

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



**UNIVERSITE DE M'SILA**

**FACULTE DES SCIENCES ET SCIENCES DE L'INGENIEUR**

**DÉPARTEMENT D'ÉLECTROTECHNIQUE**

**MEMOIRE DE FIN D'ETUDES EN VUE DE L'OBTENTION DU DIPLÔME**  
**D'INGENIEUR D'ETAT EN GENIE ÉLECTROTECHNIQUE**

**OPTION: ELECTROMECHANIQUE**

**THÈME**

**RÉSOLUTION PAR GRAFCET ÉVOLUÉ**  
**D'UN SYSTÈME AUTOMATISÉ INDUSTRIEL**  
**« BRIQUETERIE D'EL KALAA BENI HAMMAD - M'SILA »**

**Proposé et dirigé par :**

**Mr. ABDOU ABDEALHAK.**

**Présenté par :**

**Mr. LASSELAT RACHID**  
**Mr. BENIA ALI**

**ANNÉE UNIVERSITAIRE : 2008/2009**

# *Dédicaces*

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

*À mes frères À mes très chères  
soeurs*

*À toute ma famille*

*À tous mes amis*

*Benia Ali*

# *Dédicaces*

*À Ma chère Mère, pour ses  
sacrifices depuis qu'elle m'a mis au  
monde, Mon Père, qui m'a  
toujours soutenu et aidé à  
affronter les difficultés*

*A mes frères  
Et tous mes soeurs*

*À toute ma famille*

*À tous mes amis*

*Lasselat Rachid*

# Remerciements

*Cette dernière période fut enrichissante et pleines d'activité grâce au bon " Dieu " tout puissant, qui nous a donné volonté, patience et santé. Nous avons eu la chance d'évoluer parmi des personnes qui nous ont toujours assuré de leur soutien, nous tenons à remercier très sincèrement :*

*Nous exprimons notre grand remerciement à notre encadreur Mr. Abdelhak Abdou pour son aide, ses conseils, ses encouragements, sa patience.*

*Nous tenons à remercier chaleureusement monsieur le directeur du département de l'électrotechnique Mr. Bachir BENJAIMA et tous les enseignants du département, particulièrement messieurs : Salim CHECKROUN, Saïd BARKATI, Yahia LAAMARI, Mohamed BEDBOUDI, Djalal Eddine Khodja, Abdelmadjid CHOUCOU, Abdelhalim KESSAL, Fethi CHOUAF, Salah BELKHIRI et H.HAMIMIDE.*

*Sans oublier le directeur Mr. AYACHE qui nous autorisé à faire un stage et tous les personnes de la Briqueterie d'El KALAA BENI - HAMMAD, pour ces sacrifices et toutes les choses qu'il mes donne pendant notre stage, surtout Mr. Abdelnnour, Mr. Djailani ,Mr. Azelddine.*

*Nous ne pouvons terminer ces remerciements sans mentionner, tous mos amis de la promotion d'ingénieur 2008/2009 du département électrotechnique et informatique.*

*R.Lasselat & A.Benia*

# Sommaire

## Introduction générale

Introduction générale.....	1
----------------------------	---

## CHAPITRE I : L'AUTOMATE ET L' AUTOMATE PROGRAMMABLE INDUSTRIEL (API)

I.1 Introduction.....	2
I.2 Notion de système automatisé.....	2
I.2.1 Point de vue historique.....	2
I.2.2 Point de vue technique.....	2
I.2.3 Point de vue économique et social.....	2
I.3 L'automatisation.....	3
I.3.1 Les objectifs de l'automatisation.....	3
I.3.1.1 Produire à qualité constante.....	3
I.3.1.2 Fournir les quantités nécessaires.....	3
I.3.1.3 Augmenter la productivité.....	3
I.3.1.4 Améliorer les conditions de travail.....	3
I.3.2 Structure d'un système automatisé.....	4
I.3.2.1 Partie Opérative.....	4
I.3.2.2 Partie Commande.....	4
I.4 Système automatisé de production.....	5
I.4.1 Objectifs de l'automatisation des productions.....	6
I.4.2 Acquisition des données.....	7
I.5 Technologies des automatismes.....	7
I.6 Dialogue homme –machine.....	9
I.7 protection du système.....	9
I.8 Champ d'application de l'automatisme.....	9
I.9 Les automates programmables industriel.....	10
I.9.1 Historique.....	10
I.9.2 Définition et caractéristiques.....	11
I.9.2.1 Automate.....	11
I.9.2.2 Automate Programmable Industriel (API).....	11
I.9.3 Les Objectifs.....	12
I.9.4 Fonctionnement d'un automate programmable.....	12
I.9.5 Domaines d'emploi des automates.....	12
I.9.6 Architecture de l'API.....	12
I.9.6.1 Structure générale d'un automate programmable.....	12
I.9.6.2 L'unité centrale.....	13
I.9.6.2.1 Architecteur de l'unité centrale (UC).....	13
I.9.6.2.2 Processeur.....	14
I.9.6.3 Module d'alimentation.....	15
I.9.6.4 Modules d'entrées/sorties (E/S).....	15
I.9.7 Langages de la programmation.....	16
I.9.7.1 Programmation à l'aide du GRAFCET (SFC: Sequential Function Chart).....	16
I.9.7.2 Langage à contacts (LD : Ladder Diagram).....	17
I.9.7.3 Langage Blocs Fonctionnels (FBD : Function Bloc Diagram).....	18

I.9.7.4 Langage ST.....	18
I.9.7.5 Le langage IL.....	19
I.10 L'environnement de l'API.....	19
I.11 Conclusion.....	20

## **CHAPITRE II : GENERALITES SUR LE GRAFCET**

II.1 Introduction.....	21
II.2 Historique.....	21
II.3 Domaine d'application du GRAFCET.....	21
II.4 Définition du GRAFCET.....	22
II.5 Conventions et règles de construction d'un GRAFCET.....	23
II.5.1 Conventions.....	23
II.5.1.1 L'étape.....	23
II.5.1.2 Les actions associées à une étape.....	24
II.5.1.2.1 Les actions continues.....	24
II.5.1.2.2 Les actions conditionnelles.....	24
II.5.1.2.3 Les actions mémorisées.....	25
II.5.1.2.4 Les actions impulsionnelles.....	25
II.5.1.3 La transition .....	25
II.5.1.3.1 La réceptivité.....	25
II.5.1.3.2 La temporisation .....	26
II.5.1.4 Les liaisons.....	27
II.6 Structure de base.....	27
II.6.1 Séquence unique.....	27
II.6.2 Séquence simultanée : parallélisme structural.....	27
II.6.3 Sélection de séquence.....	28
II.6.3.1 Séquences exclusives.....	28
II.6.4 Saut d'étape.....	29
II.6.5 Reprise de séquence.....	29
II.6.6 Réutilisation d'une même séquence.....	30
II.7 Les règles de construction.....	30
II.7.1 Les règles d'évolution .....	31
II.7.1.1 Règle 1 : Situation initiale .....	31
II.7.1.2 Règle 2 : Franchissement d'une transition.....	31
II.7.1.3 Règle 3 : Evolution des étapes actives.....	31
II.7.1.4 Règle 4 : Evolution simultanée.....	31
II.7.1.5 Règle 5 : Activation et désactivation simultanées d'une étape.....	32
II.8 Les Macro étapes.....	32
II .8.1 Définition Les Macro étapes.....	32
II .9 GRAFCET hiérarchisés.....	33
II .9.1 Définitions.....	33
II .9.2 Fonctionnement.....	33
II.10 Séquenceur.....	35
II.11 Conclusion.....	35

## **CHAPITRE III : DESCRIPTION DE LA BRIQUETERIE**

III.1 Introduction.....	36
III.2 Importance de la brique.....	36

III.3 Présentation de la briqueterie d'EL KALAA BENI-HAMMAD.....	36
III.3.1 Fiche technique de la Briqueterie KALAA BENI-HAMMED.....	37
III.3.2 Effectif a l'unité.....	37
III.3.3 Caractéristique technique de l'unité .....	39
III.3.4 Puissances des équipements utilisés dans les processus.....	39
III.4 Maintenance et sécurité de l'unité.....	40
III.4.1 Maintenance.....	40
III.4.2 Sécurité d'usine .....	40
III.5 Processus de fabrication.....	41
III.5.1 préparation de terre .....	41
III.5.2 fabrication et séchage.....	42
III.5.3 La cuisson.....	42
III.6 conclusion.....	48

## CHAPITRE IV : SOLUTIONS PROPOSEES

IV.1 Introduction.....	49
IV.2 Présentation de l'automate.....	49
IV.3 Choix de l'automate.....	50
IV.4 Présentation des logiciels PL7.....	50
IV.4.1 Généralités.....	50
IV.4.2 Le programme PL7-PRO.....	50
IV.4.3 Les différents langage de PL7.....	50
IV.4.3.1 Langage à contacts.....	50
IV.4.3.2 Langage Grafset.....	51
IV.4.3.3 Langage liste d'instructions.....	51
IV.4.3.4 Langage littéral structuré.....	51
IV.5 Les objets disponibles sur PL7-PRO.....	52
IV.5.1 Repère.....	52
IV.5.2 Préfixe.....	52
IV.5.3 Format.....	52
IV.5.4 Numéro.....	52
IV.5.5 Front.....	53
IV.5.6 Bascule RS.....	53
IV.5.7 Temporisation.....	53
IV.5.7.1 Les modes de fonctionnements .....	53
IV.5.8 Monostable.....	54
IV.5.9 Registre.....	54
IV.5.10 Compteur.....	55
IV.5.11 Autres objets.....	55
IV.5.11.1 Bloc COMPARE.....	55
IV.5.11.2 Bloc OPERATE.....	56
IV.6 Editeur de configuration matérielle.....	56
IV.6.1 Présentation .....	56
IV.6.2 Illustration.....	56
IV.6.3 Configuration de base.....	56
IV.6.4 Module alimentation.....	57
IV.6.5 Mode d'adressage des modules.....	58

<b>IV.7 Solution par GRAFCET.....</b>	<b>59</b>
<b>IV.7.1 Hiérarchie des GRAFCET.....</b>	<b>59</b>
<b>IV.7.2 GRAFCET de sécurité.....</b>	<b>59</b>
<b>IV.7.3 GRAFCET de Conduite et Initialisation.....</b>	<b>60</b>
<b>IV.7.4 GRAFCET de coordination des tâches.....</b>	<b>61</b>
<b>IV.7.5 GRAFCET de production normale (GPN).....</b>	<b>62</b>
<b>IV.7.5.1 Tâche de préparation.....</b>	<b>62</b>
<b>IV.7.5.2 Tâche de fabrication.....</b>	<b>63</b>
<b>IV.7.5.3 Tâche de pré coupeur.....</b>	<b>64</b>
<b>IV.7.5.4 Tâche de coupe finale.....</b>	<b>65</b>
<b>IV.7.5.5 Tâche de séchage.....</b>	<b>66</b>
<b>IV.7.5.6 Tâche cuisson de brique.....</b>	<b>67</b>
<b>IV.8 Conversion écriture Grafcet en langage LD.....</b>	<b>68</b>
<b>IV.8.1 Programme de GRAFCET de sécurité.....</b>	<b>68</b>
<b>IV.8.2 Programme de GRAFCET de conduite et initialisation.....</b>	<b>69</b>
<b>IV.8.3 Programme de GRAFCET de coordination de tâche préparation.....</b>	<b>70</b>
<b>IV.8.4 Programme de tâche de préparation.....</b>	<b>72</b>
<b>IV.9 Conclusion.....</b>	<b>83</b>

## **Conclusion générale**

<b>Conclusion générale.....</b>	<b>84</b>
---------------------------------	-----------

# TABLE DES FIGURES

## CHAPITRE I : L'AUTOMATE ET L' AUTOMATEPROGRAMMABLE INDUSTRIEL (API )

Figure I.1 : Structure d'un système automatisé.....	5
Figure I.2 : Système automatisé de production.....	6
Tableau (I.1) : Comparaison entre la solution câblée et la solution programmée.....	8
Figure I.3 : Armoire de commande à logique Câblée.....	9
Figure I.4 : Armoire de commande à logique programmable "SIEMEN".....	9
Figure I.5 : Champ d'application de l'automatisme.....	10
Figure I.6 : Automate compact (Allen-bradley).....	10
Figure I.7 : Automate modulaire (Modico).....	10
Figure I.8 : Structure générale d'un automate programmable.....	13
Figure I.9 : Structure générale l'unité centrale d'un API.....	14
Figure I.10 : Partie opérative et partie commande.....	16
Figure I.11 : Exemple Programme SFC avec divergence et convergence doubles.....	17
Figure I.12 : Exemple Programme LD avec de liaison multiple à droite.....	17
Figure I.13 : Exemple Programme FBD avec de saut et d'étiquette.....	18
Figure I.14 : Exemple Programme ST.....	18
Figure I.15 : Exemple Programme IL.....	19
Figure I.16 L'environnement de l'API.....	19

## CHAPITRE II : GENERALITES SUR LE GRAFCET

Figure II.1 : Schéma système de commande.....	22
Figure II.2 structure générale d'un GRAFCET.....	23
Figure II.3 : Représentation d'une étape.....	24
Figure II.4 : Action associée à une étape.....	24
Figure II.5 : Symbole de la transition et la condition logique.....	25
Figure II.6 : Ecriture de la réceptivité.....	26
Figure II.7 : Ecriture d'une temporisation.....	26
Figure II.8 :Transition avec réceptivité toujours vrai.....	26
Figure II.9 : Séquence unique.....	27
Figure II.1 : Transition en ET convergent.....	28
Figure II.11 : Transition en ET divergent.....	28
Figure II.12 : Transition en OU divergent.....	28
Figure II.13 :Transition en OU convergent.....	28
Figure II.14 : Une Séquences exclusives.....	28
Figure II.15 : Saut d'étape.....	29
Figure II.16 : Reprise de séquence.....	29
Figure II.17 : Réutilisation d'une même séquence.....	30
Figure II.18 : Structure du GRAFCET et interprétation.....	30
Figure II.19 : Le franchissement d'une transition.....	31
Figure II.20 : Macro étapes.....	32

<b>Figure II.21 : Structure GRAFCET hiérarchisés.....</b>	<b>33</b>
<b>A - GRAFCET de sécurité (GS).....</b>	<b>33</b>
<b>B - GRAFCET de conduite (GC).....</b>	<b>34</b>
<b>C - GRAFCET de production normale (GPN).....</b>	<b>34</b>
<b>D - GRAFCET DE REMISE EN ETAT INITIAL : GREI.....</b>	<b>34</b>
<b>Figure II.22 : GRAFCET HIERACHISES.....</b>	<b>34</b>

### **CHAPITRE III : DESCRIPTION DE LA BRIQUETERIE**

<b>Figure III.1 : photos interne et externe du l'usine.....</b>	<b>37</b>
<b>Figure III.2 : organigramme de l'unité.....</b>	<b>38</b>
<b>Figure III.3 : cycle préparation de terre.....</b>	<b>41</b>
<b>Figure III.4 : cycle fabrication et séchage.....</b>	<b>42</b>
<b>Figure III.5 : cycle de cuisson.....</b>	<b>42</b>
<b>Figure III.6 : schéma synoptique de l'installation de préparation affiché sur pupitre.....</b>	<b>43</b>
<b>Figure III.7 : schéma synoptique de l'installation de fabrication affiché sur pupitre.....</b>	<b>44</b>
<b>Figure III.8 : schéma de pré coupeur.....</b>	<b>45</b>
<b>Figure III.9 : schéma de coupeur.....</b>	<b>45</b>
<b>Figure III.10 : schéma synoptique de l'installation de séchoir affiché sur microcontrôleur.....</b>	<b>46</b>
<b>Figure III.11 : schéma synoptique de l'installation de four affiché sur microcontrôleur.....</b>	<b>47</b>

### **CHAPITRE IV : SOLUTIONS PROPOSES**

<b>Figure IV.1 : automate TSX Micro.....</b>	<b>50</b>
<b>Figure IV.2 : automate TSX Premium.....</b>	<b>50</b>
<b>Figure IV.3 : exemple de réseau de contacts.....</b>	<b>50</b>
<b>Figure IV.4 : Exemple de programme en langage Grafcet.....</b>	<b>51</b>
<b>Figure IV.5 : exemple de programme en langage liste d'instructions.....</b>	<b>51</b>
<b>Figure IV.6 : exemple de programme en langage littéral structuré.....</b>	<b>51</b>
<b>Tableau (IV.1) : le préfixe sur PL7-PRO.....</b>	<b>52</b>
<b>Tableau (IV.2) : format des éléments d' indication.....</b>	<b>52</b>
<b>Figure IV.7 : indication sur la situation géographique de l'élément.....</b>	<b>52</b>
<b>Tableau (IV.3) : la forme de contact entrée front(P , N).....</b>	<b>53</b>
<b>Tableau (IV.4) : la forme de contact de sortie bascule (R , S).....</b>	<b>53</b>
<b>Tableau (IV.5) : l'indication de temporisation.....</b>	<b>53</b>
<b>Figure IV.8 : bloc temporisation.....</b>	<b>53</b>
<b>Figure IV.9 : TON, temporisation travail.....</b>	<b>53</b>
<b>Figure IV.10 : TOFF, temporisation repos.....</b>	<b>54</b>
<b>Figure IV.11 : TP, monostable.....</b>	<b>54</b>
<b>Tableau (IV.6) : l'indication de monostable.....</b>	<b>54</b>
<b>Figure IV.12 : bloc monostable.....</b>	<b>54</b>
<b>Tableau (IV.7): l'indication de registre.....</b>	<b>55</b>
<b>Figure IV.13 : bloc registre.....</b>	<b>55</b>
<b>Tableau (IV.8): l'indication compteur.....</b>	<b>55</b>
<b>Figure IV.14 : bloc compteur.....</b>	<b>55</b>
<b>Figure IV.15 : bloc compare.....</b>	<b>55</b>
<b>Figure IV.16 : bloc operate.....</b>	<b>56</b>
<b>Figure IV.17 : Configuration matérielle.....</b>	<b>56</b>

<b>Figure IV.18</b> : Configuration matérielle.....	<b>57</b>
<b>Figure IV.19</b> : Configuration de l'alimentation.....	<b>57</b>
<b>Figure IV.20</b> : Ajout des modules sur le rack.....	<b>57</b>
<b>Figure IV.21</b> : Configuration de la station automate.....	<b>58</b>
<b>Figure IV.22</b> : La hiérarchie des GRAFCET.....	<b>59</b>
<b>Figure IV.23</b> : GRAFCET de sécurité.....	<b>59</b>
<b>Figure IV.24</b> : GRAFCET de Conduit et Initialisation.....	<b>60</b>
<b>Figure IV.25</b> : GRAFCET de coordination de tâche.....	<b>61</b>
<b>Figure IV.26</b> : GRAFCET de la tâche de préparation .....	<b>62</b>
<b>Figure IV.27</b> : GRAFCET de la tâche de fabrication .....	<b>63</b>
<b>Figure IV.28</b> : GRAFCET de la tâche de pré coupeur.....	<b>64</b>
<b>Figure IV.29</b> : GRAFCET de la tâche de coupe finale.....	<b>65</b>
<b>Figure IV.30</b> : GRAFCET de la tâche de séchage.....	<b>66</b>
<b>Figure IV.31</b> : GRAFCET de la tâche de cuisson .....	<b>67</b>

## ***Introduction générale***

*L'automatique fait partie des sciences de l'ingénieur, c'est une discipline technique et scientifique traitant, d'une part, de la caractérisation des systèmes automatisés et d'autre part, du choix, de la conception, et de la réalisation du système de commande.*

*Sur le plan industriel, l'Automatique est présente dans de nombreuses grandes, moyennes et petites entreprises, afin de permettre une compétitivité globale du produit ; soit directement (coût, qualité...) soit indirectement (amélioration des conditions de travail..).*

*La réalisation et l'exploitation des automatismes industriels font appel, aux connaissances théoriques et à la maîtrise des technologies des divers outils, matériels et logiciels, qui doivent être convenablement choisis et utilisés, en vue de remplir certaines fonctions. C'est dans cet état d'esprit qu'on a jugé utile de faire une étude, sur site, d'un système automatisé industriel de production de brique d'EL KALAA BENI HAMMAD.*

*Les objectifs fixés à travers ce travail est de faire, une recherche bibliographique sur les systèmes automatisés, avoir un aperçu global sur le diagramme fonctionnel "GRAFCET ", utilisé pour la résolution des systèmes automatisés industriels. Ensuite, à partir d'un plan de fonctionnement d'une entreprise de fabrication de la brique, proposer des solutions GRAFCET pour les différentes tâches du cycle de production. Et fin, faire une approche de programmation d'automate pour une l'une des tâches tout en faisant intervenir les consignes.*

*Ce mémoire est partagé en quatre chapitres, décrit comme suit :*

*Un premier chapitre, dont le but est de donner des notions générales sur les systèmes automatisés et automate programmable industriel.*

*Le deuxième chapitre sera consacré au GRAFCET et GRAFCET évolué, solution graphique des systèmes automatisés.*

*Au troisième chapitre, nous donnerons une description de la briqueterie d'EL KALAA BENI AMMAD, et des différentes étapes de fabrication de la brique.*

*Dans le quatrième chapitre, des solutions par GRAFCET pour les différentes tâches et une approche de programmation de la tâche de préparation, seront proposées.*

*Nous terminerons par une conclusion générale qui englobera l'ensemble des travaux effectués, les problèmes rencontrés et enfin les perspectives envisagées pour la continuité de ce travail.*

## **I.1 Introduction**

*Dans un monde industriel en pleine évolution où la compétitivité est l'objectif essentiel, l'automatisation est une nécessité incontournable .*

*L'automatisation des équipements de production peut améliorer les coûts, la qualité et même la disponibilité des produits et surtout après avoir bénéficié du progrès de la micro-électronique et la micro-informatique.*

*Cependant cette automatisation doit être maîtrisée par la connaissances des notions de bases des systèmes automatisés d'une part, et par le savoir faire des technologies câblés et programmés.*

*Nous évoquerons dans ce chapitre, en premier lieu des les notions générales sur les systèmes automatisés, objectif de l'automatisation 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,[2].*

## **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),[3].

### **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 ...[3]

### **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,[3].

## **I.3 L'automatisation**

L'automatisation d'un procédé (c'est à dire une machine, un ensemble de machines ou plus généralement un équipement industriel) consiste à en assurer la conduite par un dispositif technologique.

Une « machine » ou un « système » est dit automatisé lorsque le processus qui permet de passer d'une situation initiale à une situation finale se fait sans intervention humaine et que ce comportement est répétitif chaque fois que les conditions qui caractérisent la situation initiale sont remplies, [4].

