

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTRE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE**



UNIVERSITE DE M'SILA

FACULTE DE TECHNOLOGIE

DEPARTEMENT DE GENIE ELECTRIQUE

**MEMOIRE DE FIN D'ETUDES EN VUE DE L'OBTENTION DU DIPLÔME
DE MASTER EN GENIE ELECTRIQUE**

SPECIALITE: COMMANDE DES SYSTEMES ELECTRIQUES

THEME

**COMMANDE D'UN SYSTEME AUTOMATISE DE PRODUCTION (SAP)
PAR AUTOMATE PROGRAMMABLE INDUSTRIEL (API)
« BRIQUETERIE D'EL KALAA BENI HAMMAD -M'SILA »**

Proposé et dirigé par :

Mr : Abdelhak ABDOU

Présenté par :

Lahcen. GUERRAS

Année Universitaire: 2012/2013

N° d'ordre : 082

Dédicaces

*Je dédie cet humble travail
À mes très chers parents pour
leurs grand amour et leurs
sacrifices*

À toute ma famille

À toutes mes amis

À tous ce qu'il m'a aidé

Lahcen GUERRAS

Remerciements

En préambule à ce mémoire, je souhaitais adresser mes remerciements les plus sincères aux personnes qui m'ont apporté leur aide et qui ont contribué à l'élaboration de ce mémoire.

Je tiens à remercier sincèrement Monsieur Abdelhak .ABDOU, qui, en tant qu'encadreur de mémoire, s'est toujours montré à l'écoute et très disponible tout au long de la réalisation de ce mémoire, ainsi pour l'inspiration, l'aide et le temps qu'il a bien voulu me consacrer et sans qui ce mémoire n'aurait jamais vu le jour.

Je suis aussi redevable à, tout les enseignants du département de génie électrique, ainsi qu'à Monsieur BENJAIMA. Bachir , chef du département de génie électrique.

Je ne peux oublier Je tiens également à adresser mes vives remerciements à Mr. AYACHE, directeur de la Briqueterie d'El KALAA BENI - HAMMAD, qui nous a autorisé à faire le stage au sein de l'entreprise, ainsi que tous le personnel de l'entreprise qui ont instaurer un environnement d'entraide et de soutien tant au plan technique qu'au plan humain. durant notre stage.

Une pensée particulière à mes parents pour leurs contributions, leurs soutiens et leurs patiences., que DIEU les gardent.

Enfin, j'adresse mes plus sincères remerciements à tous mes proches et amis, qui m'ont toujours soutenue et encouragé au cours de la réalisation de ce mémoire.

Lahcen. GUERRAS

Table des matières

Sommaire

<i>Dédicaces</i>	i
<i>Remerciements</i>	ii
<i>Sommaire</i>	iii
<i>Liste des figures</i>	viii
<i>Liste des tableaux</i>	ix
INTRODUCTION GENERALE.....	1
Chapitre I :automatisme et automate programmable.....	3
I-1 Introduction	3
I-2 Notion de système automatisé	3
I-2-1 Point de vue historique	3
I-2-2 Point de vue technique	3
I-2-3 Point de vue économique et social	4
I-3 Automatisation.....	4
I-3-1 Objectifs de l'automatisation	4
I-3-2 Conduite et surveillance d'un système automatisé	5
I-3-3 Structure d'un système automatisé.....	5
I-2-3-2 Partie Commande	6
I-4 Système automatisé de production.....	7
I-4-1 Objectifs de l'automatisation des productions.....	8
I-4-2 Acquisition des données.....	8
I-5 Technologies des automatismes	9
I-6 Dialogue homme –machine	10
I-7 protection du système	10
I-8 Champ d'application de l'automatisme	11
I-9 Les automates programmables industriels	11
I-9-1 Historique	11
I-9-2 Définition	12
I-9 -3 Les Objectifs.....	12
I-9-4 Fonctionnement d'un automate programmable	13
I-9-5 Domaines d'emploi des automates	13
I-9-6 Structure générale d'un automate programmable.....	14
I-9-6-1 L'unité centrale	15

I-9-6-2 Module d'alimentation	16
I-9-6-3 Modules d'entrées/sorties (E/S).....	17
I-9-7 Langages de la programmation	18
I-9-7-1 Programmation à l'aide du GRAFCET (SFC: Sequential Function Chart).....	18
I-9-7-2 Langage à contacts (LD : Ladder Diagram).....	18
I-9-7-3 Langage Blocs Fonctionnels (FBD : Function Bloc Diagram)	19
I-9-7-4 Langage ST	19
I-9-7-5 Le langage IL.....	20
I-10 L'environnement de l'API	20
I-10 Conclusion.....	21
Chapitre II : description du logiciel STEP7	23
II-1 introduction :	23
II-2 Gestionnaire de projet SIMATIC <i>manager</i>	23
II-2-1 Editeur de programme et les langages de programmation	23
II-2-2 Simulateur des programmes PLCSIM	24
II-2-3 La conception d'une structure programme compète et optimisé	25
II-3 Réalisation du programme	26
II-3-1 Création d'un projet dans SIMATIC <i>manager</i>	26
II-3-2 Configuration matérielle	27
II-3-3 Création de la table des mnémonique	30
II-3-4 Elaboration du programme S7	31
- Les blocs de code :	31
- Les blocs d'organisation(OB) :	31
- Les blocs fonctionnels (FB) :	32
- Les fonctions (FC) :	32
- Les blocs de données (DB) :	32
II-4 conclusion	32
Chapitre III : description et fonctionnement de la briqueterie	34
III-1 Introduction :	34
III-2 Présentation de la briqueterie d'EL KALAA BENI-HAMMAD :	34
III-3 Sécurité d'usine :	34
III-4 Processus de fabrication(fonctionnement) :	35
III-4-1 Hiérarchie des GRAFCET	35
III-4-1-1 GRAFCET de sécurité (ou de surveillance) :	35
III-4-1-2 GRAFCET de Conduite et Initialisation (ou GRAFCET des modes de marches).....	36

III-4-1-3 GRAFCET de coordination des tâches.....	37
III-4-2 Tâche de préparation de l'argile	38
III-4-2-1 Cahier du charge.....	38
III-4-2-2 Schéma synoptique de la tâche de préparation.....	39
III-4-2-3 Table des entrées.....	40
III-4-2-4 Table des sorties	40
III-4-2-5 GRAFCET de la tâche de préparation.....	41
III-4-3 Tâche de fabrication	42
III-4-3-1 Cahier du charge.....	42
III-4-3-2 Schéma synoptique de la tâche de fabrication.....	43
III-4-3-3 Table des entrées.....	44
III-4-3-4 Table des sorties	44
III-4-3-4 GRAFCET de la tâche de fabrication	45
III-4-4 Tâche de coupure.....	46
III-4-4-1 Cahier du charge.....	46
III-4-4-2 Table des entrées.....	47
III-4-4-3 Table des entrées.....	47
III-4-4-4 GRAFCET de la tâche de coupure	48
III-4-5 Tâche de séchage	49
III-4-5-1 Cahier du charge.....	49
III-4-5-2 Schéma synoptique de la tâche du séchage.....	50
III-4-5-3 Table des entrées.....	51
III-4-5-4 Table des sorties	51
III-4-5-5 GRAFCET de la tâche du séchage.....	52
III-4-6 Tâche de cuisson.....	53
III-4-6-1 Cahier du charge.....	53
III-4-6-2 Schéma synoptique de la tâche de la cuisson	54
III-4-6-3 Table des entrées de la tâche de cuisson	55
III-4-6-4 Table des sorties de la tâche de cuisson.....	55
III-5 Conclusion	57
Chapitre IV : programmation par STEP7.....	59
IV-1-Introduction	59
IV -2- Présentation du langage de programmation	59
IV- 2-1 Langage de programmation GRAPH	59
IV -2-2 Blocs de la commande séquentielle.....	59

IV -2-3 Interface utilisateur de GRAPH.....	60
IV -2-4 Création d'un FB GRAPH.....	61
IV-2-4-1 Création de la table des mnémoniques.....	62
IV- 2-4-2 Remplissage de la table des mnémoniques	63
IV- 2-5 Création du graphe séquentiel	63
IV- 2-5-1 Mode d'insertion directe	64
IV- 2-5-2 Mode d'insertion présélection	64
IV-3 Programmation des tâches :	65
IV-3-1 Configuration du matériel :	65
IV-3-1 Tables des mnémoniques.....	65
IV-3-2 Programmation du GRAFCET de sécurité.....	67
IV-3-4 Programmation du GRAFCET de conduite et Initialisation.....	67
IV-3-5 Programmation du GPN	68
IV-3-5 Programmation de tâche de préparation	69
IV-3-6 Programmation de tâche de fabrication	70
IV-3-7 Programmation de tâche de coupure	71
IV-3-8 Programmation de tâche de séchage.....	72
IV-3-9 Programmation de tâche de cuisson	73
IV-4 Conclusion :	75
CONCLUSION GENERALE :	77
BIBLIOGRAPHIE	79
Annexes.....	81

Table des figures et des tableaux

Liste des figures

Chapitre I :automatisme et automate programmable

Figure I. 1 :Structure d'un système automatisée	6
Figure I. 2: Système automatisé de production.	7
Figure I. 3: Armoire de commande basée sur logique Câblé	10
Figure I. 4: Armoire de commande basée sur la logique programmée.	10
Figure I. 5: Champ d'application de l'automatisme.	11
Figure I. 6: Fonctionnement cyclique d'un API.	13
Figure I. 7: Fonctionnement cyclique d'un API.	14
Figure I. 8: Fonctionnement cyclique d'un API.	15
Figure I. 9: Partie opérative et partie commande.	17
Figure I. 10: Exemple Programme SFC avec divergence et convergence doubles.	18
Figure I. 11: Exemple Programme LD avec de liaison multiple à droite.	19
Figure I. 12: Exemple Programme FBD avec de saut et d'étiquette.	19
Figure I. 13: Exemple Programme ST.	20
Figure I. 14: Exemple Programme IL.	20
Figure I. 15: L'environnement de l'API.	21

Chapitre II : description du logiciel STEP7

Figure II. 1: L'icône du logiciel STEP7.	23
Figure II. 2: Interface de simulation PLCSIM.	25
Figure II. 3: Page de démarrage de STEP7.	27
Figure II. 4: Choix du RACK.	28
Figure II. 5: Choix du CPU et de l'alimentation.	29
Figure II. 6 : Choix des embases d'E/S.	30
Figure II. 7: Insertion de configuration dans le projet	30
Figure II. 8: Création de la table des mnémoniques.	31

Chapitre III : description et fonctionnement de la briqueterie

Figure III. 1: Hiérarchie des GRAFCET.	35
Figure III. 2: GRAFCET de sécurité.	36
Figure III. 3: GRAFCET de Conduit et Initialisation.	36
Figure III. 4: GRAFCET de coordination de tâche.	37
Figure III. 5: Schéma synoptique de l'installation de préparation affiché sur pupitre.	39
Figure III. 6: GRAFCET de la tâche de préparation	41
Figure III. 7: Schéma synoptique de l'installation de fabrication affiché sur pupitre.	43
Figure III. 8: GRAFCET de la tâche de fabrication	45
Figure III. 9: GRAFCET de la tâche de coupure	48
Figure III. 10: Schéma synoptique de l'installation du séchoir affiché sur pupitre.	50
Figure III. 11: GRAFCET de la tâche de séchage	52
Figure III. 12: Schéma synoptique de l'installation de la cuisson affiché sur pupitre.	54

Figure III. 1: GRAFCET de la tâche de cuisson.56

Chapitre IV : programmation par STEP7

Figure IV. 1: Blocs de la commande séquentielle.....60
 Figure IV. 2: Interface utilisateur de GRAPH.....61
 Figure IV. 3: Méthode de création d'un FB GRAPH.....62
 Figure IV. 4: Méthode de création de mnémonique.....63
 Figure IV. 5: Remplir une table des mnémoniques.....63
 Figure IV. 5: Programme du GRAFCET de sécurité.....67
 Figure IV. 6: Programme du GCI.....67
 Figure IV. 7: Programme du GPN.....68
 Figure IV. 8: Programme de tâche de préparation.....69
 Figure IV. 9: Programme de tâche de fabrication.....70
 Figure IV. 10: Programme de tâche de coupure.....71
 Figure IV. 11: Programme de tâche de séchage.....72
 Figure IV. 12: Programme de tâche de cuisson.....74

Liste des tableaux

Chapitre I :automatisme et automate programmable

Tableau I. 1: Comparaison entre la solution câblée et la solution programmée. 9

Chapitre III : description et fonctionnement de la briqueterie

Table III. 1: Table des entrées de la tache de préparation.....40
 Table III. 2: Table des entrées de la tache de préparation.....40
 Table III. 3: Table des entrées de la tâche de fabrication.....44
 Table III. 4: Table des sorties de la tâche de fabrication.....44
 Table III. 5: Table des entrées de la tâche du coupeur.....47
 Table III. 6: Table des sorties de la tâche du coupeur.....47
 Table III. 7: Table des entrées de la tâche de séchage.....51
 Table III. 8: Table des sorties de la tâche du séchage.....51
 Table III. 9: Table des entrées de la tâche de cuisson.....55
 Table III. 10: Table des sorties de la tâche de cuisson.....55

Chapitre IV : programmation par STEP7

Table IV. 1: Table des mnémoniques.....66

Introduction Générale

INTRODUCTION GENERALE

*"Depuis toujours l'homme est en quête de bien être". Cette réflexion (qui rejoint la notion de **besoin**) peut paraître bien éloignée des Sciences Industrielles, pourtant c'est la base de l'évolution des sciences en général, et de l'automatisation en particulier. L'homme a commencé par penser, concevoir et réaliser. Lorsqu'il a fallu multiplier le nombre d'objets fabriqués, produire en plus grand nombre, l'automatisation des tâches est alors apparue : remplacer l'homme dans des actions pénibles, délicates ou répétitives.*

Automatismes industriels traite de façon claire et précise - en restant au plus près du milieu industriel - la partie commande (PC) des systèmes automatisés de production. Les outils graphiques (GRAFCET, GEMMA), les technologies actuelles sont étudiés progressivement et sont illustrés de nombreuses applications concernant les principaux secteurs d'activité du monde industriel, comme le secteur de fabrication de la brique

le secteur du bâtiment traverse aujourd'hui une phase d'innovation sans précédente. L'industrie de la brique est au cœur de cette révolution, qui met l'accent sur les performances mais aussi sur le cycle de vie des matériaux qui constituent la brique.

Au cours des dernières années, la brique de terre cuite a progressivement détrôné le parpaing en béton comme premier matériau pour les murs des maisons. La compétitivité a joué un rôle important : un usinage extrêmement précis des briques a relégué aux oubliettes la pose au mortier au profit de la pose dite « à joint mince », beaucoup plus rapide et économique sur chantier. Mais ce sont surtout les performances thermiques qui ont fait la différence. Avec une structure alvéolaire sophistiquée, la brique est devenue un bon isolant thermique : pour une même épaisseur de 20 cm, sa résistance thermique a été multipliée par quatre en six ans.

Or justement, c'est une nouvelle révolution, celle des contraintes énergétiques et environnementales, qui est en passe de changer radicalement la donne pour le secteur du bâtiment.

Vu les caractéristiques importantes citées à la fois des systèmes automatisés et du secteur de fabrication de la brique, que nous avons effectués un stage auprès de la briqueterie D'ELKALAA BENI-HAMMED .afin d'améliorer nos connaissances théoriques par le côté pratique et pour contribuer à l'amélioration de la solution automatisée existante

L'automatisme est devenu une technologie incontournable aujourd'hui de par son utilisation dans tous les domaines de fabrication. Il est donc important d'en connaître les bases et d'en suivre l'évolution. Cette formation, dans sa structure, suit le cheminement de la conception d'un système automatisé depuis la logique câblée jusqu'au dialogue homme-machine caractérisé par l'insertion des automates programmables industriels (API), [5, 8].

La force principale d'un automate programmable industriel API réside dans sa grande capacité de communication avec l'environnement industriel, ils sont aujourd'hui les constituants les plus répandus pour réaliser des automatismes. On les trouve pratiquement dans tous

les secteurs de l'industrie car ils répondent à des besoins d'adaptation et de flexibilité pour un grand nombre d'opérations. Cette émergence est due en grande partie, à la puissance de son environnement de développement et aux larges possibilités d'interconnexions.

Les systèmes automatisés de production sont de plus en plus complexes, afin de simplifier l'étude, la mise en œuvre et la maintenance du système, il est nécessaire de structurer la partie commande et la partie opérative. L'objectif essentiel de la structuration est de permettre une approche progressive du fonctionnement d'un système automatisé, tant au niveau de l'analyse qu'au niveau de la représentation ceci peut être concrétiser par le GRAFCET hiérarchisé, cette dernière étant la solution proposée pour le système de fabrication de brique (cahier de charge) et qui constitue l'objectif principale de notre travail, avec la programmation par API sous PL7, [9].

Ce mémoire comporte quatre chapitres :

Un premier chapitre, dont le but est de rappeler brièvement les caractéristiques des systèmes automatisés, nous évoquerons le principe des APIs Nous passerons ensuite à une description des langages de programmation utilisés par les APIs

Le deuxième chapitre sera consacré à la présentation du logiciel STEP7 , les différents éléments qui le compose, les langages utilisés, et nous présenterons toute les étapes de l'élaboration d'un programmation.

Au troisième chapitre, nous donnerons une description globale du fonctionnement de la briqueterie D'ELKALAA BENI-HAMMED ainsi que les solutions proposées sous forme de tâche en utilisant le GRAFCET hiérarchisée.

Quant au quatrième chapitre, il sera réservé pour la programmation, sous le logiciel STEP7 (langage GRAPH), des GRAFCETs de Sécurité (GS), GRAFCET de Conduite et d'Initialisation (GCI), et GRAFCET de Production Normale (GPN), ainsi que les différents GRAFCET des tâches de fonctionnement .

Nous terminerons par une conclusion générale et les perspectives envisagées pour la continuité de ce travail.

Chapitre I
Automatisme
et
Automates Programmables Industriels (API)

Chapitre I :automatisme et automate programmable

I-1 Introduction

Dans l'industrie il existe plusieurs systèmes automatisés, ils sont indispensables, ils effectuent quotidiennement les tâches les plus difficiles et les plus répétitives ou les plus dangereuses. Parfois, les automatismes sont d'une telle rapidité et d'une telle précision, qu'ils réalisent des actions impossibles pour un être humain. Donc l'automatisme est réalisé en vue d'apporter des solutions à des problèmes de nature technique économique ou humaine.

Nous évoquerons dans ce chapitre, en premier lieu des notions générales sur les systèmes automatisés, les systèmes de production et leurs technologies et le champ d'application des systèmes automatisés.

Puis nous passerons à la description détaillée de l'architecture des automates programmables industriels (API); domaines d'applications des API et les différents langages de programmation.

Nous terminerons par une analyse comparative des techniques câblées et programmées utilisées en automatisation des systèmes de production .

I-2 Notion de système automatisé

I-2-1 Point de vue historique

L'évolution des systèmes automatisés s'est faite en trois temps avec deux grandes ruptures. Chaque temps correspond à un type d'époque et à un type de logique de pensée. Ainsi on passe d'une vision imitative (Imitations des vivants au travers d'automates) à une vision mécaniste dans les années 50 (Opérationnalisation des machines, la machine remplace l'homme dans les tâches difficiles et répétitives, Ford) pour terminer vers une vision actuelle systémique (c'est l'ère de l'intelligence artificielle naît de la cybernétique de Wiener, la machine réfléchie, analyse et s'adapte, la spécialisation à fait place à la polyvalence et à la capacité d'adaptation), [1].

I-2-2 Point de vue technique

La fonction globale d'un système automatisé est une modification sur l'environnement pour laquelle de l'énergie est mobilisée. Sa finalité est d'apporter une valeur ajoutée à la matière d'œuvre (énergie, information, matière) qui est mise en jeu.

Les applications sont vastes : industrielles, environnement, domotique (système de régulation et de sécurité), gestion à distance , [1].

I-2-3 Point de vue économique et social

Les systèmes automatisés ont un impact économique très important dans le monde industriel (gain de temps du à la robotisation, meilleure productivité, rendement optimisé...), mais aussi ils amènent des incidences sociales fortes autant, pour le travailleur (évolution des métiers, responsabilité accentuée, sécurité renforcée, imagination accrue, adaptabilité par la polyvalence), que pour le citoyen (amélioration des tâches ménagères, communication variées, commande à distance...), [1].

I-3 Automatisation

L'automatisation de la production consiste à transférer tout ou partie des tâches de coordination, auparavant exécutées par des opérateurs humains, dans un ensemble d'objets techniques appelé partie commande.

La Partie Commande mémorise le savoir faire des opérateurs pour obtenir la suite des actions à effectuer sur les matières d'œuvre, afin d'élaborer la valeur ajoutée.

Elle exploite un ensemble d'informations prélevées sur la Partie Opérative pour élaborer la succession des ordres nécessaires pour obtenir les actions souhaitées, [3].

I-3-1 Objectifs de l'automatisation

L'automatisation permet d'apporter des éléments supplémentaires à la valeur ajoutée par le système. Ces éléments sont exprimables en termes d'objectifs par :

- Accroître la productivité du système c'est-à-dire augmenter la quantité de produits élaborés pendant une durée donnée. Cet accroissement de productivité exprime un gain de valeur ajoutée sous forme :

- . D'une meilleure rentabilité ;
- . D'une meilleure compétitivité.

- améliorer la flexibilité de production,

- améliorer la qualité du produit grâce à une meilleure répétitivité de la valeur ajoutée,

- s'adapter à des contextes particuliers :

- . Adaptation à des environnements hostiles pour l'homme (milieu salin, spatial, nucléaire...) ;
- . Adaptation à des tâches physiques ou intellectuelles pénibles pour l'homme (manipulation de lourdes charges, tâches répétitives parallélisées...)

- augmenter la sécurité, etc...

D'autres objectifs, à caractères sociaux, financiers... peuvent s'ajouter à ceux-ci, [4].

I-3-2 Conduite et surveillance d'un système automatisé

Il s'avère très difficile en pratique d'intégrer dans une Partie Commande la totalité des savoir-faire humains de sorte que l'automatisation reste souvent partielle : certaines tâches restent confiées à des intervenants humains.

A ces causes techniques viennent s'ajouter des considérations économiques de compétitivité, des considérations financières imposant un fractionnement des investissements, des considérations sociales d'automatisation douce.

Certaines tâches restent donc manuelles et l'automatisation devra donc prendre en compte la spécificité du travail humain, c'est-à-dire en particulier :

- assurer le dialogue entre les intervenants et le système automatisé,
- assurer la sécurité de ces intervenants dans l'exécution de leurs tâches manuelles.

En outre le modèle de fonctionnement de la Partie Commande, choisi par le concepteur du système, ne correspond qu'à un ensemble de situations prévues, c'est-à-dire retenues par le concepteur parmi un ensemble de situations possibles.

Or il est impératif de pouvoir faire face à des situations non prévues (donc non retenues en général pour des raisons économiques compte tenu de leur faible probabilité), voire imprévisibles.

Un seul opérateur peut alors intervenir et prendre les décisions requises par cette situation :

Il assure une fonction de conduite et de surveillance du système automatisé. Cette fonction peut être plus ou moins assistée par un ensemble de moyens (pupitres, informatique...).

Le concepteur devra alors :

- fournir à l'intervenant (ou lui permettre de prélever) toutes les informations significatives (ou indices) nécessaires à l'analyse de la situation,
- lui permettre d'agir sur le système, soit directement (dépannage..), soit indirectement (consignes de sécurité, de marches et d'arrêts...), [2].

I-3-3 Structure d'un système automatisé

Chaque système automatisé comporte deux parties ,(Fig.I.1).

- une Partie Opérative (PO) dont les actionneurs agissent sur le processus automatisé ;
- une Partie Commande (PC) qui coordonne les actions de la Partie Opérative : [5]

I-3-3-1 Partie Opérative

C'est elle qui opère sur la matière l'œuvre et le produit. Elle comporte en général, [5]:

- des outillages et moyens divers mettant en œuvre le processus d'élaboration, par exemple, moules, poinçons, outils de coupe, pompes, têtes de soudure, de marquage,... ;
- des actionneurs destinés à mouvoir ou mettre en œuvre ces moyens, par exemple :

- moteur électrique pour actionner une pompe ;
- vérin hydraulique pour fermer un moule ;
- vérin pneumatique pour mouvoir une tête de marquage.....

I-2-3-2 Partie Commande

C'est elle qui émet des ordres vers la Partie Opérative et en reçoit les signaux en retour, afin de coordonner ses actions. De plus en plus réalisée avec des technologies de traitement programmables, la Partie Commande est plus particulièrement l'objet du présent ouvrage.

Au centre de la partie commande, le « traitement » est la convergence de 3 dialogues qu'il coordonne, [5, 10]:

1) Le dialogue avec la machine

Commande des actionneurs (moteurs, vérins) via les pré actionneurs (contacteurs, distributeurs, variateur...) ; acquisition des signaux en retour par les capteurs qui rendent compte de l'évolution de la machine), [5, 10].

2) Le dialogue homme-machine

Pour exploiter, régler, dépanner la machine, le personnel émet des consignes et reçoit des informations en retour, [5, 10].

3) Le dialogue avec d'autres machines

Plusieurs machines peuvent coopérer dans une même production. Leur coordination est assurée par le dialogue entre leurs Parties Commandes. L'illustration ci-dessous, schématise l'organisation de la Partie Commande, par rapport à la Partie Opérative), [5].

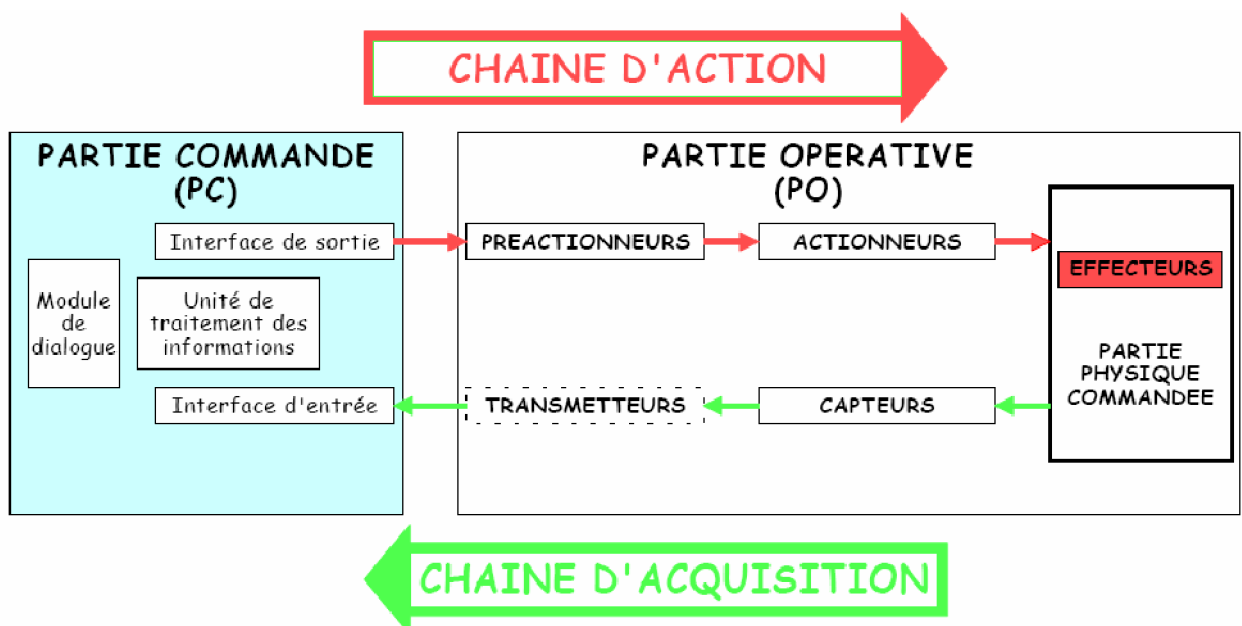


Figure I. 1 :Structure d'un système automatisé.

I-4 Système automatisé de production

Un système automatisé de production SAP (notion assez large qui inclut des systèmes de contrôle, de conditionnement, d'analyse....) reçoit un flux de matière ou de produits génère un flux de produits plus élaborés (moulés, usinés, assemblés, testés, ..etc.). Il doit aussi gérer l'alimentation en énergie ainsi que des flux auxiliaires tels les consommables. Les déchets, tout en minimisant les stocks initiaux, finaux et intermédiaires. Tout cela, ajouté à des exigences sans cesse accrues de qualité, sécurité, flexibilité entraîne un accroissement des besoins, en particulier la manipulation d'un grand nombre de variables et la gestion de véritables flux de communication, [8, 9].

