] Manuel Siemens Simatic S7 (programmation niveau 1).
- [6] ANDER Simon, " Automates Programmables Industriels " ,Edition l'élan LIEGE 1991.
- [7] Technique de l'ingénieur.
- [8] G.Michel, " Les A.P.I architecture et applications des automates programmables industriels "(édition DUNOD 1988).
- [9] J.C Bossy, P.faugere, C.merlaud.Le GRAFCET sa pratique et ses applications, edition CAS-TEILLA, FRANCE 1985.
- [10] Siemens, logiciel SIMATIC STEP7-V5-2.
- [11] Aide d'objet dans HW configuration.
- [12] Aide sur S7-PLCSIM.
- [13] aide sur le convertisseur des fichiers S5.
- [14] manuel Siemens Simatic S7 (programmation niveau 2).

FC1 - <offline>

"généralités"

Nom : Famille :
 Auteur : Version : 0.1
 Version de bloc : 2
 Horodatage Code : 07/04/2013 01:02:03
 Interface : 19/03/2013 10:23:12
 Longueur (bloc/code /données locales) : 00200 00096 00000

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

Bloc : FC1 généralités

Réseau : 1 temps d'arrêt/ temps de démarrage

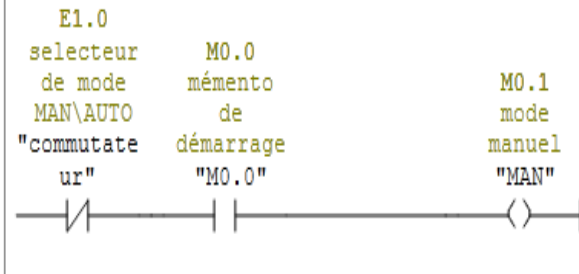
```

U   "alimentation" E0.0                    -- contacteur K2K2
L   S5T#2S
SE  "T 0"                                  T0                                    -- temps de démarrage
UN  "alimentation" E0.0                    -- contacteur K2K2
L   S5T#3S
SE  "T 1"                                  T1                                    -- temp de d'arrêt
  
```

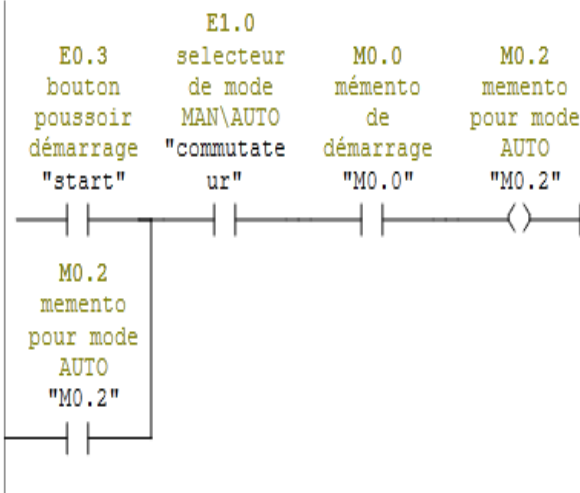
Réseau : 2 memento de démarrage
--

E0.0	T0	M0.4	M0.0
contacteur	temps de	memento	memento
K2K2	démarrage	d'alarme	de
"alimentat	"T 0"	"M0.4"	"M0.0"
ion"			

Réseau : 3 mode manuel



Réseau : 4 memento pour mode AUTO



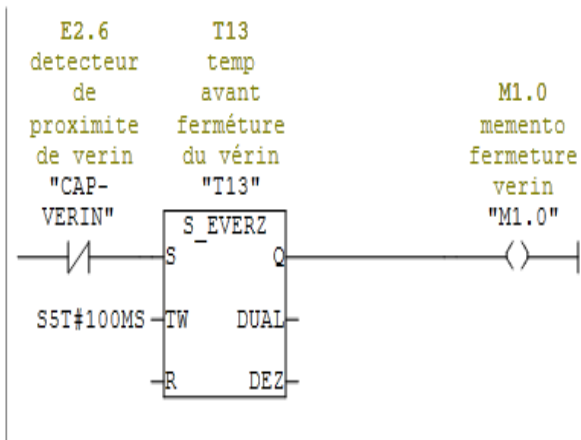
Réseau : 5 compteur

```

UN  "FDC-HAUT" E2.0          -- fin de course descenseur en HAUT
U   M   110.3
=   M   110.2
U   "FDC-HAUT" E2.0          -- fin de course descenseur en HAUT
=   M   110.3
U   M   110.2
U   "MAN"      M0.1          -- mode manuel
=   "M0.3"    M0.3          -- memento de RESETcomptage
UN  "M0.3"    M0.3          -- memento de RESETcomptage
SPB M001
L   0
T   MW   11
R   "z0"     Z0            -- compteur des cartons
M001: NOP 0

```

Réseau : 6 temp avant fermeture du vérin



FC2 - <offline>

"alarmes" défaut

Nom : Famille :