### **I.3.1 Les objectifs de l'automatisation**

La compétition économique entraîne les industriels à vivre en permanence dans un esprit de concurrence, qui oblige à toujours améliorer les performances en termes de quantité et surtout de qualité (d'où le maître mot de l'économie : rapport qualité/prix). L'automatisation des processus industriels a pour finalité de réaliser ces vœux. Ses objectifs principaux au nombre de quatre sont complémentaires et liés. Ils peuvent s'énoncer ainsi:

#### **I.3.1.1 Produire à qualité constante**

Contrairement à l'être humain, il est clair que la machine n'est pas sujette à la fatigue de fin de journée, par conséquent la qualité des produits sortant des chaînes est généralement la même, [5].

#### **I.3.1.2 Fournir les quantités nécessaires**

Cette notion fait référence à l'adaptativité, c'est-à-dire pouvoir adapter l'offre à la demande, l'objectif est de produire les quantités nécessaires à un instant donné, de façon à tendre vers la notion de stock zéro, pour pouvoir adapter l'offre à la demande, cela se fait rapidement et efficacement dans un environnement automatisé (arrêter par exemple une chaîne de fabrication ayant une faible demande, ou au contraire mettre en route d'autres pour répondre à une forte demande, [5].

#### **I.3.1.3 Augmenter la productivité**

Il s'agit donc d'augmenter le rendement. Pour se faire l'automatisation a consisté à remplacer une grande partie des opérateurs humains par des machines, qui ont des cadences de travail élevées, ne connaissons ni les pauses café ni les congés payés, [5].

#### **I.3.1.4 Améliorer les conditions de travail**

Il s'agit d'une part de remplacer l'homme par la machine pour les tâches pénibles ou qu'il ne peut pas faire (pour l'effectuer ailleurs où il est censé faire un travail plus noble), d'autre part d'augmenter les possibilités de réaliser les objectifs "a", "b", "c". En effet un employé qui mange bien, est bien soigné, et a bonnes conditions de vie et de travail, n'est pas souvent malade, n'est pas fatigué, a peu d'absentéisme, et devient donc plus rentable économiquement, [5].

### **I.3.2 Structure d'un système automatisé**

Chaque système automatisé comporte deux parties (Fig.I.1): [6].

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

#### **I.3.2.1 Partie Opérative**

C'est elle qui opère sur la matière l'œuvre et le produit. Elle comporte en général,[6] :

- 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.3.2.2 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,[6] :

##### **1) Le dialogue avec la machine**

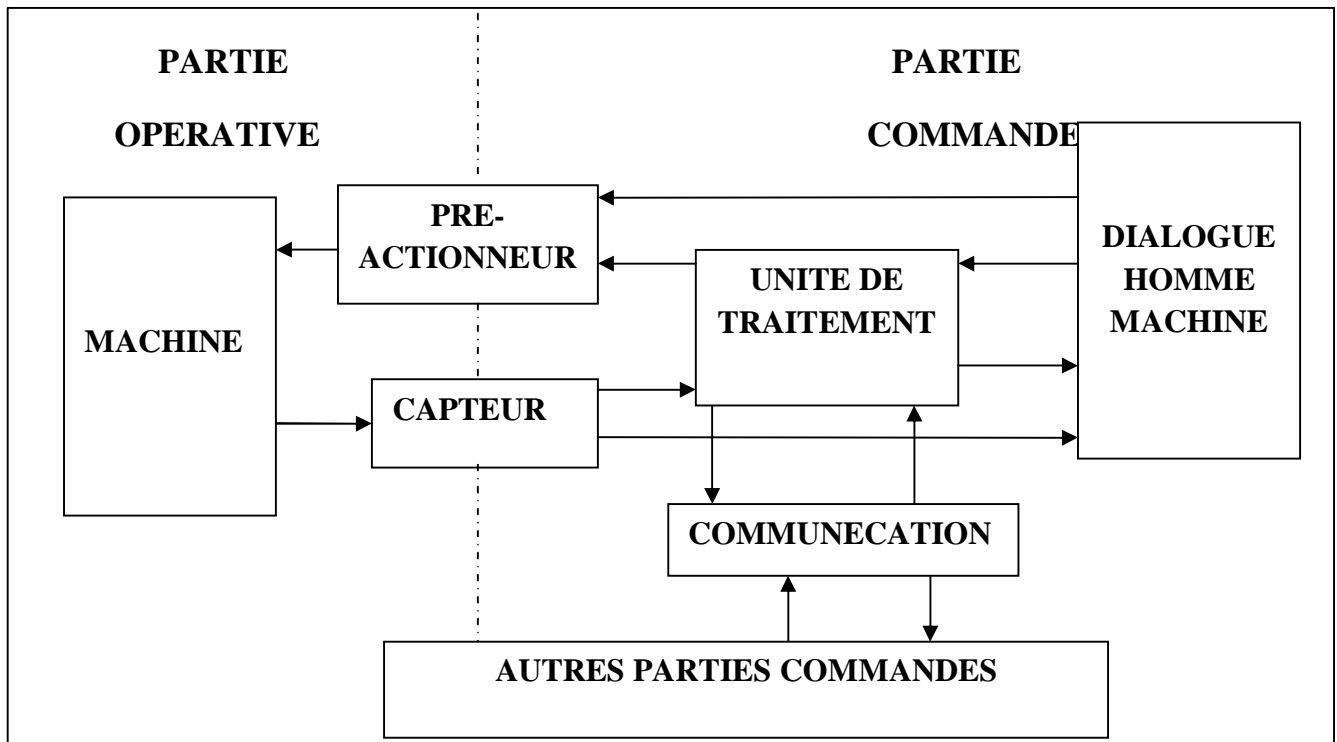
Commande des actionneurs (moteurs, vérins) via les prés actionneurs (contacteurs, distributeurs, variateur...) ; acquisition des signaux en retour par les capteurs qui rendent compte de l'évolution de la machine , [6].

##### **2) Le dialogue homme-machine**

Pour exploiter, régler, dépanner la machine, le personnel émet des consignes et reçoit des informations en retour,[6].

##### **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), [6].



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

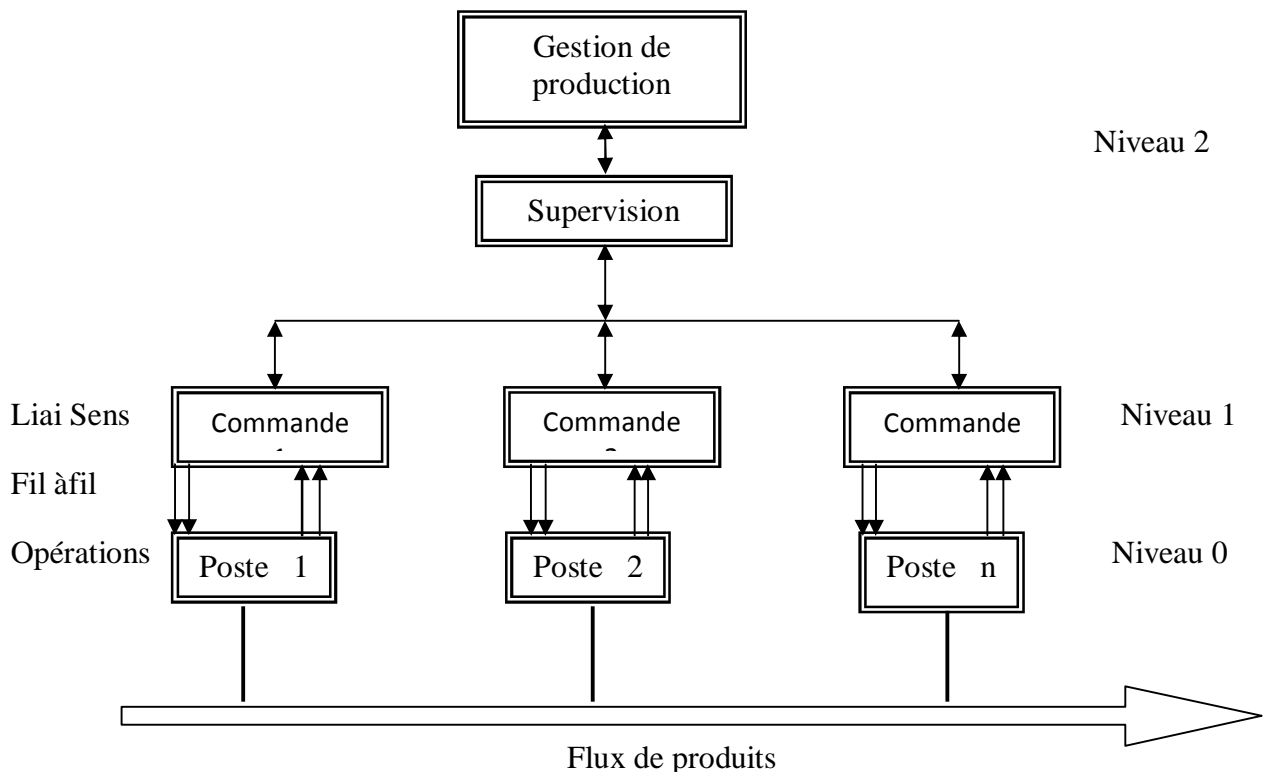
#### I.4 Système automatisé de production

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.

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, déconditionnement),[7] .



**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, [8].

### **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, [6].

### **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 , [9].

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 reliée 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 automatisme 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 à logique Câblée.



**Figure I.4 :** Armoire de commande à logique programmable "SCHNEIDER".

## 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 cours de cycle, de procéder à un arrêt d'urgence et par l'intermédiaire du système de signalisation, de contrôler en permanence le déroulement des opérations.

Cette fonction de dialogue est assurée par tous les auxiliaires de commande à intervention humaine (boutons, boîtes à boutons, commutateurs) ainsi que par des voyants de signalisation et pour les installations plus complexes, par des pupitres de commande, des tableaux synoptiques, des boutons, [8].

## 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, [8] :

- \* Électrique : surintensité (surcharge et 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 sert 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 simples 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 (fig.I.5), [6].

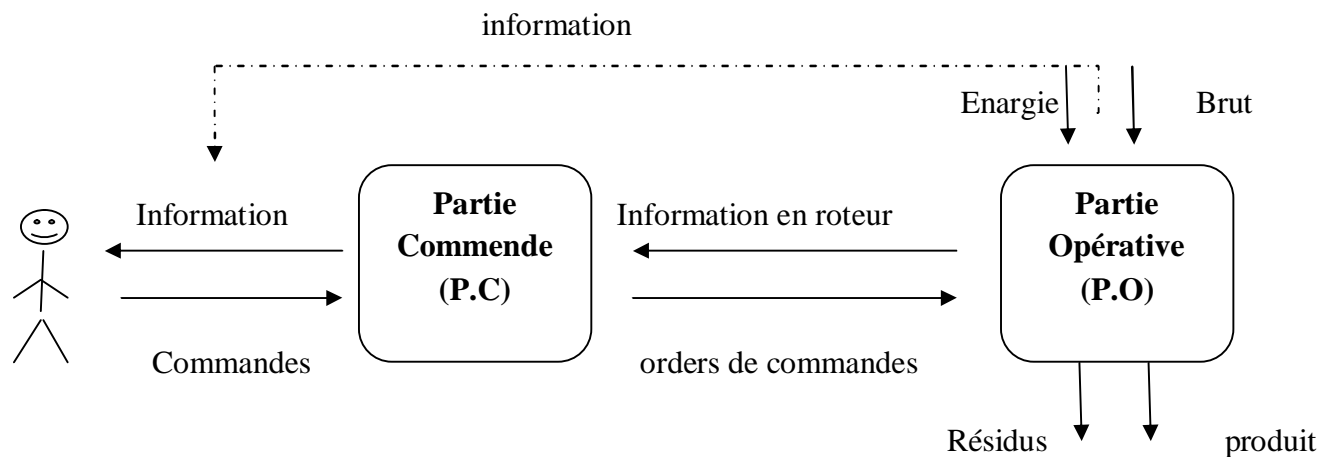


Figure I.5 : Champ d'application de l'automatisme.

## I.9 Les automates programmables industriels

### I.9.1 Historique

Le premier est apparu en 1969 aux Etats Unis sur une demande de Ford pour remplacer les grosses armoires à relais des chaînes de la construction automobile. La réponse vint d'ALLEN BRADLEY.

La NASA fait en suite un appel d'offres sur le même thème auquel MODICON répondit. Sa date de création coïncide avec le début de l'ère du microprocesseur et avec la généralisation de la logique câblée modulaire. Le premier automate français fût le PB6 de MERLIN GERIN en 1971, suivi d'ALSPA, [10].

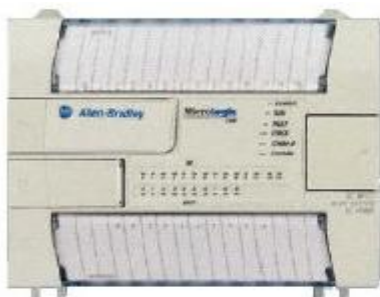


Figure I.6 : Automate compact (Allen-bradley) .



Figure I.7 : Automate modulaire (Modico).

## I.9.2 Définition et caractéristiques

### I.9.2.1 Automate

Selon le dictionnaire : l'automate est une machine qui imite le mouvement d'un être animé.  
Au sens figuré : homme sans volonté qu'on fait agir comme on veut. L'interprétation serait plutôt entre les deux : machine que l'on fait agir comme veut et qui imite les décisions systématiques d'un homme,[10].

### I.9.2.2 Automate Programmable Industriel (API)

L'automate programmable industriel est une commande conçue autour d'un microprocesseur. Sa conception et son langage de programmation sont spécialement adaptés aux contrôles de processus industriel. L'abréviation PLC de l'Anglais "Programmable logical Controller". Sa définition est donnée par la norme NFC 63-850 :

« Appareil électrique qui comporte une mémoire programmable par un utilisateur automaticien (non informaticien) à l'aide d'un langage adapté, pour le stockage interne des instruction composant les fonctions d'automatisme comme par exemple :

- Logique séquentielle et combinatoire.
- Temporisation, comptage, décomptage, comparaison.
- Calcul arithmétique.
- Réglage, asservissement, régulation, etc. pour commander, mesuré et contrôler au moyen de modules d'entrées et de sorties (logique, numérique, ou analogique) différentes sorties de machines ou de processus, en environnement industriel ».

Les automates programmables industriels sont des équipements électriques à câblage interne indépendant du processus à commander. L'automate programmable industriel est adapté à la machine ou à l'installation à commander au moyen d'un programme qui définit le mode de déroulement des opérations souhaité et d'un câblage direct entre les éléments d'entrée et de sortie de l'automate.

Les éléments d'entrée sont par exemple des auxiliaires de commande, des contacts de fin de course, des détecteurs de proximité mais aussi des tensions analogiques ou des détecteurs de courant (signaux en retour de la machine, signaux de commande issus du pupitre).

Les contacteurs, électrovannes, dispositifs de couplage, lampes, etc. sont des éléments de sortie (signaux de commande émis vers la machine, signaux de visualisation vers le pupitre).

Le nombre d'instructions traitées varie d'un automate à l'autre. Le programme est introduit à partir d'une console de programmation. Le jeu d'instruction d'un automate est orienté vers les problèmes de logique et vers les systèmes à évolution séquentielle.

Les API se caractérisent par rapport aux ordinateurs par leur fiabilité et leur facilité de maintenance. Les modules peuvent être changés très facilement et le redémarrage des API est très rapide.

Au cours de la dernière décennie, la baisse du prix des A.P.I et des micro-ordinateurs a modifié sensiblement le domaine d'emploi des A.P.I. Au déterminent de la logique câblée et des microprocesseurs. Les produits de type cartes à microprocesseurs n'ont pas eu le développements espérés.

La situation actuelle se présente de la manière, [10] :

- le plus gros API : 10 processeurs, 10000 entrées/sorties est le MDICON 584.
- le plus petit API : 1 processeur, 1 bit, 16 entrées/sorties est le TEC 16.

### **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 augmente 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 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...), [7].

### **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), [7].

### **I.9.5 Domaines d'emploi des automates**

L'évolution des techniques s'est traduite pour l'automatisation par :

- un développement massif ;
- une approche de plus en plus globale des problèmes ;
- une intégration dès la conception de l'installation.

On est ainsi passé du stade de la machine automatisée à celui du système automatisé de production, et on utilise les API dans tous les secteurs industriels pour la commande des machines (convoyage, emballage...) ou des chaînes de production (automobile, agroalimentaire...) ou il peut également assurer des fonctions de régulation de processus (métallurgie, chimie...).

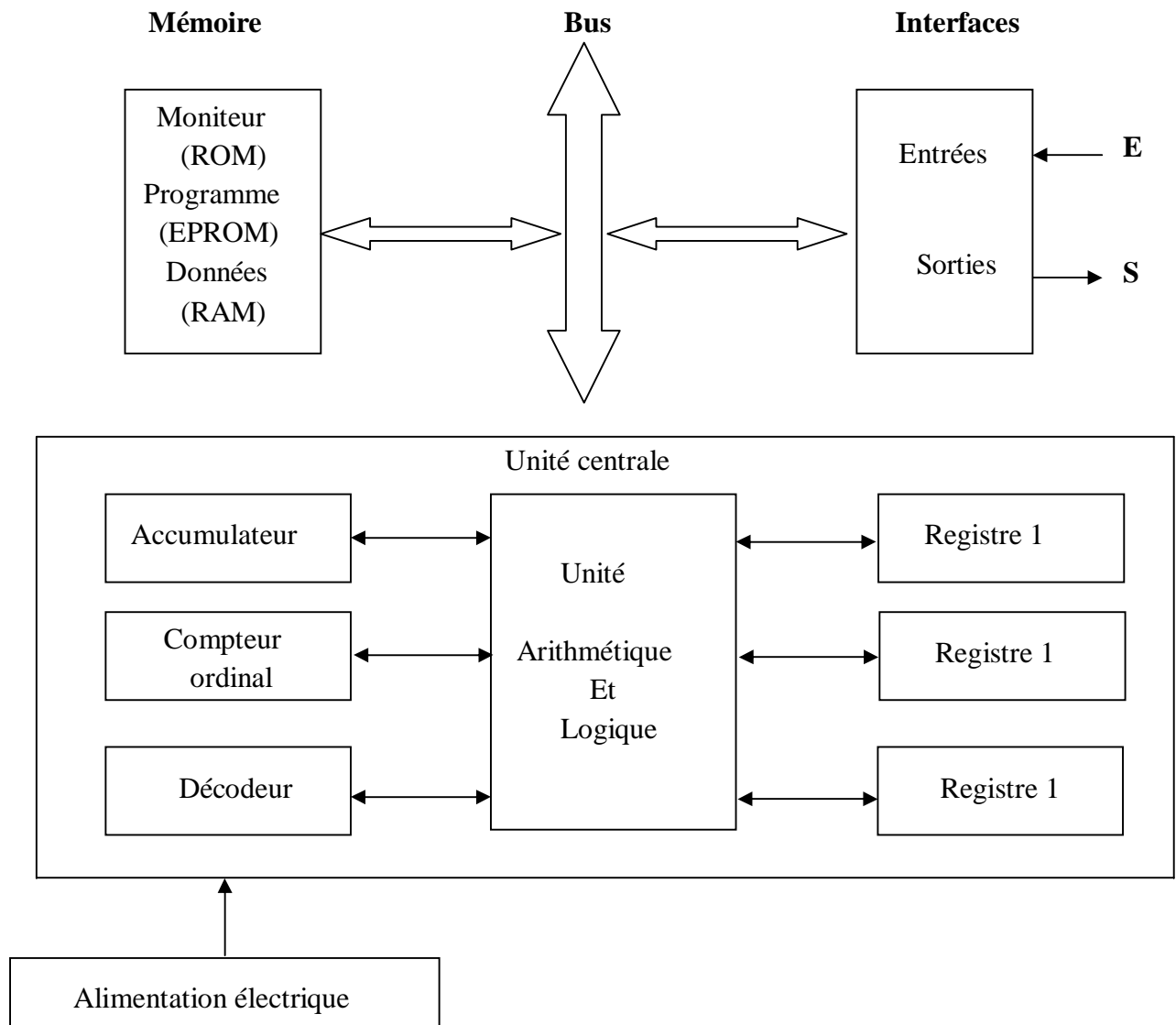
Il est de plus en plus utilisé dans le domaine du bâtiment (tertiaire et industriel) pour le contrôle du chauffage, de l'éclairage, de la sécurité ou des alarmes, [7].

### **I.9.6 Architecture de l'API**

#### **I.9.6.1 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 processif unit), la mémoire, les modules d'entrées / sorties et la console de programmation (fig.I.8), [7].



**Figure I.8 :** Structure générale d'un automate programmable.

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

### I.9.6.2 L'unité centrale

C'est l'ensemble des dispositifs nécessaire en fonctionnement logique interne de l'API, c'est le cœur l'automate, [9].

#### I.9.6.2.1 Architecteur de l'unité centrale (UC)

L'unité centrale comporte la zone mémoire et le processeur (fig.I.9), [9].

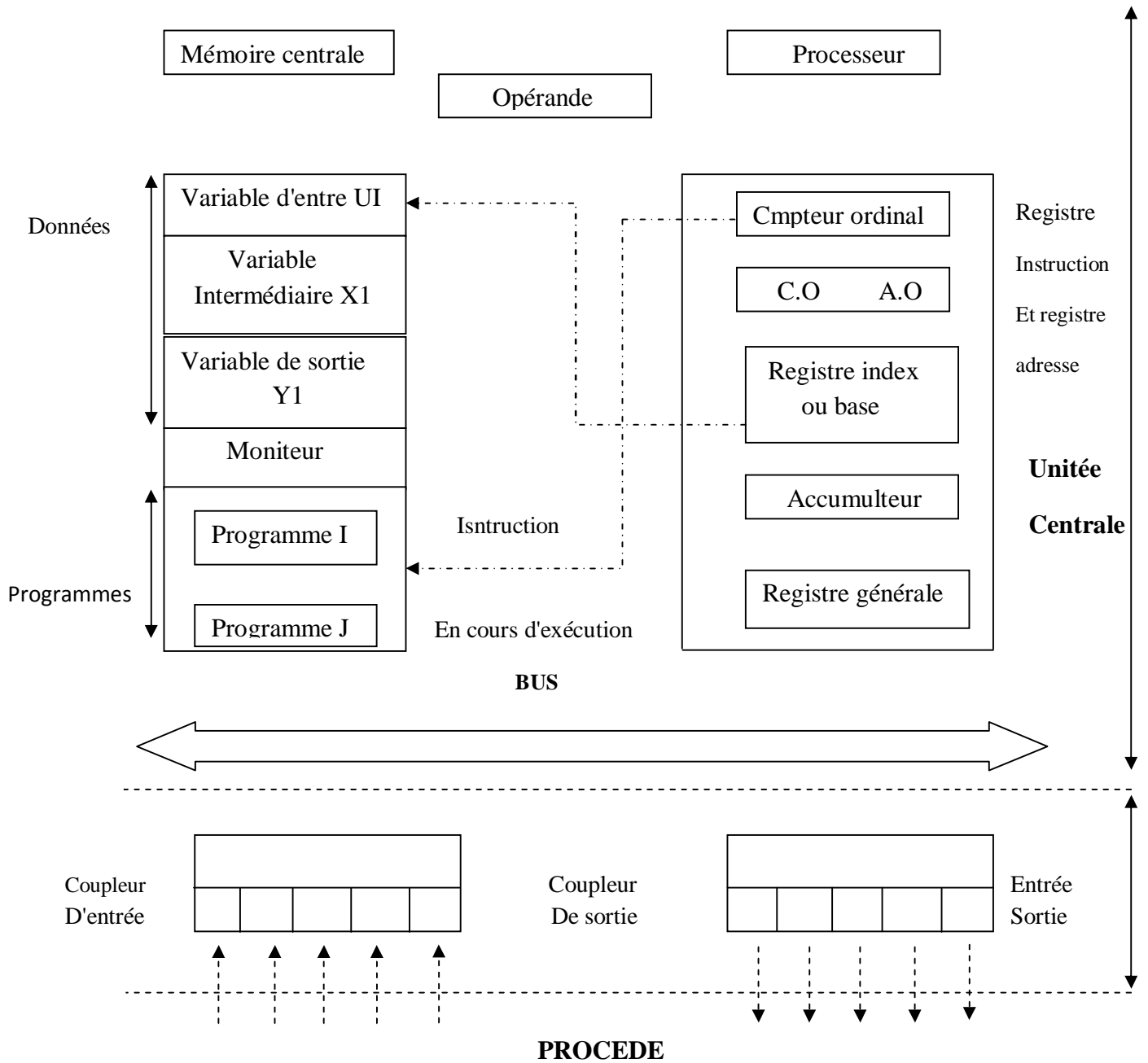


Figure I.9 : Structure générale l'unité centrale d'un API.