Cela explique que les systèmes câbles deviennent trop volumineux et trop rigides pour de telles applications, et que l'on se tourne donc vers des solutions utilisant les techniques de traitement de l'information par processeurs programmables. La solution reposant sur un processeur central unique s'étant vite révélée peu économique du point de vue câbles, complexe et interconnectés s'est aujourd'hui largement imposée.

L'architecture décentralisée qui en résulte facilite la conception et l'installation en permettant de fractionner les études, la mise en place, les tests; elle améliore aussi la maintenance (modification aisée des programmes, de parties du système automatisé) et se traduit par plus de flexibilité et de disponibilité. Elle entraîne toutefois, du fait des multiples sous – ensembles fonctionnels, un fort accroissement des besoins de communication et de gestion.

La figure I.2 : Illustre le fonctionnement d'un SAP, tel que l'on en rencontre notamment, mais pas exclusivement, dans l'industrie manufacturière (chaînes d'usinage, de montage, de conditionnement), [6].

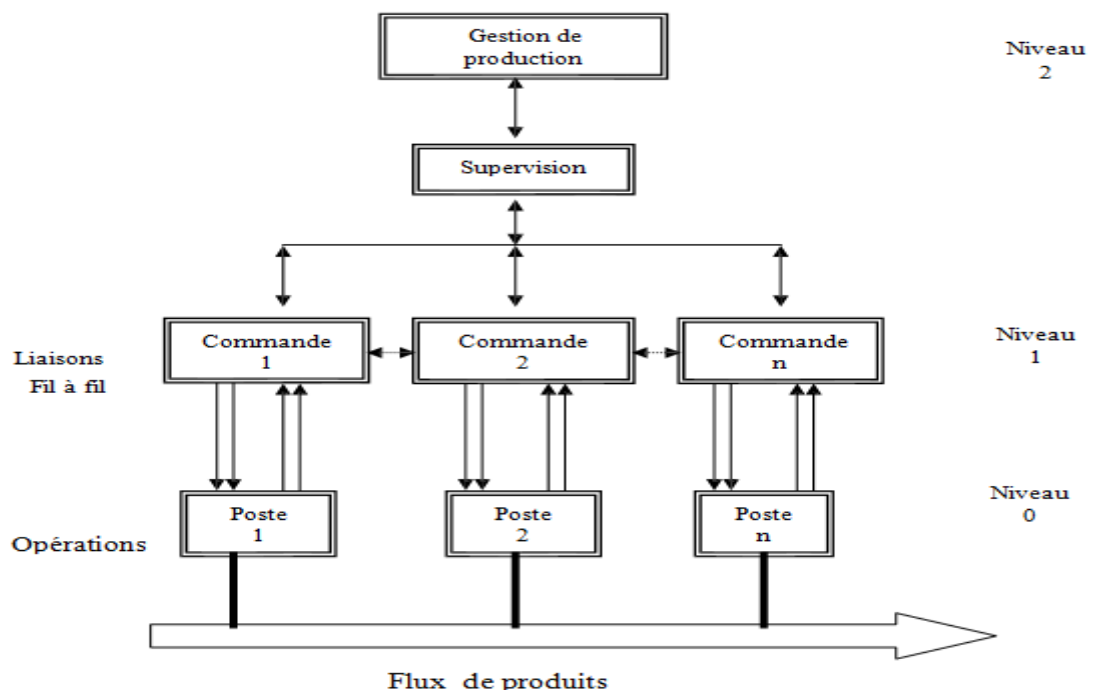


Figure I. 2:Système automatisé de production.

I-4-1 Objectifs de l'automatisation des productions

Les productions industrielles sont de plus en plus automatisées. Ces progrès concernent :

- l'automatisation d'opérations autrefois entièrement manuelles, par exemple, les assemblages, les contrôles,.... ;
- l'automatisation plus poussée d'opérations déjà partiellement automatisées, par exemple :
 - le passage en automatique de machines semi-automatiques,
 - le remplacement de machines rigides (ne fabriquant qu'un seul type de produit) par des machines flexibles susceptibles d'opérer sur plusieurs variantes de produits.

Les objectifs poursuivis par une automatisation peuvent être assez variés. Que cette liste soit limitative :

- la recherche de coûts plus bas pour le produit, par réduction des frais de main-d'œuvre, d'économie de matière, d'économie d'énergie,.... ;
- la suppression des travaux dangereux ou pénibles, et l'amélioration des conditions de travail par l'ennoblissement des tâches,..... ;
- la recherche d'une meilleure qualité du produit, en limitant le facteur humain, et en multipliant les contrôles automatisés,..... ;
- la réalisation d'opérations impossibles à contrôler manuellement ou intellectuellement, par exemple des assemblages miniatures, des opérations très rapides, des coordinations complexes, [4].

I-4-2 Acquisition des données

L'automatisation d'une machine ou d'une installation nécessite la prise en compte permanente des informations de commande, de position, de température, et de vitesse.

Ces informations sont transmises à la partie commande de la machine sous forme de messages codés peuvent être de nature:

- Binaire: L'information est transmise en tout ou – rien (TOR), il est utilisé dans les automatisations simples.
- Numérique: L'information est transmise sous la forme d'un code binaire par un mot de plusieurs bits.
- Analogique: L'information est transmise sous forme de tension ou courant proportionnelle à la grandeur mesurée et évolue entre deux valeurs limites, [5].

I-5 Technologies des automatismes

Nous disposons de nombreux outils technologiques pour réaliser l'organe de commande de notre système que l'on regroupe habituellement en deux catégories fondamentales :

Les solutions câblées (fig.I.3) et les solutions programmées (fig.I.4). Le tableau (I.1) illustre une comparaison entre ces deux solutions, [7]:

Logique Câblée	Logique Programmable
Toutes les fonctions technologiques (électromagnétiques, électroniques, pneumatiques...), sont câblées entre elles à partir d'un diagramme à caractère booléen.	Le programme est établi à partir d'un langage compris puis rangé suivant un ordre dans les mémoires. Ce programme est divisé en une succession d'ordre. Les fonctions logiques ne sont pas dépendantes du câblage mais sont reliées entre elles par un câblage binaire.
Faible capacité	Grâce à l'API, l'adaptation à une nouvelle fabrication est rapide.
Gros frais de reconversion des installations.	Moins coûteuse.
Un automate à base d'une logique câblée est prévu pour des productions industrielles de mode répétitif.	L'automate étant ; modifiable à volonté, il est programmable.
L'encombrement, le poids et le volume des composants sont des contraintes majeurs lorsqu'il s'agit de maîtriser des applications plus au moins complexes.	La flexibilité, la souplesse vis-à-vis de la mise au point de commande et de l'évolution de celles-ci et la performance font de la logique programmée, une solution préférable.
Connexion figée, valable pour une seule application	Toutes les connexions sont en soft (logique).
Longueur de câble utilisée très importante	Gain en longueur de câble électrique.
Mise au point difficile.	Diagnostic rapide : (recherche d'éventuelles pannes).
Prix de revient (installation) coûteux	Gain important en temps : (amélioration du taux de marche).

Tableau I. 1: Comparaison entre la solution câblée et la solution programmée.

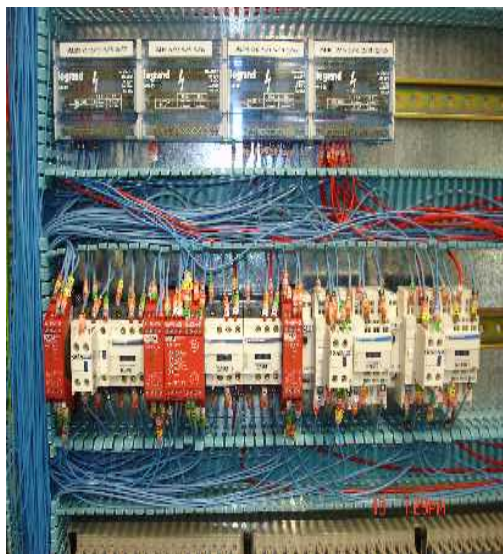


Figure I. 3: Armoire de commande basée sur logique Câblé .



Figure I. 4: Armoire de commande basée sur la logique programmée.

I-6 Dialogue homme –machine

Le dialogue homme machine est le complément indispensable de tout l'automatisme, il permet à l'opérateur d'intervenir au moment de démarrage ou en cour de cycle, de procéder à un arrêt d'urgence et par l'intermédiaire du système de signalisation, de contrôler en permanence le déroulement des opération.

Cette fonction de dialogue est assurée par tous les auxiliaires de commande à intervenir humaine (boutons , boîtes à boutons , commutateurs) ainsi que par des voyants de signalisation et pour les installation plus complexes , par des pupitres de commande, des tableaux synoptiques, des boutons, [4].

I-7 protection du système

Les appareils de protection assurent le bon fonctionnement d'une installation ou d'une machine et la sécurité des personnes contre tout dysfonctionnement d'origine, [4]:

- Electrique: surintensité (surcharge court-circuit),surtension,
- Pneumatique : surpression,
- Hydraulique pneumatique : surpression.

I-8 Champ d'application de l'automatisme

L'automatisme consiste en l'étude de la commande de système industriel, la première amélioration de travail a été de remplacer l'énergie humaine fournie par l'ouvrier par une machine (partie opérative).

L'opérateur commande la machine, et servile le résultat obtenu, il adapte ses commandes en fonction du déroulement du processus. L'automate d'étude lorsque l'on intercale entre l'opérateur et la partie opérative.

Une partie commande, qui prend des décisions (gestion automatique des cas les plus simple et les plus courants).

La partie commande se base sur les informations reçues par les capteurs à fin de piloter les actionneurs.

Le but est de prendre en compte tout ce qui est simple, complexe et répétitive, en laissant à l'opérateur les tâches nobles de réflexion.

Exemple : la partie commande doit nécessairement avoir toute information. Où devrait passer par elle.

Il reste néanmoins phénomènes difficiles à mesurer, ou dont la mesure coûte trop cher par rapport à la probabilité qu' ils se produisent, ou non prévus pour cela, l'opérateur contrôler reste nécessaire, [5].

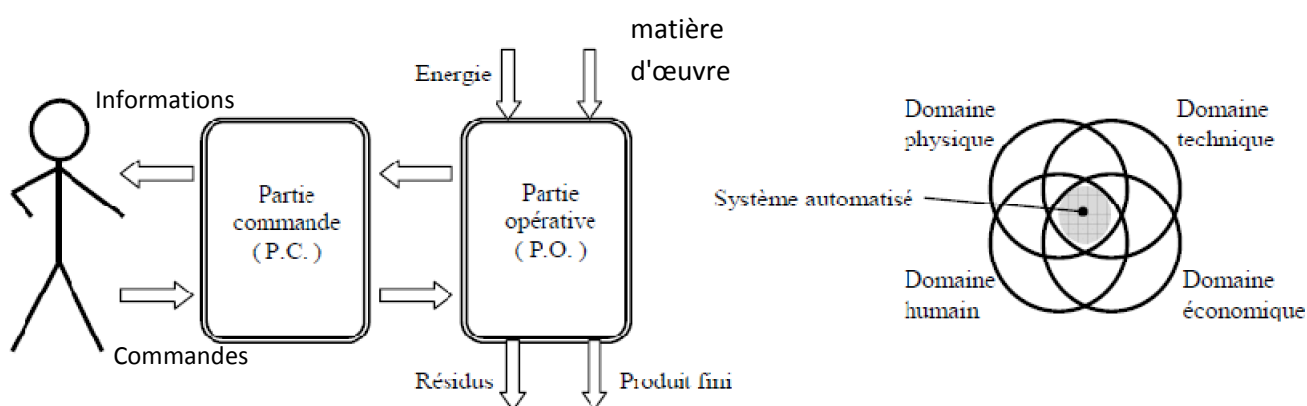


Figure I. 5: Champ d'application de l'automatisme.

I-9 Les automates programmables industriels

I-9-1 Historique

Les automates programmables industriels sont apparus à la fin des années soixante, à la demande de l'industrie automobile américaine, qui réclamait plus d'adaptabilité de leurs systèmes de commande.

Les coûts de l'électronique permettant alors de remplacer avantageusement

les technologies actuelles.

Avant : utilisation de relais électromagnétiques et des systèmes pneumatiques pour la réalisation des Parties commandes \iff Logique câblée.

Après : utilisation de systèmes à base de microprocesseurs permettant une modification aisée des systèmes automatisés \iff Logique programmée.

Les ordinateurs de l'époque étant chers et non adaptés aux contraintes du monde industriel, les automates devaient permettre de répondre aux attentes de l'industrie, [1].

I-9-2 Définition

Un automate est une machine que l'on programme comme on veut et qui limite les décisions systématiques d'un homme. Un Automate Programmable Industriel (API) est une commande conçue autour d'un microprocesseur. Sa conception et son langage de programmation sont spécialement adaptés aux contrôles des processus industriels.

Les caractéristiques d'un automate programmable fixent les limites de ses Performances qui sont, [7] :

- La capacité d'entrées / sorties;
- La capacité de programme;
- La capacité de mémorisation de données;
- La capacité de mémorisation de textes;
- Le temps de cycle et le temps de réaction;
- Les possibilités du jeu d'instructions;
- Les possibilités de communication.

I-9 -3 Les Objectifs

Les Objectifs d'automate programmable sont :

- Simplifier le travail de l'homme qui, libéré de la machine, peut se consacrer à des activités plus qualifiantes;
- Améliorer la productivité des installations en augmentant leur autonomie de fonctionnement ;
- Renforcer la qualité par une surveillance permanente des productions et augmenter la sécurité des installations;
- Réaliser, notamment grâce au développement des techniques programmables, de l'installation plus souple, plus flexible d'emploi et susceptible de former des productions différentes par un simple changement de programme.

En effet, l'automatisme peut intervenir à tous les stades d'opérations industrielles, dans des domaines aussi divers que les industries de transformations, de fabrication ou de transport, dans les machines-outils ainsi que dans la vie quotidienne (systèmes d'alarmes, automatisation de parking...), [6].

I-9-4 Fonctionnement d'un automate programmable

Lorsque le programme est introduit dans la mémoire de l'automate en utilisant soit une console de programmation, soit un PC, la phase d'exécution est alors possible. Elle est généralement obtenue après la mise en RUN de l'automate (cette commande peut être logicielle ou matérielle).

L'automate programmable reçoit les informations relatives à l'état du système et puis commande

Les pré-actionneurs suivant le programme inscrit dans sa mémoire. Généralement les automates programmables industriels ont un fonctionnement cyclique (fig.I.6), [6].

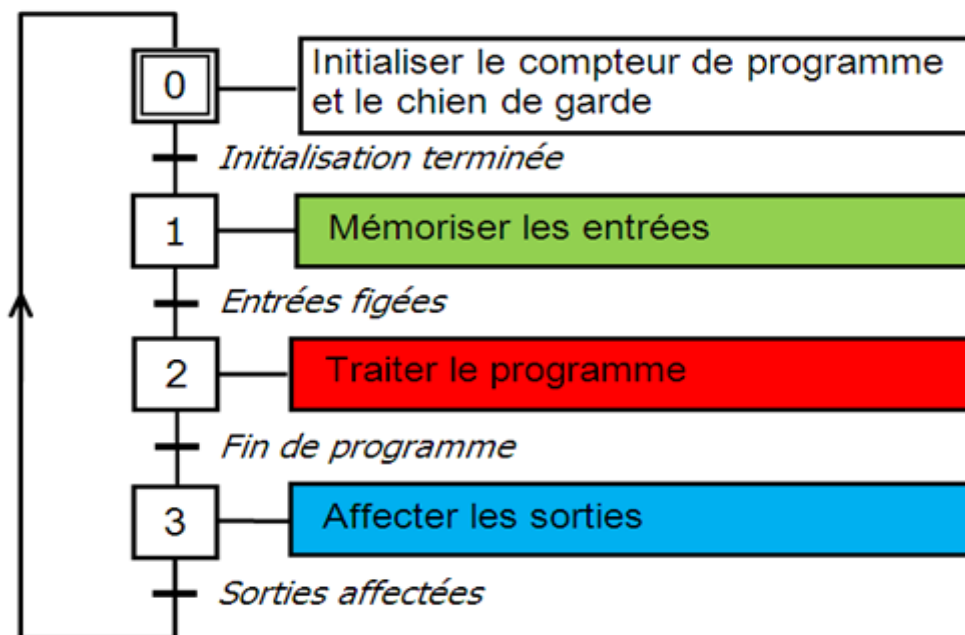


Figure I. 6: Fonctionnement cyclique d'un API.

I-9-5 Domaines d'emploi des automates

Pour les raisons qui viennent d'être évoquées, les API s'adressent à des applications que l'on trouve dans la plupart des secteurs industriels :

- **Métallurgie et sidérurgie** : pour les problèmes de gaz, de contrôle de qualité.
- **Mécanique et automobile** : commande de machines-outils, robots, machines de soudage.
- **Industries chimiques** : pilotage d'unités de production, dosage et mélange de produits.

- **Industries pétrolières** : commande et la surveillance des pipe-lines, ou affectés aux parcs de changement et la distribution des gaz et des liquides.
- **Industries agricoles et alimentaires** : aux postes de mélanges et de contrôle des produits.
- **Transports et manutention** : tri de paquets, de courrier, gestion mécanisée de parcs de stockage, emballage.

L'application des API sont des Application diverses dans toute les domaines industrielles.

I-9-6 Structure générale d'un automate programmable

Les éléments principaux que l'on rencontre toujours dans un API sont l'alimentation, l'unité centrale CPU (central processing unit), la mémoire, les modules d'entrées / sorties et la console de programmation (fig.I.7), [6].

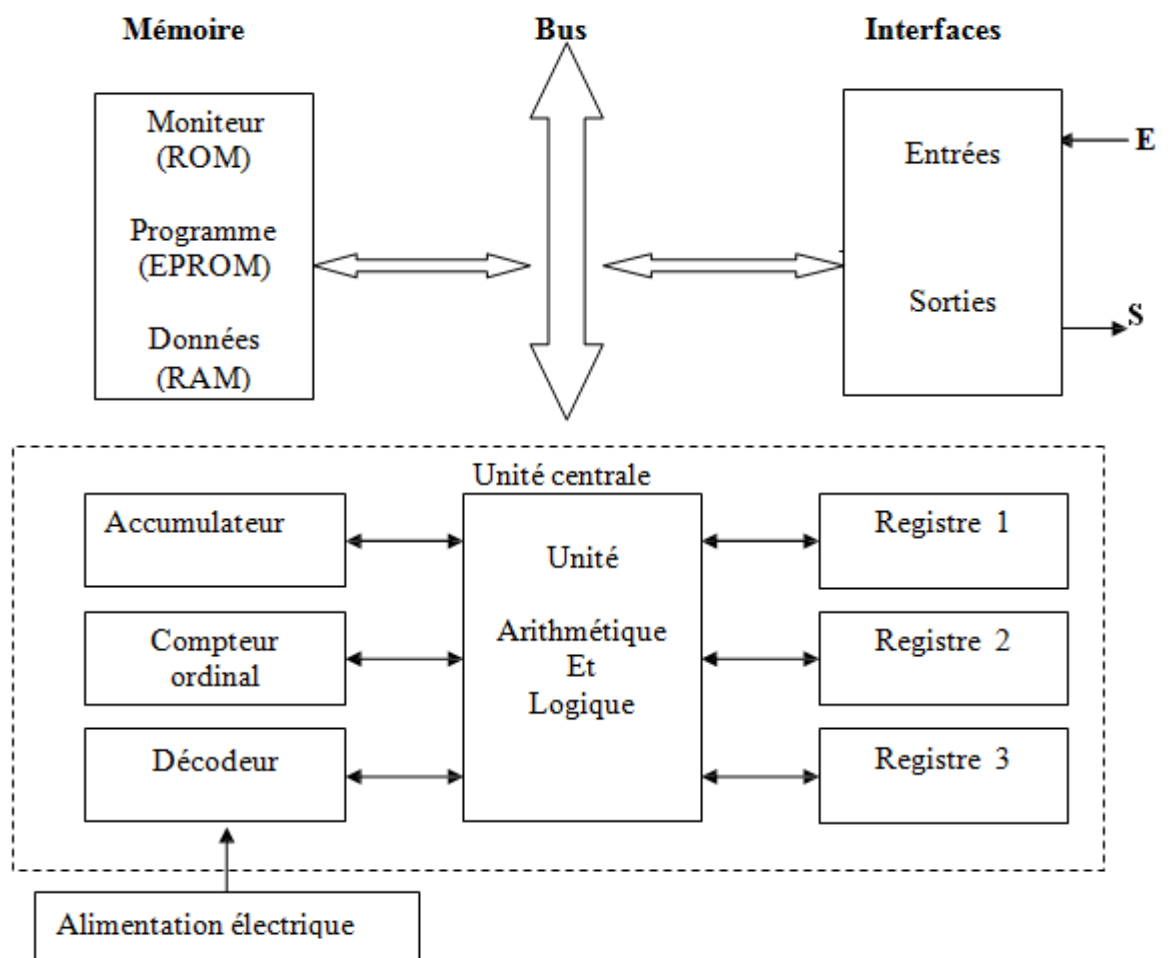


Figure I. 7: Fonctionnement cyclique d'un API.

Un API se compose de trois parties : l'unité centrale, les interfaces et l'alimentation.

I-9-6-1 L'unité centrale

C'est l'ensemble des dispositifs nécessaire en fonctionnement logique interne de l'API, c'est le cœur l'automate, [7].

I-9-6-1-1 Architecteur de l'unité centrale (UC)

L'unité centrale comporte la zone mémoire et le processeur (fig.I.8), [7].

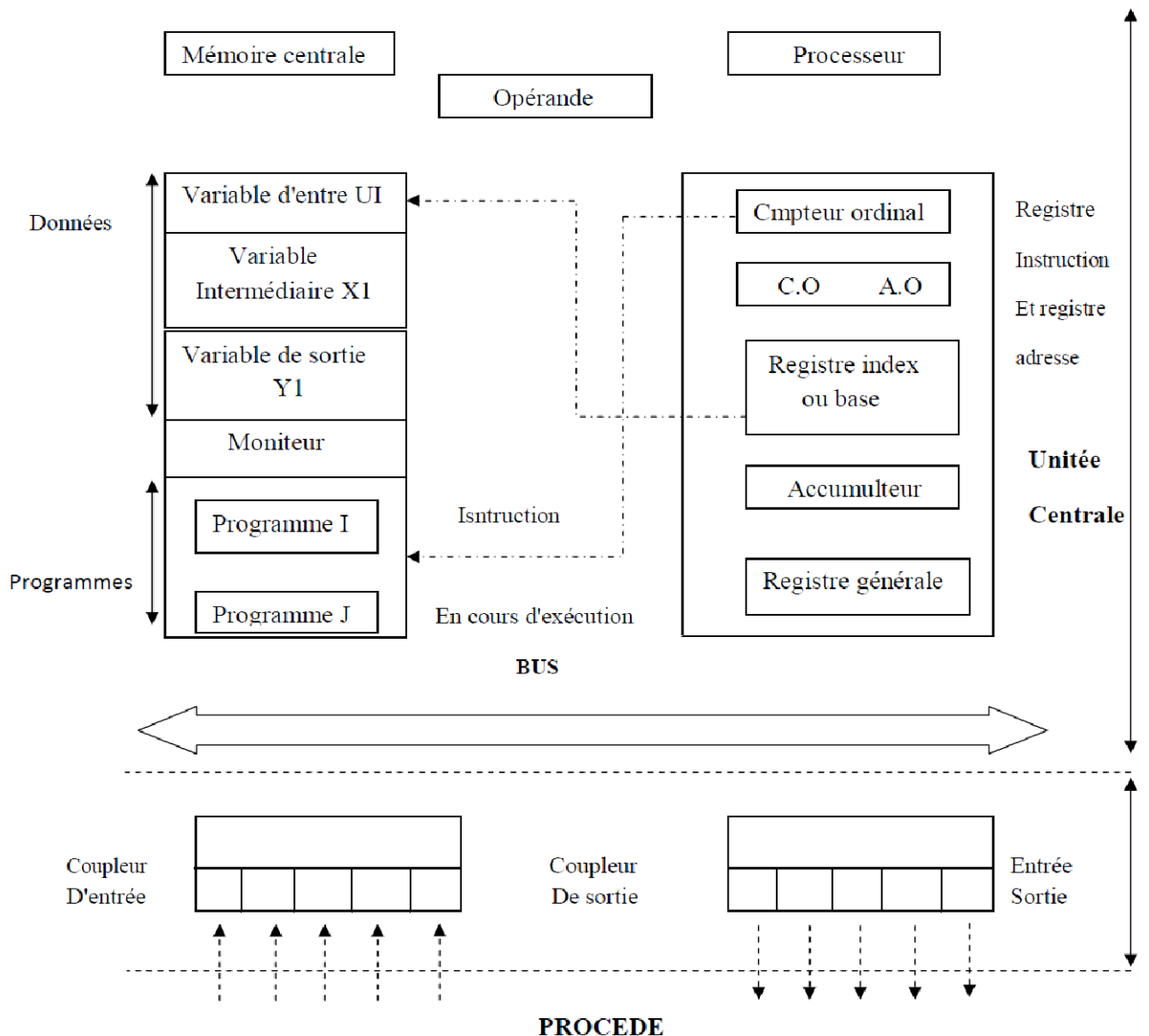


Figure I. 8: Fonctionnement cyclique d'un API.

I-9-6-1-2 Processeur

Cœur de l'appareil, dans l'unité central, ce n'est sans doute pas, paradoxalement, le point le plus caractéristique, mais il conditionne tout de même largement les performances.

Les premières API étaient équipées de processeurs spécifiques, à cycle de scrutation unique : on exécutait en permanence un programme gérant essentiellement des variables binaires. On est passé ensuite à des processeurs plus performants, issus du monde de l'informatique. Cette évolution a permis de baisser les coûts, d'accroître les possibilités opérationnelles. Les processeurs <généralistes> étant toutefois ponctuellement moins efficaces que les spécialisés, on peut penser que c'est aussi une des raisons pour laquelle les progrès (en temps de calcul par exemple) sur un ensemble d'opérations de base ont été moins spectaculaires qu'en informatique générale, avec une conséquence heureuse pour les utilisateurs : une longévité supérieure du matériel. Des progrès importants ont été accomplis sur d'autres points, sur lesquels nous reviendrons.

L'unité central UC est une carte électronique bâtie autour de la (ou des) « puce(s) » processeur(s), qui assure au moins les fonctions suivante : [7].

- opérations logique sur bits (le bit, contraction de « binary digit », étant l'information élémentaire à deux états) ou sur mots (ensemble de bits, le plus souvent 16 pour les API) ;
- temporisation et comptage.

Il existe trois technologies de réalisation :

- la technologie câblée, la plus rapide mais aussi la plus coûteuse, réservée à des usages particuliers;
- la technologie à microprocesseur, la plus économique dès lors que l'on utilise un microprocesseur standard produit en grand série ;
- la technologie mixte, certaine opération étant réalisée en câble pour en accroître la rapidité.

I-9-6-2 Module d'alimentation

Il assure la distribution d'énergie aux différents modules.

- **Le bus interne** : il permet la communication de l'ensemble des blocs de l'automate et des éventuelles extensions, [7].

- **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 type PCMCIA

A l'intérieur de l'automate il y a un bloc d'alimentation fournissant les différentes tensions nécessaires aussi bien à la logique de commande (5 V à ± 15 V) qu'aux entrées sorties (24 V à 220 V). La tension normalisée la plus courante est de 24 V (alternatif ou continu).

Généralement l'alimentation des capteurs est fournie par l'API (en 24 V) alors que celle des prés actionneurs est fournie par une alimentation externe. En courant continu, la tension

des signaux de sortie de l'API sont de l'ordre de 24, 48 ou 110 V, alors que l'intensité du courant varie entre 0.5 et 2A. En alternatif 50HZ, elles peuvent aller jusqu'à 220 V et 5A respectivement, [7].

I-9-6-3 Modules d'entrées/sorties (E/S)

Ils assurent le rôle d'interface de la partie commande (PC) dans le schéma classique de la figure 3, qui distingue une partie opérative, où les actionneurs agissent physiquement sur le processus, et une partie commande récupérant les informations sur l'état de ce processus et coordonnant en conséquence les actions pour atteindre les objectifs prescrits (matérialisés par des consignes).

Pour ce faire, ils doivent :

- regrouper des variables de même nature, pour diminuer complexité et coût ;
- assurer le dialogue avec l'UC ;
- « traduire » les signaux industriels en informations API et réciproquement, avec une protection de l'UC et un traitement adéquats. Beaucoup d'automates assurent cet interfaçage par des modules amovibles comportant un nombre fixé de voies d'un certain type (fig I.7) ; [7].

- Alimentation électrique.

Exemple : on pourra trouver un module « 16 entrées TOR 220 V alternatif ».