Auteur : Version : 0.1

Version de bloc : 2

Horodatage Code : 07/04/2013 12:22:25

Interface : 21/03/2013 09:29:31

Longueur (bloc/code /données locales) : 00236 00126 00000

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

Bloc : FC2

Réseau : 1 memento de reset de défaut

```

U "rearm" E0.2 -- bouton poussoir de rearmement
U "M110.1" M110.1 -- memento de rearmement
= "M110.0" M110.0 -- memento de reset de défaut
UN "rearm" E0.2 -- bouton poussoir de rearmement
= "M110.1" M110.1 -- memento de rearmement
UN "M110.0" M110.0 -- memento de reset de défaut
SPB m001
L 0
T MB 2
m001: NOP 0

```

Réseau : 2 défaut arrêt d'urgence

E0.1	M2.0
entrée	memento
urgence	d'arrêt
"arrêt	d'urgence
d'urgence"	"M2.0"

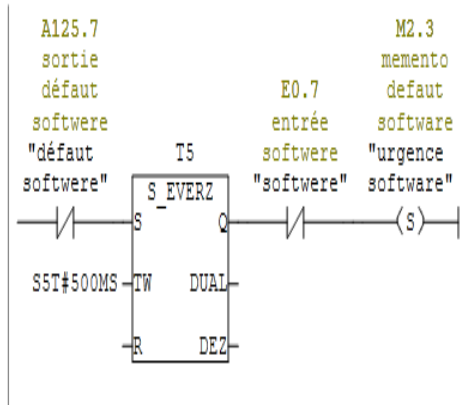
Réseau : 3 défaut thermique

E0.5	M2.1
entrée	memento
protection	de
des	protection
moteurs	thermique
thermique	"M2.1"
"defaut	
thermique"	

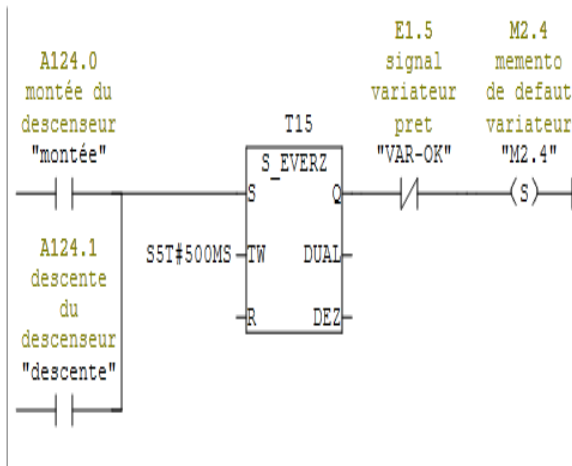
Réseau : 4 air comprimé

E0.6	M2.2
entrée	memento
d'air	manque
comprimie	d'air
"air"	"M2.2"

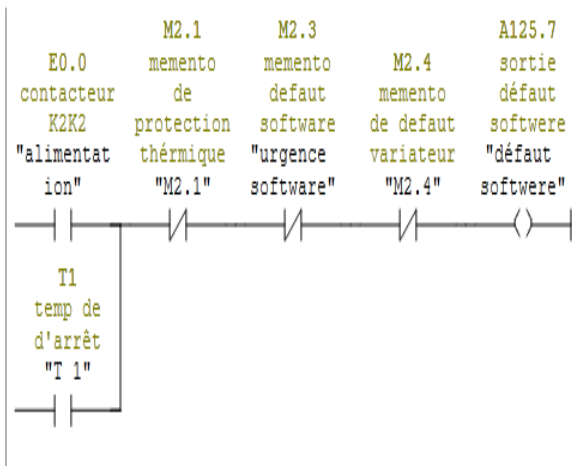
Réseau : 5 défaut software



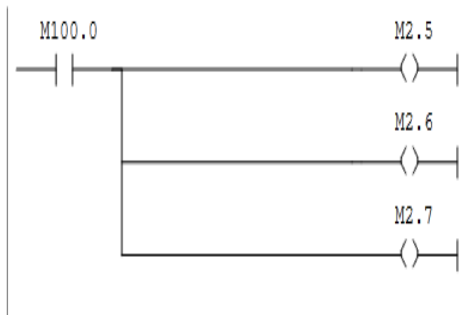
Réseau : 6 défaut variateur



Réseau : 7 génération de défaut software



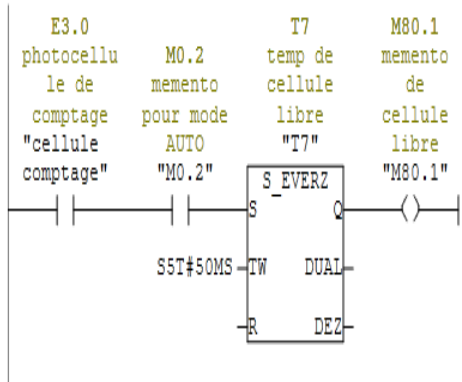
Réseau : 8 memento reserve pour programme d'autre alarmes



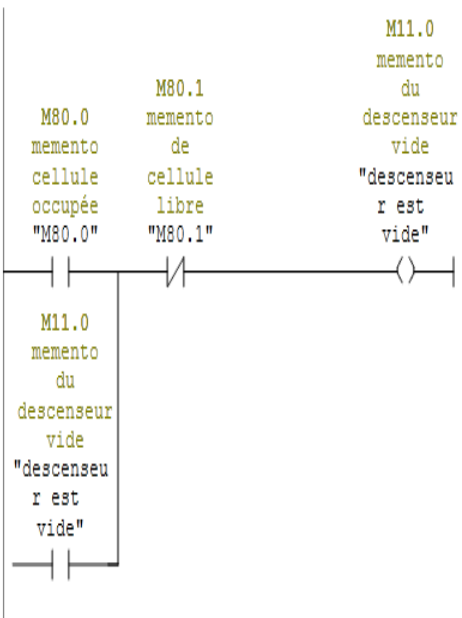
Réseau : 9 memento d'alarme

```
L   MB   2
L   0
<>I
=   "M0.4"  M0.4        -- memento d'alarme
```