### I.9.6.2.2 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 suivantes : [2].

- opérations logiques 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 grande série ;
- la technologie mixte, certaines opérations étant réalisées en câbles pour en accroître la rapidité.

### I.9.6.3 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,[2].

\* **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  $\pm 15$  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 pres 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 2 A. En alternatif 50 Hz, elles peuvent aller jusqu'à 220 V et 5 A respectivement,[2].

### I.9.6.4 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

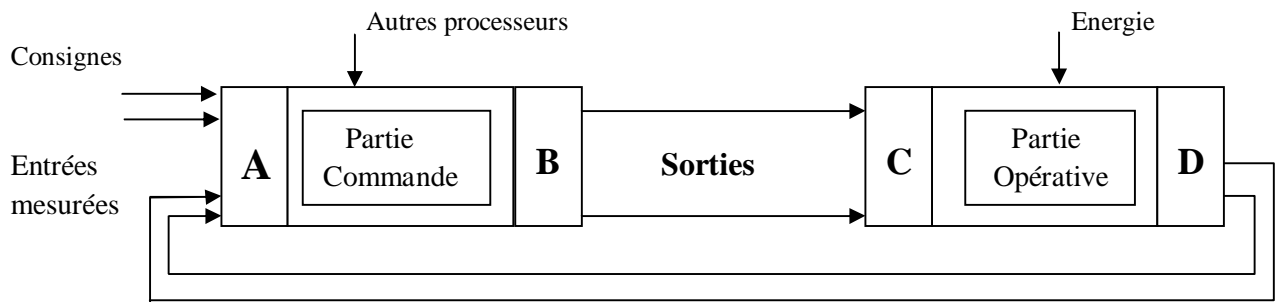
Amovibles comportant un nombre fixé de voies d'un certain type Figure (I.8), [2].

- 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 (opto couplage), [2].

- **Interfaces de sortie** : elle permet de commander les divers préactionneurs et éléments de signalisation du S.A.P. tout en assurant l'isolement électrique (fig.I.10), [2].



**A Interface d'entrée**

**B Interface de sortie**

**C Actionneurs**

**D Capteurs et détecteurs**

**Figure I.10** : Partie opérative et partie commande.

### I.9.7 Langages de la programmation

Il y a cinq langages de programmation principaux : [11].

- séquentiel function chart (SFC) ou GRAFCET pour les opérations de haut niveau ;
- fonction 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.11),[11].

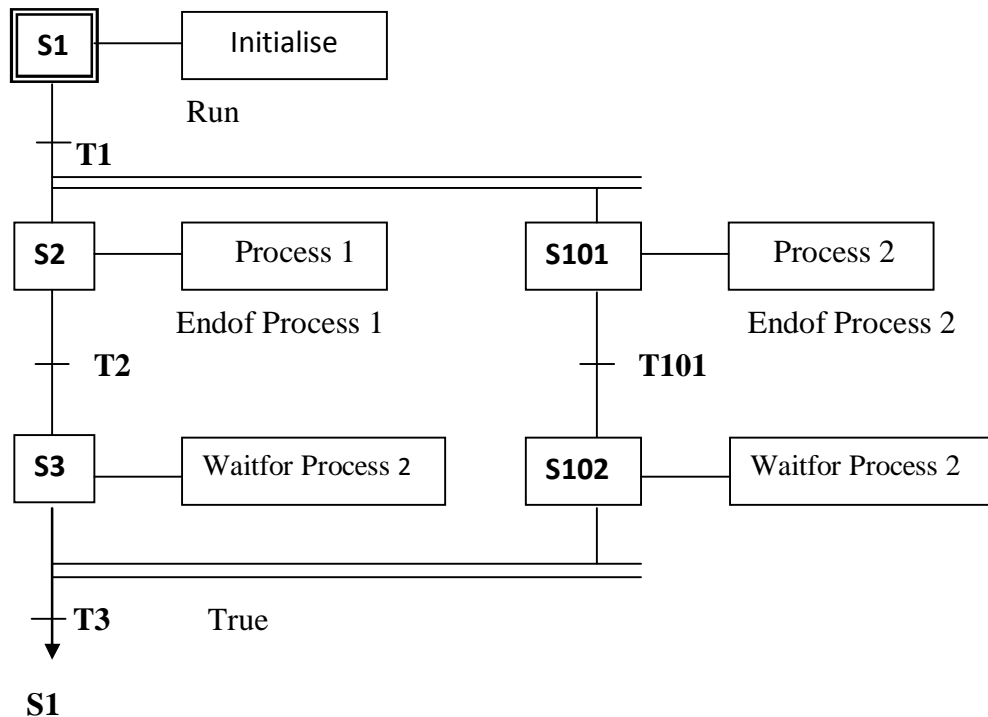


Figure I.11 : Exemple Programme SFC avec divergence et convergence doubles.

On peut également traduire un Grafset 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.12), [11] .

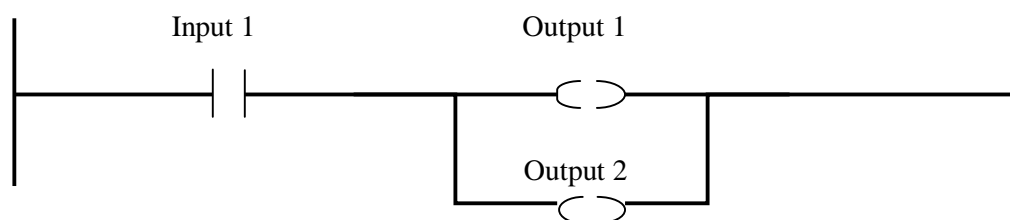
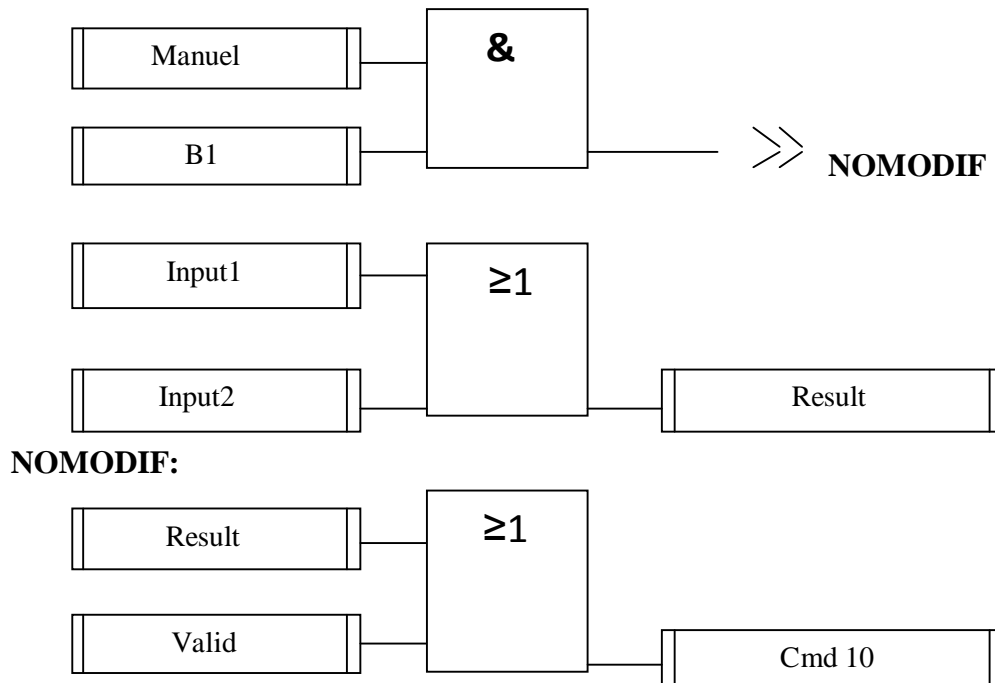


Figure I.12 : Exemple Programme LD avec de liaison multiple à droite.