- **Interface d'entrée** : elle permet de recevoir les informations du S.A.P. ou du pupitre et de mettre en forme (filtrage,...) ce signal tout en l'isolant électriquement , [7].

- **Interfaces de sortie** : elle permet de commander les divers pré actionneurs et éléments de signalisation du S.A.P. tout en assurant l'isolement électrique (fig.I.9), [7].

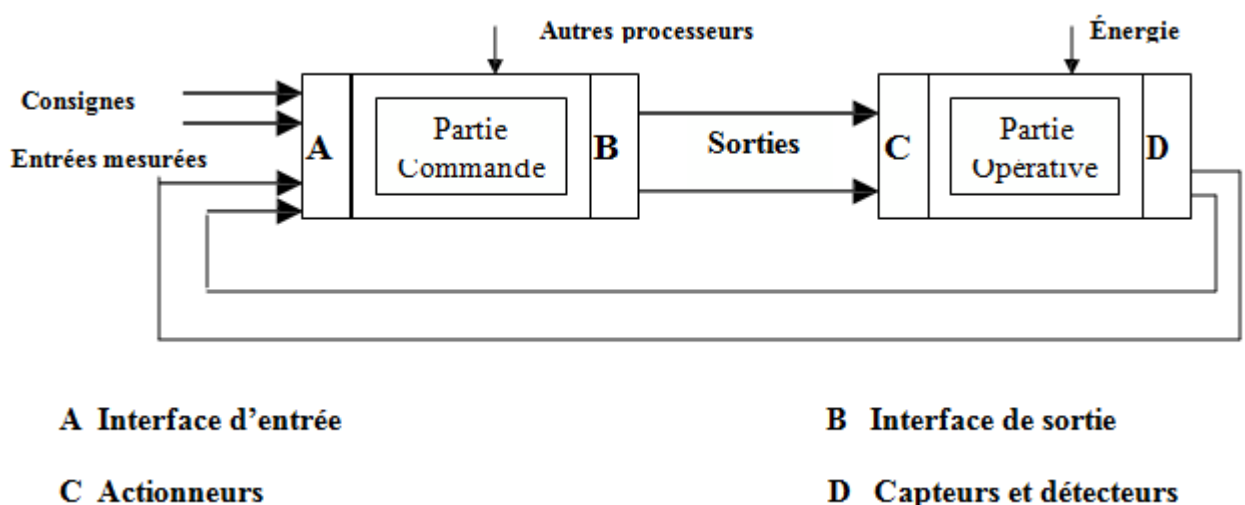


Figure I. 9: Partie opérative et partie commande.

I-9-7 Langages de la programmation

Il y a cinq langages de programmation principaux : [2].

- sequential function chart (SFC) ou GRAFCET pour les opérations de haut niveau ;
- function block diagram (FBD) ou logigramme pour les opérations cycliques complexes ;
- ladder diagram (LD) ou schéma à contacts pour les opérations cycliques booléennes ;
- structured text (ST) pour les opérations cycliques ;
- instruction list (IL) pour les opérations de bas niveau.

I-9-7-1 Programmation à l'aide du GRAFCET (SFC: Sequential Function Chart)

Le langage SFC (sequential function chart), ou GRAFCET, est un langage graphique utilisé pour décrire les opérations séquentielles. Le procédé est représenté comme une suite connue d'étapes (états stables), reliées entre elles par des transitions. Une condition booléenne est attachée à chaque transition.

Les actions dans les étapes sont décrites avec les langages ST, IL, LD ou FBD (figI.10), [2].

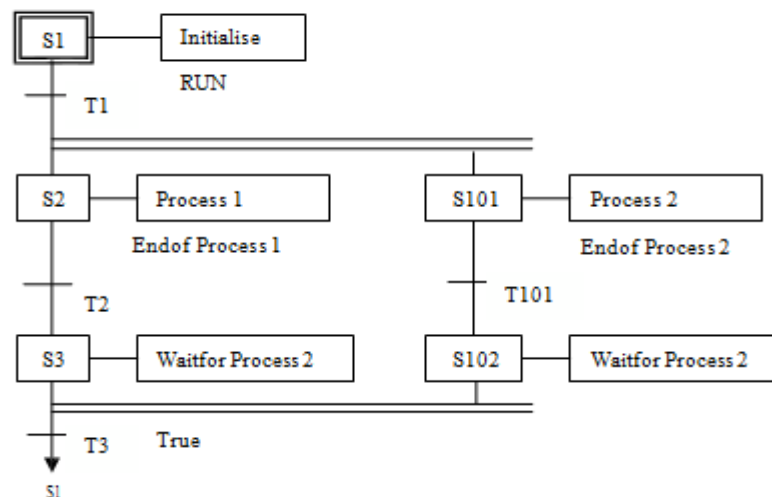


Figure I. 10: Exemple Programme SFC avec divergence et convergence doubles.

On peut également traduire un Grafcet en langage en contacts et l'implanter sur tout type d'automate. Il existe d'autres langages de programmation des automates qui sont normalisés au plan mondial par la norme CEI 61131-3.

I-9-7-2 Langage à contacts (LD : Ladder Diagram)

Le langage LD (ladder diagram) 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. Il est développé pour les électriciens. Il utilise les symboles tels que : contacts, relais et blocs fonctionnels et s'organise en réseaux (labels). C'est le plus utilisé, (figI.11), [2].

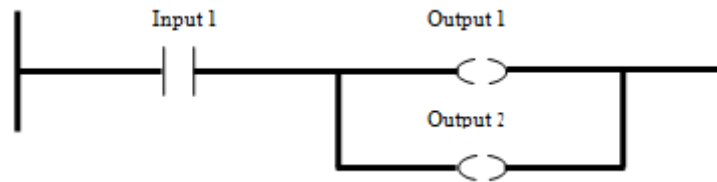


Figure I. 11: Exemple Programme LD avec de liaison multiple à droite.

I-9-7-3 Langage Blocs Fonctionnels (FBD : Function Bloc Diagram)

Le langage FBD (function block diagram) est un Langage graphique où des fonctions sont représentées par des rectangles avec les entrées à gauche et les sorties à droite. Les blocs sont programmés (bibliothèque) ou programmables. Utilisé par les automaticiens (figI.12), [2].

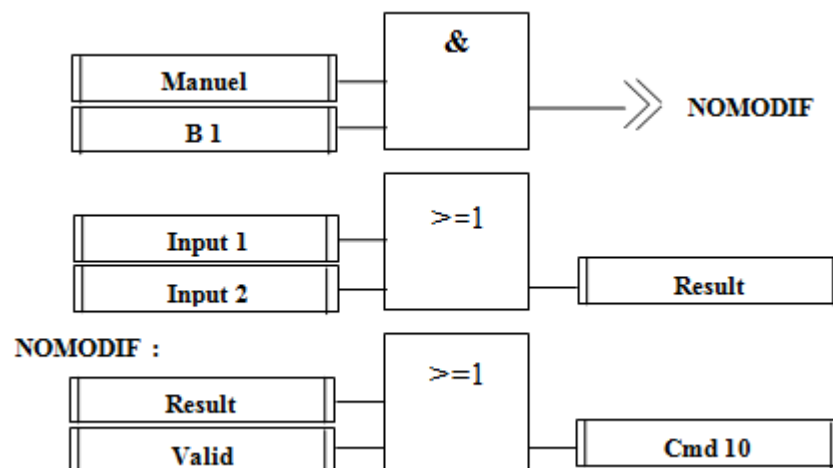


Figure I. 12: Exemple Programme FBD avec de saut et d'étiquette.

I-9-7-4 Langage ST

Le langage ST (structured text) 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 associées aux transitions du langage SFC. (Fig.I .13), [2].

```

Q: =false;
      CV: =0
      RETURN ;(*fin du programme*)
      End_if;
If RESET then
      CV: =0;
      Else
      If (CV<PV) then
      CV: = CV+1;
      End_if;
      End_if ;
      Q: = (CV>=PV);

```

Figure I. 13: Exemple Programme ST.

I-9-7-5 Le langage IL

Le langage IL (instruction list), 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 le type d'opération à effectuer entre le résultat courant et l'opérande. Le résultat de l'opération est stocké à son tour dans le résultat courant (fig. I 14), [2].

Etiquette	Opérateur	Opérande	Commentaire
Début :	LD	IX 1	(*bouton poussoir*)
	ANDN	MX 5	(*Commande valide*)
	ST	QX 2	(*lance moteur*)

Figure I. 14: Exemple Programme IL.

I-10 L'environnement de l'API

Pour conduire sa mission vis-à-vis d'un procédé, un API doit disposer des liaisons connectée Temporairement ou non, directement ou non, à l'API. Ces dispositifs peuvent être, par exemple, les périphériques et les auxiliaires des API (l'imprimante, les mémoires de masse...), les consoles d'exploitation, etc.... (fig.I.15), [6].

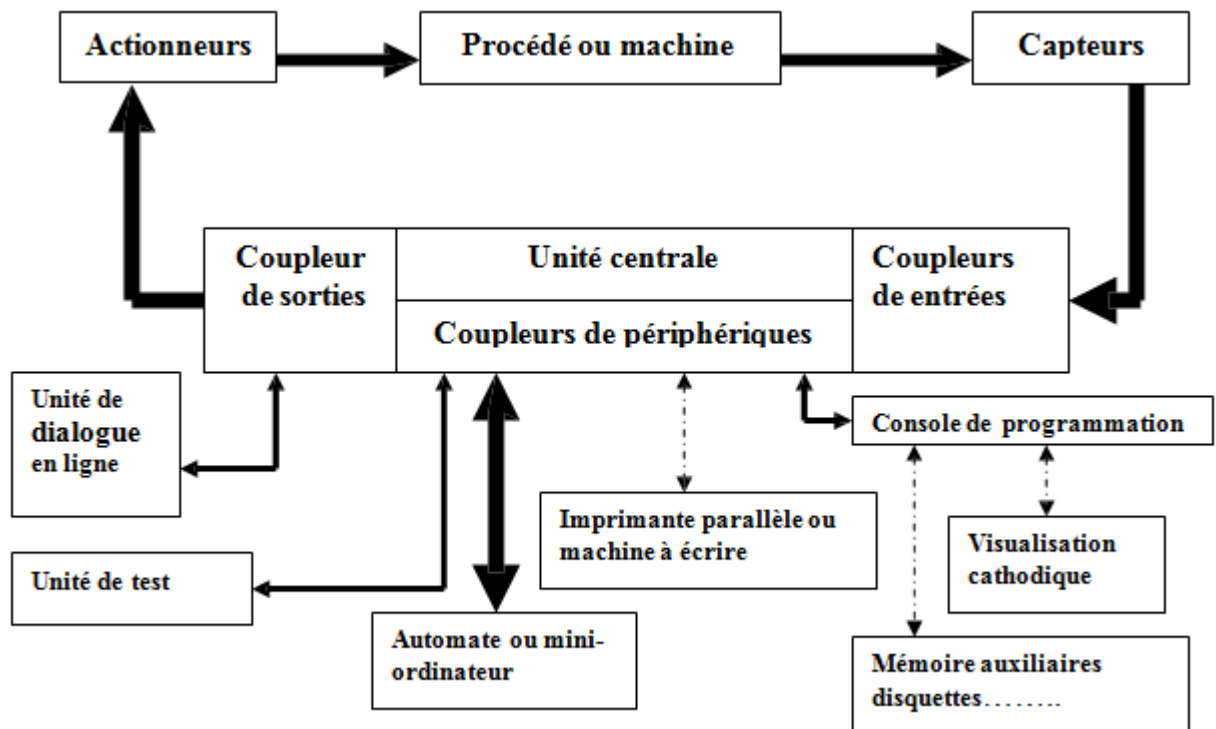


Figure I. 15: L'environnement de l'API.

I-10 Conclusion

L'automatisation s'est généralisée à l'ensemble des activités de production, tant dans l'industrie, que les activités de services.

Quelque soit son domaine d'application et les technique auxquelles elle fait appel, l'automatisation s'est constamment développé dans l'unique but de réduire la pénibilité du travail humain et d'améliorer la productivité du travail.

Les besoins de l'automatisation se caractérise par :

- * Un développement massif;*
- * Une approche de plus en plus globale des problèmes;*
- * Une intégration dès la conception de l'installation des automates programmables, on est ainsi passé du stade de la machine automatisée à celui du système automatisé de production.*

Les fabricants d'automatismes industriels étant nombreux, l'un des plus développés dans ce domaine, à savoir SIEMENS.

Etant donné que le logiciel STEP7 permet l'accès "de base" aux automates Siemens., nous avons jugé utile de lui consacré le deuxième chapitre.

Chapitre II

DESCRIPTION DU LOGICIEL STEP7

Chapitre II : description du logiciel STEP7

II-1 introduction :

STEP7 est le progiciel de base pour la configuration et la programmation de systèmes d'automatisation SIMATIC S300 et S400. Il fait partie de l'industrie logicielle SIMATIC. Le logiciel de base assiste dans toutes les phases du processus de création de la solution d'automatisation, La conception de l'interface utilisateur du logiciel STEP7 répond aux connaissances ergonomiques modernes.

Dans ce chapitre on va donner une description du logiciel STEP7, ces langages de programmations, et les différents blocs qui lui compose.

II-2 Gestionnaire de projet SIMATIC manager



Figure II. 1: L'icone du logiciel STEP7.

SIMATIC Manager constitue l'interface d'accès à la configuration et à la programmation. Ce gestionnaire de projets présente le programme principal du logiciel *STEP7* il gère toutes les données relatives à un projet d'automatisation, quelque soit le système cible sur lequel elles ont été créées. Le gestionnaire de projets *SIMATIC* démarre automatiquement les applications requises pour le traitement des données sélectionnées.

II-2-1 Editeur de programme et les langages de programmation

Les langages de programmation CONT, LIST et LOG, font partie intégrante du logiciel de base.

- **Langage CONT :** Le schéma à contacts (CONT) est un langage de programmation graphique. La syntaxe des instructions fait penser aux schémas de circuits électriques.

Le langage CONT permet de suivre facilement le trajet du courant entre les barres d'alimentation passant par les contacts, les éléments complexes et les bobines, [11].

- **Langage LIST :** La liste d'instructions (LIST) est un langage de programmation textuel proche de la machine. Dans un programme LIST, les différentes instructions correspondent, dans une large mesure, aux étapes par lesquelles la CPU traite le programme, [11].
- **Langage LOG :** Le logigramme (LOG) est un langage de programmation graphique qui utilise les boîtes de l'algèbre de Boole pour représenter les opérations logiques. Les fonctions complexes, comme par exemple les fonctions mathématiques, peuvent être représentées directement combinées avec les boîtes logiques, [11].

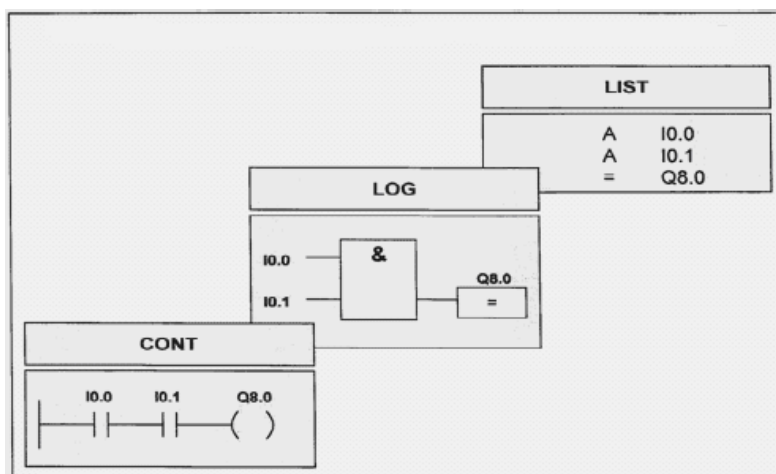


Figure II. 2: Mode de représentation des langages basiques de programmation STEP7.

II-2-2 Simulateur des programmes PLCSIM

L'application de simulation de modules S7-PLCSIM permet d'exécuter et de tester le programme dans un Automate Programmable (AP) qu'on simule dans un ordinateur ou dans une console de programmation. La simulation étant complètement réalisée au sein du logiciel STEP7, il n'est pas nécessaire qu'une liaison soit établie avec un matériel S7 quelconque (CPU ou module de signaux). L'AP S7 de simulation permet de tester des programmes destinés aux CPU S7-300 et aux CPU S7-400, et de remédier à d'éventuelles erreurs, [12].

S7-PLCSIM dispose d'une interface simple permettant de visualiser et de forcer les différents paramètres utilisés par le programme (comme, par exemple, d'activer ou de désactiver des entrées). Tout en exécutant le programme dans l'AP de simulation, on a 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,[13].

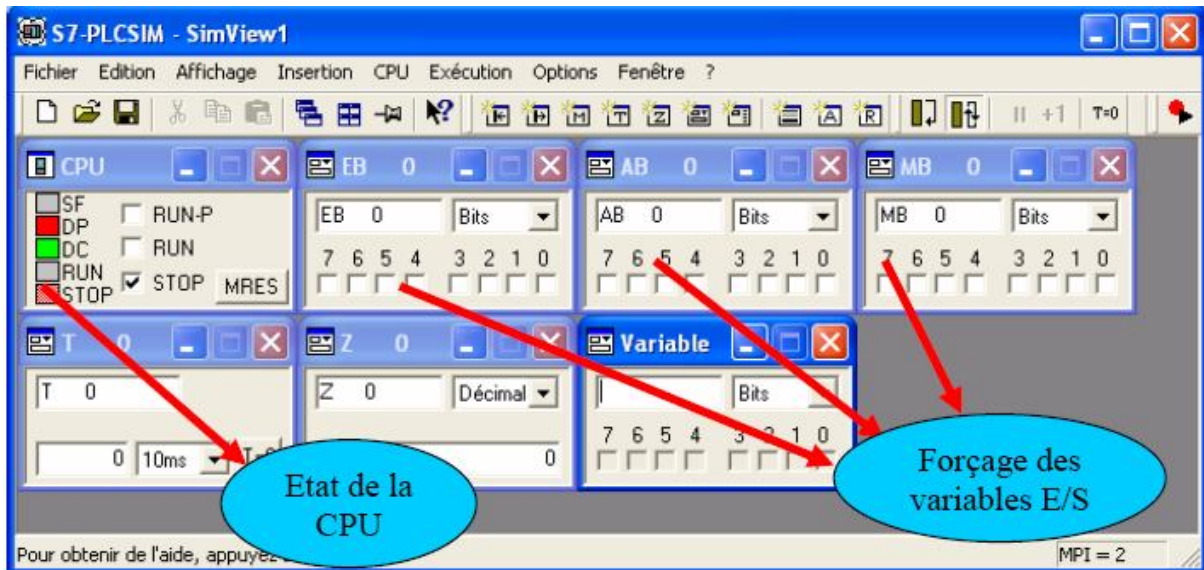


Figure II.3 : Interface de simulation PLCSIM.

II-2-3 La conception d'une structure programme compétè et optimisé

La mise en place d'une solution d'automatisation avec *STEP7* nécessite la réalisation des taches fondamentales suivantes :

- Création du projet *SIMATIC STEP7*
- Configuration matérielle *HW Config*

Dans une table de configuration, on définit les modules mis en œuvre dans la solution d'automatisation ainsi que les adresses permettant d'y accéder depuis le programme utilisateur, pouvant en outre, y paramétrer les caractéristiques des modules.

- Définition des mnémoniques

Dans une table des mnémoniques, on remplace des adresses par des mnémoniques locales ou globales de désignation plus évocatrice afin de les utiliser dans le programme.

- Création du programme utilisateur

En utilisant l'un des langages de programmation mis à disposition, on crée un programme affecté ou non à un module, qu'on enregistre sous forme de blocs, de sources ou de diagrammes.

- Exploitation des données:

Création des données de références : Utiliser ces données de référence afin de faciliter le test et la modification du programme utilisateur et la configuration des variables pour le " contrôle commande".

- Test du programme et détection d'erreurs

Pour effectuer un test, on a la possibilité d'afficher les valeurs de variables depuis le programme utilisateur ou depuis une CPU, d'affecter des valeurs à ces variables et de créer une table des variables qu'on souhaite afficher ou forcer.

- Chargement du programme dans le système cible

Une fois la configuration, le paramétrage et la création du programme terminés, on peut transférer le programme utilisateur complet ou des blocs individuels dans le système cible (module programmable de la solution matérielle). La CPU contient déjà le système d'exploitation.

- Surveillance du fonctionnement et diagnostic du matériel

La détermination des causes d'un défaut dans le déroulement d'un programme utilisateur se fait à l'aide de la « Mémoire tampon de diagnostic », accessible depuis le *SIMATIC Manager*.

II-3 Réalisation du programme

II-3-1 Création d'un projet dans SIMATIC manager

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

En sélectionnant l'icône *SIMATIC Manager*, on affiche la fenêtre principale, pour sélectionner un nouveau projet et le valider.

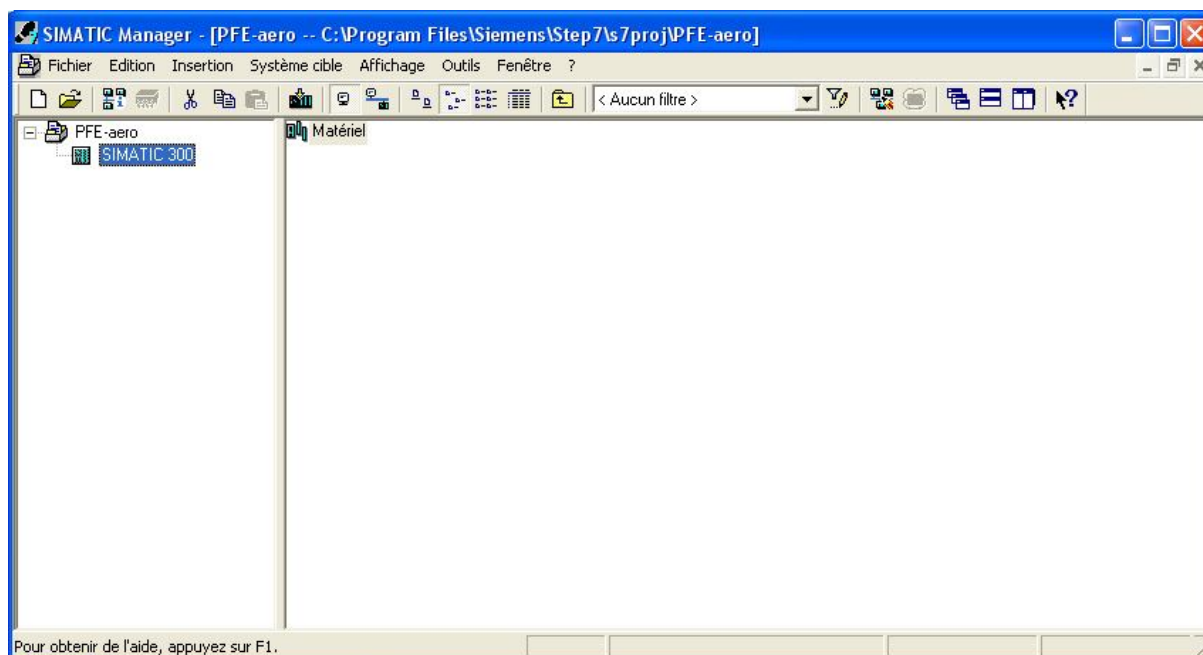


Figure II. 4: Page de démarrage de STEP7.

Comme le projet est vide il nous faut insérer une station *SIMATIC 300*.

Deux approches sont possibles. Soit on commence par la création du programme puis la configuration matérielle ou bien l'inverse.

II-3-2 Configuration matérielle

C'est une étape importante, qui correspond à l'agencement des châssis, des modules et de la périphérie décentralisée.

Les modules sont fournis avec des paramètres définis par défaut en usine.

Une configuration matérielle est nécessaire pour :

- Modifier les paramètres ou les adresses pré-règles d'un module,
- Configurer les liaisons de communication.

Le choix du matériel *SIMATIC S300* avec une CPU314 nous conduit à introduire la hiérarchie suivante :

On commence par le choix du châssis selon la station choisie auparavant, Pour la station *SIMATIC S300*, on aura le châssis « RACK-300 » qui comprend un rail profilé.

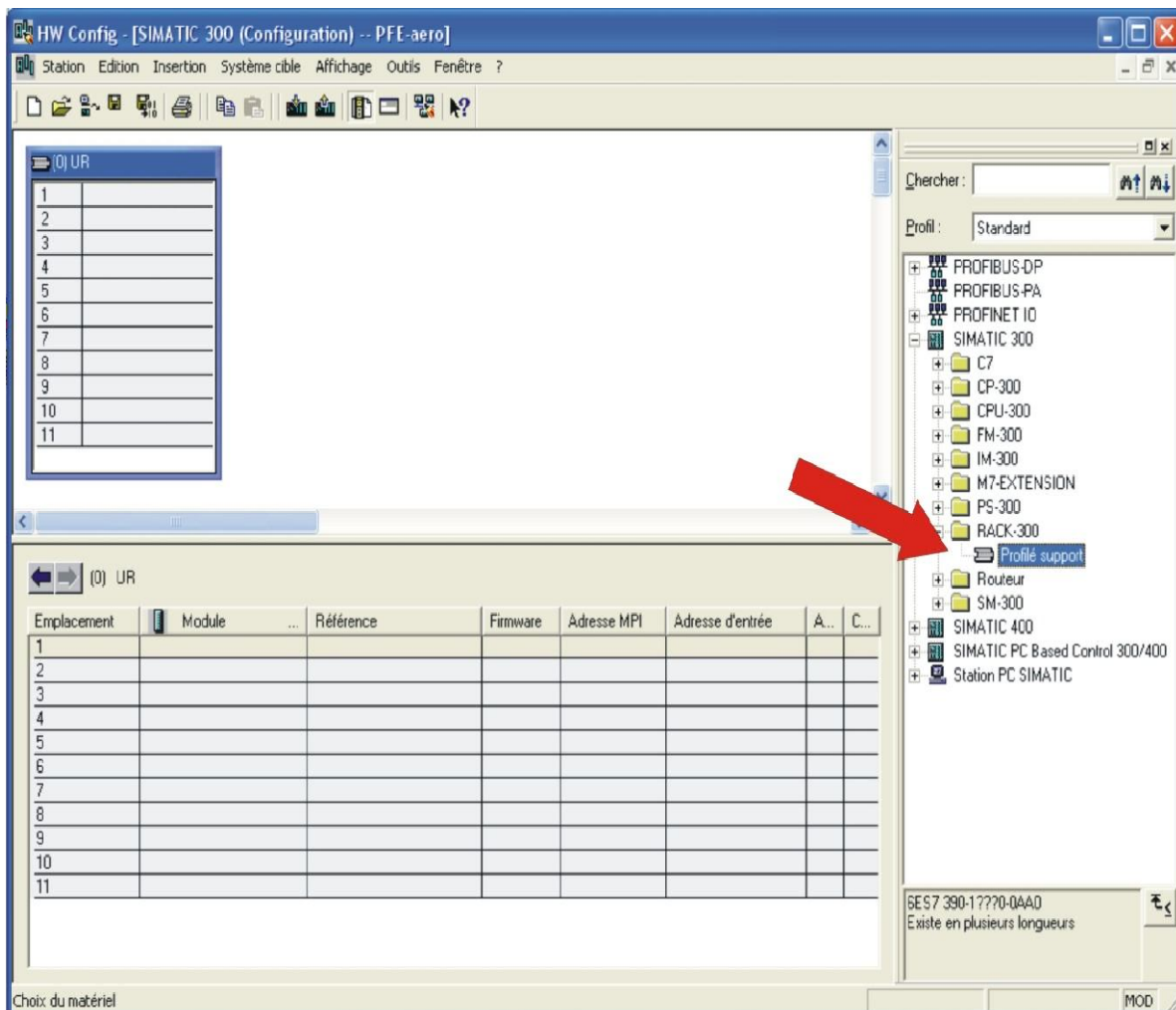


Figure II. 5: Choix du RACK.

Sur ce profile, l'alimentation préalablement sélectionnée se trouve dans l'emplacement n°1.

Parmi celles proposées notre choix s'est porté sur la « PS-307 5A ».

La « CPU 314 » est impérativement mise à l'emplacement n°2.

L'emplacement n°3 est réservé comme adresse logique pour un coupleur dans une configuration multi-châssis.