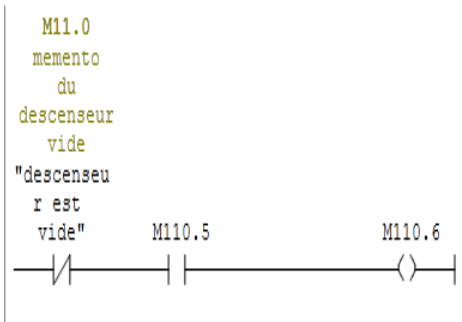

Réseau : 2 temp de cellule libre



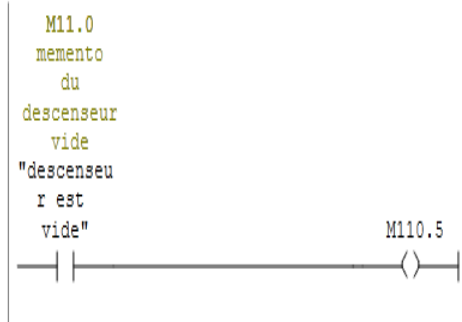
Réseau : 3 memento du descenseur vide



Réseau : 4



Réseau : 5



Réseau : 6 compteur des cartons

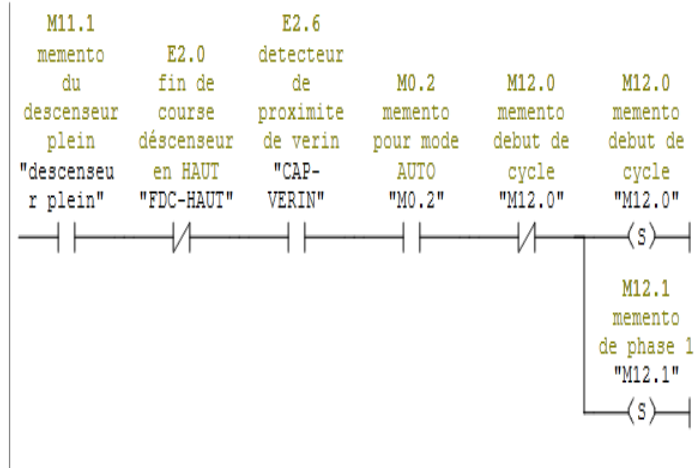
```

U   M   110.6
ZV  "z0"      Z0      -- compteur des cartons
L   "z0"      Z0      -- compteur des cartons
L   5
>=I
S   "descenseur plein" M11.1      -- memento du descenseur plein

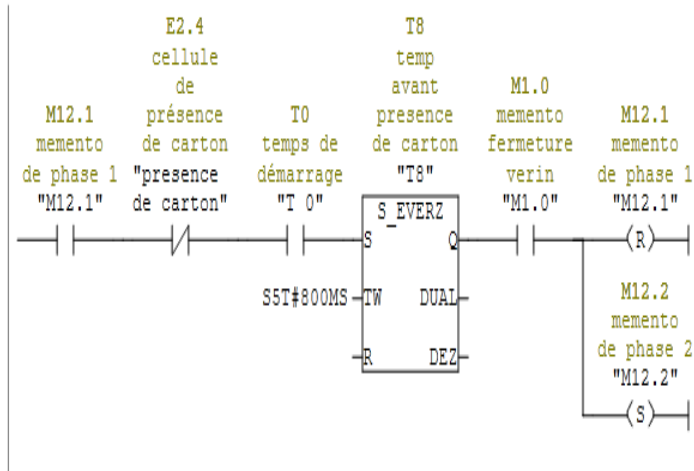
```