### 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.13), [11].



**Figure I.13 :** 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.14),[11].

```

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.14 :** 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 registreIL)

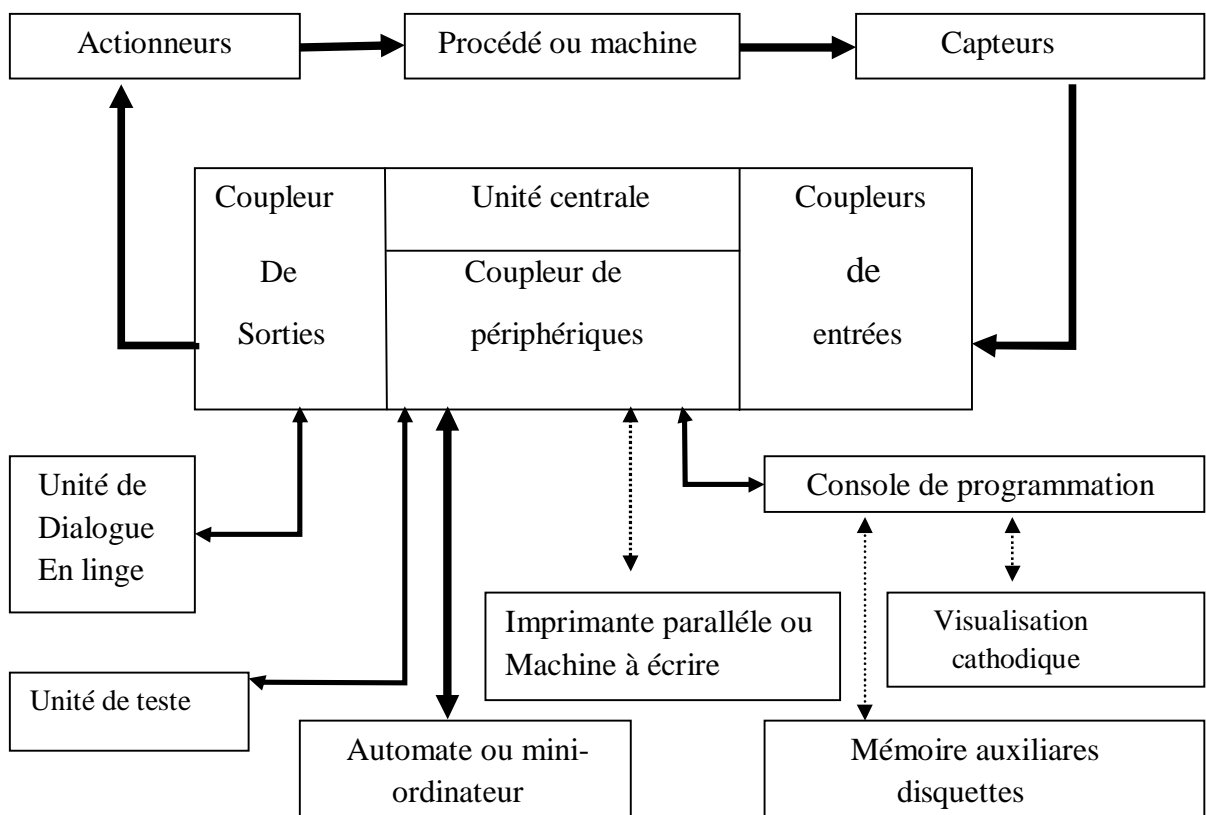
L'opérateur indique le type d'opération à effectuer entre le résultat courant et l'opérande. Le résultat de l'opération est stocké à son tour dans le résultat courant (fig. I. 15), [11].

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

**Figure I.15 :** 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 console d'exploitation, etc.... (fig.I.16), [7].



**Figure I.16** L'environnement de l'API.

## ***I.10 Conclusion***

*L'automatisation s'est généralisée à l'ensemble des activités de production, industrielle, 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.*
- Étant donné que le GRAFCET ( Graphe Fonctionnel de Commande Étapes-Transitions ) est l'outil description de base des systèmes automatisé, nous avons jugé outil de lui consacré le deuxième chapitre , [2].*

## II.1 Introduction

*La résolution des systèmes automatiques deviennent de plus en plus complexes avec un nombre élevé de variables d'entrées, et la résolution par méthode du tableau de KARNAUGH n'est pas toujours aisée, d'où la nécessité d'un outil souple " GRAFCET " qui propose des solutions adéquates pour ces systèmes.*

*Le GRAFCET est un outil graphique qui permet de décrire le fonctionnement d'un automatisme séquentiel. Il peut être utilisé pour représenter l'automatisme dans toutes les phases de la conception : de la définition du cahier des charges, à la mise en œuvre (programmation d'un automate programmable industriel, utilisation de séquenceurs ou autres technologies ) en passant par l'étude des modes de marches et d'arrêts, en utilisant le guide graphique « GRAFCET ».*

*Le GRAFCET repose sur l'utilisation d'instructions précises, l'emploi d'un vocabulaire bien défini, le respect d'une syntaxe rigoureuse, et l'utilisation de règles d'évolutions. Il permet, entre autre, d'adopter une démarche progressive dans l'élaboration de l'automatisme.*

*C'est dans cet état d'esprit que s'inscrira notre deuxième chapitre. Nous aborderons en premier lieu le domaine d'application et la description d'éléments de bases GRAFCET. On passera ensuite à différentes structures du GRAFCET et les règles d'évolutions et nous termineront par un aperçu des GRAFCET hiérarchisés, [1].*

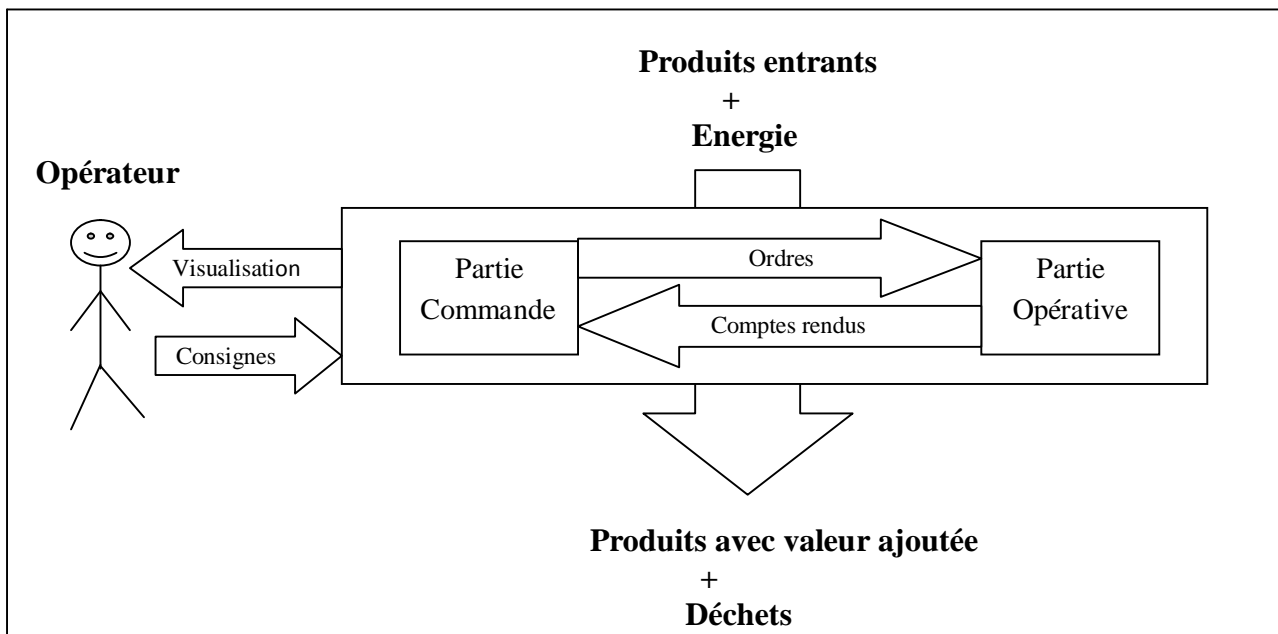
## II.2 Historique

**Le GRAFCET : GR**aphe **F**onctionnel de **C**ommande **E**tapes-**T**ransitions, est né en 1977 des travaux de l'AFCEC (Association Française pour la Cybernétique Economique et Technique), en tant que synthèse théorique des différents outils existants à cette époque.

Mis sous sa forme actuelle par l'ADEPA, en 1979, normalisé en France en 1982 (Norme Française NFC 03-190 juin 1982), le Grafcet est aujourd'hui normalisé sur le plan international sous l'appellation diagramme fonctionnel. Depuis sa création le Grafcet est en perpétuelle évolution, suite à différents travaux de recherche Actuellement la norme est cours de révision, [1].

## II.3 Domaine d'application du GRAFCET

Le GRAFCET (Graphe Fonctionnel de Commande Etape Transition), également appelé Diagramme Fonctionnel en Séquence ou Séquentiel Function Chart, permet "...l'établissement des descriptions de la fonction et du comportement des systèmes de commandes en établissant une représentation graphique indépendant de la réalisation technologique.....", (FigII.1), [1].



**Figure II.1** : Schéma système de commande.

Le GRAFCET est destiné représenter des automatismes logiques séquentiels, c'est à dire des systèmes événementiels dans lesquels les informations sont de type booléennes (tout ou rien) ou peuvent s'y ramener (numériques). Le GRAFCET est utilisé généralement pour spécifier et concevoir le comportement souhaité de la partie commande d'un système de commande mais il peut également être utilisé pour spécifier le comportement attendu de la partie opérative ou bien de tout le système de commande, Destiné à être un moyen de communication entre l'automaticien et son client, le GRAFCET est un outil utilisé pour la rédaction du cahier des charges d'un automatisme.

Cependant un des points forts du GRAFCET est la facilité de passer du modèle l'implantation technologique de celui-ci dans un automate programmable industriel.

Le GRAFCET passe alors du langage de spécification au langage d'implémentation utilisé pour la réalisation de l'automatisme.

## II.4 Définition du GRAFCET

Lorsque le mot GRAFCET (en lettre capitale) est utilisé, il fait référence à l'outil de modélisation. Lorsque le mot grafcet est écrit en minuscule, il fait alors référence à un modèle obtenu à l'aide des règles du GRAFCET.

Le GRAFCET permet de construire des modèles ayant une structure graphique (Représentation statique) à laquelle on associe une interprétation (Elle correspond à l'aspect fonctionnel du grafcet). De plus, ce modèle possède un comportement dicté par des règles d'évolution (représentation dynamique), complétées pour l'implémentation par des algorithmes d'application de ces règles.

Où le *GRAFCET* c'est un diagramme de description du comportement déterministe de la partie commande d'un système automatisé, (Fig.II.2).

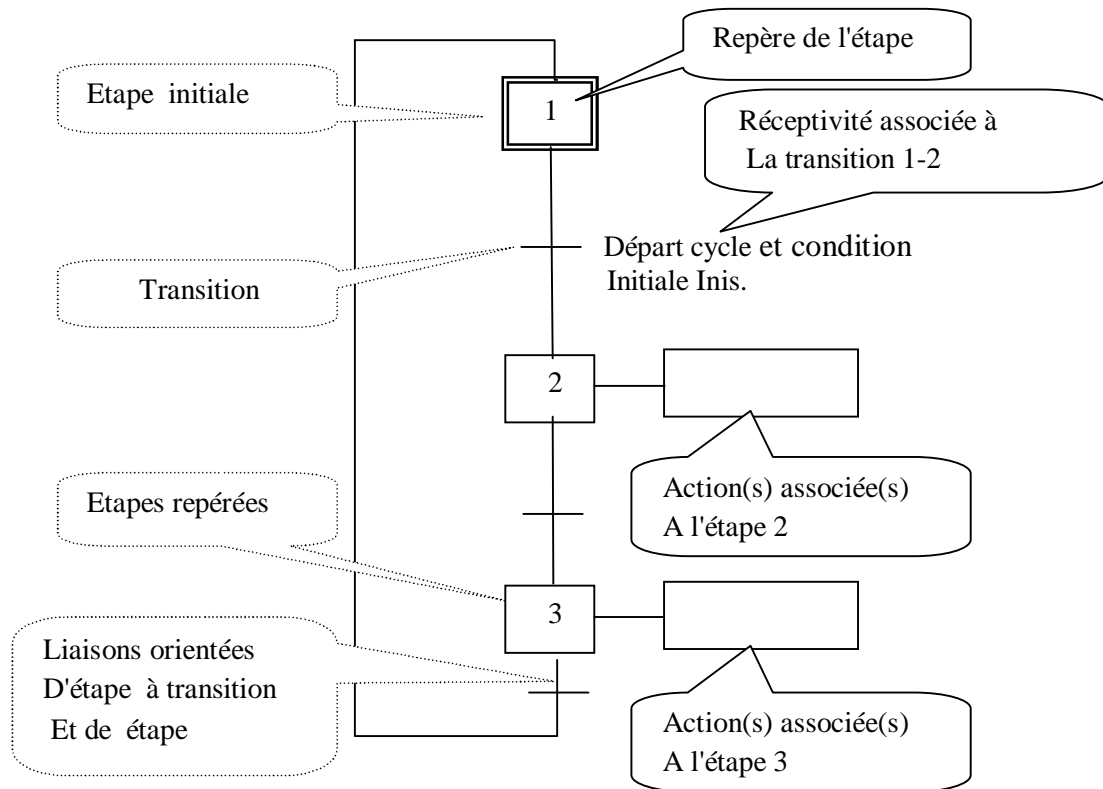


Figure II.2 structure générale d'un GRAFCET.

## II.5 Conventions et règles de construction d'un GRAFCET

Le GRAFCET permet de représenter le cahier des charges d'un automatisme, ce cahier de charges décrit les relations entre la partie commande et la partie opérative et les conditions d'utilisation et de fonctionnement de l'automatisme.

Le fonctionnement d'un automatisme séquentiel peut être décomposé en un certain nombre d'étapes. Le passage (ou transition) d'une étape à une autre étape se fait à l'arrivée d'un événement particulier (réceptivité) auquel le système est réceptif.

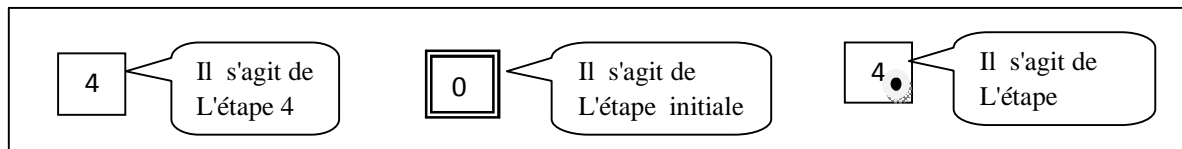
Le fonctionnement d'un système automatisé peut être représenté graphiquement par un grafcet qui est un graphe cyclique composé alternativement d'étapes et de transitions, reliées entre elles par des liaisons ou arcs orientés : [12].

- Aux étapes sont associées des actions ;
- Aux transitions sont associées des réceptivités.

### II.5.1 Conventions

**II.5.1.1 L'étape :** symbolise un état ou une partie de l'état du système, correspond à une situation élémentaire ayant un comportement stable ou invariant. Elle se représente par un carré numéroté de manière unique (repéré par un chiffre) .

Une étape peut être initiale, représentée par un carré double (initialement active en début de fonctionnement). Une étape est soit active soit inactive, lorsqu'elle est active, elle contient un jeton, [12].



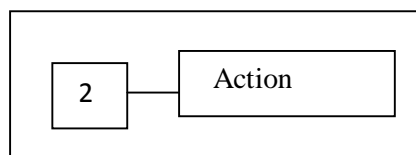
**Figure II.3 :** Représentation d'une étape.

L'ensemble des étapes actives (appelé situation) définit entièrement l'état du système. On précise pour chaque étape, les actions à effectuer. Ces actions ne sont effectuées que lorsque l'étape correspondante est active. On peut associer une condition à ces actions, l'action n'est alors réalisée que si l'étape est active et la condition remplie, [12].

### II.5.1.2 Les actions associées à une étape

On peut associer une interprétation à une étape grâce à une action. C'est un ordre vers la partie opérative du système. Lorsqu'une étape est active alors l'action associée est exécutée, on précise à l'intérieur d'un rectangle les actions à effectuer lorsque l'étape est active

Il n'est pas indispensable qu'une étape possède des actions (exemple : étape d'attente)



**Figure II.4 :** Action associée à une étape.

Une action est une sortie du système logique que nous modélisons, elle correspond donc à une composante du vecteur de sortie. Ces actions peuvent être de trois types : [12].

- Les actions continues,
- Les actions conditionnelles, qui peuvent être classées-en :
  - Action conditionnelle simple ;
  - Action conditionnelle retardée ;
  - Action conditionnelle limitée dans le temps.
- Une action mémorisée.

#### II.5.1.2.1 Les actions continues

La sortie S correspondante est émise à vrai tant que l'étape associée est active. Lorsque l'étape devient inactive la sortie est émise à faux , [12].

#### II.5.1.2.2 Les actions conditionnelles

N'est exécutée que si l'étape associée est active et si la condition associée est vraie. Cette condition est une expression booléenne portant sur l'ensemble des entrées, des variables d'étapes et exprimée à l'aide des opérateurs logiques ET, OU, ou NON. On fait apparaître dans le cadre de l'action attaché à l'étape le caractère conditionnel qui peut être de 3 types: [12].

- Une condition simple ( $a > 10$ ).

- Action retardée d' un certain temps à partir de l' activation de l' étape associée.
- Limitée dans le temps, c' est à dire que l' action est exécutée dès l' activation de l' étape mais avec un temps limité.

### II.5.1.2.3 Les actions mémorisées

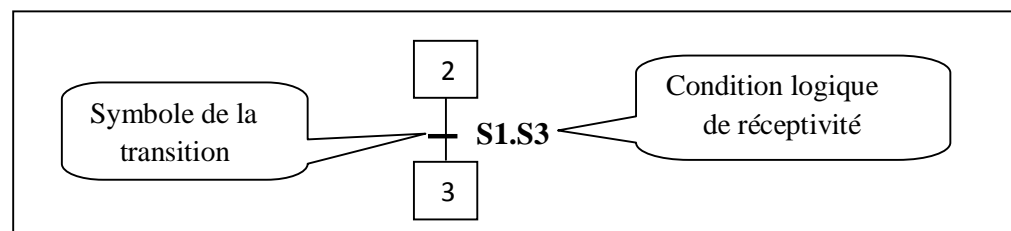