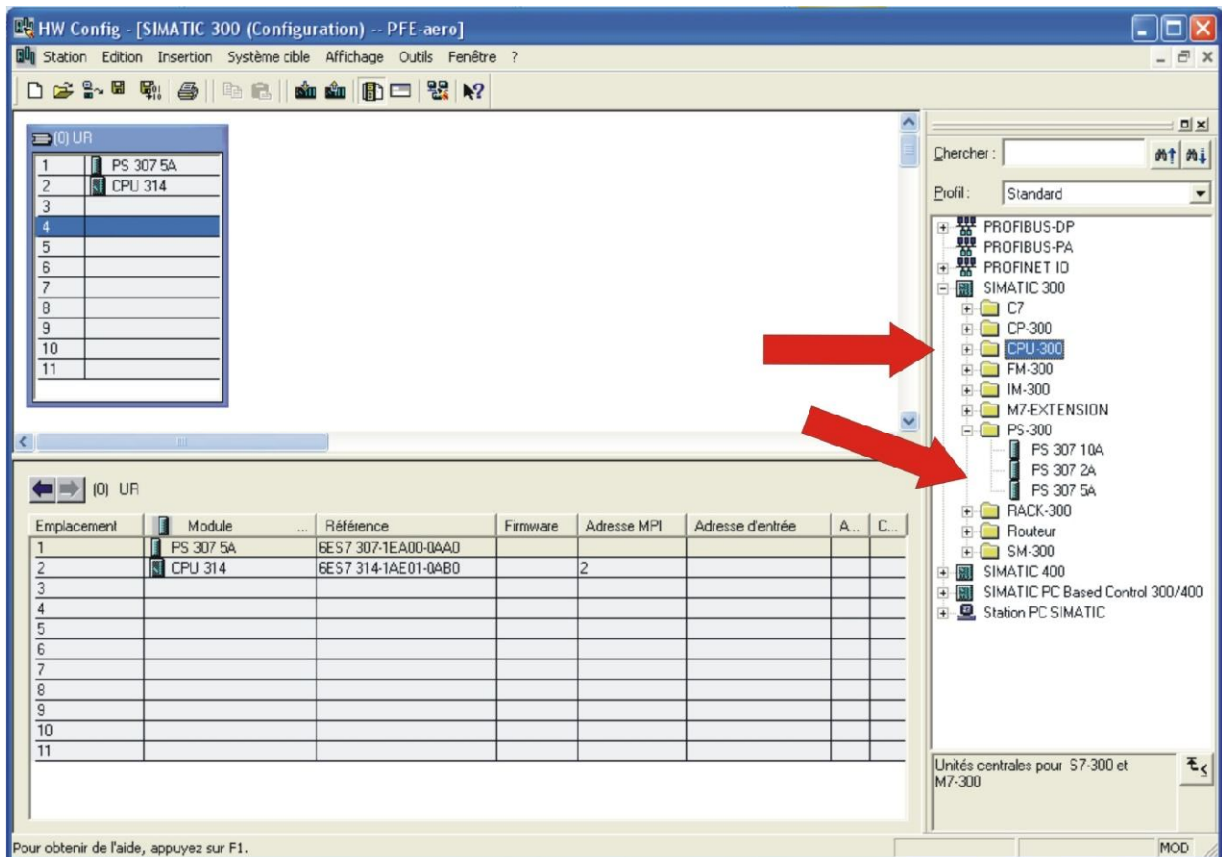


Figure II.6 : Choix du CPU et de l'alimentation.

A partir de l'emplacement 4, il est possible de monter au choix jusqu'à 8 modules de signaux (SM), processeurs de communication (CP) ou modules fonctionnels (FM), [13].

Nous allons y mettre les modules d'entrées et de sorties analogiques et numériques ; D'après l'identification des E/S du PLC dans le chapitre III il y a :

- Une embase de 32 entrées numériques (32 DI).
- Une embase de 32 sorties numériques (32 DO).

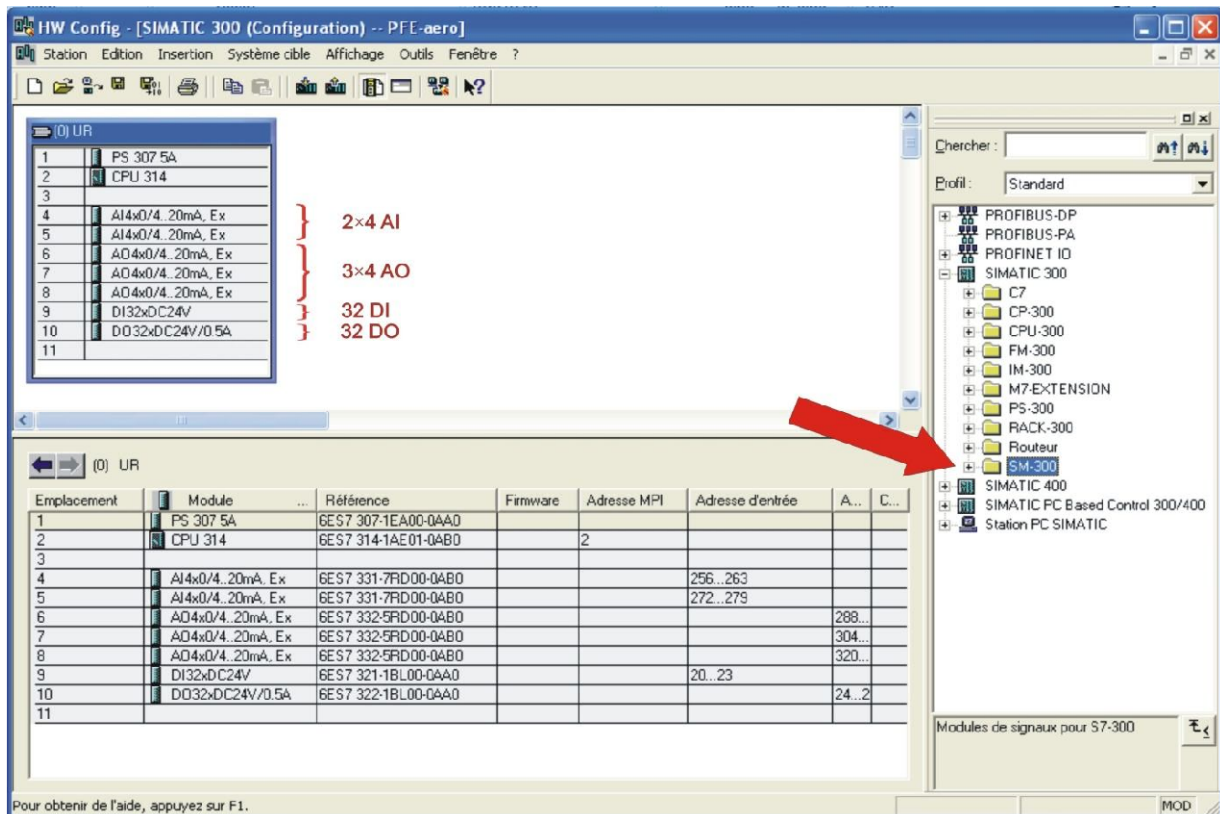


Figure II.7 : Choix des embases d'E/S.

Après cela il ne nous reste qu'à enregistrer et compiler.

La configuration matérielle étant terminée, un dossier « Programme S7 » est automatiquement inséré dans le projet, comme indique dans la figure suivante :

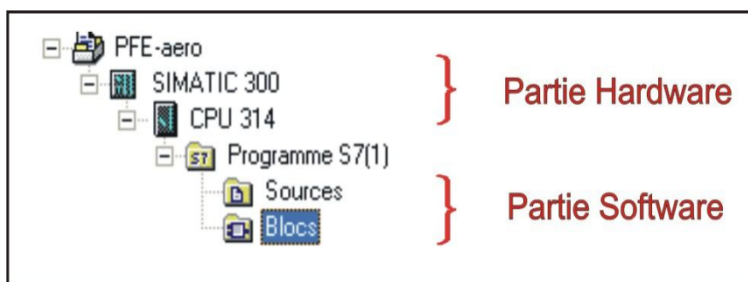


Figure II. 8: Insertion de configuration dans le projet .

II-3-3 Création de la table des mnémonique

Dans tout programme il faut définir la liste des variables qui vont être utilisées lors de la programmation. Pour cela la table des mnémoniques est créée. L'utilisation des noms

appropriés rend le programme plus compréhensible est plus facile à manipuler. Ce type d'adressage est appelé « relatif ».

Pour créer cette table, on suit le cheminement suivant :

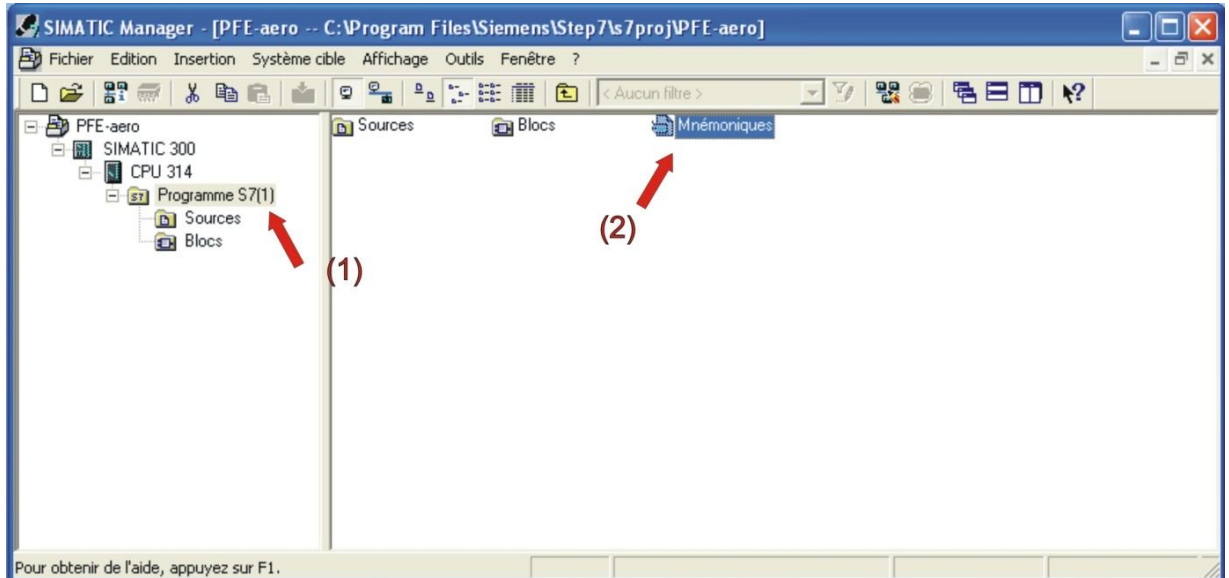


Figure II.9: Création de la table des mnémoniques.

II-3-4 Elaboration du programme S7

-Les blocs de code :

Le dossier bloc, contient les blocs que l'on doit charger dans la CPU pour réaliser la tâche d'automatisation, il englobe :

- Les blocs de code (OB, FB, SFB, FC, SFC) qui contiennent les programmes,
- Les blocs de données DB d'instance et DB globaux qui contiennent les paramètres du programme.

-Les blocs d'organisation(OB) :

Les OB sont appelés par le système d'exploitation, on distingue plusieurs types :

- ceux qui gèrent le traitement de programmes cycliques
- ceux qui sont déclenchés par un événement,
- ceux qui gèrent le comportement à la mise en route de l'automate programmable
- et en fin, ceux qui traitent les erreurs. [12]

Le bloc OB1 est généré automatiquement lors de la création d'un projet. C'est le programme cyclique appelé par le système d'exploitation. [12]

-Les blocs fonctionnels (FB) :

Le FB est un sous programme écrit par l'utilisateur et exécuté par des blocs de code. On lui associe un bloc de données d'instance relatif à sa mémoire et contenant ses paramètres.

Les SFB système sont utilisés pour des fonctions spéciales intégrées dans la CPU, [12].

-Les fonctions (FC) :

La FC contient des routines pour les fonctions fréquemment utilisées. Elle est sans mémoire et sauvegarde ses variables temporaires dans la pile de données locales. Cependant elle peut faire appel à des blocs de données globaux pour la sauvegarde de ses données, [12].

Les SFC sont utilisées pour des fonctions spéciales, intégrées dans la CPU S7, elle est appelée à partir du programme.

-Les blocs de données (DB) :

Ces blocs de données servent uniquement à stocker des informations et des données mais pas d'instructions comme les blocs de code. Les données utilisateurs stockés seront utilisés par la suite par d'autres blocs.

II-4 conclusion

Le STEP7 permet de programmer individuellement un automate (en différents langages). Il prend également en compte le réseau des automates, ce qui permet d'accéder à tout automate du réseau (pour le programmer), et éventuellement aux automates de s'envoyer des messages entre eux.

Il permet de décrire les comportements attendus de l'automatisme en imposant une démarche rigoureuse, éventuellement hiérarchisée, est ce le but recherché les systèmes industriel de production

Le chapitre suivant sera consacré à fois, à la description et la fonctionnement d'une unité de production de brique d'EL KALAA BENI-HAMMAD, et à la solution automatique basée sur le GRAFCET hiérarchisé.

Chapitre III

DESCRIPTION ET FONCTIONNEMENT DE LA BRIQUETERIE

Chapitre III : description et fonctionnement de la briqueterie

III-1 Introduction :

Le stage d'une mois au sein de la briqueterie D'ELKALAA BENI-HAMMED à pour objectifs d'enrichir nos connaissances théoriques par les solutions technologiques réelles sur le lieu de productions d'une part, et d'autre part avoir un aperçu sur les systèmes automatisé industriel de productions, qui sont en mutation continuelle, et forment le pilier de toutes l'industries actuelles.

Dans ce troisième chapitre on donneras en premier des notions générales sur l'unité de production briqueterie d'EL KALAA BENI-HAMMAD, puis on passeras à la description des différent étapes de fabrication de la brique

- *préparation de l'argile,*
- *fabrication de la brique,*
- *coupure,*
- *séchage,*
- *cuisson.*

Puis on termineras par les solutions graphiques de ces étapes(GRAFCET).

III-2 Présentation de la briqueterie d'EL KALAA BENI-HAMMAD :

La briqueterie d'EL KALAA BENI-HAMMAD est située au lieu dit : LOUIZA à la commune de METARFA WILAYA.M'SILA, coté nord de la RN n°40 et Au est, ouest, sud terroirs inexploitées, le site occupe une superficie de 60.000 m², la briqueterie est équipée par la société française CERIC qui a fait la prise en charge du projet et la mise en service de cette dernière en février 2007. La briqueterie à une capacité de production de 75000T/année

III-3 Sécurité d'usine :

Pour protéger l'unité et par personnel , un réseau de canalisation d'eau est installé à fin de l'utilisé en cas d'incendie, en plus d'une bonne distribution des extincteurs couvrant toute l'unité. Tout les systèmes de protection est contrôlé par les agents de la protection civile, et en plus de la vérification périodique de date rechargement soit pour le respect des notes de sécurité et d'interdictions des zones pour ces personelles.

Avant toute intervention plusieurs règles de sécurité doivent être appliques le système automatisé de l'unité est doté des options de sécurité très sophistiqués (alarmes , signalisation , arrêt d'urgence , coupure au touché , sectionnent rapide et accessible.....ext),

III-4 Processus de fabrication(fonctionnement) :

III-4-1 Hiérarchie des GRAFCET

Les GRAFCET hiérarchisés forment une structure de type maître esclave (père fils) dans laquelle le GRAFCET maître donne des ordres à un ou plusieurs GRAFCET esclaves (on parle alors de GRAFCET de tâche ou de sous programme GRAFCET) et les GRAFCET esclaves renvoient un accusé d'exécution en fin de tâche. A la différence d'une macro-étape les GRAFCET de tâche peuvent être appelés de différents endroits du GRAFCET maître. Cependant ils exécuteront une nouvelle tâche seulement lorsqu'ils auront terminé celle en cours.

En fait, il est souvent nécessaire de placer à un niveau hiérarchiquement supérieur des GRAFCETs de gestions Les principaux GRAFCETs que l'on peut trouver sont :

- Le GRAFCET de Sûreté (GS) maître par rapport aux deux autres,
- Le GRAFCET de Conduite et Initialisation (GCI), maître par rapport au GPN
- Le GRAFCET de Production Normale (GPN).

Les actions d'un GRAFCET sur un autre peuvent être représentées par ordres de forçage. L'émission d'un ordre de forçage vers un autre implique une hiérarchie ces GRAFCET. Cette hiérarchie peut être représentée par (fig III.1)

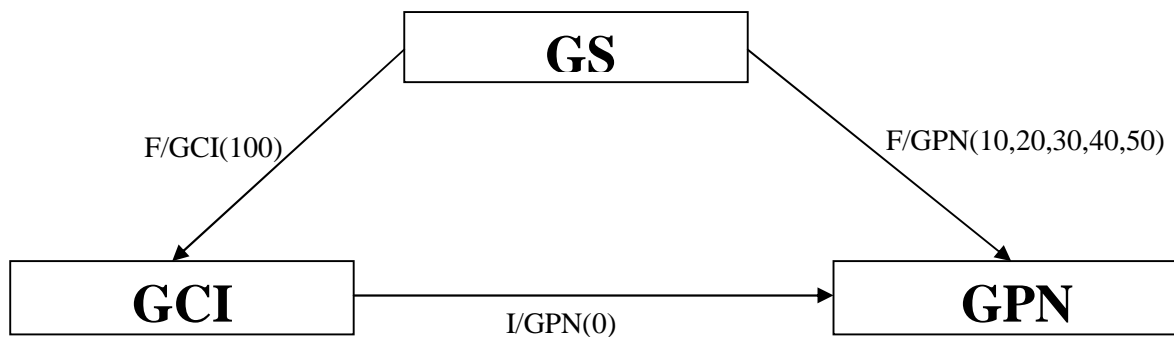


Figure III. 1: Hiérarchie des GRAFCET.

III-4-1-1 GRAFCET de sécurité (ou de surveillance) :

Ce GRAFCET décrit l'ensemble des procédures de sécurité du système, c'est le GRAFCET hiérarchiquement le plus important. L'arrêt d'urgence et les procédures de mise en route sont décrits dans ce GRAFCET., (fig III.2).

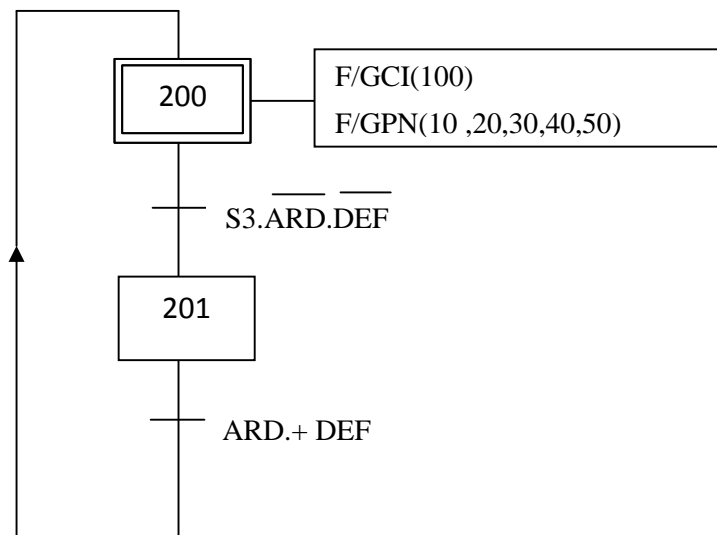


Figure III. 2: GRAFCET de sécurité.

F/GC (100): Indique que le GRAFCET de sûreté est hiérarchiquement supérieur au GRAFCET de conduite et initialisation (GCI). Cet ordre provoque l'activation et le maintien actif de l'étape 100 du GRAFCET de conduite et initialisation et la désactivation de toutes les autres étapes de ce même GRAFCET, et le même ordre de forçage est donné pour le GRAFCET de production normal F/GPN (10,20, 30, 40, 50). Ce GRAFCET est réceptif à une consigne d'arrêt de sécurité (arrêt d'urgence: AU) et ce depuis tout les états.

III-4-1-2 GRAFCET de Conduite et Initialisation (ou GRAFCET des modes de marches)

ce GRAFCET décrit l'ensemble des procédures de Marches (auto, Cycle/Cycle, Manuel,...) et des arrêts normaux ainsi que la mise en référence et l'initialisation du système , fig III.3.

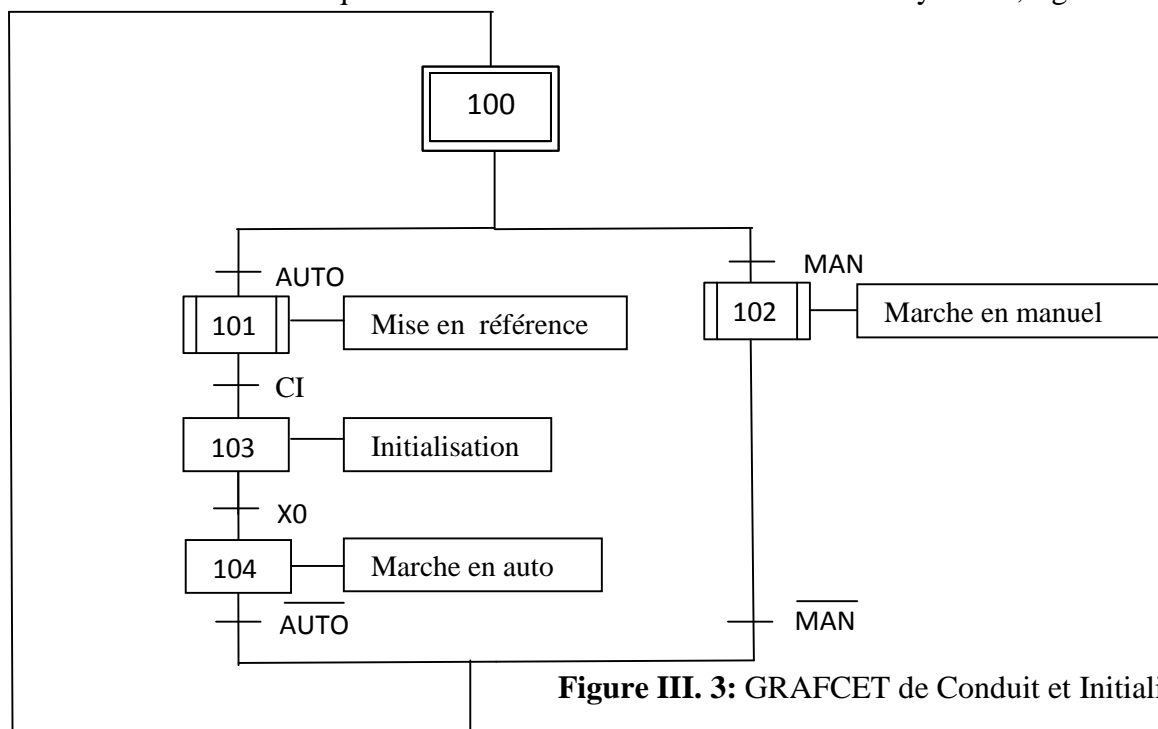


Figure III. 3: GRAFCET de Conduit et Initialisation.

III-4-1-3 GRAFCET de coordination des tâches

ce GRAFCET est le niveau de description du fonctionnement normal de l'automatisme. Ce GRAFCET est en général décomposé en plusieurs tâches représentant les différentes fonctions de l'automatisme.

Nôtre système de production se compose de cinq tâches:

- 1- Tâche de préparation,
- 2- Tâche de fabrication,
- 3- Tâche de pré coupeur et coupeur ,
- 4- Tâche de séchoir,
- 5- Tâche de four.

Le GRAFCET de coordination des tâches est donné par la figure (fig III.4).

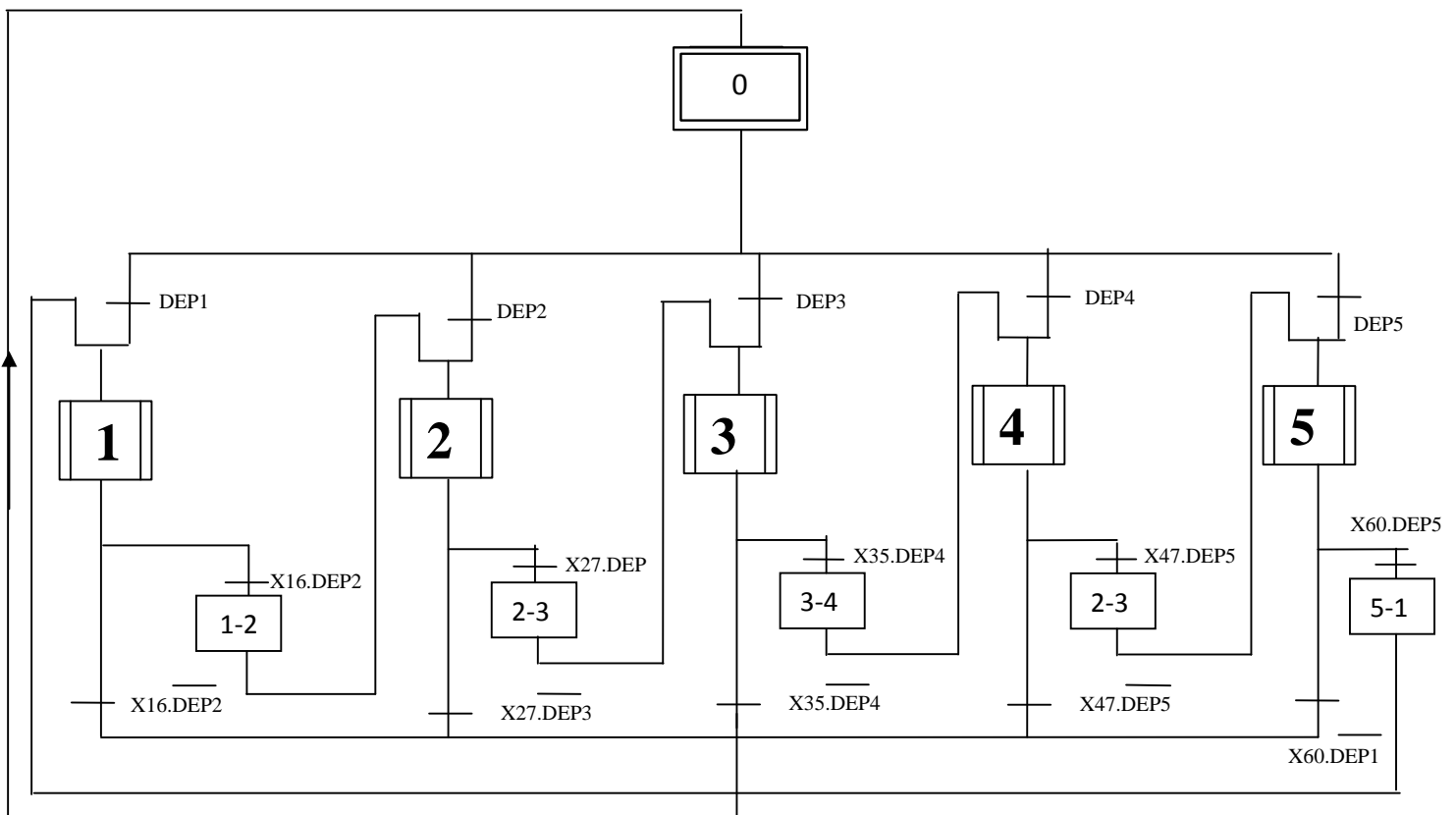


Figure III. 4: GRAFCET de coordination de tâche.

III-4-2 Tâche de préparation de l'argile :

l'opération de préparation est réalisée selon les étapes suivantes :

Carrière: Extraction d'argile.

Stockage: Plus de six mois.

Ajout de sable : Doseur argile 80% , doseur sable 20%.

Désagréation: Granulométrie moins de (10 mm).

Mouilleur: humidification et malaxage.

III-4-2-1 Cahier du charge :

- Pour démarrer la tâche de préparation on appuyer sur le bouton DEP1 , ce qui implique un démarrage d'une alarme de 5 s.
- démarrage des deux grands moteurs (MD et MB) et l'électrovanne d'eau, et une nouvelle temporisation après la fin de l'alarme ,avec la condition de présence d'air.
- Avec la présence d'eau et après la fin de la deuxième temporisation, un démarrage des convoyeurs (CP18,CP16,CP15,CP14) et du moteur MM qui reste en marche avec les deux moteurs MD et MD durant toute la tâche.
- lorsque le racleur du répartiteur (RRP) actionnée, démarrage de convoyeurs (CP13,CP12), et du moteur (MR) qui reste en marche durant toute la tâche.
- le convoyeur CP12 fait tirer le racleur des distributeurs (trémies) 1 et 2 (RD1,RD2) ,ce qui provoque le démarrage des deux trémies 1 et 2 (DT1,DT2),et une temporisation de 5s.
- La fin de la tâche juste après la fin de la temporisation.