Réseau : 7 cycle fonctionnement descenseur

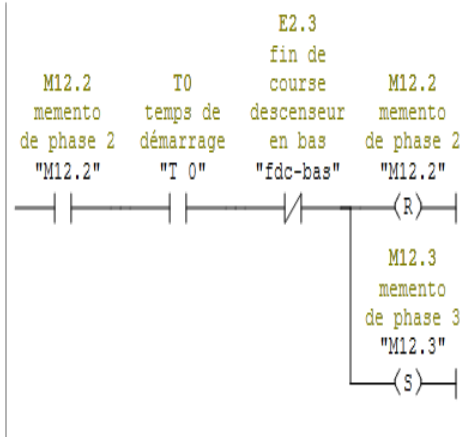
phase



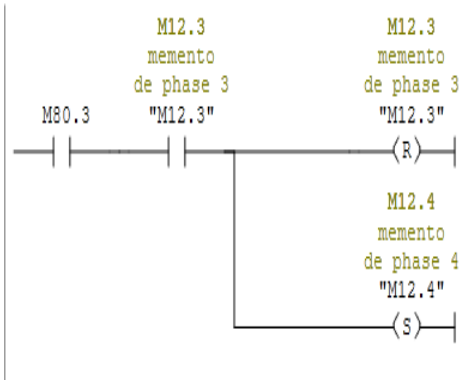
Réseau : 8 phase 1



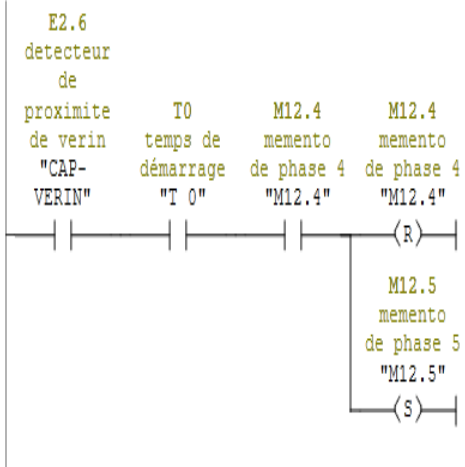
Réseau : 9 memento de phase 2



Réseau : 10 memento de phase 3



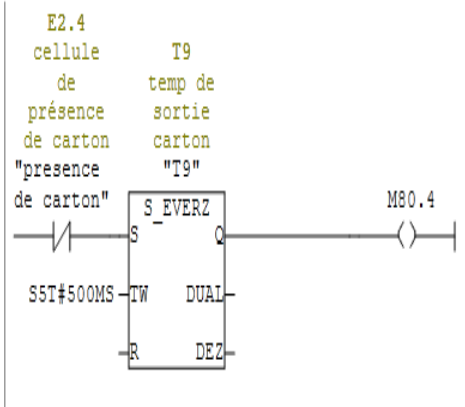
Réseau : 11 memento de phase 4



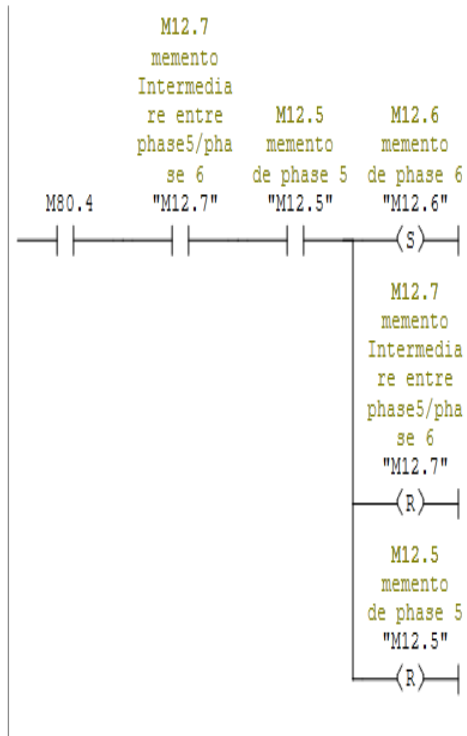
Réseau : 12 memento phase 5



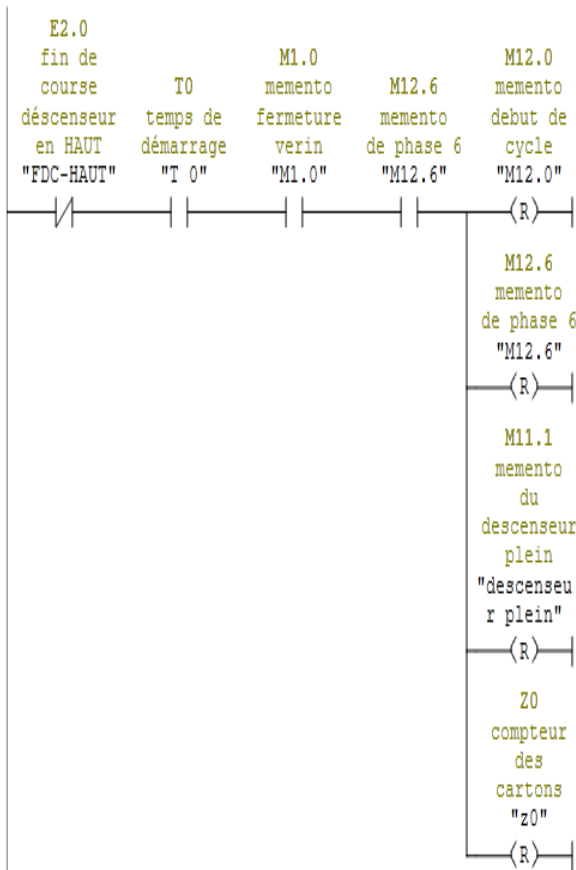
Réseau : 13 memento de phase 6



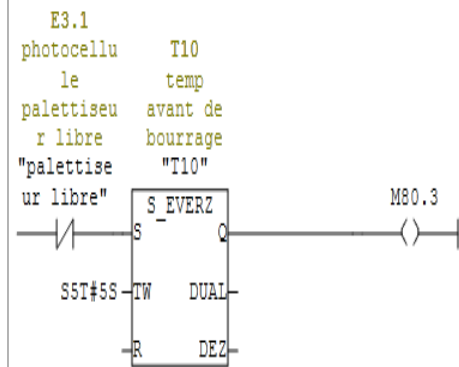
Réseau : 14 memento de phase 6



Réseau : 15 initialisation de cycle



Réseau : 16 controle bourrage palettiseur



FC4 - <offline>

"sorties" sorties

Nom : Famille :