Permet de mettre la sortie correspondante dans un état spécifié lors de son activation. La désactivation de l' étape ne remet pas la sortie associée à son état d'origine. Le passage dans un autre état de cette sortie devra être décrit explicitement par une autre étape, [12].

### II.5.1.2.4 Les actions impulsives

Sont des actions temporisées de durée très petite dont la valeur est sans importance mais suffisante pour obtenir l' effet désiré, [12].

**II.5.1.3 La transition :** indique la possibilité d'évolution d'une étape active à l'étape suivante, elle représente une possibilité de changement de comportement du système. Ce passage correspond au franchissement d' une transition. Une transition se représente par un trait (tiret) horizontal.

On peut associer une interprétation à la transition grâce à une réceptivité (la proposition logique qui conditionne la transition). C'est une information provenant de la partie opérative et/ou de l'état du système et qui détermine si l'évolution correspondante à la transition est possible ou non, [12].



**Figure II .5 :** Symbole de la transition et la condition logique.

### II.5.1.3.1 La réceptivité

La réceptivité est une fonction combinatoire d'informations telles que :

- Etats de capteurs.
- Action de boutons-poussoirs par l'opérateur.
- Action d'un temporisateur, d'un compteur.
- Etat actif ou inactif d'autres étapes.

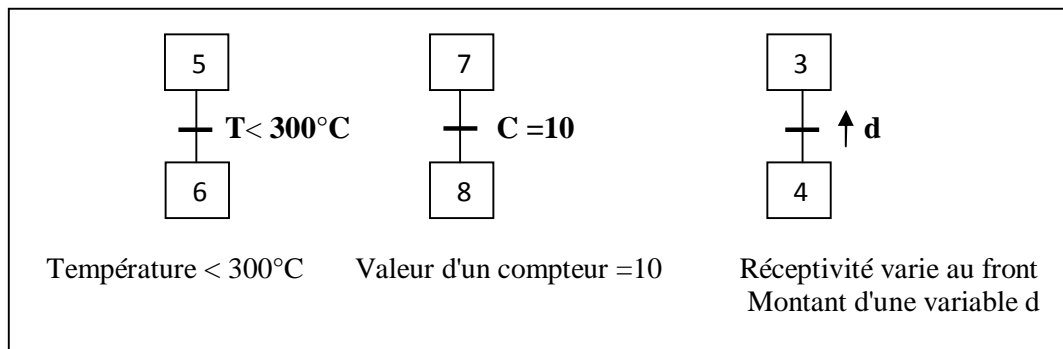


Figure II .6 : Ecriture de la réceptivité.

C'est une condition qui détermine la possibilité ou non d'évolution du système par cette transition, on distingue deux cas pour la réceptivité : [12].

II.5.1.3.2 La temporisation

Pour faire intervenir le temps dans une réceptivité, il suffit d'indiquer après le repère "t", son origine et sa durée. L'origine sera l'instant de début de la dernière activation de l'étape précédente. Lorsqu'une étape se trouve être origine d'un temps, il peut être utile de l'indiquer comme une action associée à cette étape.

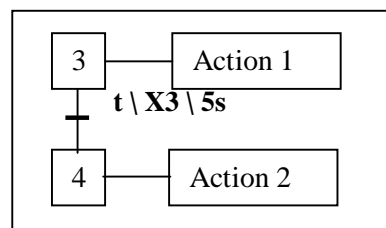


Figure II .7 : Ecriture d'une temporisation.

Les temporisations utilisées en GRAFCET font référence aux variables d'étapes et donc s'écrivent sous la forme (t \ X3 \ 5s) qui signifie que 5 secondes sont écoulées depuis la dernière activation de l'étape 3.

\_ La réceptivité « =1 »

Une réceptivité toujours vraie est écrite « =1 »

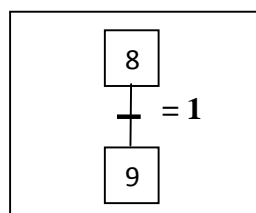


Figure II .8 : Transition avec réceptivité toujours vrai.

Une transition est dite validée lorsque toutes les étapes reliées en amont de cette transition sont actives, [12].

#### II.5.1.4 Les liaisons

Les liaisons indiquent les voies d'évolution du GRAFCET. Ceux qui se font du haut vers le bas ne comportent pas de flèches alors qu'il faut utiliser des flèches pour les liaisons de bas en haut.

Chaque liaison orientée relie une étape à une transition ou une transition à une étape, mais jamais deux étapes ou deux transitions entre elles : il y a toujours alternance stricte étape-transition, [12].

## II.6 Structure de base

Les structures de base les plus utilisées sont décrites ci – après.

### II.6.1 Séquence unique

Une séquence unique est composée d'une suite d'étapes pouvant être activées les une après les autres. Chaque étape n'est suivie que par une seule transition et chaque transition n'est validée que par une seule étape, (fig.II.9), [13].

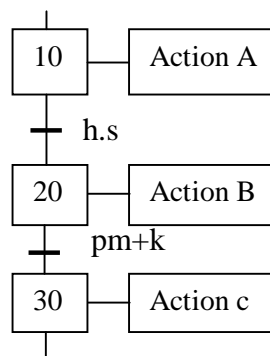


Figure II. 9 : Séquence unique.

### II.6.2 Séquence simultanée : parallélisme structural

Une transition peut supporter plusieurs étapes en amont et plusieurs étapes en aval : la notion de transitions en ET divergent, ET convergent est introduite. Le ET convergent permettent de décrire deux ou plusieurs séquences parallèles. Le franchissement d'une transition conduit à activer plusieurs séquences en même temps, ces séquences sont dites séquences simultanées.

Après l'activation simultanée de ces séquences, les évolutions des étapes actives dans chacune des séquences deviennent alors indépendantes. Pour assurer la synchronisation de la désactivation de plusieurs séquences en même temps, des étapes d'attente réciproques sont généralement prévues, (fig.II.10) et (fig.II.11), [13].

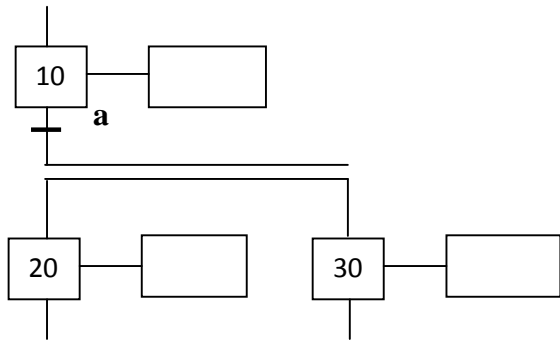


Figure II.11 : Transition en ET divergent.

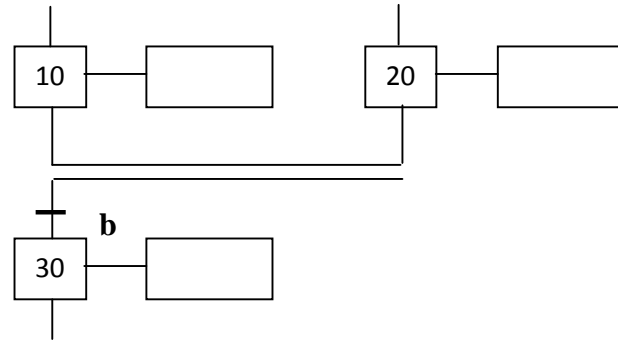


Figure II.10 : Transition en ET convergent.

II.6.3 Sélection de séquence

Une étape peut être reliée à plusieurs transitions en amont ou en aval : La notion de transition en OU convergent OU divergent est introduite. Le OU divergent permet de prendre en compte un choix, un (aiguillage) entre deux fonction. Une sélection ou un choix d'évolution entre plusieurs étapes ou séquences se présente, à partir d'une ou plusieurs étapes. Autant de transitions validées qu'il y a d'évolutions possibles (figII.12) (figII.13), [13].

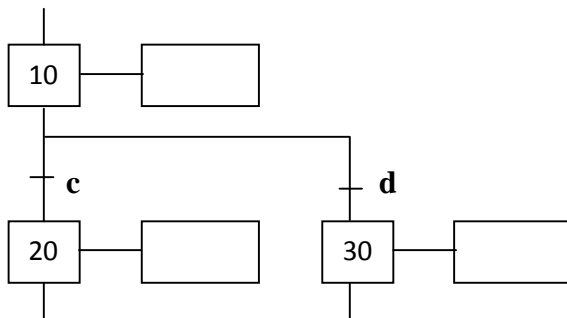


Figure II.12 : Transition en OU divergent.

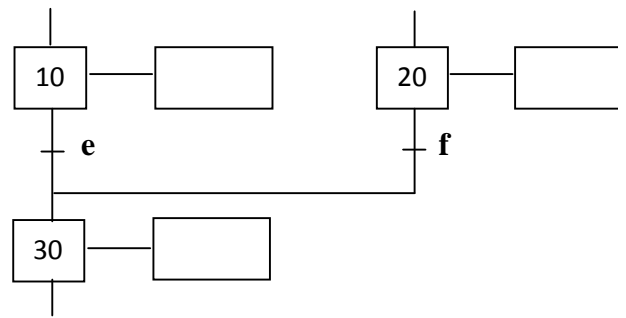


Figure II.13 : Transition en OU convergent.

II.6.3.1 Séquences exclusives

Pour obtenir une sélection exclusive entre plusieurs évolutions possibles à partir d'une même étape, il est nécessaire de s'assurer que toutes les réceptivités associées aux transitions sont exclusives, c'est-à-dire qu'elles ne peuvent pas être vraies simultanément, ( fig.II.14 ), [13].

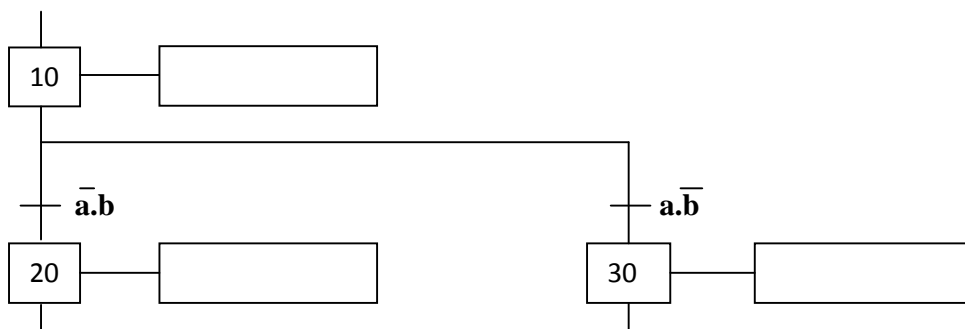
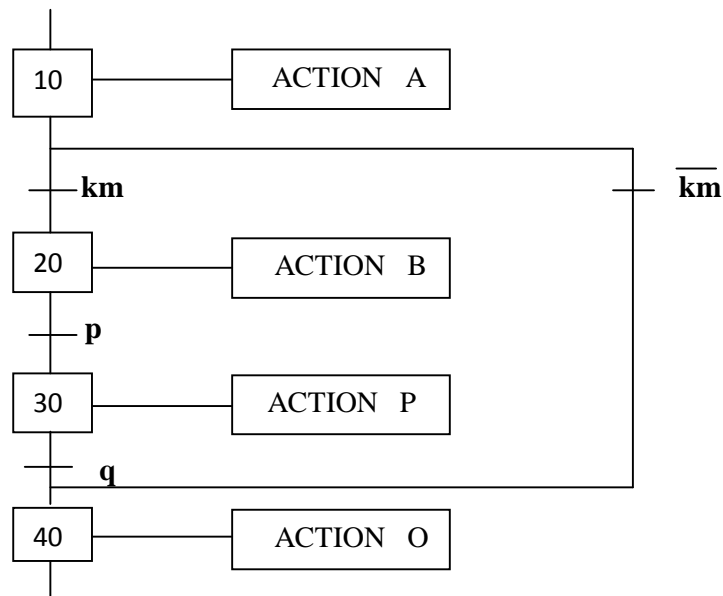


Figure II.14 : Une Séquences exclusives.

**II.6.4 Saut d'étape**

Saut d'étapes : permet de sauter une ou plusieurs étapes lorsque les actions à effectuer dans ces étapes deviennent inutiles ou sans objet, (fig.II.15), [13].

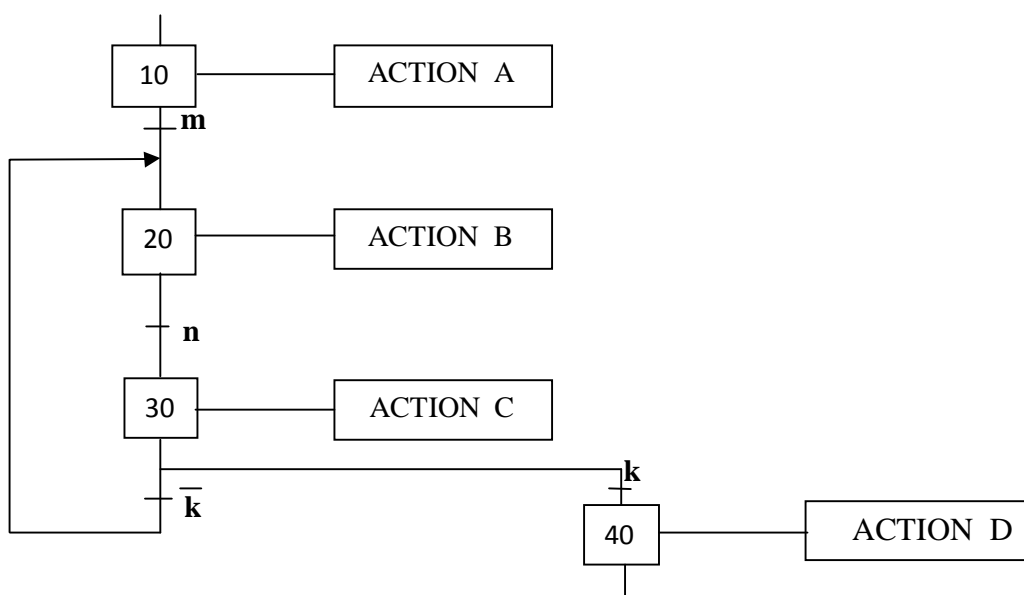


**Figure II.15** : Saut d'étape.

**II.6.5 Reprise de séquence**

Reprise de séquence : permet de recommencer plusieurs fois la même séquence tant que la condition fixée n'est pas obtenue.

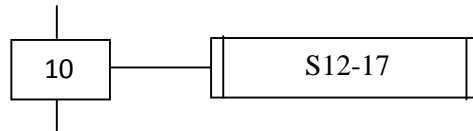
L'activation de l'étape 30 permet la reprise de la séquence 20-30 par la réceptivité  $Kc'$  tant que la réceptivité  $Kc$  n'est pas vraie, (fig.II.16), [13].



**Figure II .16** : Reprise de séquence.

**II.6.6 Réutilisation d'une même séquence**

Réutilisation d'une même séquence: une même séquence répétée plusieurs fois peut être traitée comme un sous programme, (fig.II.17), [13].



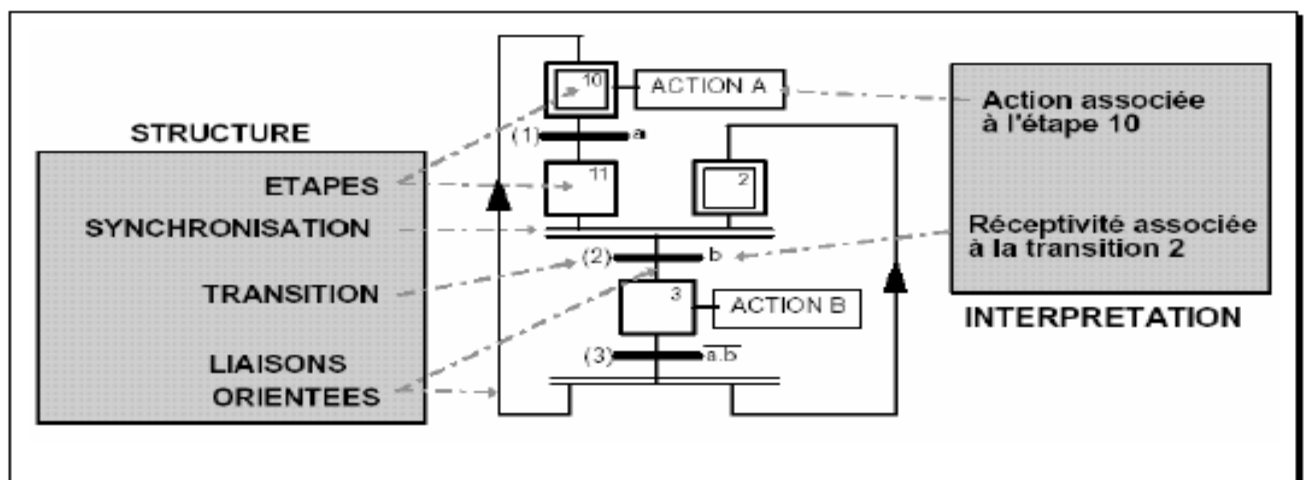
**Figure II. 17 :** Réutilisation d'une même séquence.

A l'étape 10 est associé le sous programme S12-17 qui comprend les étapes de 12 à 17. Ce sous programme doit faire l'objet d'un diagramme séparé.

**II.7 Les règles de construction**

- On relie les étapes et les transitions, qui doivent strictement alterner, grâce à des arcs orientés. Par convention, les étapes et les transitions sont placées suivant un axe vertical.
- Si plusieurs étapes doivent être reliées vers une même transition, alors on regroupe les arcs issus de ces étapes à l'aide d'une double barre horizontale appelée convergence "en ET" (ou synchronisation).
- Si plusieurs étapes doivent être issues d'une même transition, alors on regroupe les arcs allant vers ces étapes à l'aide d'une double barre horizontale appelée divergence.
- Lorsque plusieurs transitions sont reliées à une même étape dans le sens "vers étape" (respectivement dans le sens "d'étape"), on regroupe les arcs par un simple trait horizontal et l'on parle de convergence (respectivement divergence), [12].

On parle d'étape amont (respectivement d'étape aval) à une transition lorsque cette étape est avant (respectivement après) la transition au sens de la liaison orientée. De même on parlera de transition amont et de transition aval à une étape.



**Figure II. 18 :** Structure du GRAFCET et interprétation.

**II.7.1 Les règles d'évolution**

La modification de l'état de l'automatisme est appelée évolution (comportement dynamique du grafcet) qui est soumise à cinq règles qui précisent les causes et les effets du franchissement des transitions : [12].

**II.7.1.1 Règle 1 : Situation initiale**

L'initialisation précise le(s) étape(s) active(s) au début du fonctionnement, l'étape initiale est activée inconditionnellement en début de cycle.

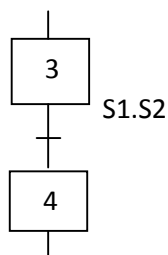
Cette situation caractérise le comportement initial de la partie commande vis-à-vis de la partie opérative, de l'opérateur et/ou des éléments extérieurs, [12].

**II.7.1.2 Règle 2 : Franchissement d'une transition**

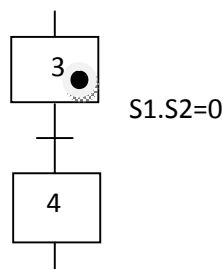
Une transition est dite validée lorsque toutes les étapes immédiatement précédentes reliées à cette transition sont actives, sinon elle est non validée.

Le franchissement d'une transition se produit:

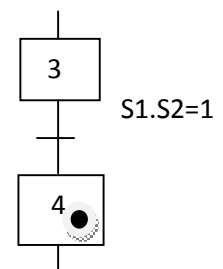
- lorsque la transition est validée.
- Et que la réceptivité associée à cette transition est vraie.



Transition non validée  
L'étape 3 n'étant pas Active, la transition 3-4 ne peut être validée.



Transition validée  
la transition 3-4 est validée mais ne peut être franchie car la réceptivité est nulle.



Transition franchie  
l'étape 3 était active et la réceptivité=1. C'est donc l'étape 4 qui est active.

Figure II .19 : Le franchissement d'une transition.