III-4-2-2 Schéma synoptique de la tâche de préparation:

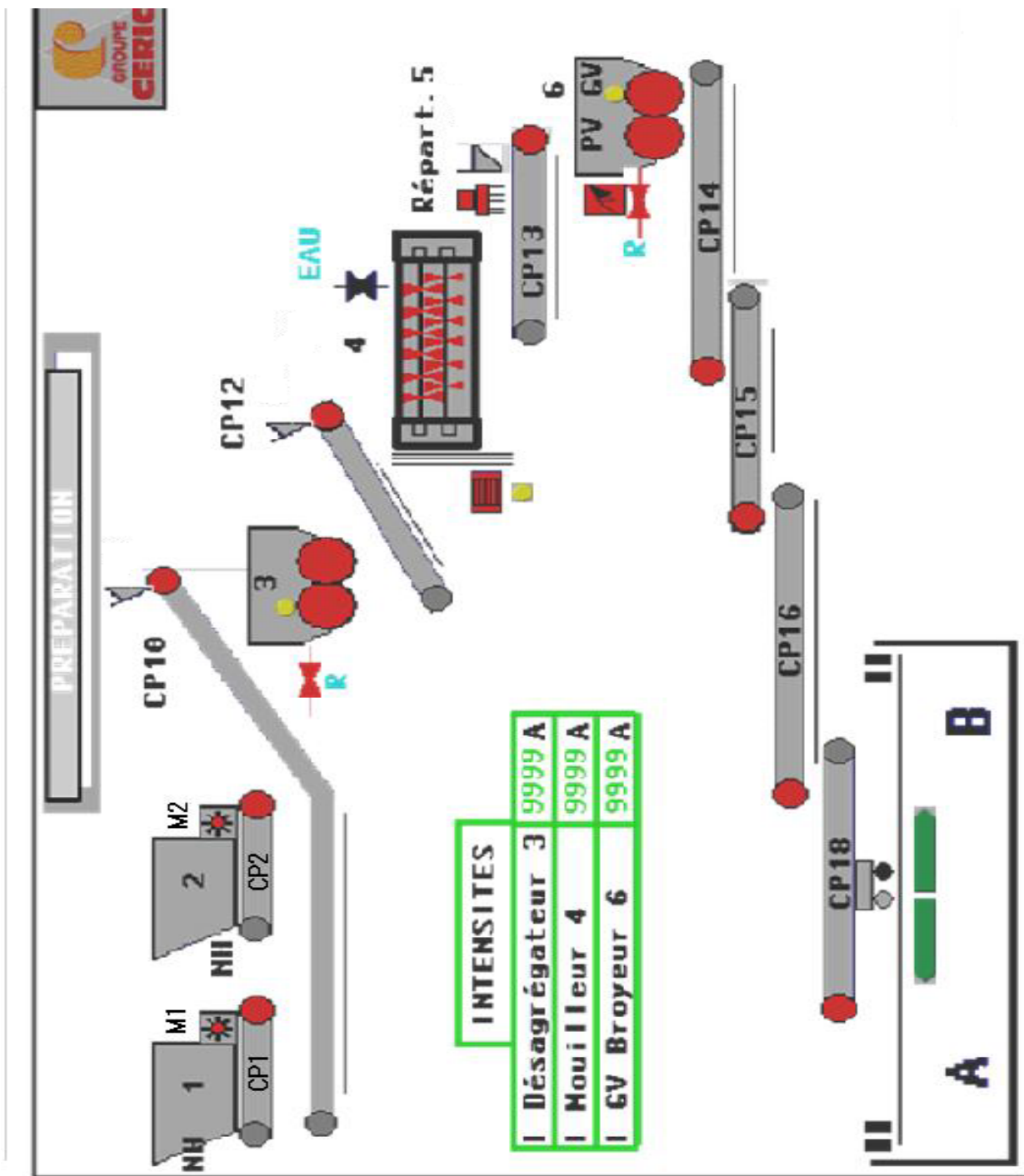


Figure III. 5: Schéma synoptique de l'installation de préparation affiché sur pupitre.

III-4-2-3 Table des entrées : toutes les entrées de la tâche de préparation sont classées dans la table (tab III.1)

Numéro	L'entrée	Commentaire
1	X1	Appel de la tâche de préparation
2	X104	Marche en auto
3	DEP1	Bouton de démarrage de la tâche
4	Pair	Présence d'air
5	Pe	Présence d'eau
6	RRPA	Racleur du répartiteur actionné
7	RD1	Racleur distributeur 01 actionné
8	RD2	Racleur distributeur 02 actionné

Table III. 1: Table des entrées de la tâche de préparation.

III-4-2-4 Table des sorties : toutes les sorties sont classées dans la table (tabIII.2).

Numéro	sortie	Commentaire
1	ALARME	Démarrage de l'alarme
2	MD	Moteur désagrégateur
3	MB	Moteur broyeur
4	EV	Electrovanne d'eau
5	MM	Moteur mouilleur
6	CP18	Convoyeur cp18
7	CP16	Convoyeur cp16
8	CP15	Convoyeur cp15
9	CP14	Convoyeur cp14
10	MR	Moteur répartiteur
11	CP13	Convoyeur cp13
12	CP12	Convoyeur cp12
13	DT1	Démarrage trémie 01
14	DT2	Démarrage trémie 02

Table III. 2: Table des entrées de la tâche de préparation.

III-4-2-5 GRAFCET de la tâche de préparation :

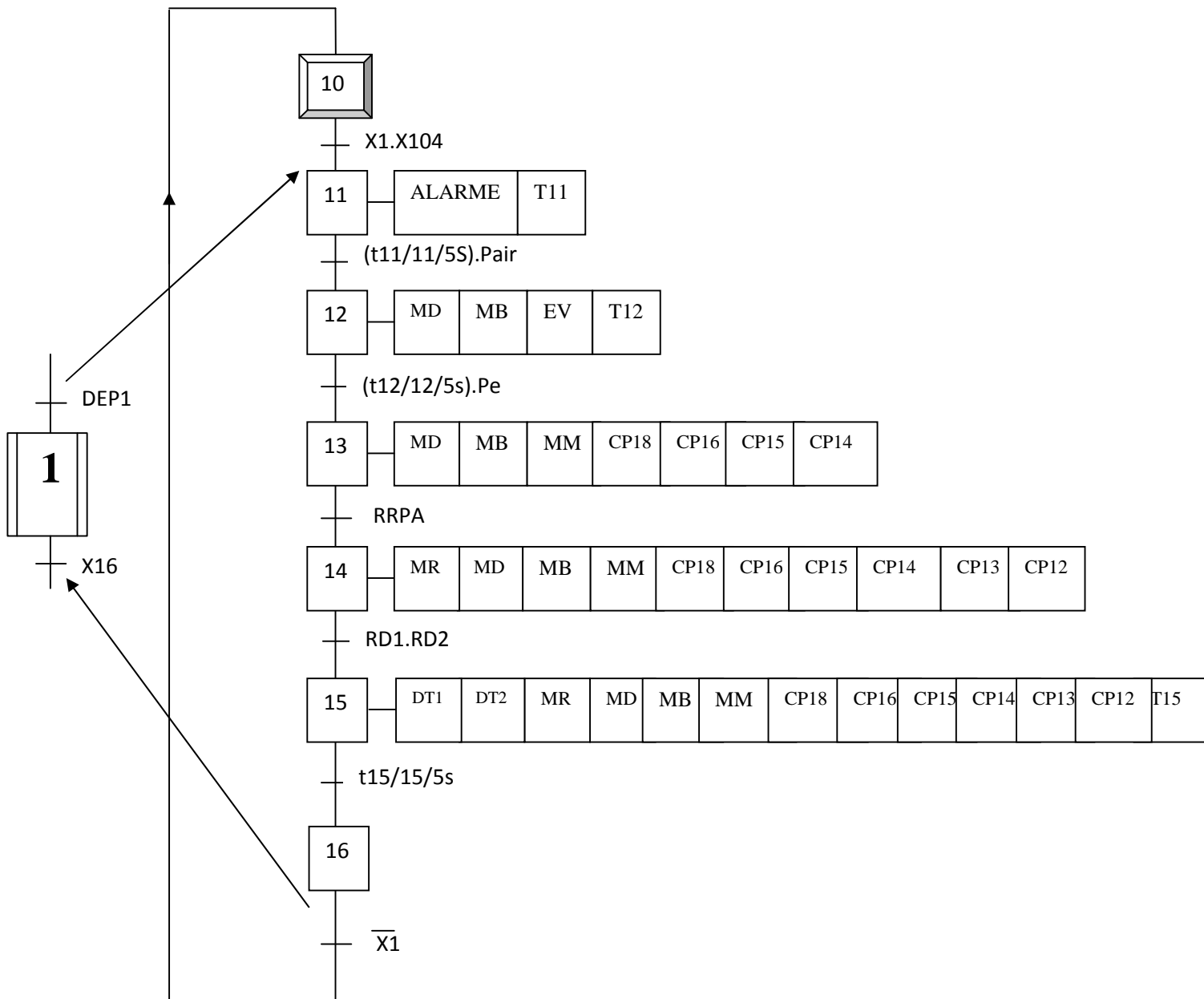


Figure III.6 : GRAFCET de la tâche de préparation .

III-4-3 Tâche de fabrication :

elle est basée sur les étapes suivantes :

Pré finisseur: Granulométrie de 04 mm.

Finisseur: Granulométrie de 02 mm (entre 1 et 1.5 mm).

Malaxage et Humidification: Un taux d'humidité de 16 à 24 %.

Mouleuse(étirage): Avec une pression pas moins de 18 bar.

III-4-3-1 Cahier du charge :

-pour démarrer la tâche de fabrication on appuie sur le bouton DEP2,ce qui engendre un démarrage d'une alarme de 5s.

-lorsque la trémie 14 est en niveau bas (T14B) et avec la présence de l'argile (terre) dans la trémie 11 (pt) ,juste après la fin de l'alarme ,une démarrage des convoyeurs (CF13,CF12).

- le convoyeur CF12 fait tirer le racleur du broyeur pré finisseur (Rbp), d'où le démarrage du moteur de broyeur (MBP).

-Démarrage de la trémie 11 lorsque le racleur RT11 est actionnée, la trémie 11 reste en marche avec les convoyeurs (CF11,CF10) et l'EV jusqu'à la trémie 14 devient en niveau haut(T14H)

-lorsque T14H et avec la présence d'eau ,démarrage de la malaxeur et de MBF et du convoyeurs (CF14,CD11,CD10).

-la présence d'argile (Parg) permet le démarrage du mouleuse (MMOL) et l'activation de T26 .

-la fin de la tâche après la fin de la temporisation.

III-4-3-2 Schéma synoptique de la tâche de fabrication:

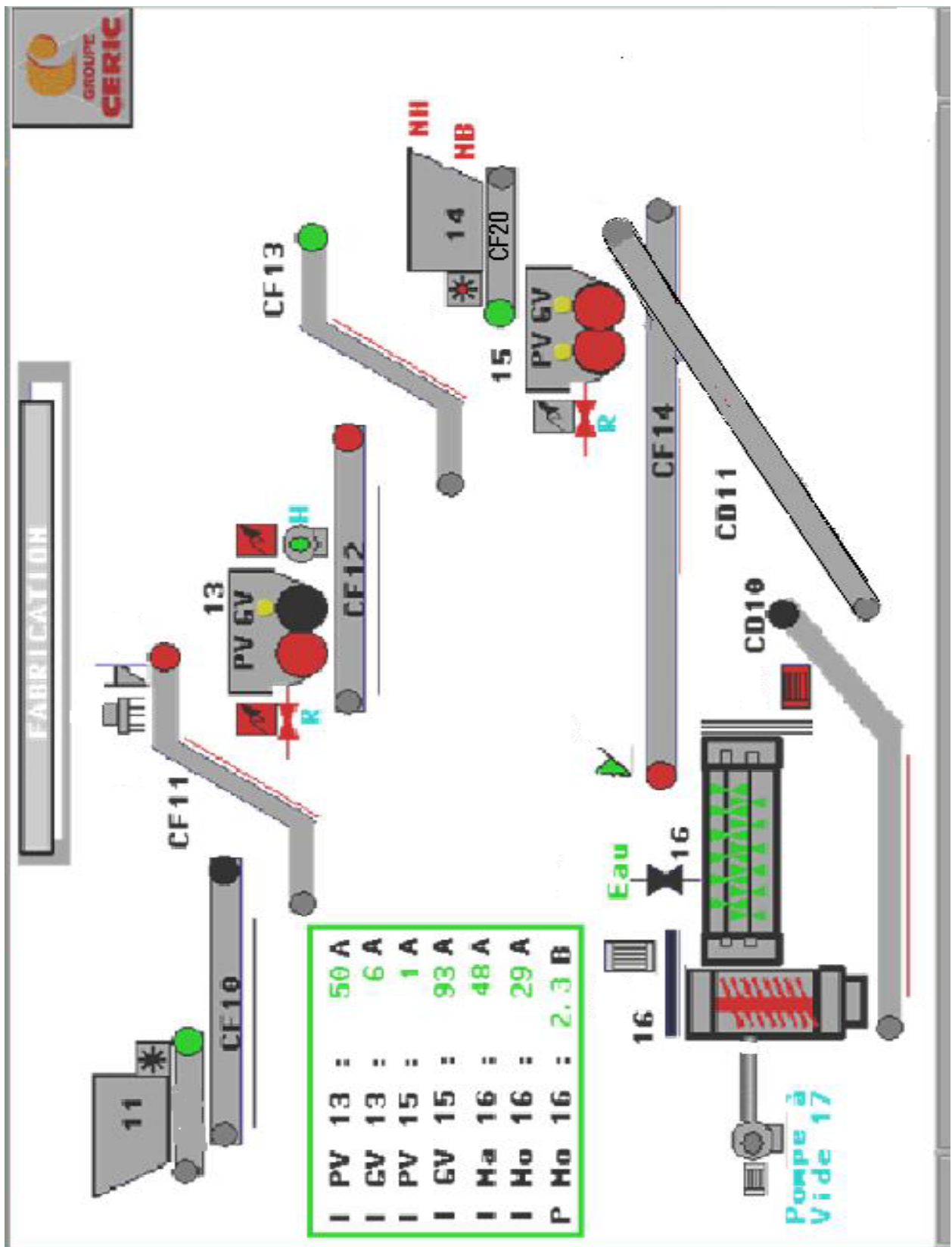


Figure III. 7: Schéma synoptique de l'installation de fabrication affiché sur pupitre.

III-4-3-3 Table des entrées : les entrées de la tâche de fabrication sont illustrées par la table(tabIII.3).

numéro	entrée	commentaire
1	X2	Appel de la tache de fabrication
2	X104	Marche en auto
3	DEP2	Botton de démarrage de la tache
4	Pt	Présence terre
5	T14B	Trémie14 niveau bas
6	RBP	Racleur broyeur pré finisseur
7	RT11	Racleur trémie11
8	T14H	Trémie14 niveau haut
9	Pe	Présence d'eau
10	Parg	Présence d'argile

Table III. 3: Table des entrées de la tâche de fabrication.

III-4-3-4 Table des sorties : les sorties de la tâche de fabrication sont données par la table (tab III.4).

numéro	sortie	Commentaire
1	ALARM	Démarrage de l'alarme
2	CF13	Convoyeur cf13
3	CF12	Convoyeur cf12
4	MBP	Moteur broyeur pré finisseur
5	DT11	Démarrage trémie11
6	CF11	Convoyeur cf11
7	CF10	Convoyeur cf10
8	EV	Electrovanne d'eau
9	MMAL	Moteur malaxeur
10	MBF	Moteur broyeur finisseur
11	CF14	Convoyeur cf14
12	CD11	Convoyeur cd11
13	CD10	Convoyeur cd10
14	MMOL	Moteur mouleuse

Table III. 4: Table des sorties de la tâche de fabrication.

III-4-3-4 GRAFCET de la tâche de fabrication :

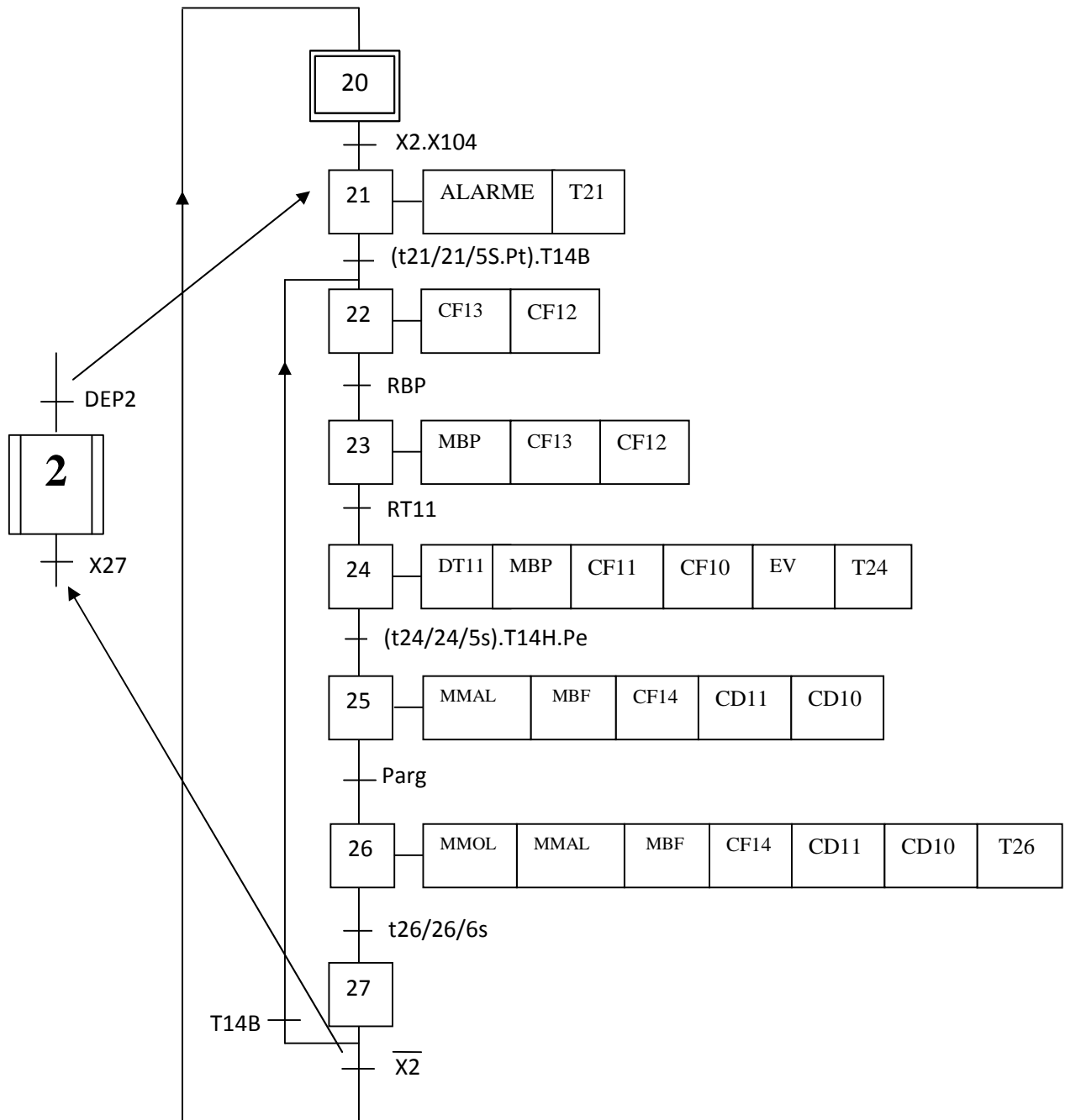


Figure III.8 : GRAFCET de la tâche de fabrication .

III-4-4 Tâche de coupure :

La dimension de 300 mm de longueur [(largeur et hauteur) filière].

III-4-4-1 Cahier du charge :

-pour démarrer la tâche de coupeur on appuyer sur le bouton DEP3.

- démarrage du moteur de pré coupeur M1 et sortie de vérin (V1+) avec les conditions suivants :

présence de la matière (brique) au niveau 1 (PM1), présence du fil (PF) et les deux portes au niveau 1 sont fermées (2PF).

-l'entrée de vérin 1 (V1-) et la sortie de vérin 2 (V2+) juste après la fin de pré coupeur.

-démarrage du moteur de tapie roulant (MTB) et l'entrée de vérin 2 (V2-) avec les conditions suivants : présence de la matière au niveau2 (PM2) et les quatre portes au niveau 2 sont fermées (4PF).

- démarrage de moteurs de la coupe finale M1 et M2 lorsque le capteur B1B9 détecte la présence de la matière.

-la fin de la tâche, lorsque les capteurs B1S1et B1S2 détectent la fin de la coupe finale.

III-4-4-2 Table des entrées: les entrées de la tâche de coupure sont citées par la table (tab III.5).

numéro	entrée	Commentaire
1	X3	Appel de la tache de coupeur
2	X104	Marche en auto
3	DEP3	Démarrage de la tâche
4	PM1	Présence de la matière niveau1
5	PF	Présence fil
6	2PF	Les deux portes de niveau1 fermées
7	FC	Fin du pré coupeur
8	PM2	Présence de la matière niveau 2
9	4PF	Les quatre portes de niveau2 fermées
10	B1B9	Présence de la matière à la fin du niveau 2
11	B1S1	Fin du coupeur(lame1)
12	B1S2	Fin du coupeur(lame2)

Table III. 5: Table des entrées de la tâche du coupeur.

III-4-4-3 Table des entrées: les sorties de la tâche de coupure sont illustrés par la table (tab III.6).

numéro	sorties	Commentaire
1	M1	Moteur de pré coupeur
2	V1	Vérin de fixation de la matière
3	V2	Vérin de la poussage
4	MTR	Moteur de tapie roulant
5	M2	Moteur du coupeur (lame1)
6	M3	Moteur du coupeur (lame2)

Table III. 6: Table des sorties de la tâche du coupeur.

III-4-4-4 GRAFCET de la tâche de coupure :

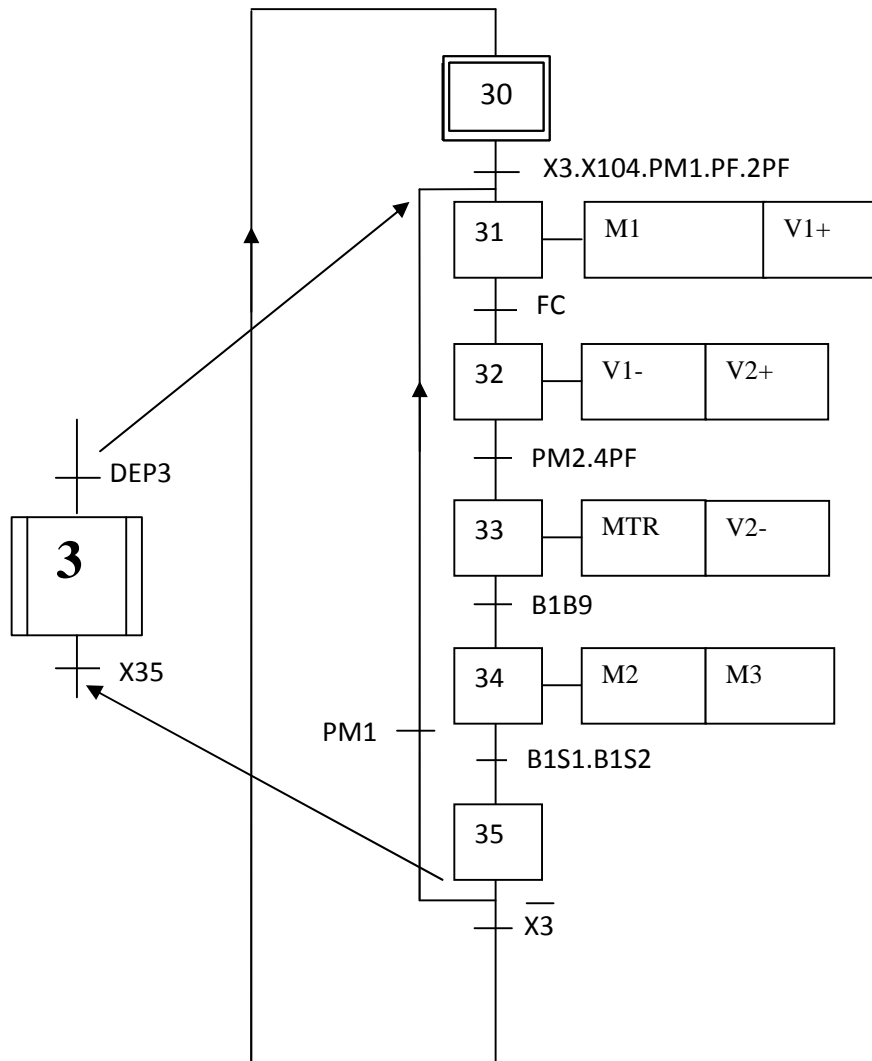


Figure III. 9: GRAFCET de la tâche de coupure .

III-4-5 Tâche de séchage :

Un cycle de séchage rapide de quatre heures a une montée de température jusqu'à 80°C (séchoir ANJOU a balancelle).

III-4-5-1 Cahier du charge :

- pour démarrer la tâche de séchage on appuyer sue le bouton DEP4 ,s'implique un démarrage de ventilateur (VN1) .
- démarrage de ventilateur VN2 lorsque le retour du signal de VN1 est vérifiée(VN1A) .
- démarrage du moteur MR lorsque le retour du signal de VN2 est vérifiée (VN2A).
- démarrage du moteur MAD lorsque le retour du signal de MR est vérifiée (MRA).
- l'ouverture de l'EVGA et l'EVA et l'allumage du briquet lorsque les conditions de présence du gaz et d'air de combustion sont vérifiées.
- démarrage du balanceur de briques lorsque la température du canal d'admission est supérieur ou égale à 80 (TEMPCAD 80).
- fin de la tâche si le balanceur reste plus de 45 min dans le séchoir.

III-4-5-2 Schéma synoptique de la tâche du séchage:

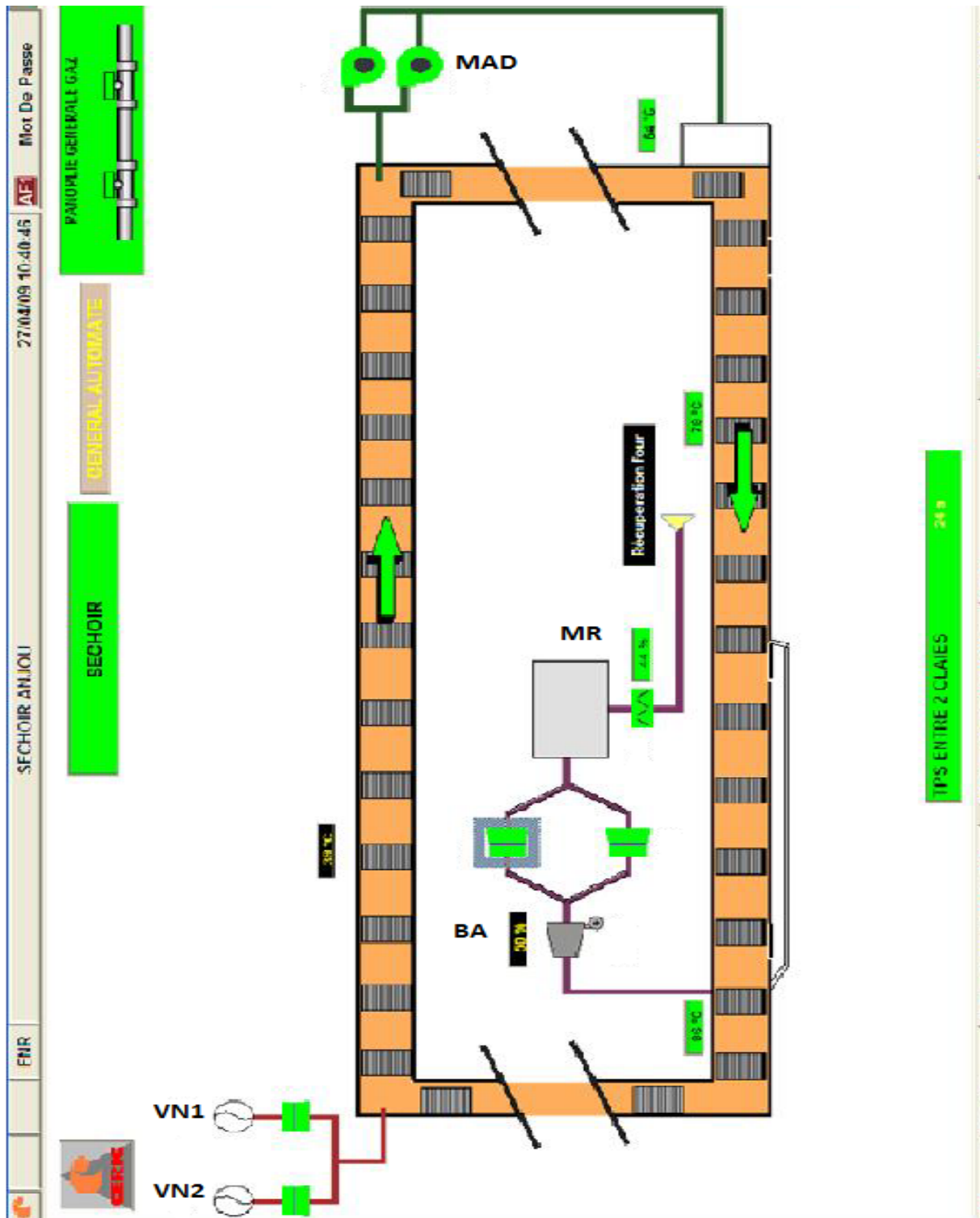


Figure III. 10: Schéma synoptique de l'installation du séchoir affiché sur pupitre.