Auteur : Version : 0.1

Version de bloc : 2

Horodatage Code : 07/04/2013 12:20:56

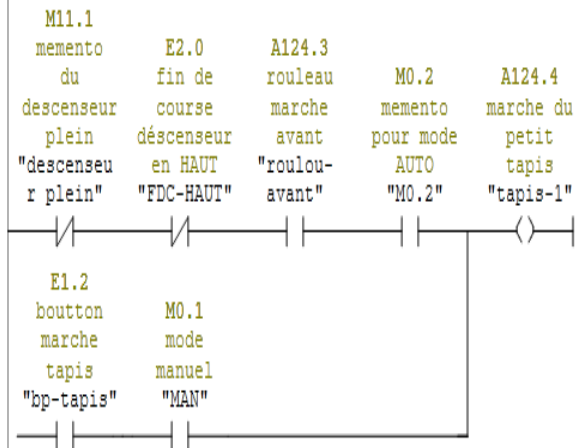
Interface : 28/03/2013 09:09:56

Longueur (bloc/code /données locales) : 00406 00280 00000

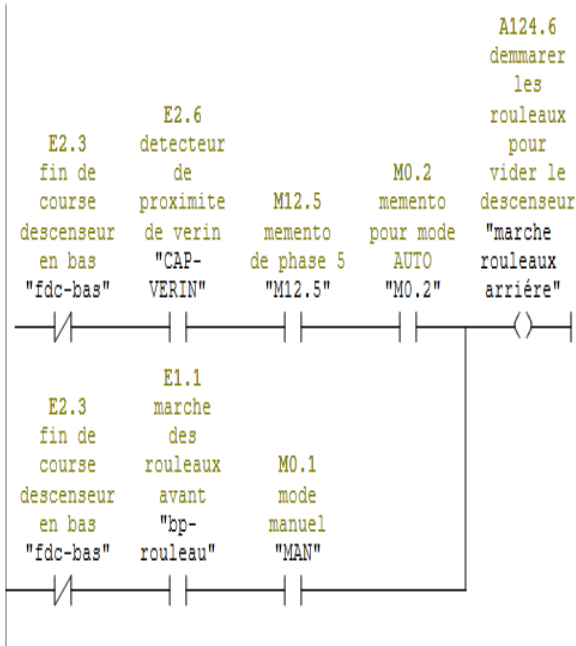
Nom	Type de données	Adresse	Commentaire
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Bloc : FC4