**II.7.1.3 Règle 3 : Evolution des étapes actives**

Le franchissement d'une transition entraîne simultanément la désactivation de toutes les étapes amont et l'activation de toutes les étapes aval, [12].

**II.7.1.4 Règle 4 : Evolution simultanée**

Plusieurs transitions simultanément franchissables sont simultanément franchies, [12].

### II.7.1.5 Règle 5 : Activation et désactivation simultanées d'une étape

Si au cours du fonctionnement la même étape est simultanément activée et désactivée, elle reste active ; une temporisation ou un compteur actionnés par cette étape ne seraient pas réinitialisés. Cette règle est prévue pour lever toute ambiguïté, et se rencontre très rarement dans la pratique.

Nous constatons que la durée de franchissement d'une transition ne peut être rigoureusement nulle, elle peut être rendue aussi petite que possible. Il en est de même de la durée d'activation d'une étape , [12].

## II.8 Les Macro étapes

Le concept de macro-étape permet des descriptions par niveau de détail successif. Ainsi Plusieurs niveaux de représentation peuvent être mis en oeuvre. Le premier niveau exprimant globalement la fonction à remplir sans se soucier de tous les détails superflus qui seront décrit dans les niveaux suivants, correspondant à une analyse plus fine. Finalement le dernier niveau pourra être celui correspondant à l'implémentation de la partie commande dont on spécifie le comportement, [1].

### II .8.1 Définition Les Macro étapes

Une macro-étape est l'unique représentation d'un ensemble unique d'étapes et de transitions nommé Macro-expansion .L'expansion de la macro-étape commence par une seule étape d'entrée et se termine par une seule étape de sortie.

On représente une macro-étape à l'aide de double barre dans le symbole d'étape. On repère une macro-étape à l'aide d'un identificateur commençant par la lettre M.

Lors de l'interprétation d'un grafcet (programmation), on remplace les macro-étapes par leur Macro expansion afin de pouvoir appliquer les règles d'évolution du GRAFCET. Une macro-étape sera dite "active" si au moins une étape de l'expansion est active.

Il n'y a pas d'action associée à une macro-étape cependant on peut faire figurer un commentaire donnant une indication de la fonction réalisée par la macro-étape, (fig.II.20), [1].

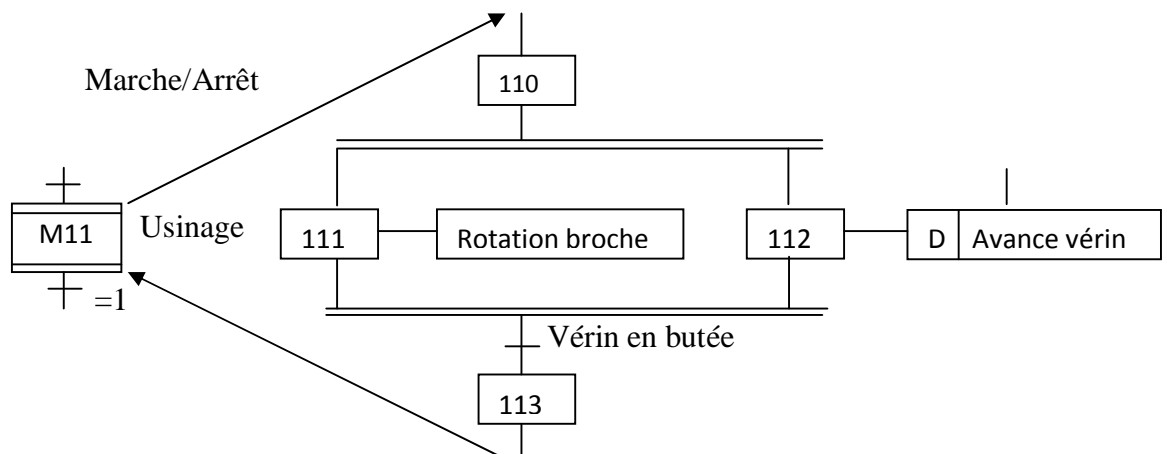


Figure II.20 : Macro étapes.

II .9 GRAFCET hiérarchisés

II .9.1 Définitions

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 le 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, (fig.II.21), [1].

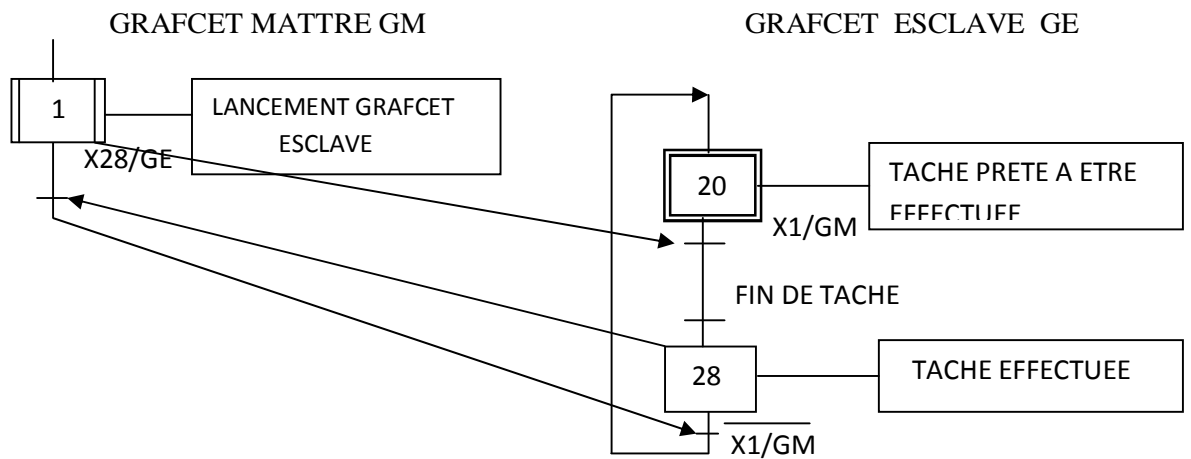
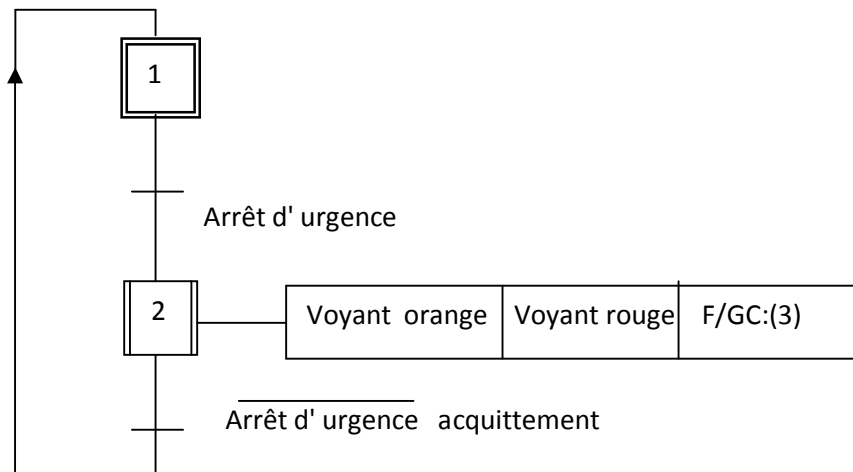


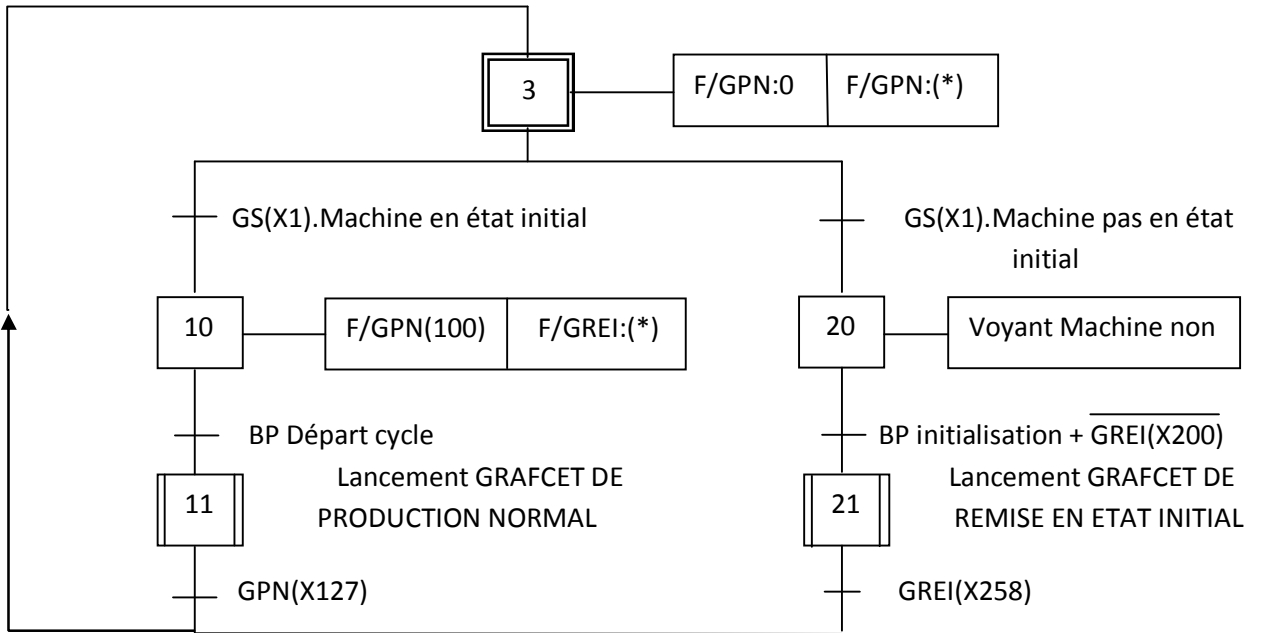
Figure II.21 : Structure GRAFCET hiérarchisés.

II .9.2 Fonctionnement

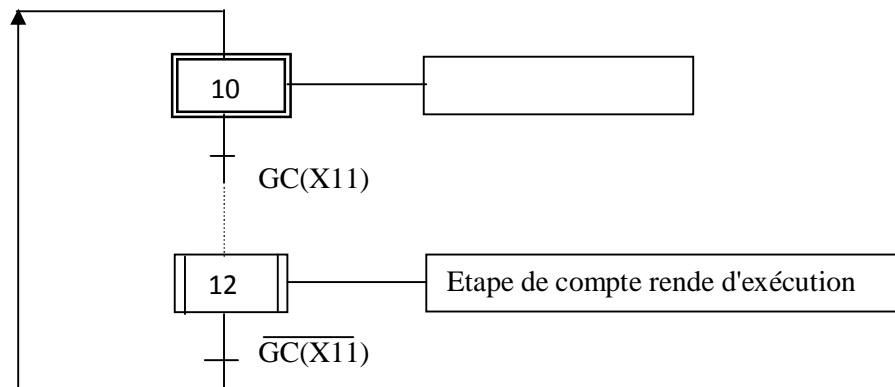
La double barre verticale indique qu'un ordre est donné à un GRAFCET esclave d'exécuter sa tâche. Cela se traduit par la présence de la réceptivité X1/GM (Appel) en transition de départ du GRAFCET de tâche. Lorsque le GRAFCET de tâche est terminé il renvoi un accusé d'exécution. Cela se traduit par la présence de la réceptivité X28/GE en transition de tâche effectuée. Le GRAFCET maître vient ensuite repositionner la tâche à l'étape 20 afin qu'elle puisse à nouveau être lancée, (fig.II.22), [12].



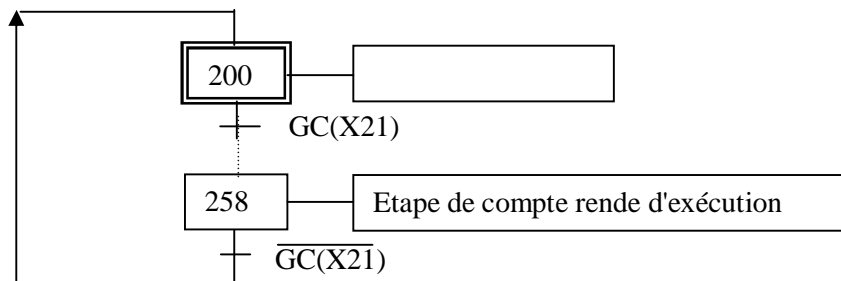
A - GRAFCET de sécurité (GS).



B - GRAFCET de conduite (GC).



C - GRAFCET de production normale (GPN).



D - GRAFCET DE REMISE EN ETAT INITIAL : GREI

Figure II.22 : GRAFCET HIERACHISES.

On peut noter sur cet exemple un forçage du GRAFCET de conduite à son étape initiale F/GC : (3) en cas d'arrêt d'urgence .Ce la entraîne l'extinction du GRAFCET de production normale F/GPN : ( ) Et le figeage du GRAFCET de remise en état initial F/GREI : (\*). Le GRAFCET de production normale est réinitialisé à l'étape 10 du GRAFCET de conduite F/GPN : (100) tandis que le GRAFCET de remise en état initial est maintenu figé. En cas d'arrêt d'urgence pendant une procédure de remise en état initial celle-ci se trouve stoppée mais sa situation est mémorisée. Son état mémorisé est restitué lorsque le bouton d'arrêt d'urgence est relâché et que l'on appui sur l'acquiescement d'arrêt d'urgence. La procédure de remise en état initial reprendra alors la ou elle avait été stoppée (voir réceptivité entre 200 et 201), [1].

## **II.10 Séquenceur**

Un séquenceur est un ensemble technologique qui commande et contrôle l'exécution des actions d'un cycle conformément à la chronologie des étapes du diagramme fonctionnel GRAFCET.

C'est l'organe de la partie commande qui dialogue pas à pas avec la partie opérative de la machine ou de l'installation. Il est constitué par l'association des modules d'étape du cycle à automatiser.

Aux étapes du GRAFCET correspond des modules d'étapes (relais bistable ou mémoire électronique ou pneumatique) interconnectés par un câblage représentatif du fonctionnement de la machine.

Les séquenceurs sont conçus en technologie:

- électrique,
- pneumatique,
- électronique.

Ces différentes technologies forment les équipements de base en logique câblés, mais avec l'émergence de la technologie programmée, les systèmes automatisés font appel progressivement aux automates (relier les entrées et sorties aux automates) pour remplacer les dispositifs câblés déjà existants,[1].

## **II.11 Conclusion**

*LE GRAFCET est un outil de description graphique du fonctionnement des systèmes automatisés.*

*Il permet de décrire les comportements attendus de l'automatisme en imposant une démarche rigoureuse, éventuellement hiérarchisée, évitant ainsi les incohérences, les blocages ou les conflits dans le fonctionnement.*

*Le chapitre suivant sera consacré à la description d'une unité de production de brique, dont le système automatisé industriel est basé sur la résolution par GRAFCET.*

### **III.1 Introduction**

Notre stage de deux mois au sein de la briqueterie D'ELKALAA BENI-HAMMED WILAYA DE M'SILA, 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 continue, et forment le pilier de toutes l'industries actuelles.

Dans le troisième chapitre nous donnerons en premier des notions générales sur l'unité de production briqueterie d'EL KALAA BENI-HAMMAD, puis nous terminerons par la description des différent étapes de fabrication de la brique

- préparation de l'argile,
- fabrication de la brique,
- cuisson,
- évacuation du produit finis,
- stockage.

### **III.2 Importance de la brique**

L'utilisation de la brique pour la construction en ALGERIE tend à se généraliser en remplacement du parpaing qui se trouve être un matériau lourd, sans est esthétique et produit par des moyens rudimentaires. Par contre la brique qui du point de vue coût est peut-être un peu plus chère, à une esthétique meilleure, se trouve être un matériau isolant et est produit en grande quantité par des usines de gros tonnage.

### **III.3 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 des terroirs inexploitées, le site occupe une superficie de 60.000 m<sup>2</sup>, 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.

Vu son emplacement, l'unité de production de briques d'EL KALAA BENI-HAMMAD (BKBH) a une grande importance sur le plan économique :

- Couvre toute le territoire de la wilaya de M'sila et partiellement Médéa, Bouira, Béjaia, Tizi-ouzou, et Tieret
- Plus proche d'Alger et du centre qui contient les plus grands chantiers, et en mètre temps elle proche de que Biskra (le premier alimentateur dans la région).

Remarque : La BKBH a influé considérablement sur les prix des briques dans cette région.

\_ Adresse : SARL Briqueterie El KALAA BENI-HAMMAD Route nationale n°40 M'TARFA W.M'SILA.

\_ Tel : 030 50 03 20 / 0770 75 25 22.

\_ Fax : 030 50 03 21.



**Figure III.1** : photos interne et externe du l'usine.

**III.3.1 Fiche technique de la Briqueterie KALAA BENI-HAMMED**

Superficie : 06 Ha.

Superficie couverte : 01 Ha.

Coût de projet : 700 000 000 DA.

Délais de réalisation : 24 mois.

Mode de réalisation : Lot par lot, équipements importé de France.

Emplois créés : 90 (Directs) ; Taux d'intégration 45%

Date en essai : Janvier 2007.

Date de réception : Février 2007.

Début de production: Mars 2007.

Capacité installée : 75 000 Tonnes/an.

Gisement d'argile : située à 10 Km de la briqueterie /réservé : +100ans.

Mélange : 80% Argile et 20% Sable.

Gammes de production : Brique creuses.

Types normalisés : 200\*150\*300 (12 Trous).

200\*100\*300 (10 Trous).

200\*300\*65 (06 Trous jumelés).

**III.3.2 Effectif a l'unité**

Les cadres : 09	Les maitres : 11	Les exécuteurs : 55	Nombre total de travailleur : 75
-----------------	------------------	---------------------	-------------------------------------

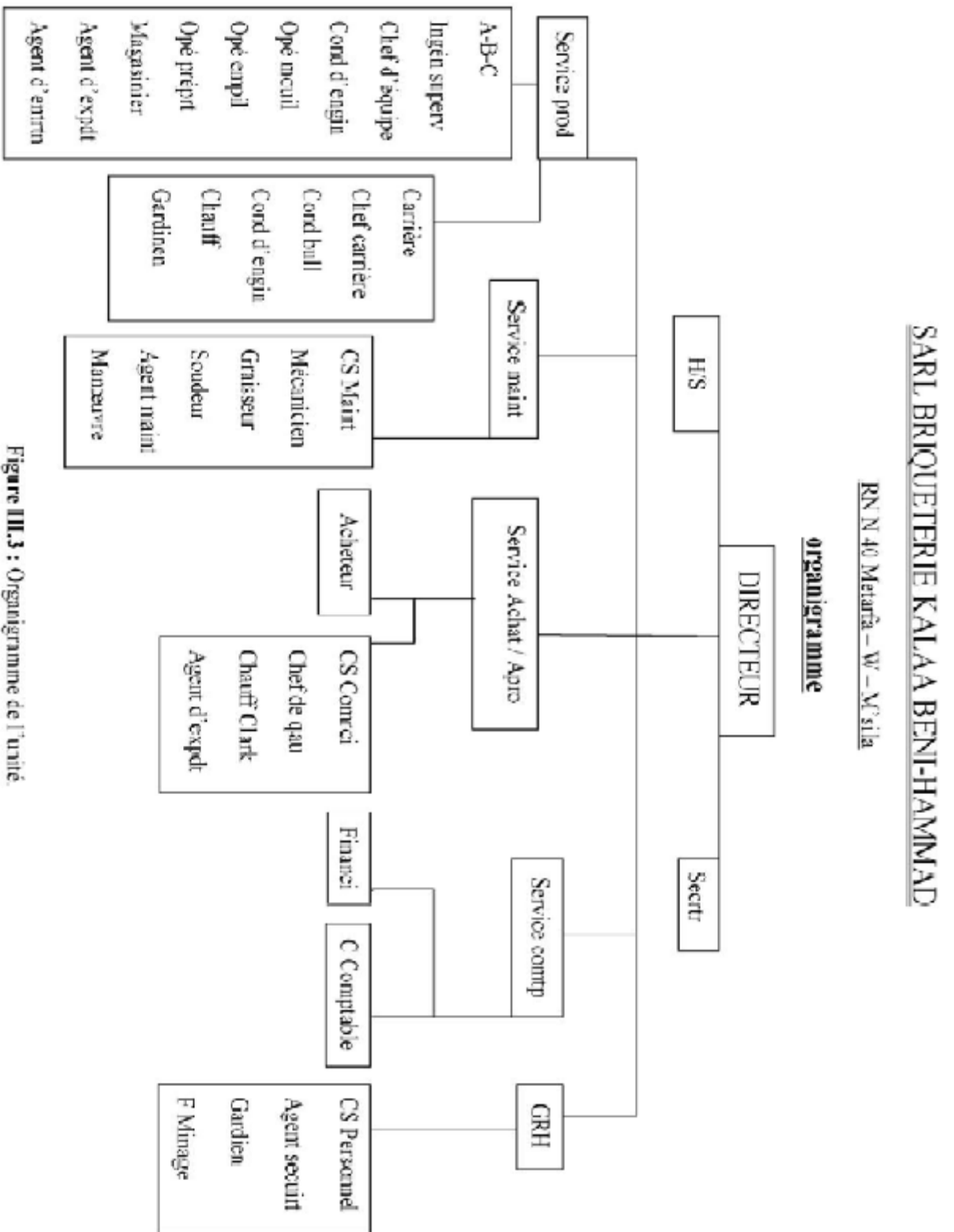


Figure III.3 : Organigramme de l'unité

**III.3.3 Caractéristique technique de l'unité**

1) les capacités de production (75 tonnes /an ) contractuelles

- préparation :

- 35 T/H en moyenne .
- 2 postes de 7 H 30 /jour .
- 5.5 jours /semaine .

- Fabrication / Séchage :

- 18.5 T / H en moyenne .
- 2 postes de 7 H 30 / jour .
- 6 jour / semaine .

- Cuisson :

- 336 jours / an , a 24 H / 24H .