III-4-5-3 Table des entrées: les entrées de la tâche du séchage sont citées par la table (tab III.7)

numéro	entrées	Commentaire
1	X4	Appel de la tâche de séchage
2	X104	Marche en auto
3	DEP4	Démarrage de la tâche
4	VN1A	Ventilateur d'extraction1 actionnée
5	VN2A	Ventilateur d'extraction2 actionnée
6	MRA	Moteur de récupération d'air actionnée
7	PGAZ	Présence du gaz
8	PAC	Présence d'air de combustion
9	TEMPCAD	Température du canal d'admission
10	CTB	Capteur de translation du balanceur de brique

Table III.7 : Table des entrées de la tâche de séchage.

III-4-5-4 Table des sorties : les sorties de cette tâche sont donné par la table (tab III.8).

numéro	sortie	Commentaire
1	VN1	Ventilateur d'extraction1
2	VN2	Ventilateur d'extraction1
3	MR	Moteur de récupération d'air
4	MAD	Moteur d'admission
5	EVGA	Electrovanne du gaz
6	EVA	Electrovanne d'air de combustion
7	BA	Briquet allumée
8	MBB	Moteur du balanceur de brique

Table III. 8: Table des sorties de la tâche du séchage.

III-4-5-5 GRAFCET de la tâche du séchage :

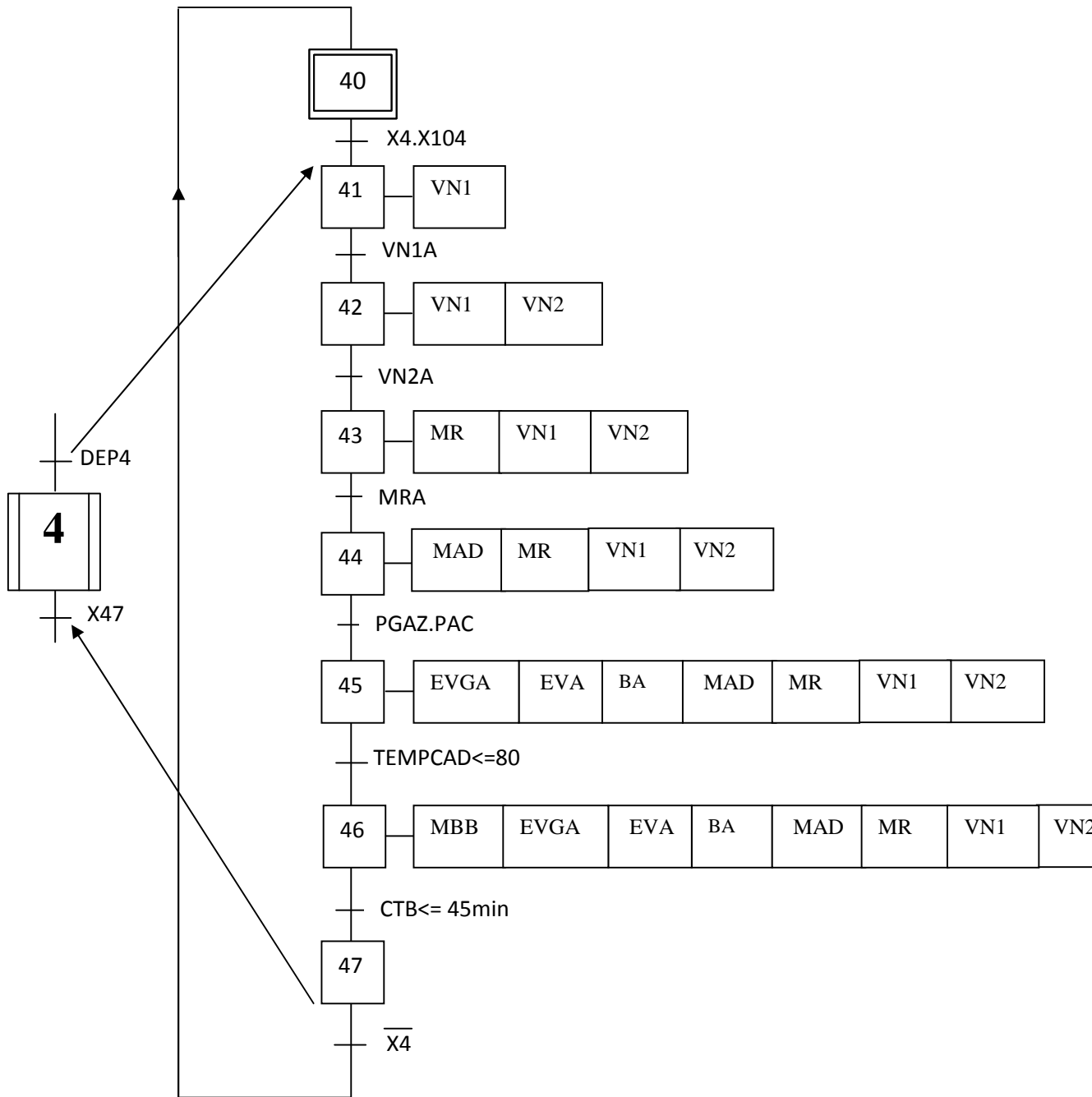


Figure III.11 : GRAFCET de la tâche de séchage .

III-4-6 Tâche de cuisson :

elle se fait dans un four tenu à une température de cuisson de 900°C.

III-4-6-1 Cahier du charge :

-pour démarrer la tâche de cuisson on appuie sur le bouton DEP5, ce qui provoque le démarrage du moteur MAC.

-chaque retour du signal d'un moteur implique le démarrage du prochain moteur comme suit :

MACA → démarrage du moteur ME.

MEA → démarrage du moteur MTF.

MTFA → démarrage du moteur MSP.

MSPA → démarrage du moteur MRA.

MRAA → démarrage du moteur MSW.

MSWA → démarrage du moteur MRF.

MRFA et présence du gaz PG → démarrage de pamoplie général du gaz et les jets et les fleuret (JA,FA).

PGGA, Fa et Ja → démarrage de transporteur du briques et T59.

-fin de la tâche après la fin de temporisation.

III-4-6-2 Schéma synoptique de la tâche de la cuisson:

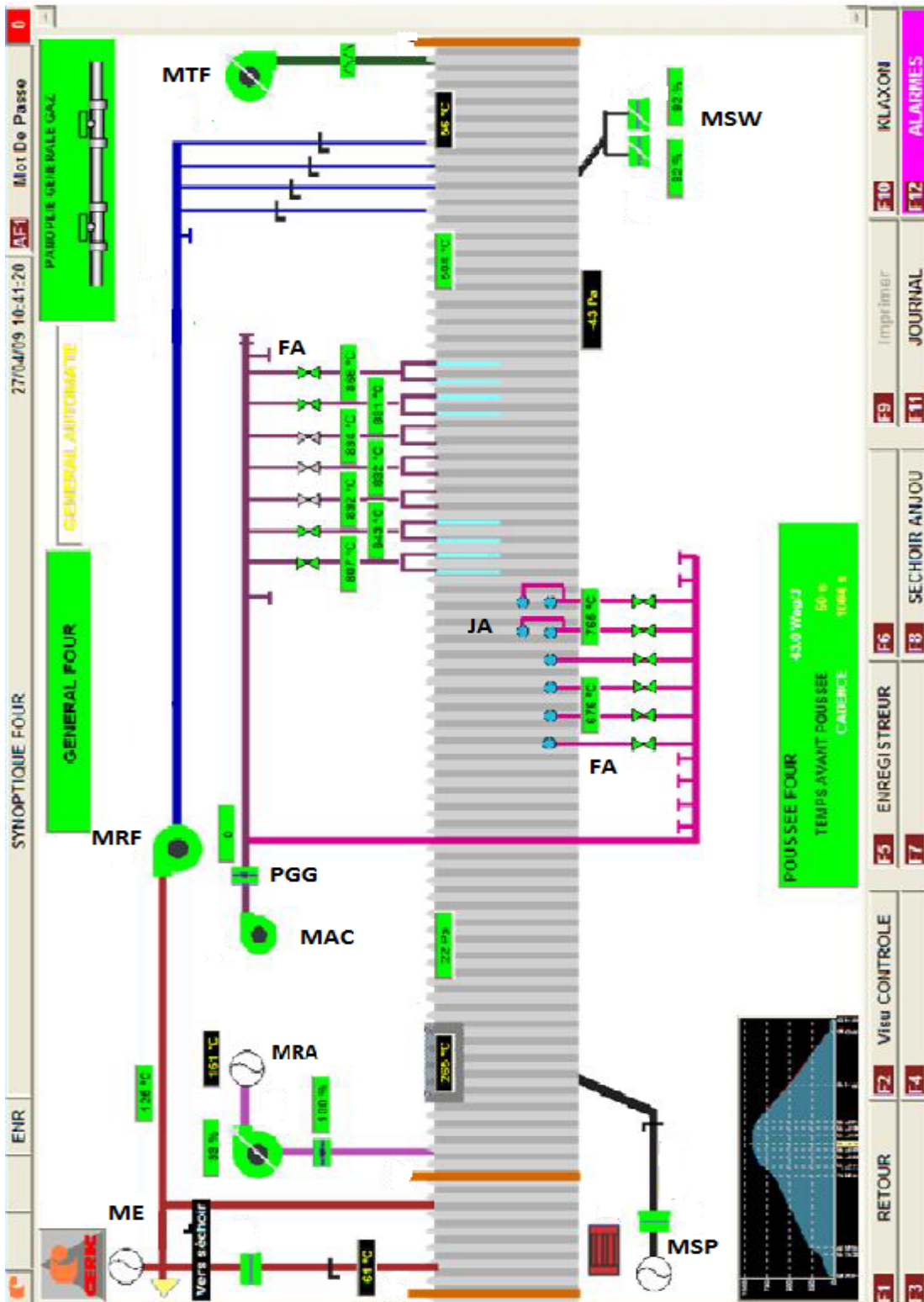


Figure III. 12: Schéma synoptique de l'installation de la cuisson affiché sur pupitre.

III-4-6-3 Table des entrées de la tâche de cuisson : les entrées de la dernière tâche sont présentés par la table (tab III.9).

numéro	entrée	Commentaire
1	X5	Appel de la tâche de cuisson de la brique
2	X104	Marche en auto
3	DEP5	Démarrage de la tâche
4	MACA	Moteur de l'air de combustion actionné (retour du signal)
5	MEA	Moteur d'extraction actionné (retour du signal)
6	MTFA	Moteur de tirage feu actionné (retour du signal)
7	MSPA	Moteur de sur pression actionné (retour du signal)
8	MRAA	Moteur de récupération d'air actionné (retour du signal)
9	MSWA	Moteur de soufflage wagon actionné (retour du signal)
10	MRFA	Moteur de refroidissement du feu actionné (retour du signal)
11	PG	Présence du gaz
12	PGGA	Pamoplie général du gaz actionné
13	Ja	Jet actionné
14	Fa	Fleuret actionné

Table III. 9: Table des entrées de la tâche de cuisson.

III-4-6-4 Table des sorties de la tâche de cuisson : les sorties de la dernière tâche sont présentés par la table (tab III.10).

numéro	Sortie	Commentaire
1	MAC	Moteur de l'air de combustion
2	ME	Moteur d'extraction
3	MTF	Moteur de tirage feu
4	MSP	Moteur de sur pression
5	MRA	Moteur de récupération d'air
6	MSW	Moteur de soufflage wagon
7	MRF	Moteur de refroidissement du feu
8	EVG	Electrovanne du gaz
9	PGG	Pamoplie général du gaz
10	JA	Jet
11	FA	Fleuret
12	DT	Démarrage transporteur

Table III. 10: Table des sorties de la tâche de cuisson.

III-4-6-5 GRAFCET de la tâche de cuisson :

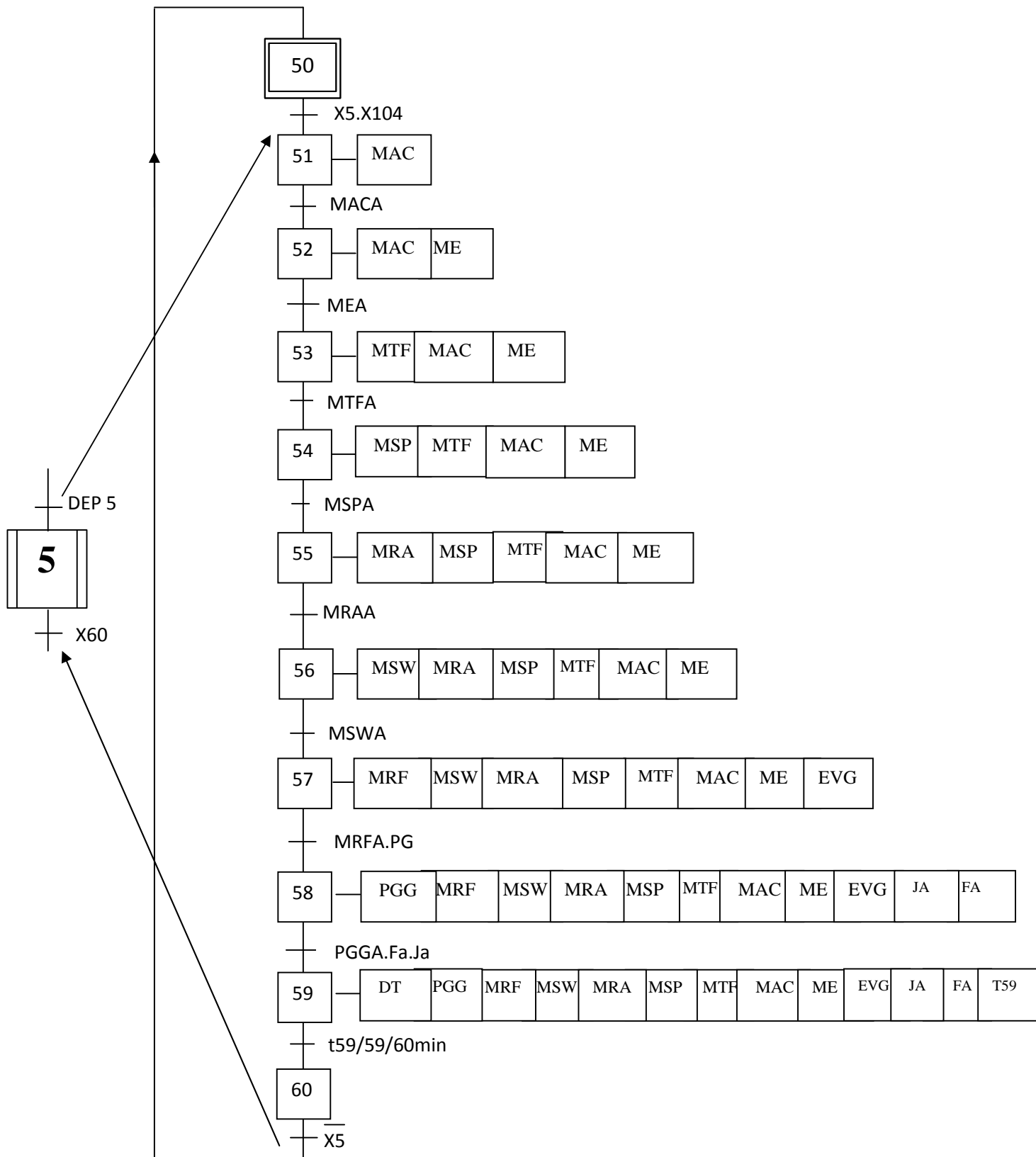


Figure III.13 : GRAFCET de la tâche de cuisson.

III-5 Conclusion

Dans ce chapitre, nous avons donné un aperçu sur l'entreprise de la Briqueterie D'EL KALAA/HAMMAD, puis après étude, sur cite, des différentes étapes de fabrication de la brique, nous avons proposés des solutions graphiques basées sur la notion de GRAFCET hiérarchisé, ces solutions englobent les GRAFCET des différentes taches et et les GRAFCET GS, GCI et CPN

*Pour concrétiser les tâches citées au paravent on doit passé par la programmation par le logiciel **STEP7**, et je choisira la langage **GRAPH**, ce qui fera l'objet du chapitre suivant.*

CHAPITRE IV

PROGRAMMATION PAR STEP7

Chapitre IV : programmation par STEP7

IV-1-Introduction :

Après une étude approfondie du fonctionnement de l'usine, nous avons optés pour programmation avec le logiciel STEP7, en utilisant le langage GRAPH, qui s'applique à tout système logique de commande d'automatisme industriel.

Dans ce chapitre, nous décrivons le langage de programmation choisi, les différentes parties qui le constitue.

Nous consacrerons la dernière partie à la programmation de toutes les tâches de production toute faisant intervenir les ordres de forçages et d'initialisation émanant de GRAFCET de sécurité, de conduite et de coordination des tâches.

IV -2- Présentation du langage de programmation :

IV- 2-1 Langage de programmation GRAPH

Le langage de programmation GRAPH s'ajoute à l'éventail des fonctions de STEP 7. Il permet de programmer graphiquement les commandes séquentielles. Ces commandes séquentielles pourront être ensuite commandées à l'aide d'un automate programmable SIMATIC.

Les actions à exécuter sont associées aux étapes, tandis que des transitions régissent l'évolution entre deux étapes successives (conditions de franchissement). Pour définir les réceptivités des transitions ainsi que les verrouillages ou les surveillances d'étape, vous aurez à utiliser un nombre restreint d'éléments du langage de programmation CONT (schéma à contacts) ou LOG (logigramme).

IV -2-2 Blocs de la commande séquentielle

Une commande séquentielle conduit le processus dans un ordre prédéfini et en fonction de certaines conditions. Le nombre de blocs de la commande séquentielle dépend de la complexité de la tâche d'automatisation à réaliser. Mais elle doit comporter toujours au moins les trois blocs suivants :

- un bloc STEP 7 (1) dans lequel le FB GRAPH est appelé ; il peut s'agir d'un OB, d'une FC ou d'un autre FB ;
- un FB GRAPH (2) contenant un ou plusieurs graphes séquentiels, avec leurs étapes et leurs transitions ainsi que les actions et les réceptivités qui y sont associées ;
- un DB d'instance (3) contenant les données et les paramètres de la commande

séquentielle ; il est associé au FB GRAPH et peut être généré automatiquement par le système.

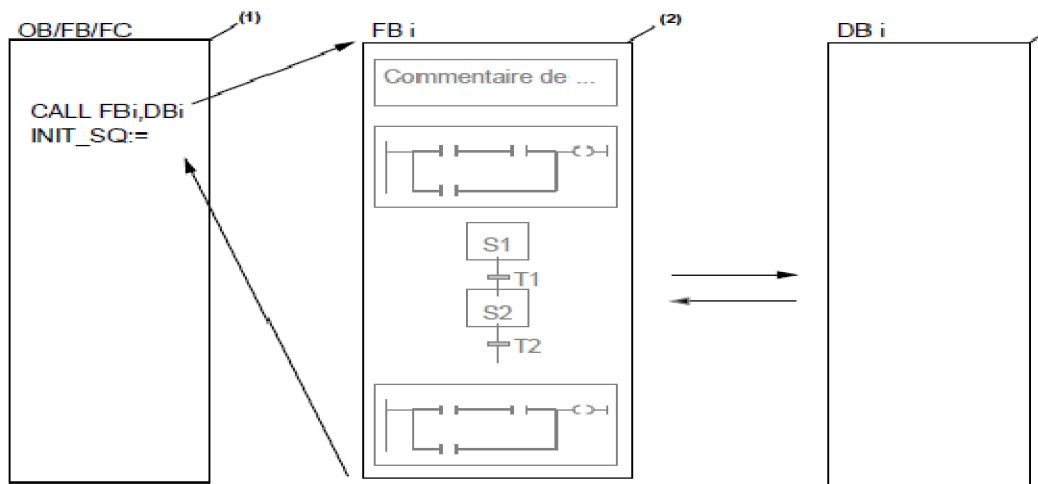
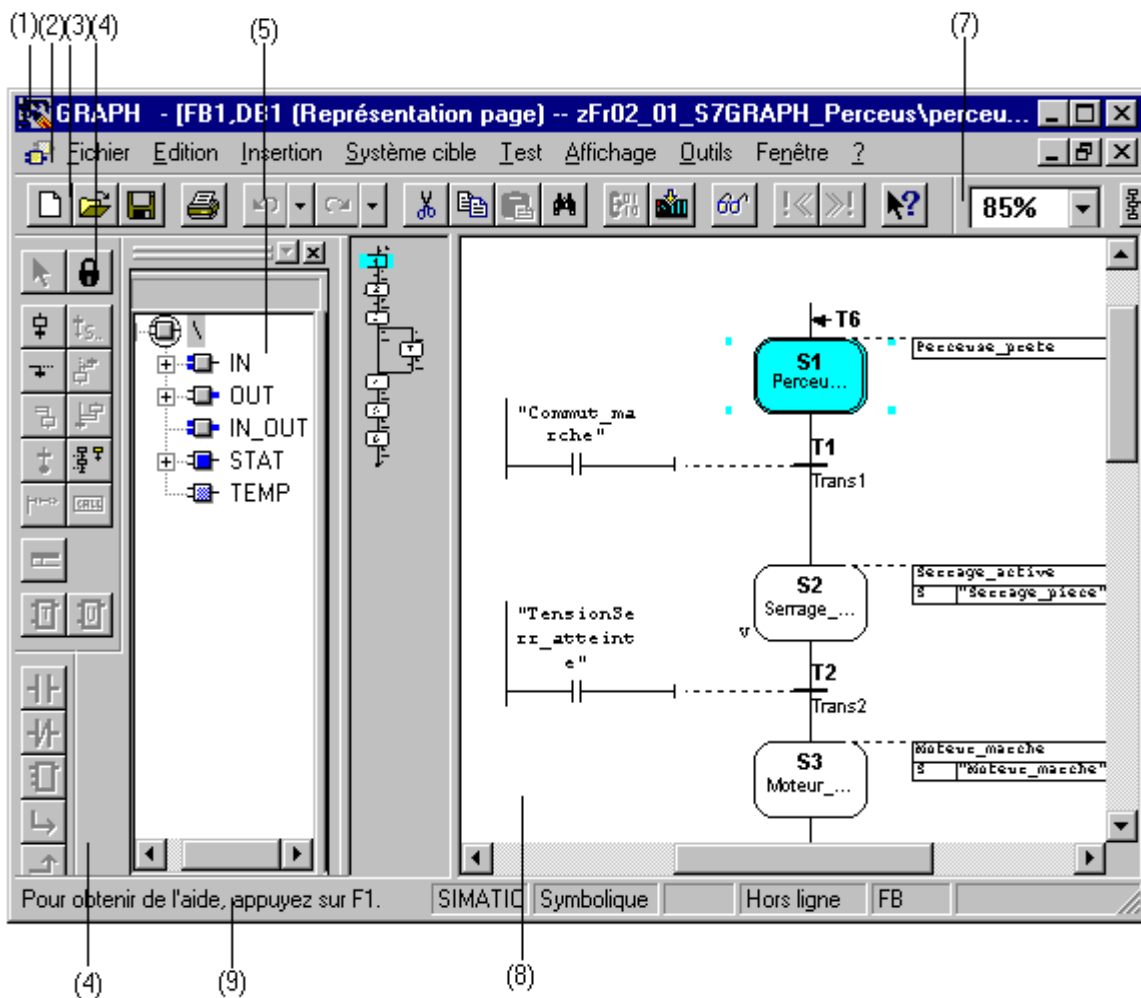


Figure IV. 1: Blocs de la commande séquentielle.

IV -2-3 Interface utilisateur de GRAPH



- barre de titre de l'éditeur GRAPH (1),
- barre des menus (2),
- barres d'outils ou d'éléments (3), (4), (6), (7),
- fenêtre de déclaration de variables (5),
- zone de travail (8),
- barre d'état (9),

Figure IV. 2: Interface utilisateur de GRAPH.

IV -2-4 Création d'un FB GRAPH

On peut créer un FB GRAPH soit dans SIMATIC Manager, soit directement dans GRAPH.

- Dans GRAPH, on crée un nouveau bloc fonctionnel avec la commande **Fichier > Nouveau**.

Il est créé dans le langage de programmation GRAPH.

- Dans SIMATIC Manager, on crée un nouveau bloc fonctionnel avec la commande **Insertion > Blocs S7 > Bloc fonctionnel**. On choisit le langage de programmation GRAPH.

La figure(IV.3) explique la méthode de création d'un FB GRAPH.

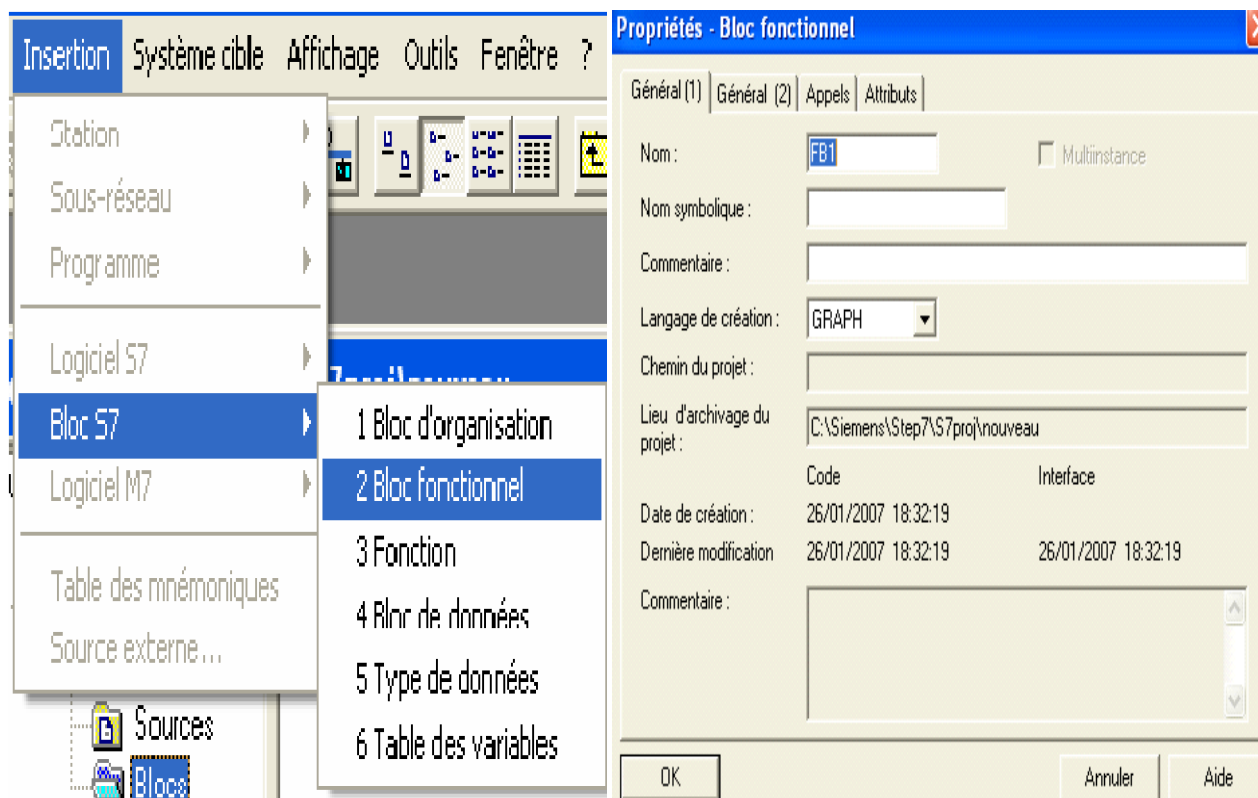


Figure IV. 3: Méthode de création d'un FB GRAPH.

IV-2-4-1 Création de la table des mnémoniques

La figure(IV.4) donne une aperçu sur la méthode de création de la table des mnémoniques.