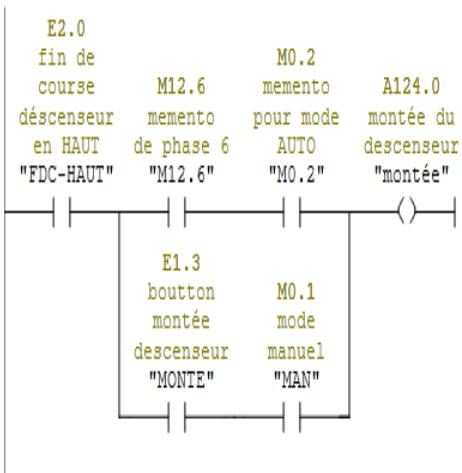
Réseau : 1 marche du petit tapis



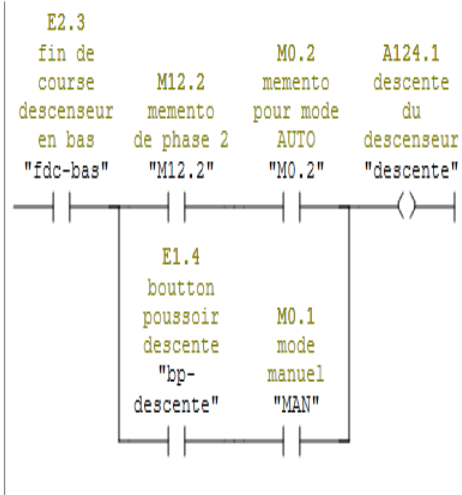
Réseau : 4 déchargement des cartons



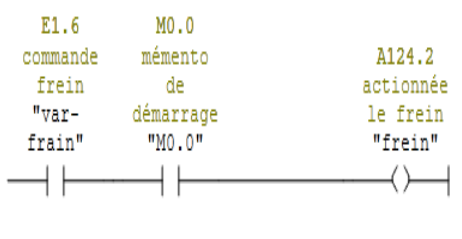
Réseau : 5 montée du descenseur



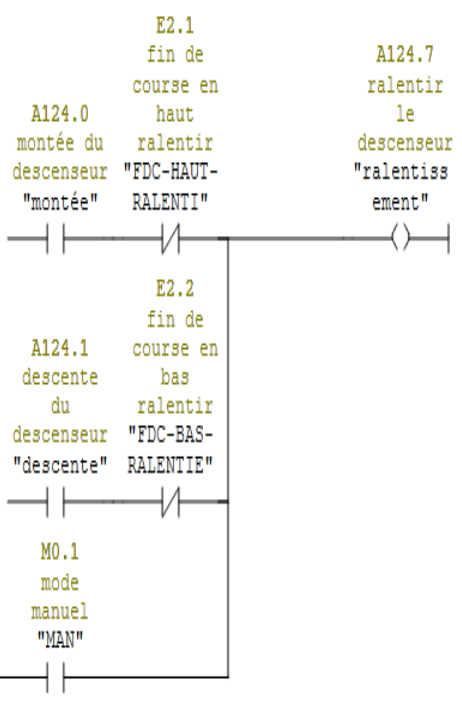
Réseau : 6 descente



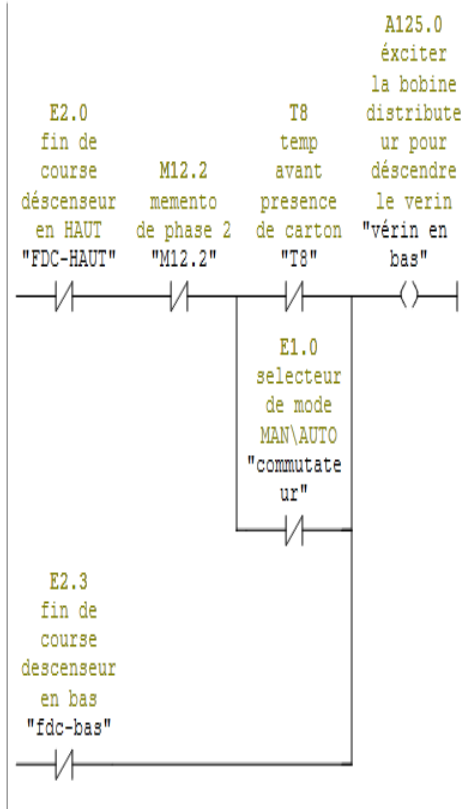
Réseau : 7 actionnée le frein



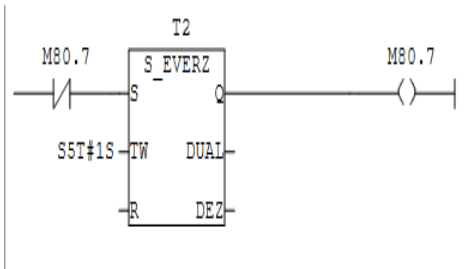
Réseau : 8 ralentir le descenseur



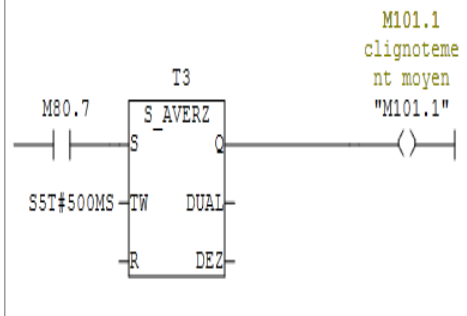
Réseau : 9 actionnée le distributeur du vérin



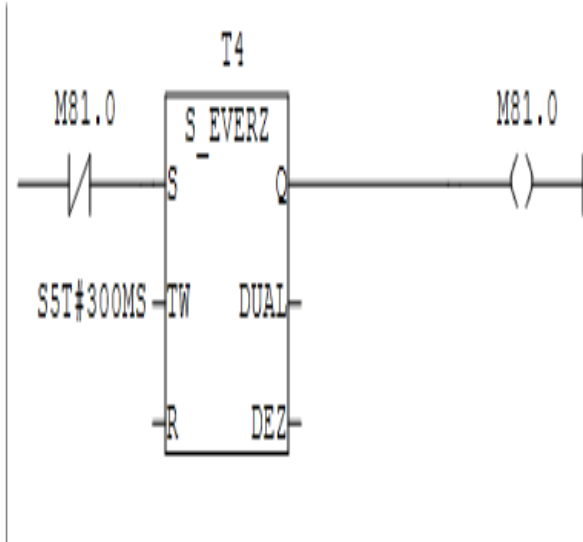
Réseau : 10



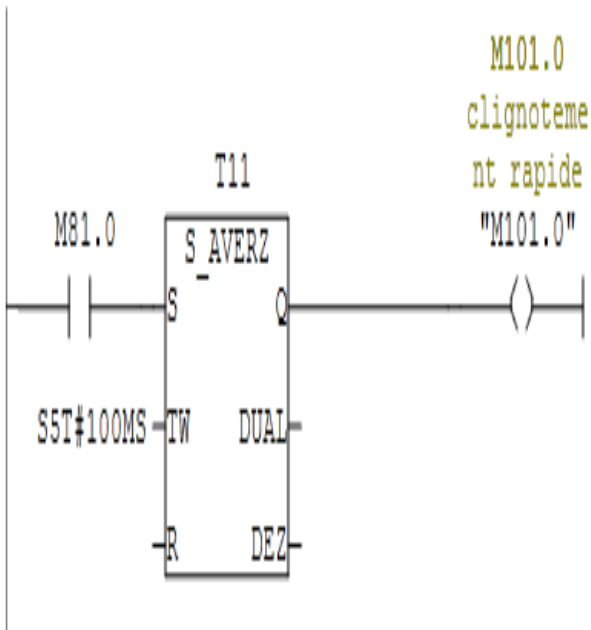
Réseau : 11 memento pour clignotement moyen