- Production :

- 230 tonnes / jour en briques creuses . soit 27.8 wagons par jour en briques de 10 / 20 / 30 .

Remarque :

- Argile + 20 % sable .
- 3 % rebut (casse) .

**III.3.4 Puissances des équipements utilisés dans les processus**

La briqueterie contient sept (07) processus alimenté a partir des transformateurs de puissances de  $P = 3 \times 630 \text{ KVA}$ , Couplage Yn11 (Tension du réseau  $U = 30 \text{ KV}$ ).

Les grandes opérations sont essentiellement :

- \_ Opération de préparation de terre : 250 KW ; 496 A.
- \_ Opération de fabrication : 510KW ; 1075 A.
- \_ Opération de séchoir : 275 KW ; 470 A.

- \_ Opération de chargeur séchoir : 15 KW ; 30 A.
- \_ Opération empileur : 25 KW ; 43 A.
- \_ Opération manutention four : 25 KW ; 43 A.
- \_ Opération four : 100 KW ; 150 A.

### **III.4 Maintenance et sécurité de l'unité**

#### **III.4.1 Maintenance**

L'unité contient un service maintenance qui assure leur maintenance avec une équipe de (13) personne et magasin considérable ,ordonnés comme suit :

- (01) Chef service maintenance : ingénieur en maintenance industriel,
- (03) mécanicien soudeur et (03) magasinier : postés avec les équipes de production,
- (02) mécanicien graisseur,
- (01) mécanicien responsable du parc ( engins, camion, véhicule),
- (01) mécanicien auto,
- (01) mécanicien soudeur,
- (01) agent d' entretien.

Tout l'équipe travaille seulement le jour ( sauf les postés ) ; et assure le préventif et curatif de l' unité :

- Les gros travaux préventif sont programme la nuits (heurs d' arrêt de fabrication ).
- Les graissage et lubrification se font aux heures d' arrêt de l' unité ( sauf possible ).
- La programmation des travaux se fait par coordination avec le service production .
- La maintenance des appareillages électriques est assuré par les quatre ingénieurs cuisseurs qui s' occupent de la salle de contrôle et l' intervention ( panne électrique )
- Les mécanicien soudeur et les magasins postés assurent le dépannage de l' unité en dehors des heures de surface .
- L'agent d'entretien assure la propreté de l'unité.
- L'unité contient un magasin géant pour pièces de rechange et outillage.

#### **III.4.2 Sécurité d'usine**

Pour protéger l'unité et par personnel , un réseau de canalisation d'eau est installé a 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 , sectionnement rapide et accessible.....ext).

### III.5 Processus de fabrication

#### Le schéma technologique de l'unité

Se compose de trois atelier : **Préparation , fabrication et la cuisson.**

#### III.5.1 préparation de terre

Les étapes de la réparation de la terre sur illustré par la figure (III.2).

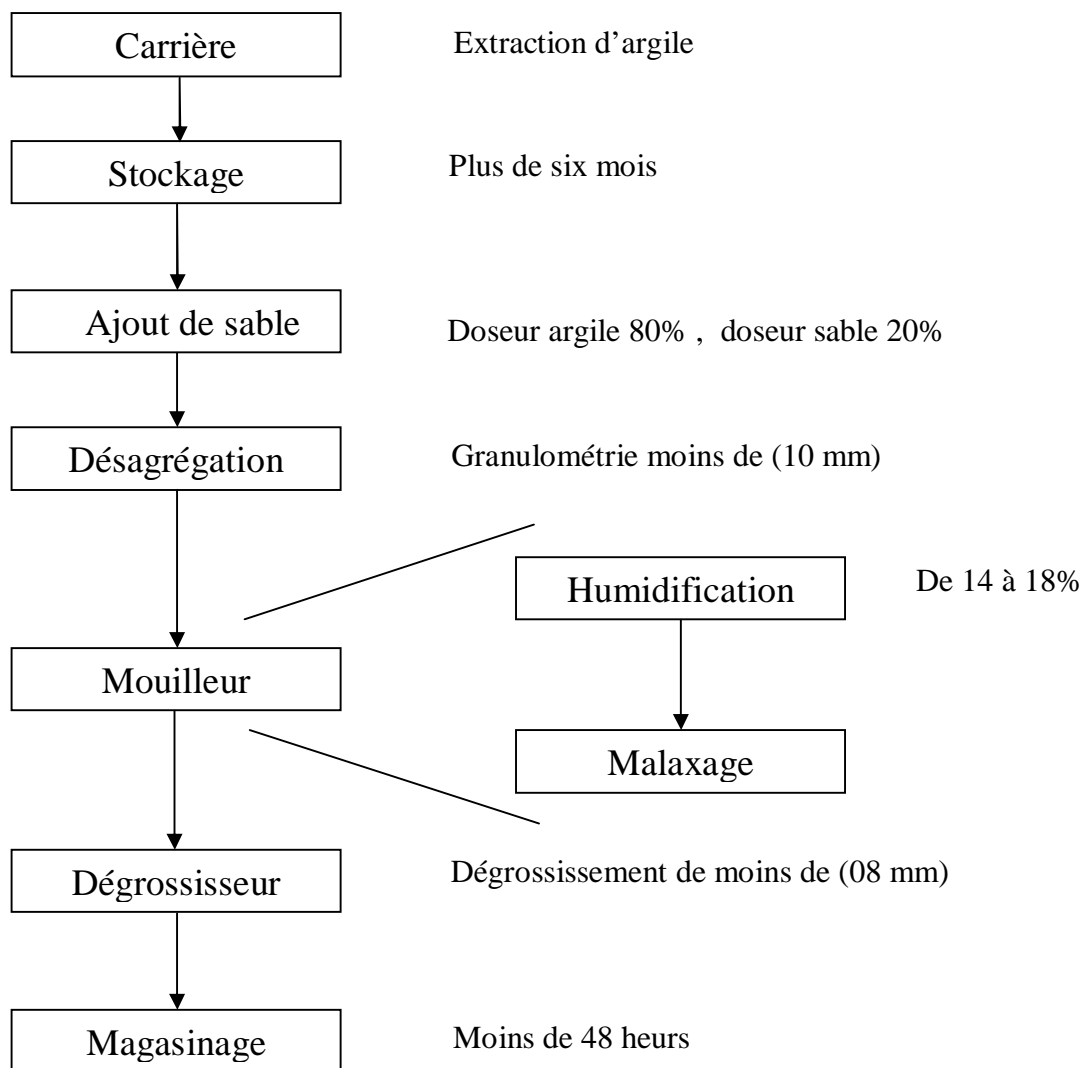


Figure III.3 : cycle préparation de terre .

III.5.2 fabrication et séchage

La Figure( III.3), donne un apprécier sur l' étape de fabrication et séchage.

Reprise du meulage homogène

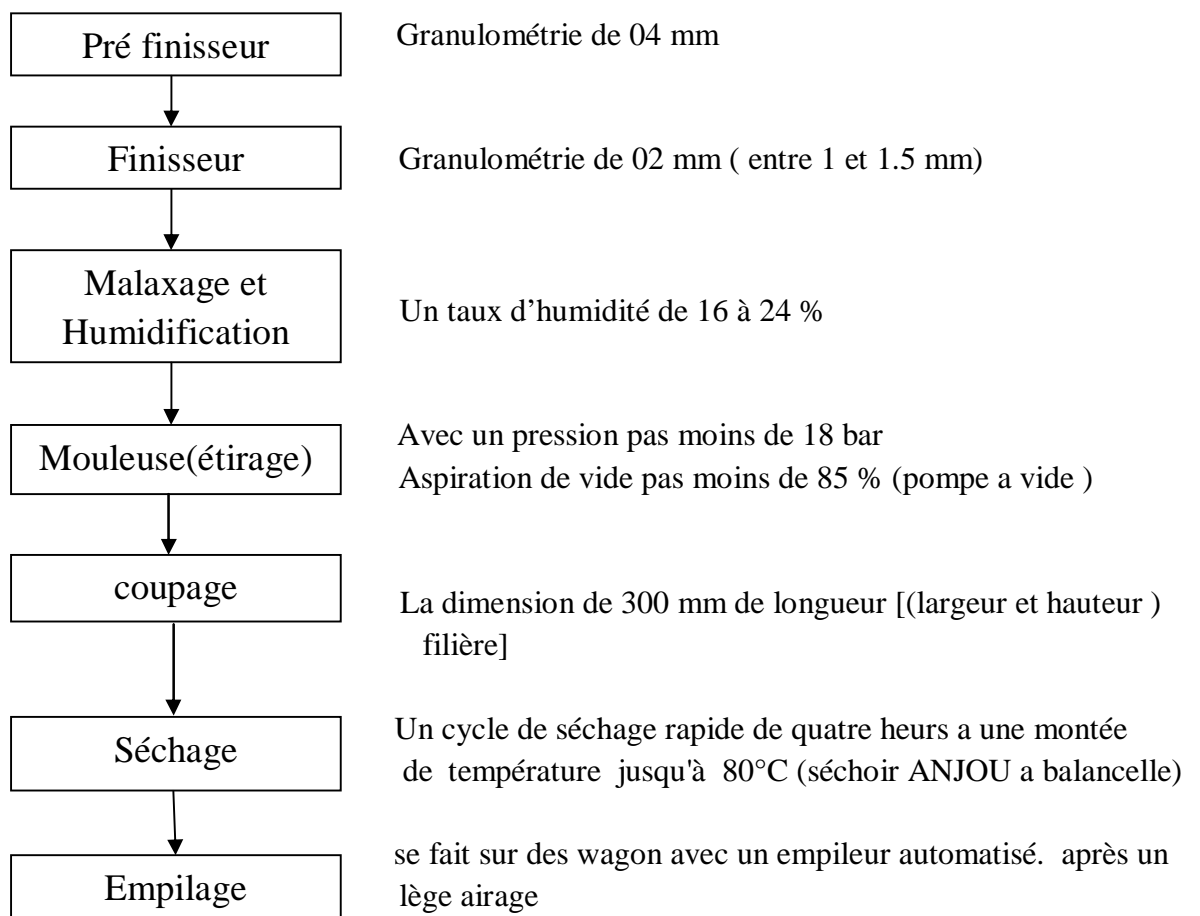
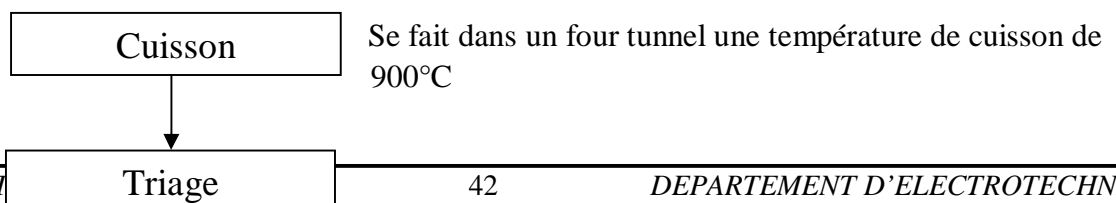


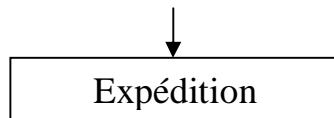
Figure III.4 : cycle fabrication et séchage.

III.5.3 La cuisson

La tâche de cuisson est présenté par la figure (III.4).



Déchets de 1 % à peu près



**Figure III.5 :** cycle de cuisson.



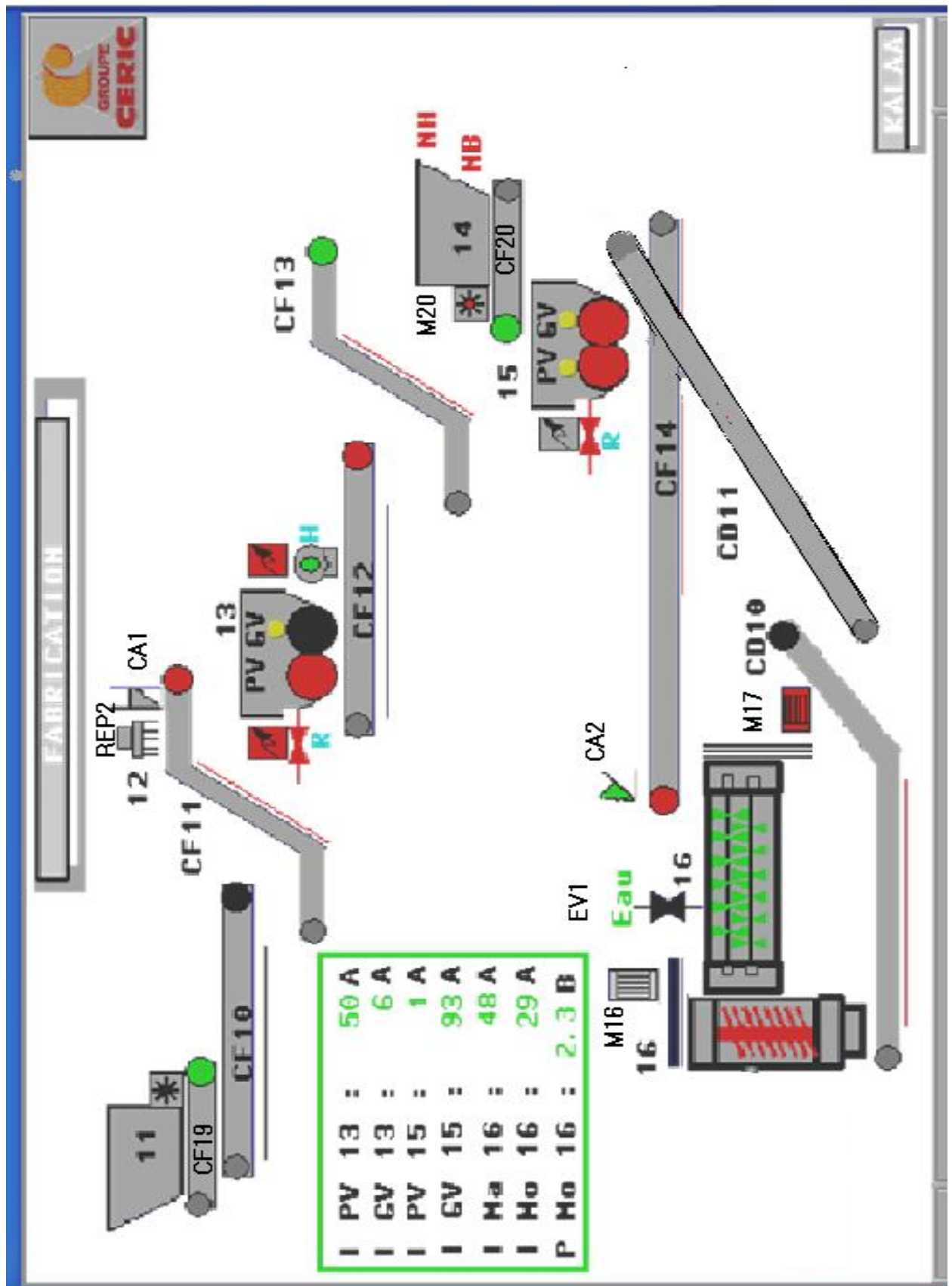


Figure III.7 : schéma synoptique de l'installation de fabrication affiché sur pupitre.

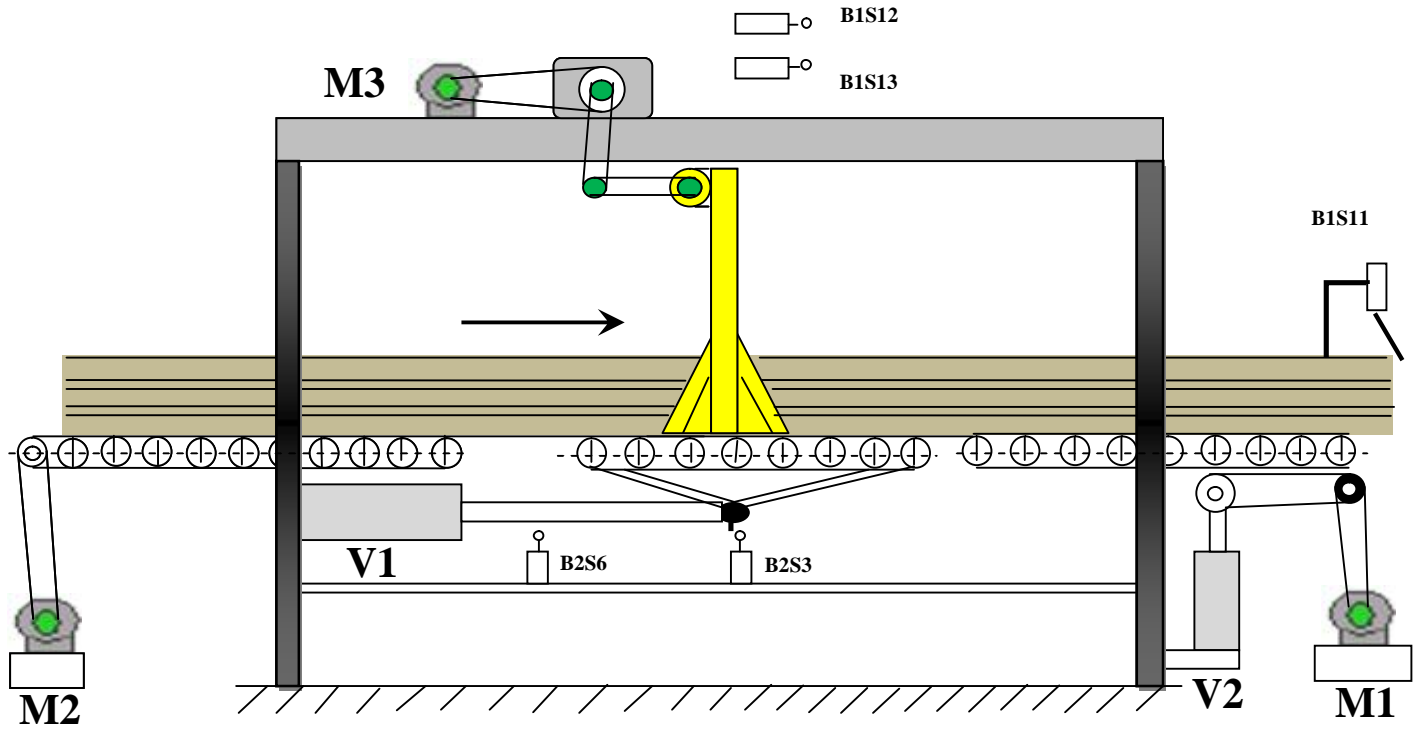


Figure III.8 : schéma du pré coupeur.

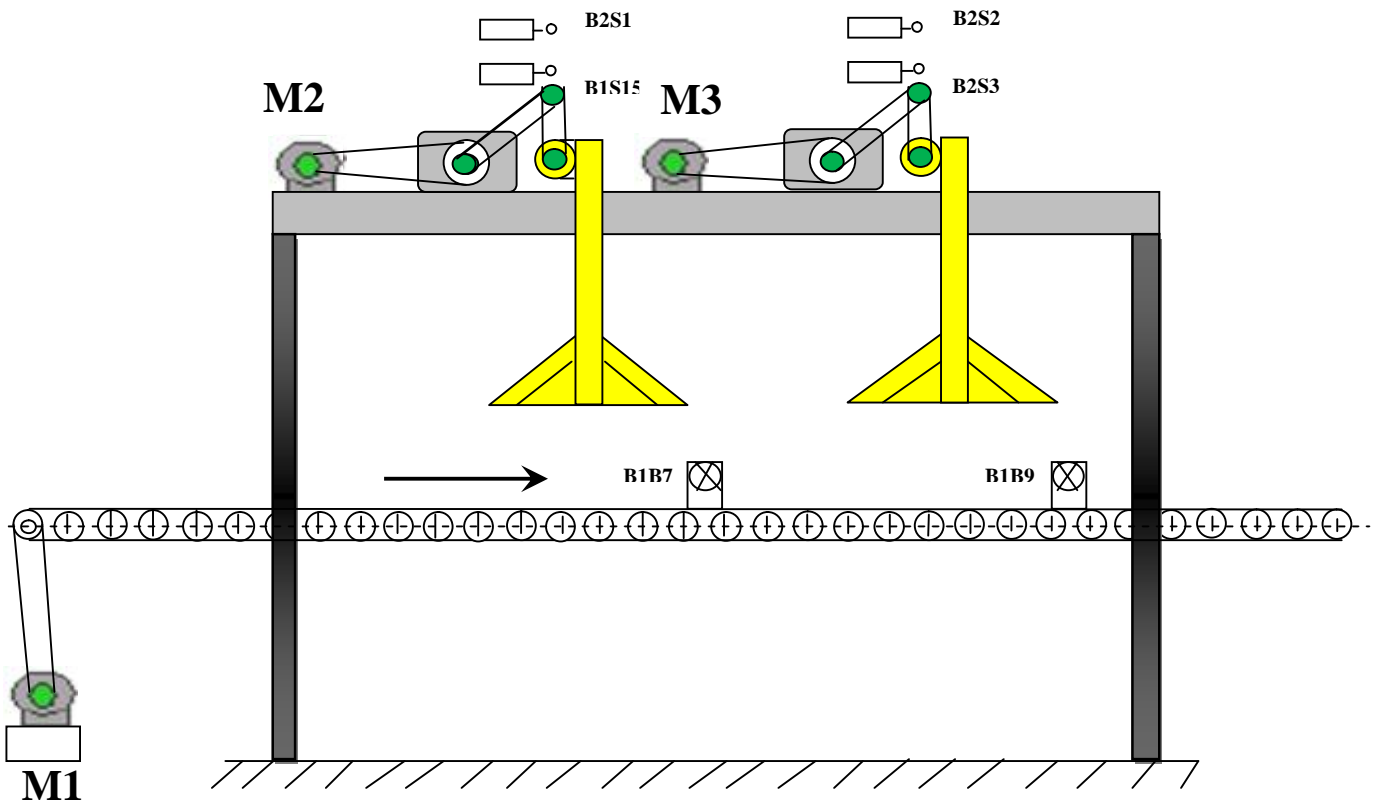


Figure III.9 : schéma du coupeur.

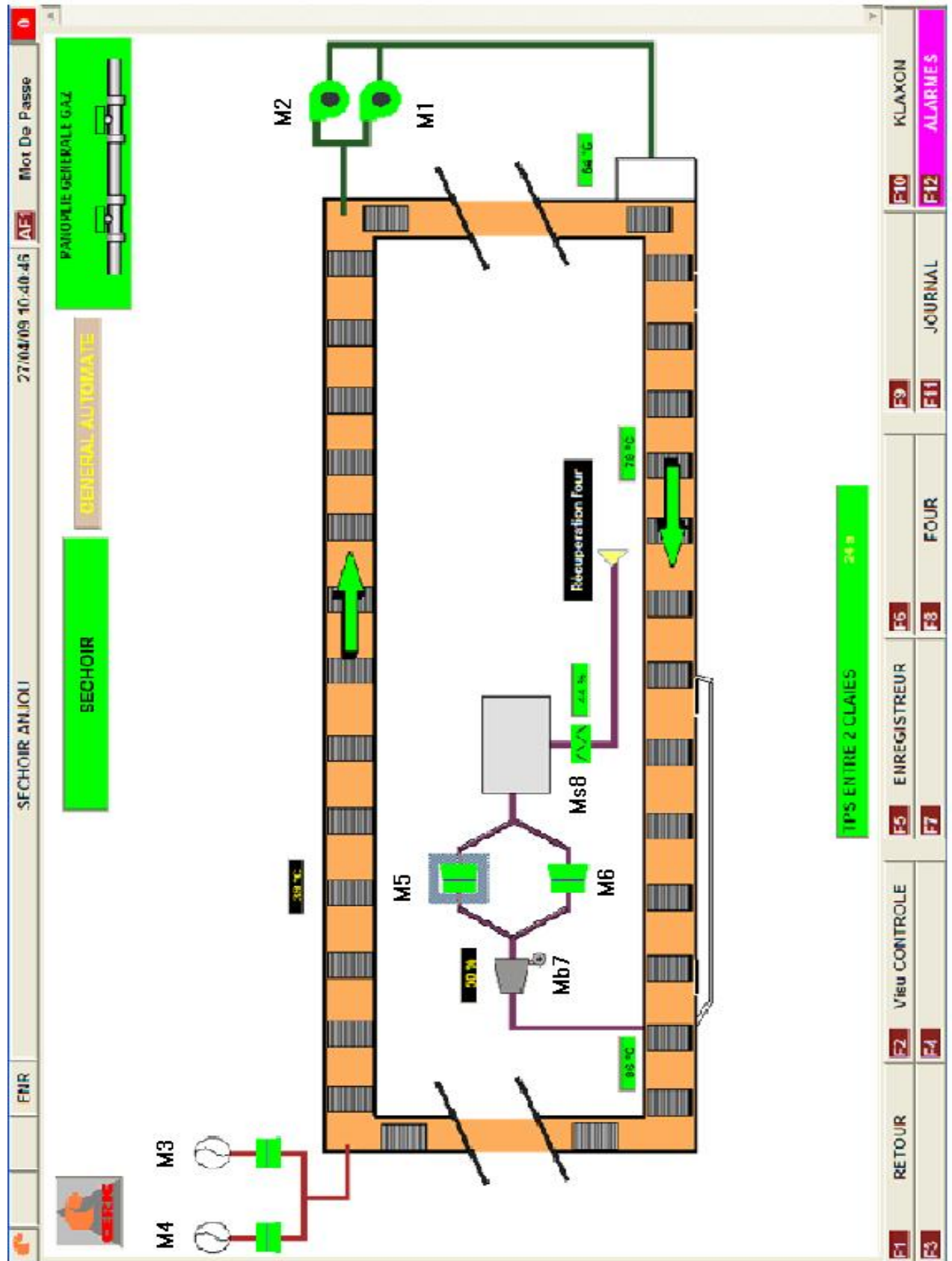


Figure III.10 : schéma synoptique de l'installation de séchoir affiché sur microcontrôleur .

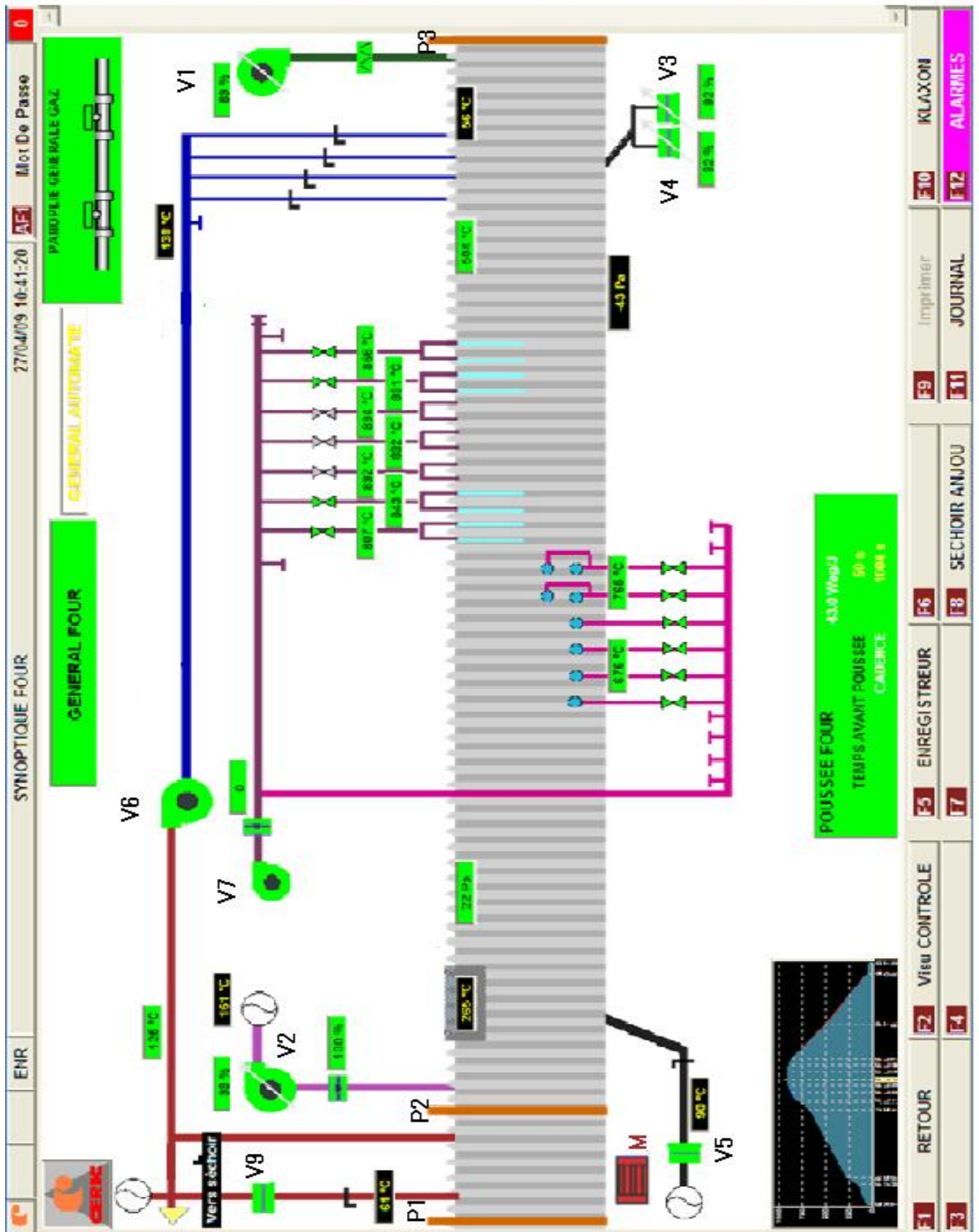


Figure III.11 : schéma synoptique de l'installation de four affiché sur microcontrôleur .

### **III.6 Conclusion**

*Dans ce chapitre, on a donné un aperçu sur l'entreprise de la Briqueterie D'EL KALAA/HAMMAD, ainsi que les statistiques qui reflètent le taux de production et on à ensuite présenter les étapes de fabrication de la brique.*

- *Préparation de terre ( 80% argile et 20% sable ),*
- *Fabrication et séchage la brique mouillé ,*
- *Cuisson la brique et en fin expédition ,*

*Pour concrétiser les tâches cités au paravent on doit passé par la résolution GRAFCET de ces dernière, ce qui fera l'objet du chapitre suivant.*

## IV.1 Introduction

*Pour concevoir, réaliser et exploiter un automatisme, il est indispensable d'en décrire le comportement. Les outils ou langages qui permettent cette description peuvent, au choix, être littéraires symboliques ou graphiques. Il est important de connaître ces outils qui se complètent pour permettre une expression claire.*

*Après une étude approfondie de l'usine, nous avons optés pour la solution graphique GRAFCET, qui s'applique à tout système logique de commande d'automatisme industriel, quelle qu'en soit la technologie (électrique, électronique, pneumatique), que cette commande soit câblée ou programmée.*

*C'est dans cet état d'esprit que s'inscrit notre quatrième chapitre, à savoir trouver les solutions GRAFCET des différentes tâches qui composent l'unité de production du briqué .*

*Dans ce chapitre, nous décrirons l'automate choisi, les différentes parties qui le constitue ainsi qu'une présentation du logiciel qui permet sa configuration et sa programmation, puis nous présenterons avec plus de détails le GRAFCET différentes tâches qui forment la chaîne de production.*

*Nous consacrerons la dernière partie à la programmation de la tâche de préparation toute faisant intervenir les ordres de forçages et d'initialisation émanant des Grafcets de sécurité, d'initialisation et de coordination des tâches. Pour répondre à ces exigences on a choisit le matériel Schneider,*

## IV.2 Présentation de l'automate

### a) Schneider électrique dans le monde

L'expertise d'un leader mondial, le service d'un partenaire de proximité

- Spécialiste mondial de la gestion de l'électricité et des automatismes
- Une offre complète de produits et de service pour répondre aux besoins de quatre grands marchés : Energie et infrastructure, industrie, bâtiment et résidentiel.
- Trois marques mondiales :Merlin Gerin, Télémécanique, Square D
- Présence dans 130 pays
- 75000 collaborateurs
- Chiffre d'affaires de 10 milliards d'euros
- Recherche et développement :5.2% du chiffre d'affaires

Une offre toujours plus complète pour répondre aux besoins des marchés satisfaire les attentes des clients avec des solutions toujours plus complètes, innovantes et différenciées dans toutes les applications, [14].

- Energie et infrastructures,
- Industrie,
- Bâtiments,
- Résidentiel.

### b) Schneider electric Algérie

Plus de 45 ans de présence

- Une présence en Algérie au travers des marques Merlin Gerin et Télémécanique depuis 45 ans,
- Création du bureau de liaison Schneider électrique en 1994,
- Inauguration de la filiale en octobre 2002,
- Siege de la filiale situé à chéraga à 10km à l'ouest d'Alger, [14].

### IV.3 Choix de l'automate

Il existe deux types de l'automate, automate **TSX Micro** et **TSX Premium**

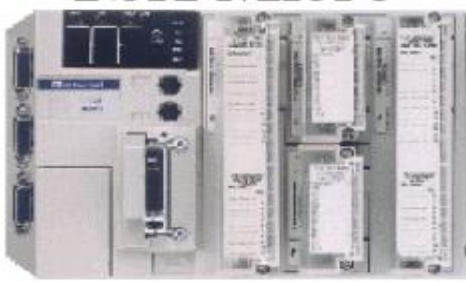


Figure IV.1 : automate TSX Micro .



Figure IV.2 : automate TSX Premium .

### IV.4 Présentation des logiciels PL7

#### IV.4.1 Généralités

La conception et la mise en oeuvre des applications pour automates Micro et Premium se réalisent à l'aide des logiciels PL7. Il est proposé 3 types de logiciels PL7, [14]:

- \_ PL7 Micro
- \_ PL7 Junior
- \_ PL7 Pro

#### IV.4.2 Le programme PL7-PRO

PL7-PRO est le langage que nous allons utiliser pour programmer nos automates. Il permet l'accès à tous les éléments des TSX. Le programme sera écrit sur ordinateur puis transféré sur l'automate. Il ne pourra être validé qu'en présence de celui-ci, [14].

#### IV.4.3 Les différents langage de PL7

##### IV.4.3.1 Langage à contacts

Le langage à contacts (LD) est un langage graphique. Il permet la transcription de schémas à relais, il est adapté au traitement combinatoire.

Il offre les symboles graphiques de base : contacts, bobines, blocs. L'écriture de calculs numériques est possible à l'intérieur de blocs opérations, [14].

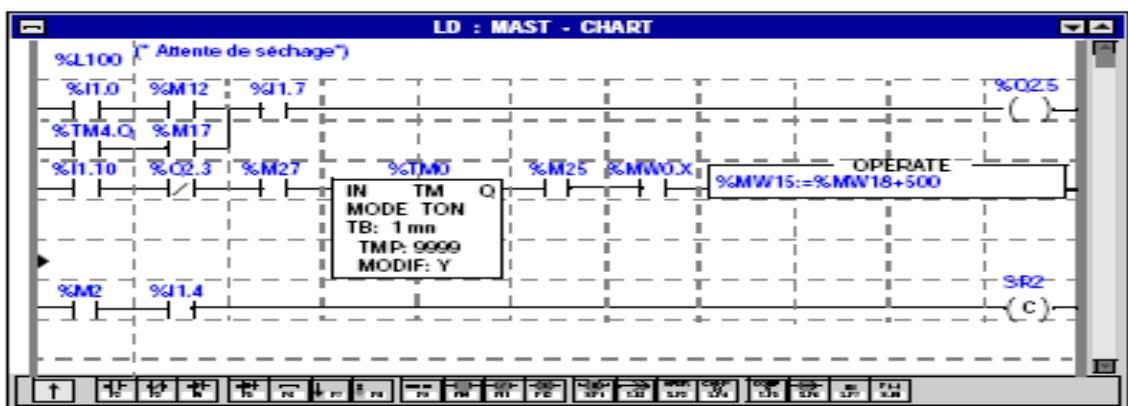


Figure IV.3 : exemple de réseau de contacts .

### IV.4.3.2 Langage Grafcet

Le langage Grafcet permet de représenter graphiquement et de façon structurée le fonctionnement d'un automate séquentiel, [14].

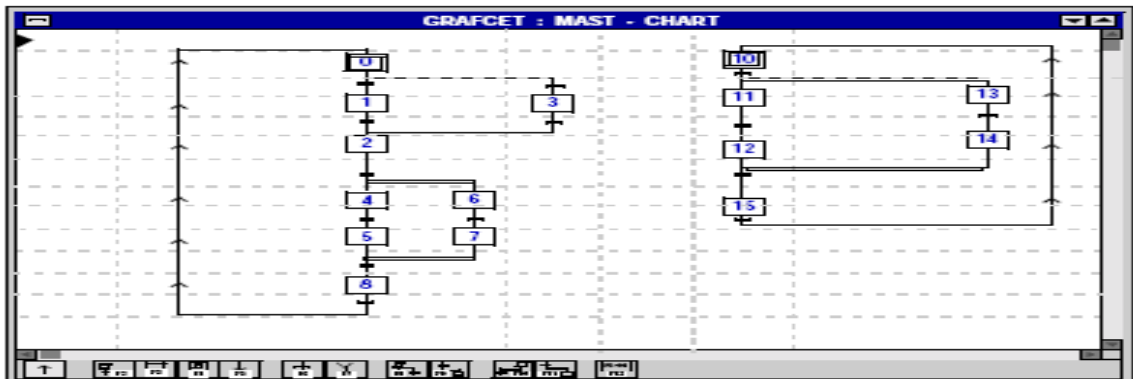


Figure IV.4 : Exemple de programme en langage Grafcet.

### IV.4.3.3 Langage liste d'instructions

Le langage liste d'instructions (IL) est un langage "machine" booléen qui permet l'écriture de traitements logiques et numériques, [14].



Figure IV.5 : exemple de programme en langage liste d'instructions .

### IV.4.3.4 Langage littéral structuré

Le langage littéral structuré (ST) est un langage de type "informatique" permettant l'écriture structurée de traitements logiques et numériques, [14].

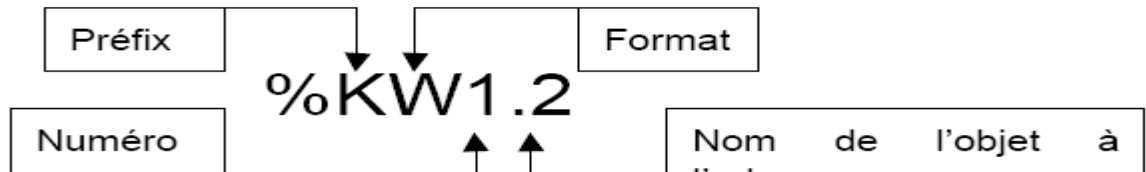


Figure IV.6 : exemple de programme en langage littéral structuré .

### IV.5 Les objets disponibles sur PL7-PRO

#### IV.5.1 Repère

Tous les objets manipulés ont un repère qui commence par le caractère % suivi de lettres ou de nombre qui nous informe sur son identité, [14].