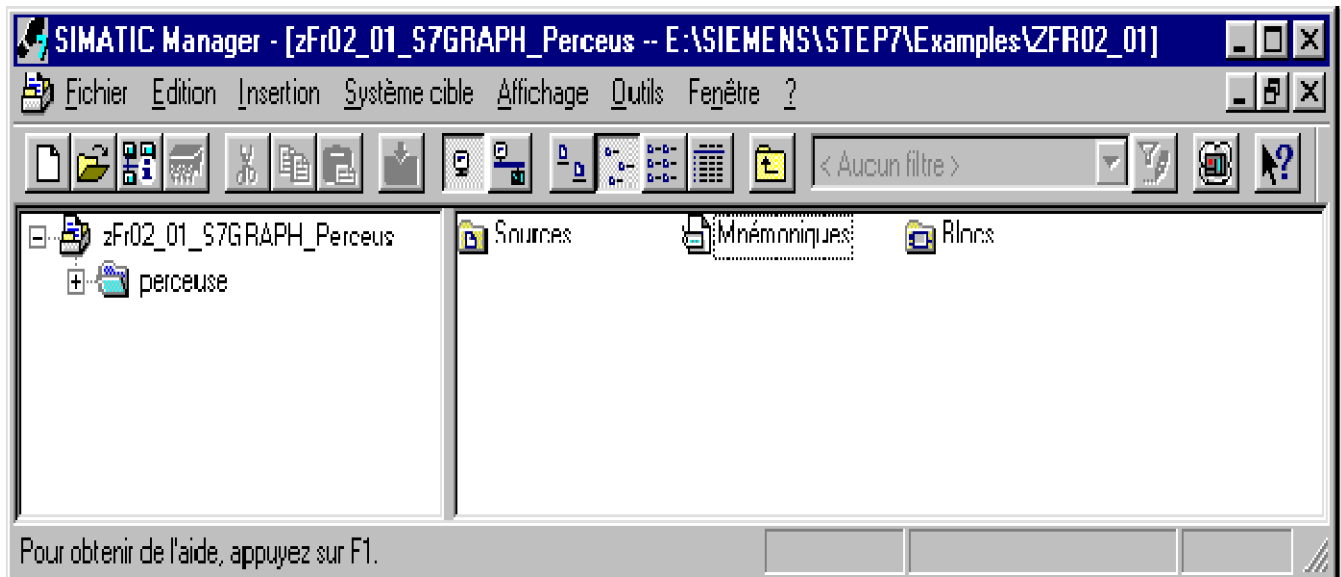


Figure IV. 4: Méthode de création de mnémonique.

IV- 2-4-2 Remplissage de la table des mnémoniques

La figure(IV.5) explique comment remplir la table des mnémoniques.

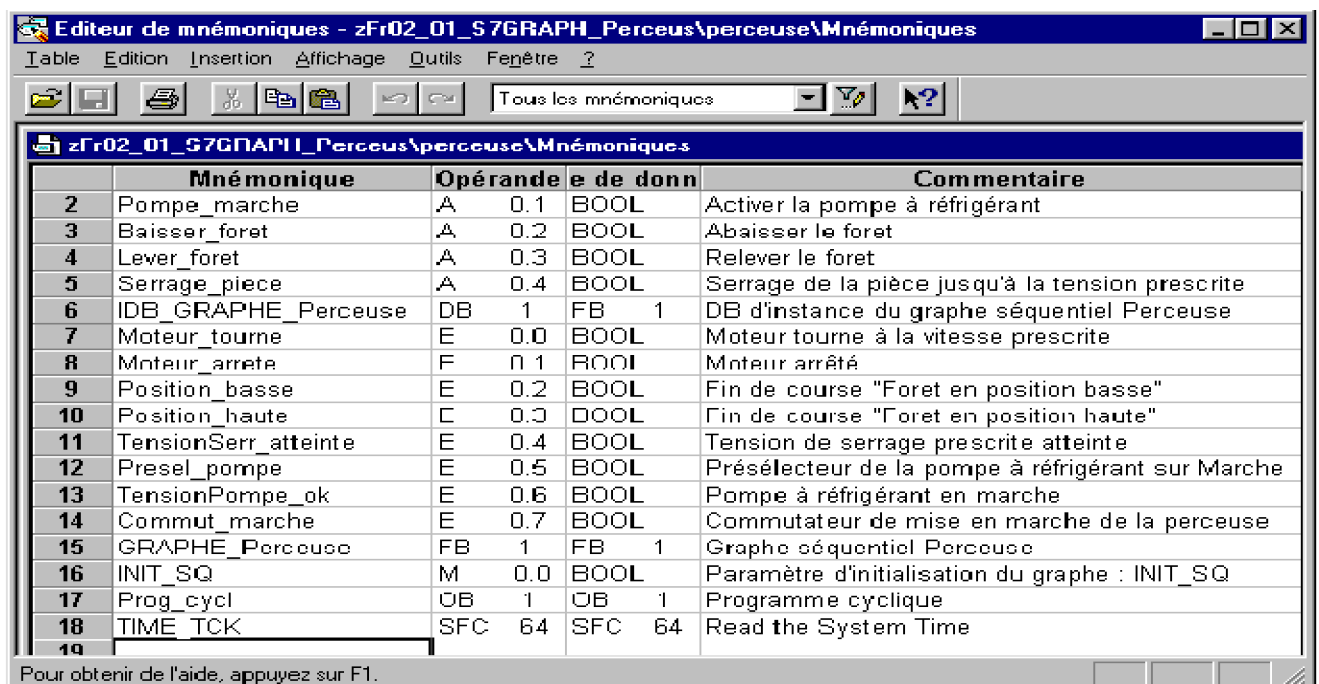


Figure IV. 5: Remplir une table des mnémoniques.

IV- 2-5 Création du graphe séquentiel

Il ya deux méthodes d'insertion **mode d'insertion directe** et **mode Présélection**.

IV- 2-5-1 Mode d'insertion directe

On sélectionnez la transition 1 et cliquons sur l'icône :



Insérer une étape+transition.



Ouvrir branche OU.



Fermer branche OU.



Insérer un saut.

IV- 2-5-2 Mode d'insertion présélection

On choisissons la commande **insertion> présélection** avec la souris on sélectionnons :



Insérer une étape+transition.



Ouvrir branche OU.



Fermer branche OU.



Insérer un saut.

IV-3 Programmation des tâches :

IV-3-1 Configuration du matériel :

Avec les méthode de création du programme et de choix du rack , module d'alimentation et de CPU qui sont expliqués en détail dans le 2ème chapitre, on configure notre automate comme suit :

- RACK-300.

-CPU 314.

-Module d'alimentation : PS-307 5A.

-Module d'entrée : Une embase de 32 entrées numériques (32 DI).

-Module de sortie : Une embase de 32 sorties numériques (32 DO).

IV-3-1 Tables des mnémoniques :

la table des mnémonique est donné comme suit :

	Etat	Mnémonique ^	Opérande	Type de d	Commentaire
1		2PF	E 4.2	BOOL	les 2 portes fermées
2		4PF	E 4.5	BOOL	les 4 portes fermées
3		ALARM	A 0.1	BOOL	démarrage de l'alarme
4		ALARM.	A 1.7	BOOL	démarrage de l'alarme
5		ALARM_S	SFC 18	SFC 18	Generate Permanently Acknowledged Block-Related Messages
6		ALARM_SQ	SFC 17	SFC 17	Generate Block-Related Messages with Acknowledgment
7		ARD	E 8.3	BOOL	arrêt d'urgence
8		AUTO	E 9.1	BOOL	MARCHE EN AUTO
9		B1B9	E 4.6	BOOL	capteur de présencede matière
10		B1S1	E 4.7	BOOL	capteur de fin de coupure 1
11		B1S2	E 5.0	BOOL	capteur de fin de coupure 2
12		BA	A 5.1	BOOL	briquet alumée
13		CD10	A 3.2	BOOL	convoyeur
14		CD11	A 3.3	BOOL	convoyeur
15		CF10	A 2.0	BOOL	convoyeur
16		CF11	A 2.1	BOOL	convoyeur
17		CF12	A 2.2	BOOL	convoyeur
18		CF13	A 2.3	BOOL	convoyeur
19		CF14	A 2.4	BOOL	convoyeur
20		CI	E 9.3	BOOL	les conditions initiales
21		CP12	A 1.4	BOOL	convoyeur
22		CP13	A 1.3	BOOL	convoyeur
23		CP14	A 1.1	BOOL	convoyeur
24		CP15	A 1.0	BOOL	convoyeur
25		CP16	A 0.7	BOOL	convoyeur
26		CP18	A 0.6	BOOL	convoyeur
27		CTB	E 6.2	BOOL	capteur de translation du balanceur de brique
28		Cycle Execution	OB 1	OB 1	
29		DB-graph	DB 1	FB 1	DB d'instance du graph
30		DEF	E 8.2	BOOL	défauts
31		DEP1	E 0.3	BOOL	bouton de démarrage de tâche 1
32		DEP2	E 1.3	BOOL	bouton de démarrage de tâche 2
33		DEP3	E 3.7	BOOL	bouton de démarrage de tâche 3
34		DEP4	E 5.3	BOOL	bouton de démarrage de tâche 4
35		DEP5	E 6.5	BOOL	bouton de démarrage de tâche 5

36	DT	A	6.6	BOOL	démarrage transporteur
37	DT1	A	1.5	BOOL	démarrage distr1
38	DT11	A	2.6	BOOL	démarrage T11
39	DT2	A	1.6	BOOL	démarrage distr2
40	EV	A	0.4	BOOL	électrovanne d'eau
41	EV.	A	2.7	BOOL	électrovanne
42	EVA	A	5.0	BOOL	électrovanne d'air de combustion
43	EVG	A	6.2	BOOL	électrovanne du gaz
44	EVGA	A	4.7	BOOL	électrovanne du gaz
45	Fa	E	8.0	BOOL	fleuret actionné
46	FA.	A	6.5	BOOL	démarrage de fleuret
47	FC	E	4.3	BOOL	fin de coupure
48	G7_STD_3	FC	72	FC 72	
49	graphe prép	FB	1	FB 1	graphe séquentiel préparation
50	INIT_SQ	M	0.0	BOOL	initialisation du graph
51	Ja	E	7.7	BOOL	jet actionné
52	JA.	A	6.4	BOOL	démarrage de jet
53	M1	A	3.5	BOOL	moteur 1
54	M2	A	4.1	BOOL	moteur 2
55	M3	A	4.2	BOOL	moteur 3
56	MAC	A	5.3	BOOL	moteur d'air de combustion
57	MACA	E	6.6	BOOL	mac actionné
58	MAD	A	4.6	BOOL	moteur d'admission
59	MAN	E	9.2	BOOL	MANUEL
60	MB	A	0.3	BOOL	moteur broilleur
61	MBB	A	5.2	BOOL	moteur balanceur de brique
62	MBF	A	3.1	BOOL	moteur broyeur finisseur
63	MBP	A	2.5	BOOL	moteur broyeur préfinisseur
64	MD	A	0.2	BOOL	moteur de désagrégateur
65	ME	A	5.4	BOOL	moteur d'extraction d'air
66	MEA	E	6.7	BOOL	ME actionné
67	MM	A	0.5	BOOL	moteur moelleur
68	MMAL	A	3.0	BOOL	moteur malaxeur
69	MMOL	A	3.4	BOOL	moteur moleuse
70	MR	A	1.2	BOOL	moteur de répartiteur
71	MR.	A	4.5	BOOL	moteur de récupération d'air
72	MRA	E	5.6	BOOL	moteur de récupération d'air
73	MRA.	A	5.7	BOOL	moteur de récupération d'air actionné
74	MRAA	E	7.2	BOOL	moteur de récupération d'air actionné
75	MRF	A	6.1	BOOL	moteur de refroidissement
76	MRFA	E	7.4	BOOL	moteur de refroidissement actionné
77	MSP	A	5.6	BOOL	moteur de surpression
78	MSPA	E	7.1	BOOL	moteur de surpression actionné
79	MSW	A	6.0	BOOL	moteur de soufflage wagon
80	MSWA	E	7.3	BOOL	moteur de soufflage wagon actionné
81	MTF	A	5.5	BOOL	moteur tirage feur
82	MTFA	E	7.0	BOOL	moteur tirage feur actionné
83	MTR	A	4.0	BOOL	moteur de tapie roulant
84	PAC	E	6.0	BOOL	présence d'air de combustion
85	Pair	E	0.4	BOOL	présostat de présence d'air
86	Parg	E	2.2	BOOL	présence d'argile
87	Pe	E	0.5	BOOL	capteur de présence d'eau
88	Pe.	E	2.1	BOOL	capteur de présence d'eau
89	PF	E	4.1	BOOL	présence du fil
90	PG	E	7.5	BOOL	présence du gaz
91	PGAZ	E	5.7	BOOL	présence du gaz
92	PGG	A	6.3	BOOL	démarrage de pamoplie du gaz
93	PGGA	E	7.6	BOOL	PGG actionné
94	PM1	E	4.0	BOOL	présence de matière au niveau 1
95	PM2	E	4.4	BOOL	présence de matière au niveau 2
96	Pt	E	1.4	BOOL	présence terre
97	RBP	E	1.6	BOOL	racleur de broyeur préfinisseur
98	RD1	E	0.7	BOOL	racleur de distributeur1 act
99	RD2	E	1.0	BOOL	racleur de distributeur2 act
100	RRPA	E	0.6	BOOL	racleur de répartiteur actionnée
101	RT11	E	1.7	BOOL	racleur de T11
102	S3	E	8.1	BOOL	réarement
103	T14B	E	1.5	BOOL	trémie 14 en niveau bas
104	T14H	E	2.0	BOOL	trémie 14 en niveau haut
105	TEMPCAD	E	6.1	BOOL	température du canal d'admission
106	TIME_TCK	SFC	64	SFC 64	Read the System Time
107	V1	A	3.6	BOOL	vérin 1
108	V2	A	3.7	BOOL	vérin 2
109	VN1	A	4.3	BOOL	ventilateur 1
110	VN1A	E	5.4	BOOL	ventilateur 1 actionné
111	VN2	A	4.4	BOOL	ventilateur 2
112	VN2A	E	5.5	BOOL	ventilateur 2 actionné
113	WR_USMSG	SFC	52	SFC 52	Write a User-Defined Diagnostic Event to the Diagnostic Buffer

Table IV. 1: Table des mnémoniques.

IV-3-2 Programmation du GRAFCET de sécurité :

Le programme du GRAFCET de sécurité est donné par la figure (fig. IV.6).

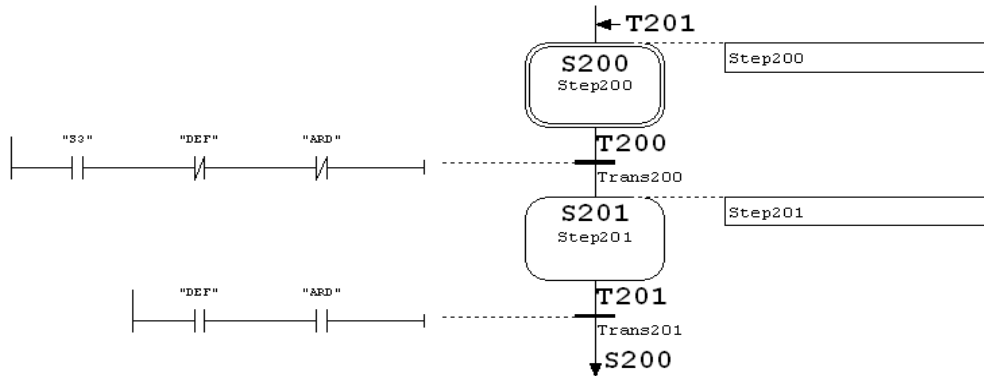


Figure IV. 6: Programme du GRAFCET de sécurité.

IV-3-4 Programmation du GRAFCET de conduite et Initialisation :

Le GCI est illustré par la figure (fig IV.7).

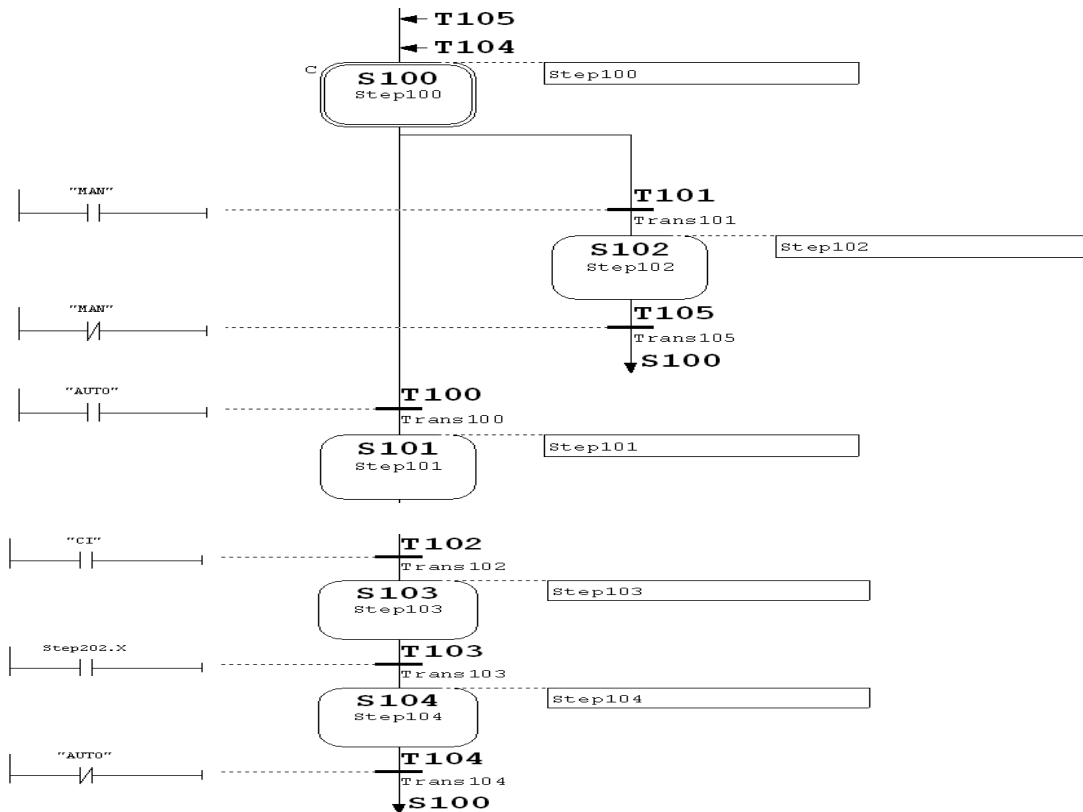


Figure IV. 7: Programme du GCI.

IV-3-5 Programmation du GPN :

le GRAFCET de production normale (GPN) est présenté par la figure : fig(IV.8).

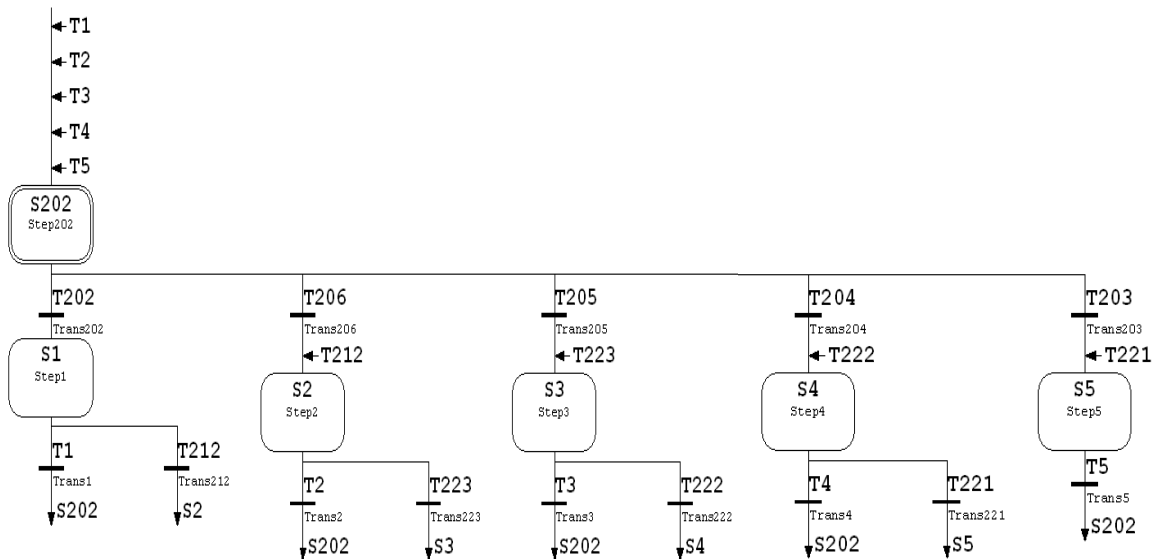
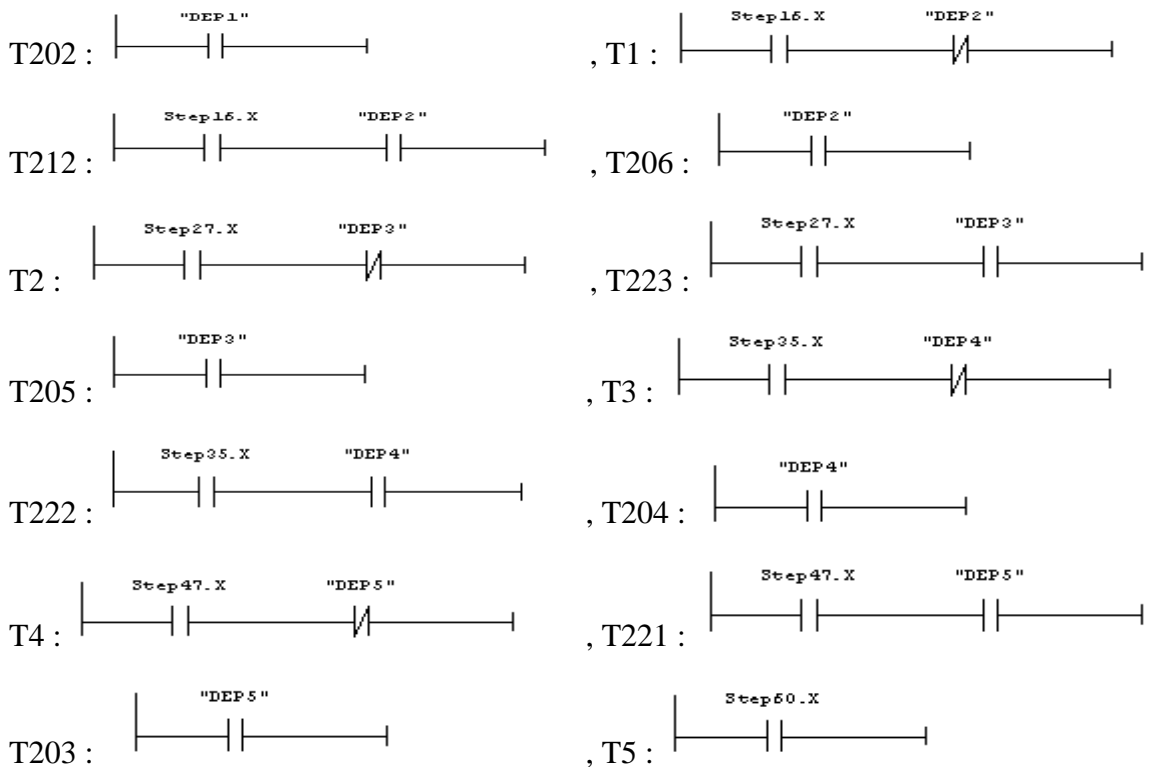


Figure IV. 8: Programme du GPN.

Tel que :



IV-3-5 Programmation de tâche de préparation :

La figure (fig IV.9) présente le programme de la tâche de préparation

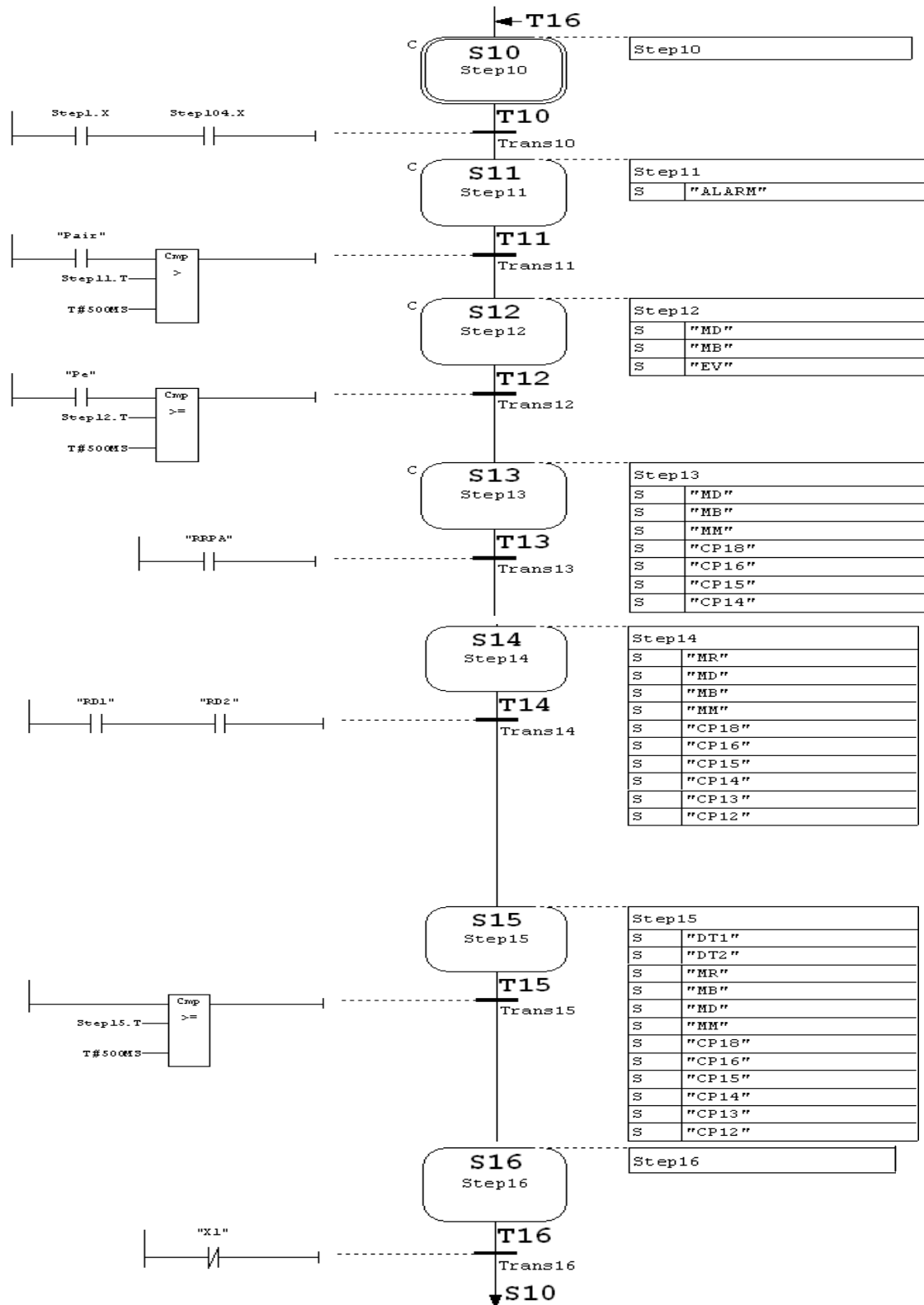


Figure IV.9 : Programme de tâche de préparation.

IV-3-6 Programmation de tâche de fabrication :

Le programme de la tâche de fabrication est illustré par la figure : fig(IV.10).