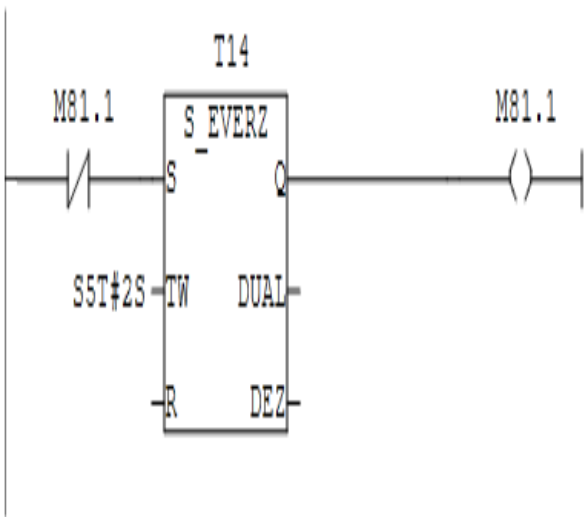
Réseau : 12



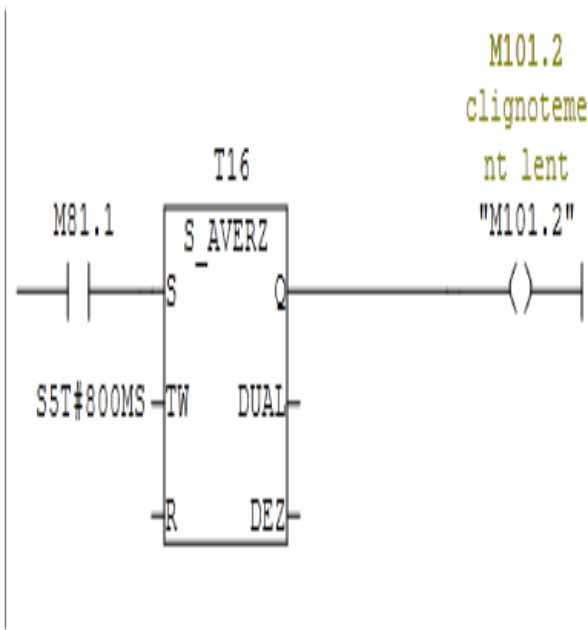
Réseau : 13 memento pour clignotement rapide



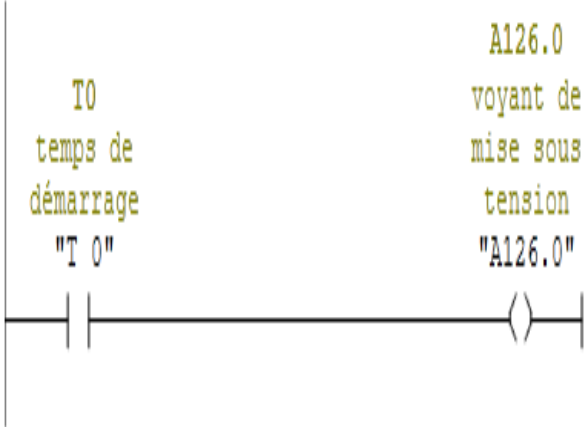
Réseau : 14



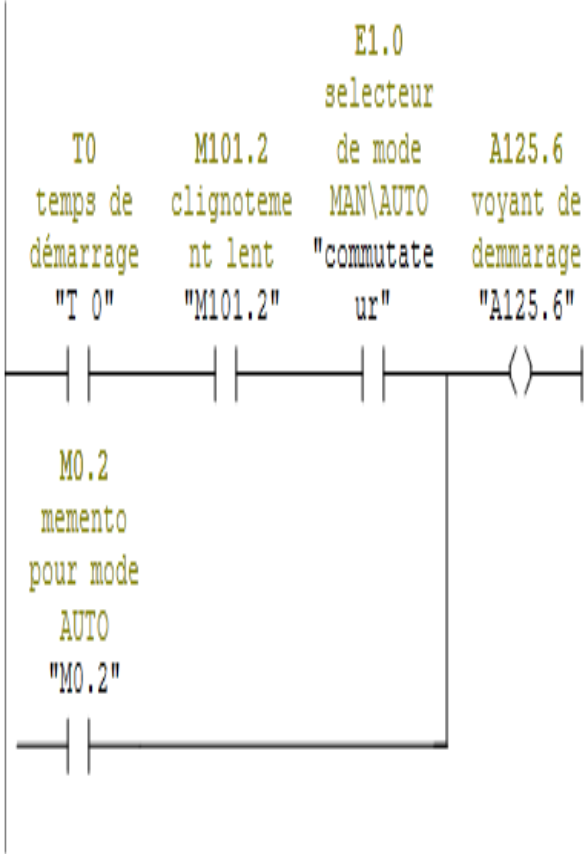
Réseau : 15 memento pour clignotement lent



Réseau : 16 voyon pour la mise sous tension



Réseau : 17 temoin start



Propriétés de la table des mnémoniques

Nom :	Mnémoniques
Auteur :	
Commentaire :	
Date de création :	10/04/2013 11:30:45
Dernière modification :	02/05/2013 11:24:31
Dernier filtre sélectionné :	Tous les mnémoniques
Nombre de mnémoniques :	80/80
Dernier tri :	Mnémonique ordre croissant