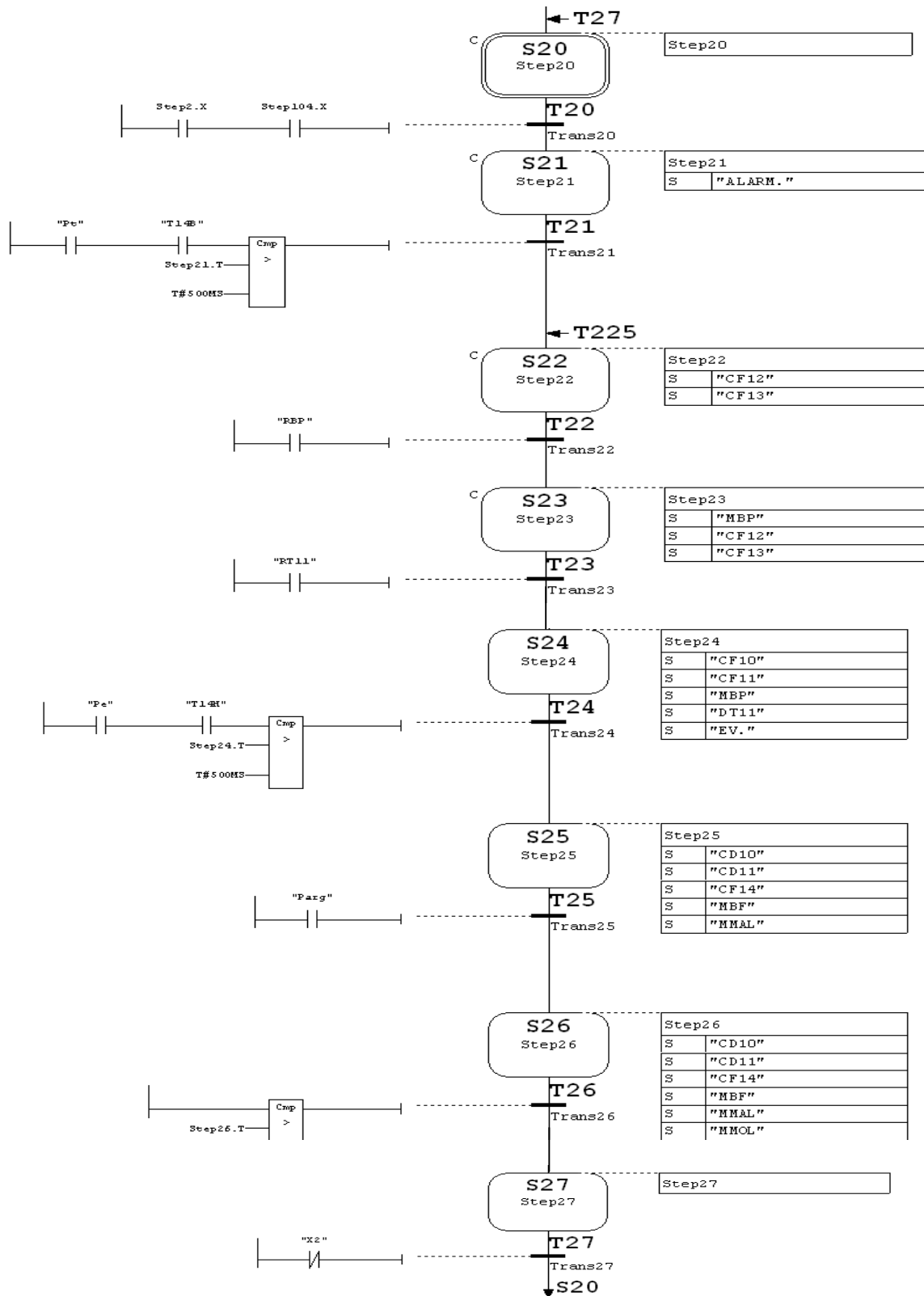


Figure IV. 10: Programme de tâche de fabrication.

IV-3-7 Programmation de tâche de coupure :

Le programme de la tâche de coupure est illustré par la figure suivante :

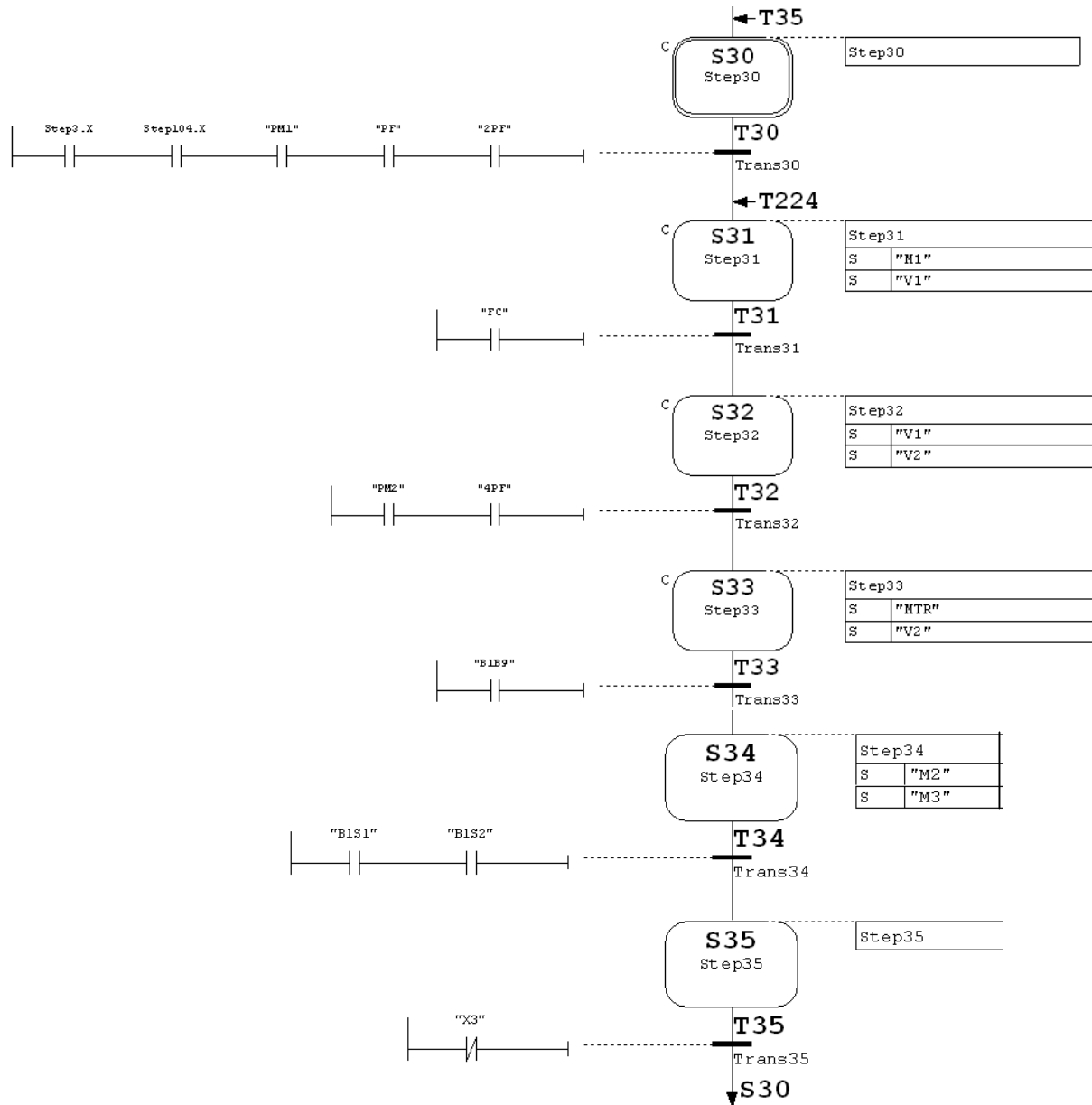


Figure IV. 11: Programme de tâche de coupure.

IV-3-8 Programmation de tâche de séchage :

Le programme de la tâche de séchage est réalisé par la figure : fig(IV.12).

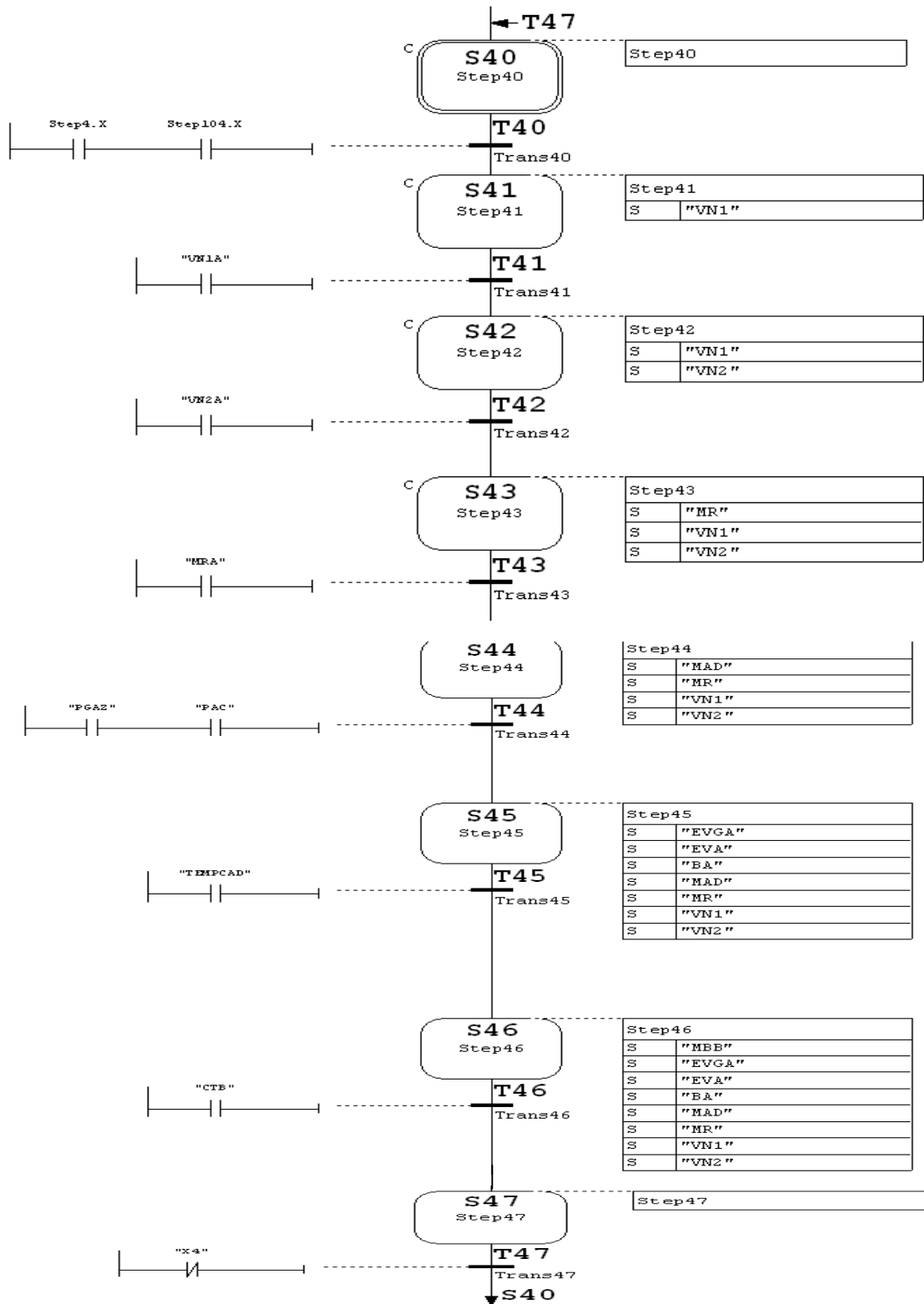
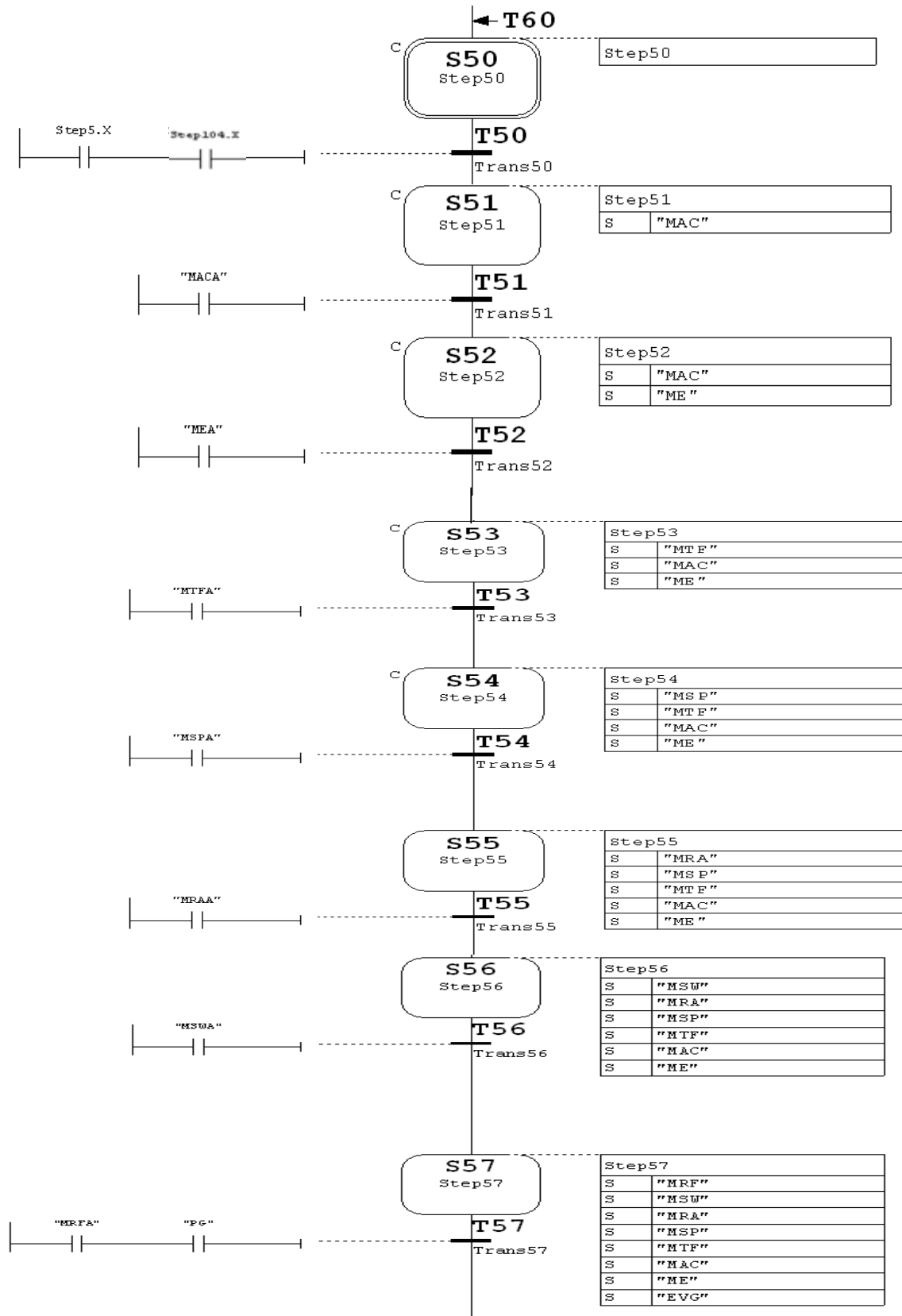


Figure IV. 12: Programme de tâche de séchage.

IV-3-9 Programmation de tâche de cuisson :

Le programme de la tâche de cuisson est donné par la figure : (fig IV.13).



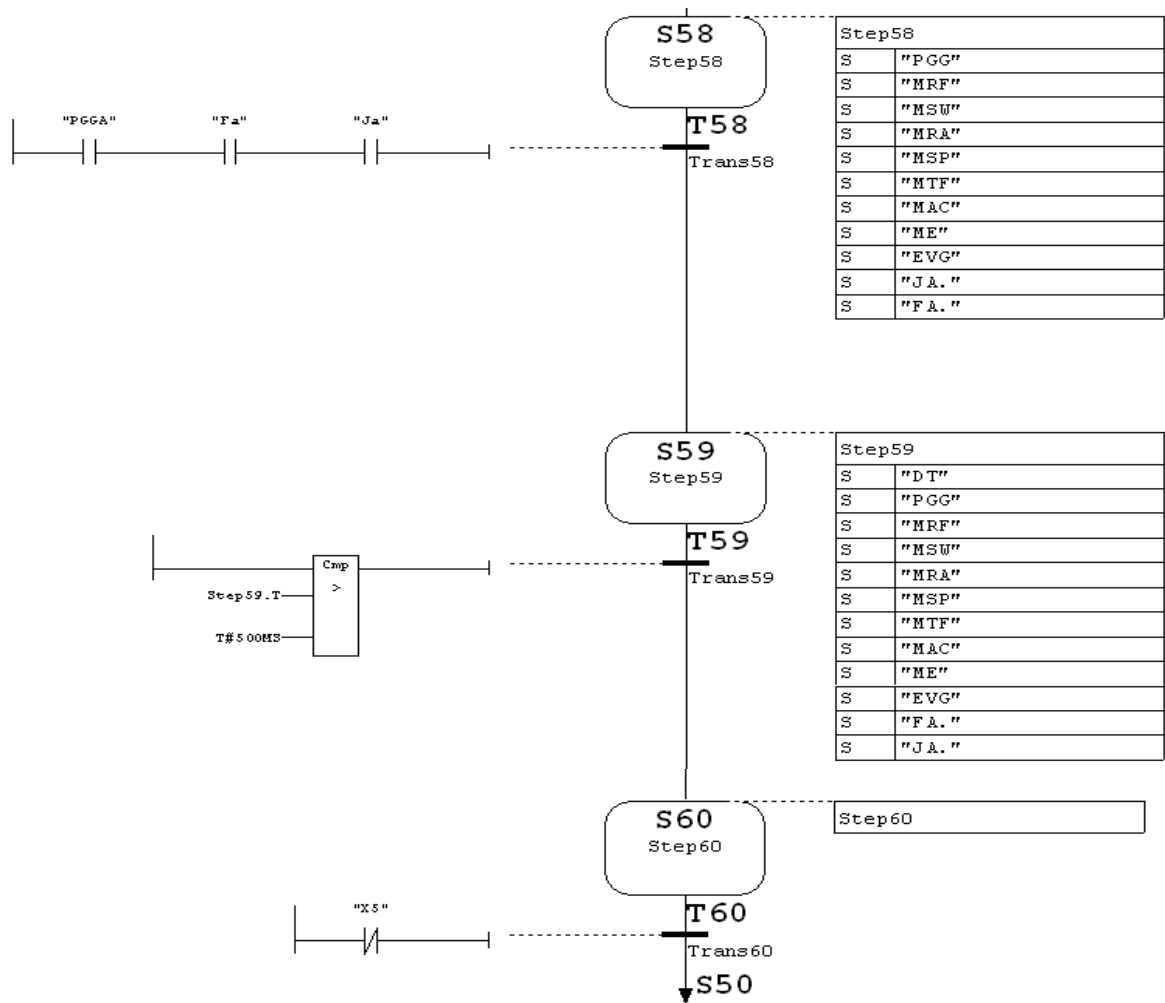


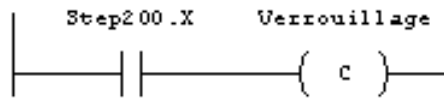
Figure IV. 13: Programme de tâche de cuisson.

IV-3-10 Programmation de relation entre les GRAFCETS (GS,GCI,GPN)

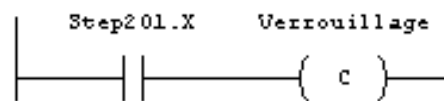
Pour chaque GRAFCET de production on programme le verrouillage des étapes, comme suit :

1-GS :

-Pour les étapes initiales :

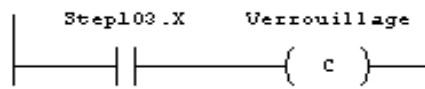


-pour les autres étapes :

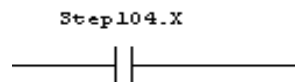


2-GCI :

-pour l'initialisation :



-pour la marche en auto



IV-4 Conclusion :

Dans ce chapitre nous avons exposé une étude de programmation d'un système automatisé industriel de fabrication de brique (BRIQUETTERIE) à l'aide de logiciel STEP7 et nous sommes arrivés à mettre en évidence l'intérêt de l'utilisation des GRAFCETS hiérarchisés pour la résolution des système de production importants , car ses derniers présente un nombre important de variable d'entrées et de sorties d'où la nécessité de subdivisé le cycle de production en plusieurs tâche, puis nous avons programmé par automates programmables industriels (API), sous le logiciel STEP7 avec le langage GRAPH, toutes les tâches ainsi que l'introduction des GRAFCET de sécurité (GS), commandes et initialisation (GCI) et production normale (GPN)

CONCLUSION GENERALE

CONCLUSION GENERALE :

Compte tenu des objectifs définis au début de ce mémoire, la première partie de notre travail a été consacrée à la recherche bibliographique sur les systèmes automatisés et sur les automates programmable industriel (API) du point de vue constitution et différents langages de programmations utilisés par les API .

Partant du fait que la firme SIEMENS est l'une des entreprise leader dans le domaine de l'automatisation industriel , ces automates sont gérés par le logiciel, la deuxième partie a été utilisé pour la description des éléments de bases du logiciel STEP7, les langages de programmation, et aux étapes de programmations

Dans la troisième partie, on a présenté l'usine de KALAA BENI-HAMMADE, les conditions de fonctionnement, les processus de fabrication.et on a proposé la solution graphique des différentes tâches du cycle de fabrication de la brique.

Quant à la quatrième partie, elle a été consacrée, à la programmation sous STEP des différentes tâches tout en introduisant le principe des GRAFCET hiérarchisé, c'est a dire introduction des GRAFCET de sécurité (GS), commandes et initialisation (GCI) et production normale (GPN)

Une continuité éventuelle de ce travail peut être envisagée, pour l'amélioration de la solution proposée par GRAFCET HIERARCHISE,

- *intégration du GEMMA « Guide d'Etude de Mode de Marches et d'Arrêts » pour plus de perfectionnement;*
- *utilisation d'autre type de programmation;*
- *Conception de prototype de systèmes automatisé miniatures ;*
- *Amélioration des connaissances théoriques par des stages de courte et moyenne durée dans les unités de productions commandées par automates programmable.*

BIBLIOGRAPHIE

BIBLIOGRAPHIE

Chapitre I :automatisme et automate programmable

- [1] **BENARIBAMMAR. BELMILOUD ABD ELAZIZ**, « Étude de réalisation d'un API a bas d'un microcontrôleur » Mémoire d'ingénieur, université, de M'sila 2006.
- [2] **BENYAHIA ISMAHANE. BOUMELIT ASMA**, « Application du GRAFCET pour l'automatisation d'une remplisseuse d'un liquide »,Mémoire du technicien Supérieur, I.N.S.F.P.BATNA, MARS 2008.
- [3] **C.BOURBOUNNE. J.COJEAN**, « Les systèmes automatisés de la connaissance à la conception », Tome 1, les éditions, Foucher, 1984.
- [4] **HAMDI Hocine**, « Automatismes logiques », Les éditions de l'université Mentouri Constantine, 1999.
- [5] **Daniel BOUTEILLE**, «les automatismes programmables », cepadues éditions, 2 édition, 1988.
- [6] **MICHEL BERTRAND**, «Automate programmable industrielle » Docteur-Ingénieur. École Nationale Supérieure, d'Arts et Métiers ENSAM, Centre d'Enseignement et de Recherche de Lille.
- [7] **TECHNIQUES DE L'INGENIEUR**, «Les systèmes automatise »,., « Langage de programmation pour API norme IEC 1131-3 ».
- [8] **G. CHEVALIER**, « le GRAFCET et les fonction d'automatisme », dunod,1981.
- [9] **H. Ney**, « automatisme et informatique industriel », nathan , 1998.
- [10] **Nourddine ATALLAOUI, Rachid FRAHTIA**, «automatisation d'une petite unité de production par la résolution GRAFCET» Mémoire d'ingénieur, université de M'sila 2008.
- [11] **Jargot P.**, « Langages de Programmation pour API. Norme IEC 1131-3. », Techniques de l'Ingénieur. S 8 030, 2006.,
- [12] **Manuel SIEMENS**, Programmation avec STEP7., 2000
- [13] **Manuel SEMENS**, *STEP7 PLCSIM*, Testez vos Programmes, 2002.

ANNEXES

Annexes

Programme de la tâche de préparation par PL7

(* A ENLEVER APRES ESSAIS *)

(*coupure secteur*)

IF %S0 OR %S1 THEN SET

%MW1000:X0;%MW4048:=%MW4047;%MW18[%KW15]:=%KW14;%MW18[%KW115]:=%KW114;

%MW221:=0;%MW321:=0;

END_IF;

(*recopie du bit systeme seconde *)

%M34:=%S6;

(*tempo flip flop toutes les 2 secondes*)

IF RE %M34 THEN DEC %MW4040;

END_IF;

(* flip flop *)

IF(%MW4040<=0)AND %M35 THEN RESET %M35;%MW4040:=%MW4039;

END_IF;

(* flip flop *)

IF(%MW4040<=0)AND NOT %M35 THEN SET %M35;%MW4040:=%MW4039;

END_IF;

DEBUT GESTION PREPARATION *)

%L90:

IF(%MW4215>0)THEN DEC %MW4215;

END_IF;

(* tps arret unite centrale *)

(*acquit Klaxon ==> BP"Acquit Klaxon" ou Mode Manuel et Position 0 des sélecteurs + BP"Marche"*)

IF(%I\1.2.3\1.15 AND NOT %MW4405:X0)OR

(%M21 AND((%MW4115=0 AND %I\1.2.4\0.4)OR(%MW4116=0 AND %I\1.2.4\0.12)OR(%MW4117=0 AND %I\1.2.4\1.4)))

THEN %MW1004:=0;

END_IF;

(* mémoire acquit klaxon *)

%MW4405:X0:=%I\1.2.3\1.15;

(* gestion des défauts sr40 *)

SR40;

(* acquit des défauts ==> BP"Acquit Défauts" ou Mode Manuel et Position 0 des sélecteurs + BP"Marche"* *)

IF(%I\1.2.3\1.14 AND NOT %MW4405:X1)OR(%MW4117=14 AND %I\1.2.4\1.6)OR

(%M21 AND((%MW4115=0 AND %I\1.2.4\0.4)OR(%MW4116=0 AND %I\1.2.4\0.12)OR(%MW4117=0 AND %I\1.2.4\1.4)))

THEN %MW4405:X1:=%I\1.2.3\1.14;

ELSE JUMP %L120;

END_IF;

(* acquit des défauts des mots d'état et effacement défauts Magelis*)

%MW1000:50:=16#7000 AND %MW1000:50;%MW4280:=0;

RESET %M24;RESET %M26;%MW227:3:=0;%MW327:3:=0;

IF %MW230:X0 THEN RESET %MW230:X0;RESET %MW330:X0;END_IF;

IF %MW230:X1 THEN RESET %MW230:X1;RESET %MW330:X1;END_IF;

RESET %M25;RESET %M27;%MW4015:=0;%MW4405:X1:=%I\1.2.3\1.14;

JUMP %L190;

%L120:(*memoire acquit défauts *)

%MW4405:X1:=%I\1.2.3\1.14;

(* general en service *)

%MW1000:X12:=(%MW1000:X13 OR %MW1000:X14)AND NOT %MW1000:X1 AND NOT %MW1004:X2;

(* zones en manu *)

IF %MW1000:X13 THEN %MW1005:44:=%MW1005:44 AND 16#9FFF;%MW1005:44:=%MW1005:44 OR 16#2000;

END_IF;

(* zones en auto *)

IF %MW1000:X14 THEN %MW1005:44:=%MW1005:44 AND 16#9FFF;%MW1005:44:=%MW1005:44 OR 16#4000;

END_IF;

(* zones en stop *)

IF NOT %MW1000:X13 AND NOT %MW1000:X14 THEN %MW1005:44:=%MW1005:44 AND 16#9FFF;

END_IF;

(* ----- FIN GESTION PREPARATION ----- *)

%L190:

(* gestion des moteurs SR1 et des alarmes *)

SR5;SR1;SR0;SR3;

(* Mémorisation du bit d'alimentation Navette *)

%M5:=%M6;

%L300:(* mise a jour des sorties *)

IF %MW4048>0 THEN DEC %MW4048;SET %S49;ELSE RESET %S49;

END_IF;

%L999:

MEMOIRE DE FIN D'ETUDES EN VUE DE L'OBTENTION DU DIPLOME
DE MASTER EN GENIE ELECTRIQUE
OPTION: COMMANDE DES SYSTEMES ELECTRIQUES

THEME

**COMMANDE D'UN SYSTEME AUTOMATISE DE PRODUCTION (SAP)
PAR AUTOMATE PROGRAMMABLE INDUSTRIEL (API)
« BRIQUETERIE D'EL KALAA BENI HAMMAD -M'SILA »**

Proposé et dirigé par :

Mr : Abdelhak ABDOU

Présenté par :

Lahcen. GUERRAS

RESUME:

L'automatisation est considérée comme l'étape d'un progrès technique où apparaissent des dispositifs techniques susceptibles de seconder l'homme, non seulement dans ses efforts musculaires, mais également dans son travail intellectuel de surveillance et de contrôle.

La réalisation et l'exploitation des automatismes industriels font appel à divers outils, matériels et logiciels, qui doivent être convenablement choisis et utilisés, en vue de remplir certaines fonctions, parmi ces outils on distingue l'Automate Programmable Industriel (API).

Dans ce mémoire on a présenté d'une manière générale les systèmes automatisés et les principes de base du GRAFCET hiérarchisé. A partir d'un cahier de charge d'une unité de production de brique, on est arrivé en premier lieu à la solution graphique par GRAFCET hiérarchisé, puis on a programmé les différentes tâches du cycle de production par API sous le logiciel STEP7.

MOTS CLES :

- **Système automatisé;**
- **GRAFCET hiérarchisé;**
- **Automate Programmable Industriel (API);**
- **STEP7.**