Mnémonique	Opérande	Type de données	Commentaire
A125.6	A 125.6	BOOL	voyant de demarrage
A126.0	A 126.0	BOOL	voyant de mise sous tension
air	E 0.6	BOOL	entrée d'air comprimie
alarmes	FC 2	FC 2	défaut
alimentation	E 0.0	BOOL	contacteur K2K2
arrêt d'urgence	E 0.1	BOOL	entrée urgence
bp-descente	E 1.4	BOOL	boutton poussoir descente
bp-rouleau	E 1.1	BOOL	marche des rouleaux avant
bp-tapis	E 1.2	BOOL	boutton marche tapis
CAP-VERIN	E 2.6	BOOL	decteur de proximite de verin
cellule comptage	E 3.0	BOOL	photocellule de comptage
commutateur	E 1.0	BOOL	selecteur de mode MAN\AUTO
défaut softwere	A 125.7	BOOL	sortie défaut softwere
defaut thermique	E 0.5	BOOL	entrée protection des moteurs thermique
descenseur est vide	M 11.0	BOOL	memento du descenseur vide
descenseur plein	M 11.1	BOOL	memento du descenseur plein
descente	A 124.1	BOOL	descente du descenseur
E5.1	E 5.1	BOOL	entree d'arret d'urgence
fdc-bas	E 2.3	BOOL	fin de course descenseur en bas
FDC-BAS-RALENTIE	E 2.2	BOOL	fin de course en bas ralentir
FDC-HAUT	E 2.0	BOOL	fin de course descenseur en HAUT
FDC-HAUT-RALENTI	E 2.1	BOOL	fin de course en haut ralentir
frein	A 124.2	BOOL	actionnée le frein
généralités	FC 1	FC 1	
gestion	FC 3	FC 3	
grand-tapis	A 124.5	BOOL	grand tapis en marche
M0.0	M 0.0	BOOL	memento de démarrage
M0.2	M 0.2	BOOL	memento pour mode AUTO
M0.3	M 0.3	BOOL	memento de RESETcomptage
M0.4	M 0.4	BOOL	memento d'alarme
M1.0	M 1.0	BOOL	memento fermeture verin
M101.0	M 101.0	BOOL	clignotement rapide
M101.1	M 101.1	BOOL	clignotement moyen
M101.2	M 101.2	BOOL	clignotement lent
M110.0	M 110.0	BOOL	memento de reset de default
M110.1	M 110.1	BOOL	memento de rearmement
M12.0	M 12.0	BOOL	memento debut de cycle
M12.1	M 12.1	BOOL	memento de phase 1

Mnémonique	Opérande	Type de données	Commentaire
M12.2	M 12.2	BOOL	memento de phase 2
M12.3	M 12.3	BOOL	memento de phase 3
M12.4	M 12.4	BOOL	memento de phase 4
M12.5	M 12.5	BOOL	memento de phase 5
M12.6	M 12.6	BOOL	memento de phase 6
M12.7	M 12.7	BOOL	memento Intermediare entre phase5/phase 6
M2.0	M 2.0	BOOL	memento d'arrêt d'urgence
M2.1	M 2.1	BOOL	memento de protection thermique
M2.2	M 2.2	BOOL	memento manque d'air
M2.4	M 2.4	BOOL	memento de défaut variateur
M80.0	M 80.0	BOOL	memento cellule occupée
M80.1	M 80.1	BOOL	memento de cellule libre
MAN	M 0.1	BOOL	mode manuel
marche rouleaux arrière	A 124.6	BOOL	demarrer les rouleaux pour vider le descenseur
MONTE	E 1.3	BOOL	boutton montée descenseur
montée	A 124.0	BOOL	montée du descenseur
palettiseur libre	E 3.1	BOOL	photocellule palettiseur libre
presence de carton	E 2.4	BOOL	cellule de présence de carton
ralentissement	A 124.7	BOOL	ralentir le descenseur
rearm	E 0.2	BOOL	boutton poussoir de rearmement
rouleau-arriere	A 126.4	BOOL	marche des rouleaux arrière
roulou-avant	A 124.3	BOOL	rouleau marche avant
software	E 0.7	BOOL	entrée software
sorties	FC 4	FC 4	sorties
start	E 0.3	BOOL	bouton poussoir démarrage
T 0	T 0	TIMER	temps de démarrage
T 1	T 1	TIMER	temp de d'arrêt
T10	T 10	TIMER	temp avant de bourrage
T13	T 13	TIMER	temp avant fermeture du vérin
T6	T 6	TIMER	temp de cellule occupée
T7	T 7	TIMER	temp de cellule libre
T8	T 8	TIMER	temp avant presence de carton
T9	T 9	TIMER	temp de sortie carton
tapis-1	A 124.4	BOOL	marche du petit tapis
tapis-2	M 124.5	BOOL	marche du grand tapis
urgence software	M 2.3	BOOL	memento defaut software
var-frain	E 1.6	BOOL	commande frein
VAR-OK	E 1.5	BOOL	signal variateur pret
vérin en bas	A 125.0	BOOL	exciter la bobine distributeur pour descendre le verin
voyant-alarme	A 125.5	BOOL	temoin d'alarme
z0	Z 0	COUNTER	compteur des cartons
Z1	Z 1	COUNTER	dyhdydy