#### IV.5.2 Préfixe

En fonctionnement, les entrées ne sont modifiables que par la partie opérative. C'est l'automate qui fixe les valeurs des sorties. L'automate peut lire et écrire sur des mémoires, mais uniquement lire les objets système et les constantes, [14].

Mémoire	M	Système	S	Constantes	K
Entrée	I	Sortie	Q		

Tableau (IV.1) : le préfixe sur PL7-PRO.

#### IV.5.3 Format

Pour les entrées-sorties, le numéro nous donne une indication sur la situation géographique de l'élément. Pour les autres éléments, il donne juste une indication sur le rang de l'élément, [14].

Booléen (1 bit)		Byte (8 bits)	B	Word (16 bit)	W
Dword	D	Réel	F		

Tableau (IV.2) : format des éléments d'indication .

#### IV.5.4 Numéro

Pour les entrées-sorties, le numéro nous donne une indication sur la situation géographique de l'élément. Pour les autres éléments, il donne juste une indication sur le rang de l'élément, [14].

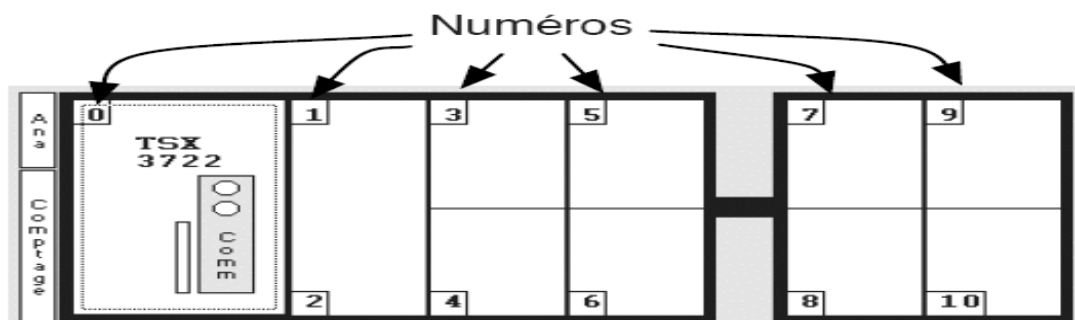
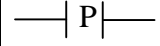



Figure IV.7 : indication sur la situation géographique de l'élément.

**IV.5.5 Front**

Pour les entrées et sorties uniquement (%Ii,j et %Qi,j), on peut détecter leur front montant, en utilisant les contacts suivants, [14] :

Front montant		Front descendant	
---------------	-----------------------------------------------------------------------------------	------------------	-------------------------------------------------------------------------------------

**Tableau (IV.3)** : la forme de contact entrée front(P , N) .

**IV.5.6 Bascule RS**

Une bascule RS est accessible pour les sorties booléens (%Ii,j) et les mémoires booléennes (%Mi) en utilisant les contacts suivants, [14] :

Set		Reset	
-----	-----------------------------------------------------------------------------------	-------	-------------------------------------------------------------------------------------

**Tableau (IV.4)** : la forme de contact de sortie bascule (R , S).

**IV.5.7 Temporisation**

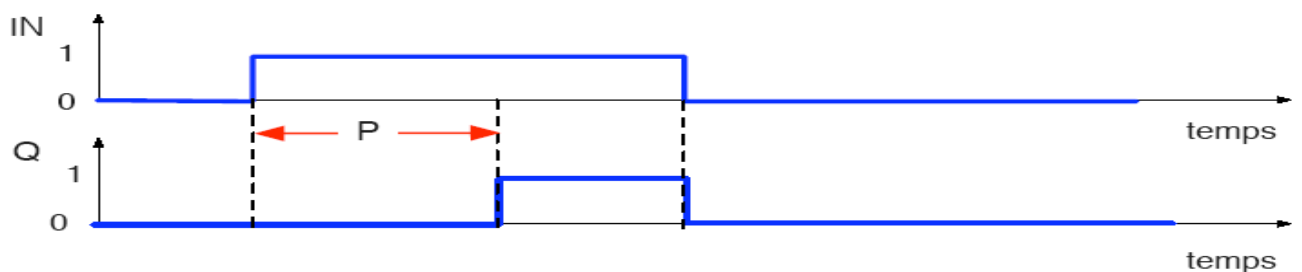
La temporisation (%Tmi) s’insère dans le réseau comme un objet avec une entrée et une sortie. La sortie fournira une valeur logique en retard par rapport aux entrées. La variable %Tmi.V fournit la valeur du temps, [14].

IN	Entrée			<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p style="text-align: center;">%TM1</p> <p>IN    TM    Q</p> <p>MODE : TON                      TB : 1mn                      TM.P : 9999                      MODIF : Y</p> </div>
Q	Sortie	% Tmi.Q		
MODE	Mode de fonctionnement		TON ,TOFF,TP	
TB	Base de temps		10ms,100ms,1s,1mn	
P	Présélection	% Tmi.P	De 0 à 9999	

**Tableau (IV.5)** :l’indication de temporisation .

**Figure IV.8** : bloc temporisation.

**IV.5.7.1 Les modes de fonctionnements**



**Figure IV.9** : TON, temporisation travail.

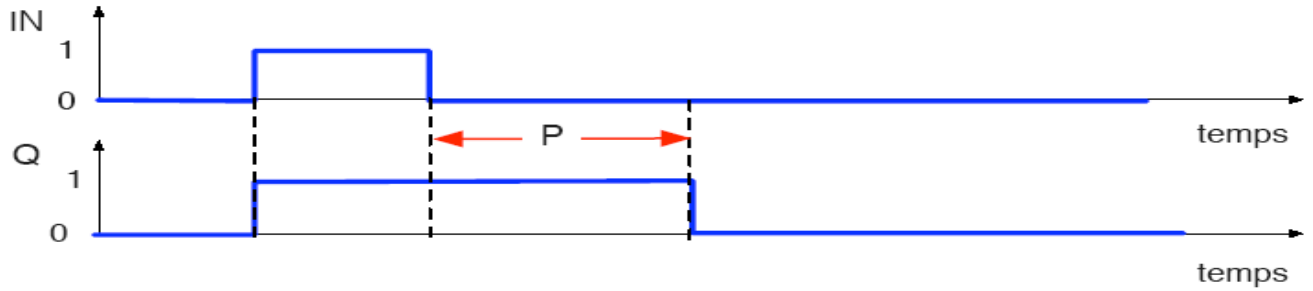


Figure IV.10 : TOFF, temporisation repos.

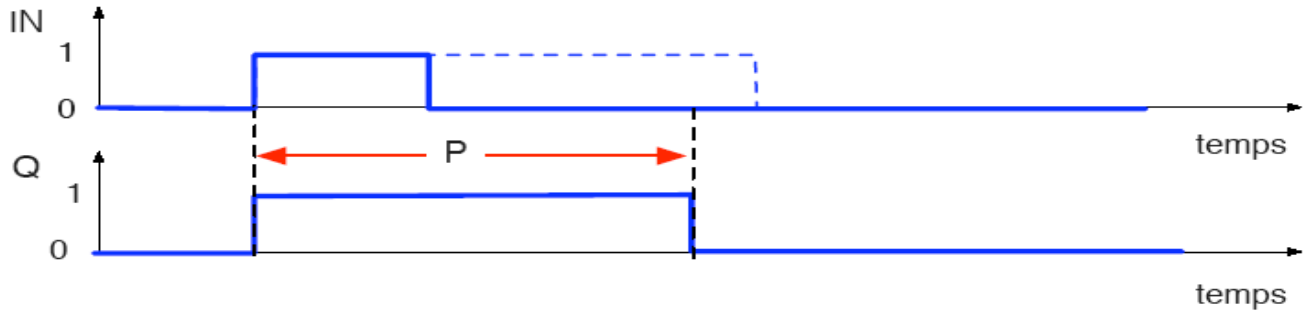


Figure IV.11 : TP, monostable.

IV.5.8 Monostable

Le monostable (%MNi) s’insère dans le réseau comme un objet avec une entrée et une sortie. La sortie fournira une valeur logique d’une durée paramétrable. La variable %MNi.V fournit la valeur du temps restant avant retour à zéro, [14].

S	Mise à 1 sur front montant		
R	Sortie	%MNi.R	=0 si MNi.V= 0
TB	Base de temps		1mn, 1s, 100ms, 10ms
MN.P	Valeur de présélection	%MNi.P	De 0 à 9999

Tableau (IV.6) : l’indication de monostable .

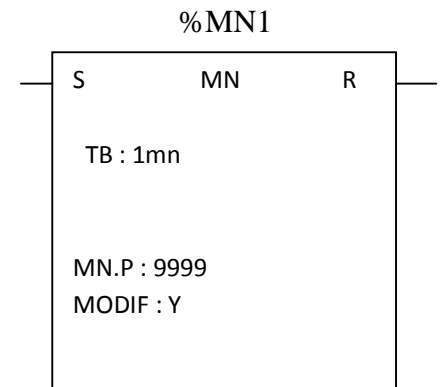


Figure IV.12 : bloc monostable .

IV.5.9 Registre

Un registre (%Ri) est un bloc mémoire permettant de stocker jusqu’à 255 mots de 16 bits de deux manières différentes, [14] :

- file d’attente (premier entré, premier sorti) appelée pile FIFO (First In, First Out),
- pile (dernier entré, premier sorti) appelée pile LIFO (Last In, First Out).

R	Remise à zéro	
I	In , provoque l'entrée du mot %Ri.I dans le registre	
O	Out, provoque la sortie d'un mot du registre dans %Ri.Q	
TYP	FIFO ou LILO	
LEN	Longueur	1 à 255
E	Registre vide	%Ri.E
F	Registre plein	%Ri.F

Tableau (IV.7): l'indication de registre .

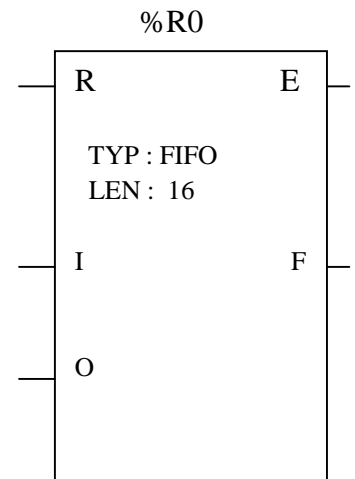


Figure IV.13 : bloc registre.

### IV.5.10 Compteur

Le compteur (%Ci) s'insère comme un objet avec quatre entrées et trois sorties dans le schéma à contacts. Les sorties fourniront des valeurs logiques calculées à partir de son état. La variable %Ci.V fournie la valeur du compteur, [14].

R	Remise à zéro		%Ci.V := 0
S	Remise à présélection		%Ci.V := %Ci.P
C.P.	Présélection	%Ci.P	
CU	Comptage(le front montant incrémente la valeur courante)		%Ci.V := %Ci.V +1
CD	Décomptage(le front montant décrémente la valeur courante)		%Ci.V := %Ci.V -1
E	Débordement décomptage	%Ci.E	
D	Présélection atteinte	%Ci.D	
F	Débordement comptage	%Ci.F	

Tableau (IV.8): l'indication compteur.

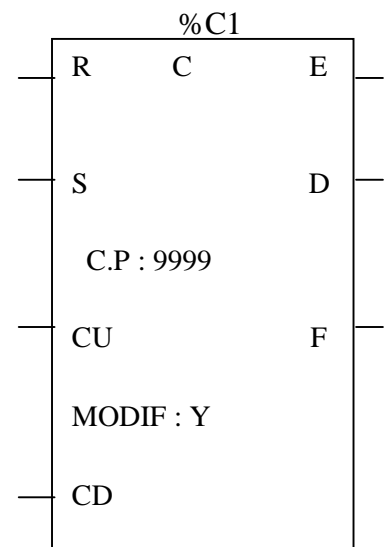


Figure IV.14 : bloc compteur.

### IV.5.11 Autres objets

#### IV.5.11.1 Bloc COMPARE

Le bloc compare permet de faire des comparaisons entre des nombres de mêmes types, entiers ou réels. Le circuit est fermé quand la comparaison est vraie, ouvert sinon. Pour une égalité on utilisera le signe =., [14].

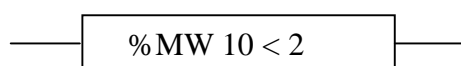


Figure IV.15 : bloc compare .

### IV.5.11.2 Bloc OPERATE

A l'aide de l'opérateur 'OPERATE', l'utilisateur est à même de faire des calculs (produit, division, somme et soustraction) entre des entiers ou des réels. Attention, pas de mélange des genres, les éléments présents dans une formule doivent être de mêmes types. Comme il s'agit d'une affectation et non d'une comparaison, on utilisera le := pour affecter le résultat du calcul de droite à l'élément de gauche, [14].

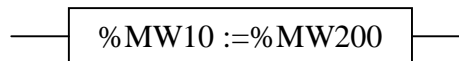


Figure IV.16 : bloc operate .

## IV.6 Editeur de configuration matérielle

### IV.6.1 Présentation

L'éditeur de configuration matérielle (figure IV.17) de PL7 permet de manière intuitive et graphique, de déclarer et configurer les différents éléments constitutifs de l'automate, [14] :

- le rack,
- l'alimentation,
- le processeur,
- le module métier.

### IV.6.2 Illustration

La configuration matérielle se présente ainsi, [14] :

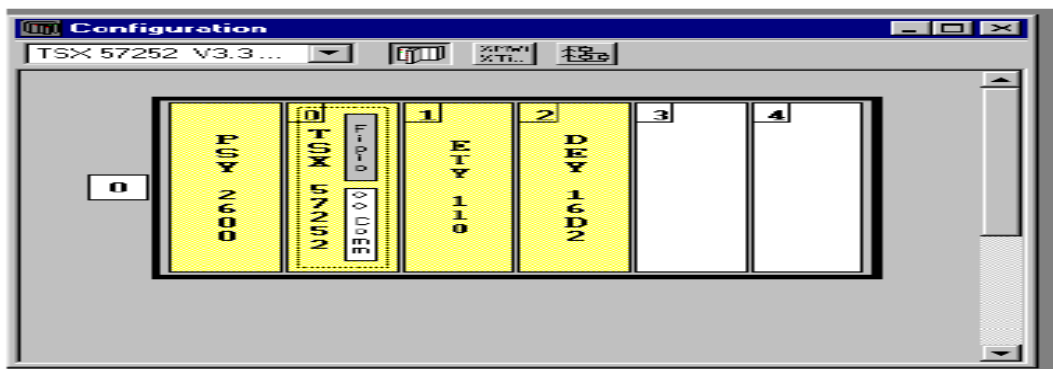


Figure IV.17 : Configuration matérielle .

La position 0 est attribuée au processeur et au module intégrés (cas du TSX 37-22).

### IV.6.3 Configuration de base

La configuration matérielle (figure IV.18) de l'application est accessible via l'éditeur de configuration du navigateur d'application. La configuration matérielle de base est constituée du module d'alimentation et du module processeur défini lors de la création de l'application, [14].

La configuration matérielle de base se présente ainsi :

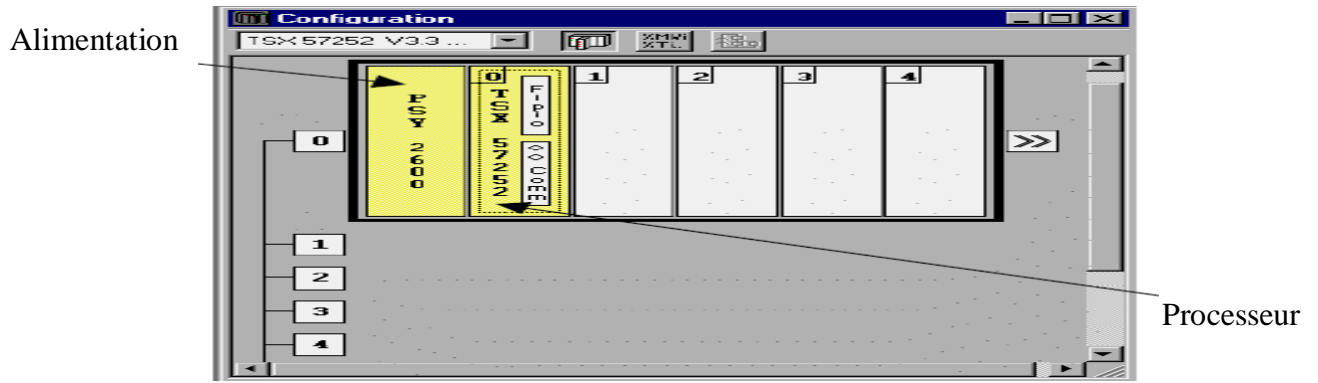


Figure IV.18 : Configuration matérielle .

Le point de connexion 0 est toujours attribué au gestionnaire du bus. La configuration met ici en évidence le module d'alimentation PSY 2600 et le module processeur TSX 57252 permettant d'établir la configuration du réseau FIPIO et du module de communication.

**IV.6.4 Module alimentation**

Une alimentation (figure IV.19) est définie par défaut lors création de l'application. Il existe en réalité 2 types de modules d'alimentation pour ce type de processeur, [14].

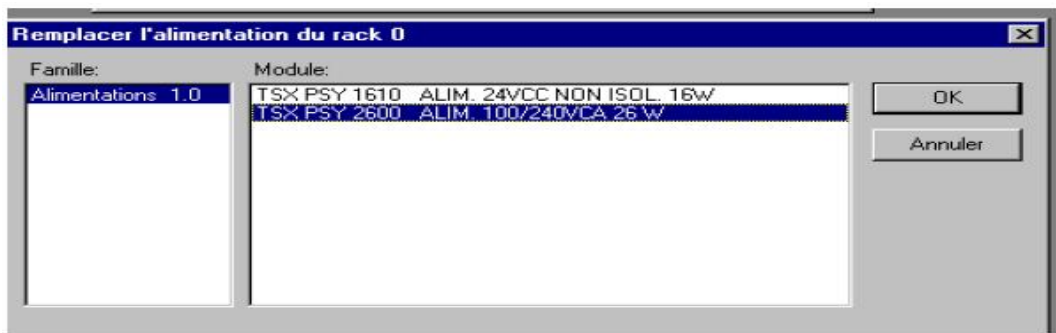


Figure IV.19 : Configuration de l'alimentation.

L'ajout des modules sur le rack (figure IV.20) de la station automate s'effectue en cliquant via la souris sur sa position dans le rack.

Les modules sont structurés en 9 familles

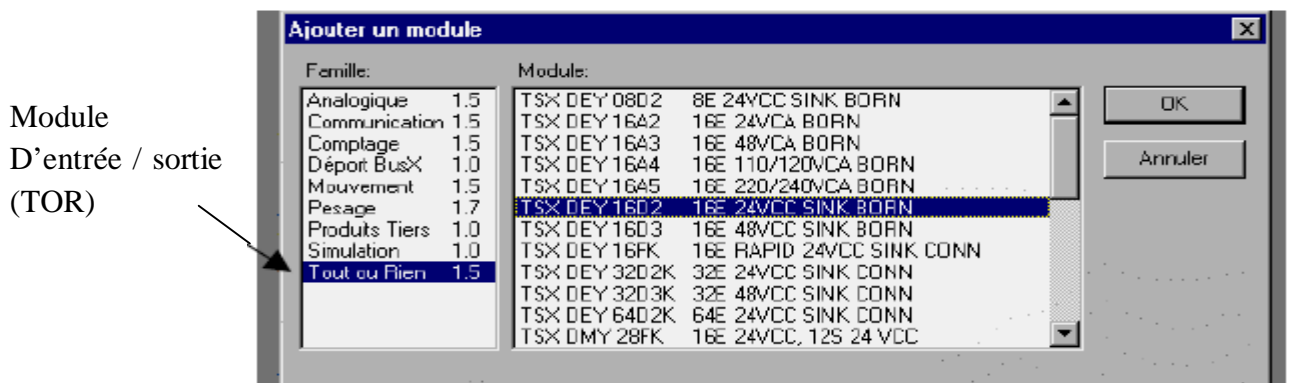
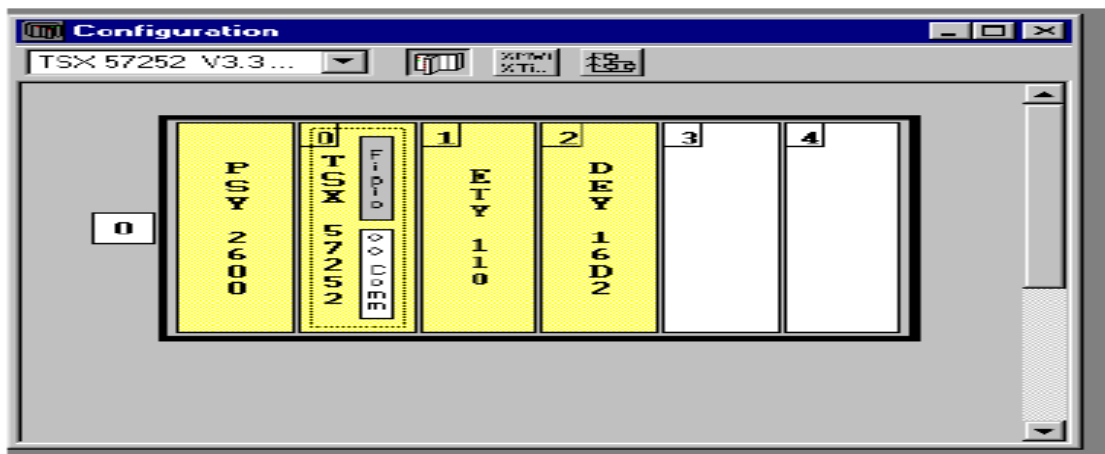


Figure IV.20: Ajout des modules sur le rack .

La configuration générale du rack de la station automate se présente ainsi (figure IV.21):



**Figure IV.21:** Configuration de la station automate .

#### IV.6.5 Mode d'adressage des modules

Les adresses des modules présent sur le rack de la station automate sont définis ainsi, [14] :

- 0 :TSX 57252,
- 1 :ETX 110,
- 2 :DEY 16D2.

## IV.7 Solution par GRAFCET

### IV.7.1 Hiérarchie des GRAFCET

Le GRAFCET complet définit le comportement du système réalisé. Il comprend :

- Le GRAFCET de Sûreté (GS) maître par rapport aux deux autres,
- Le GRAFCET de Conduite et Initialisation (GCI),
- 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, figure (IV.22).

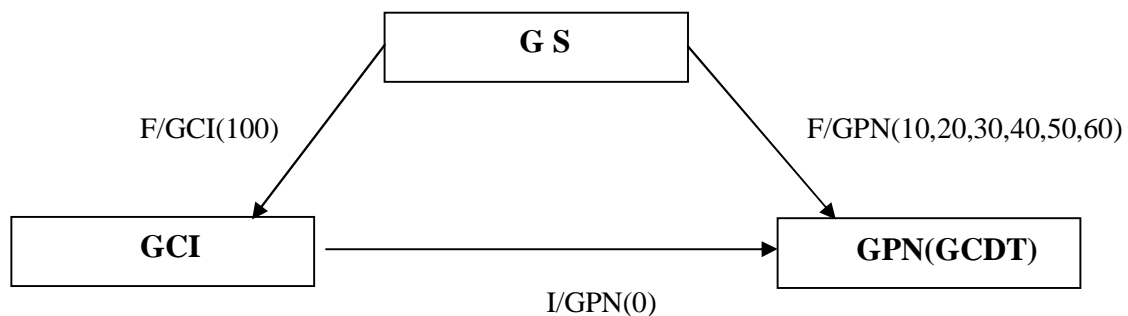


Figure IV.22 : La hiérarchie des GRAFCET.

### IV.7.2 GRAFCET de sécurité

La figure (IV.23) illustre le GRAFCET de sécurité.

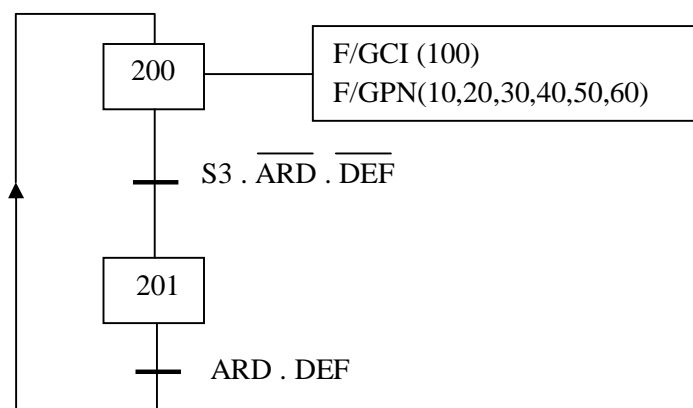


Figure IV.23 : GRAFCET de sécurité.

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 maintient 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, 60). Ce GRAFCET est réceptif à une consigne d'arrêt de sécurité (arrêt d'urgence: AU) et ce depuis tout les états.

IV.7.3 GRAFCET de Conduite et Initialisation

Il permet d'assurer la bonne marche du système automatisé en intégrant les dispositions pour le bas fonctionnement du système automatisé " briqueterie ", fig (IV.24).

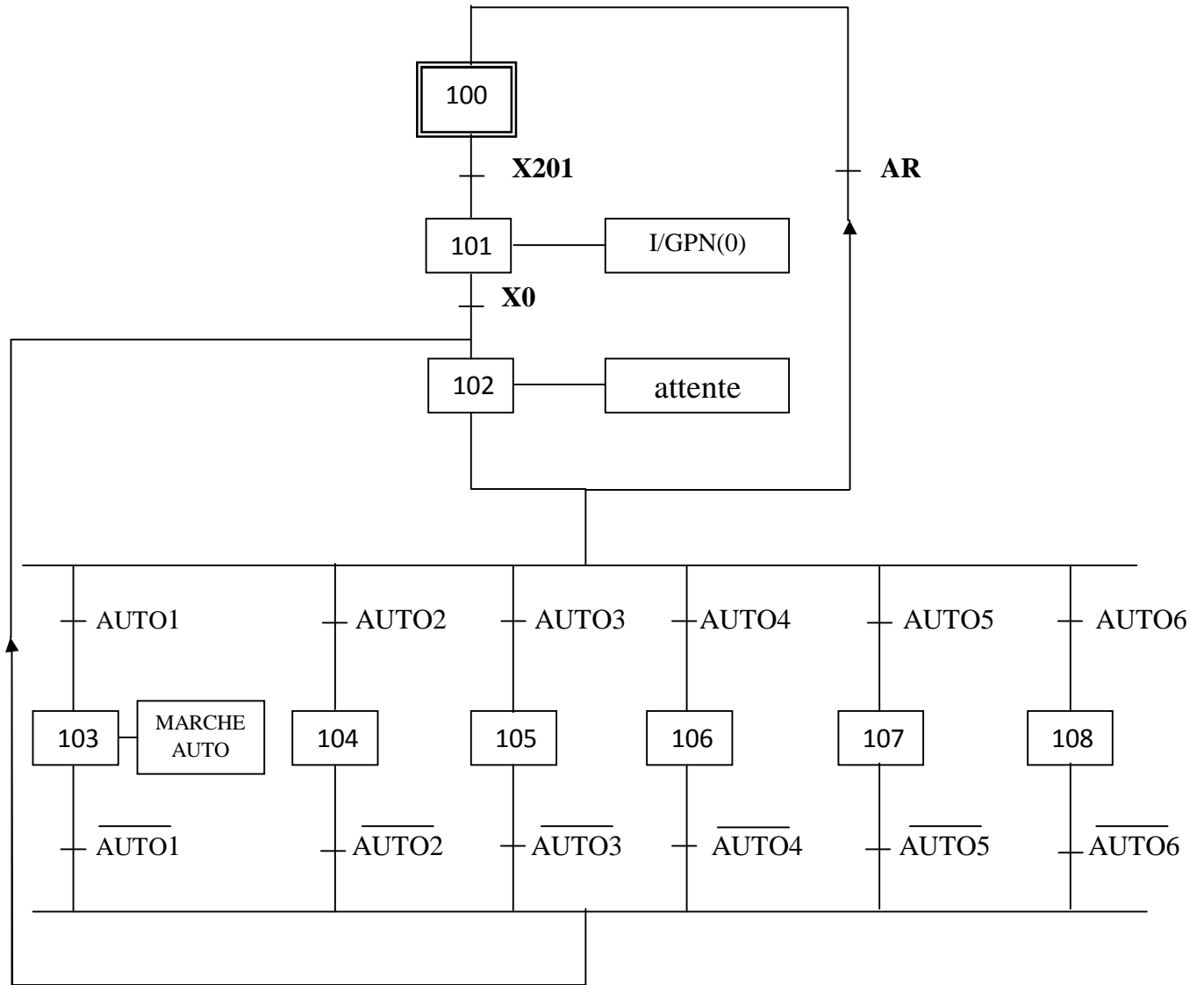


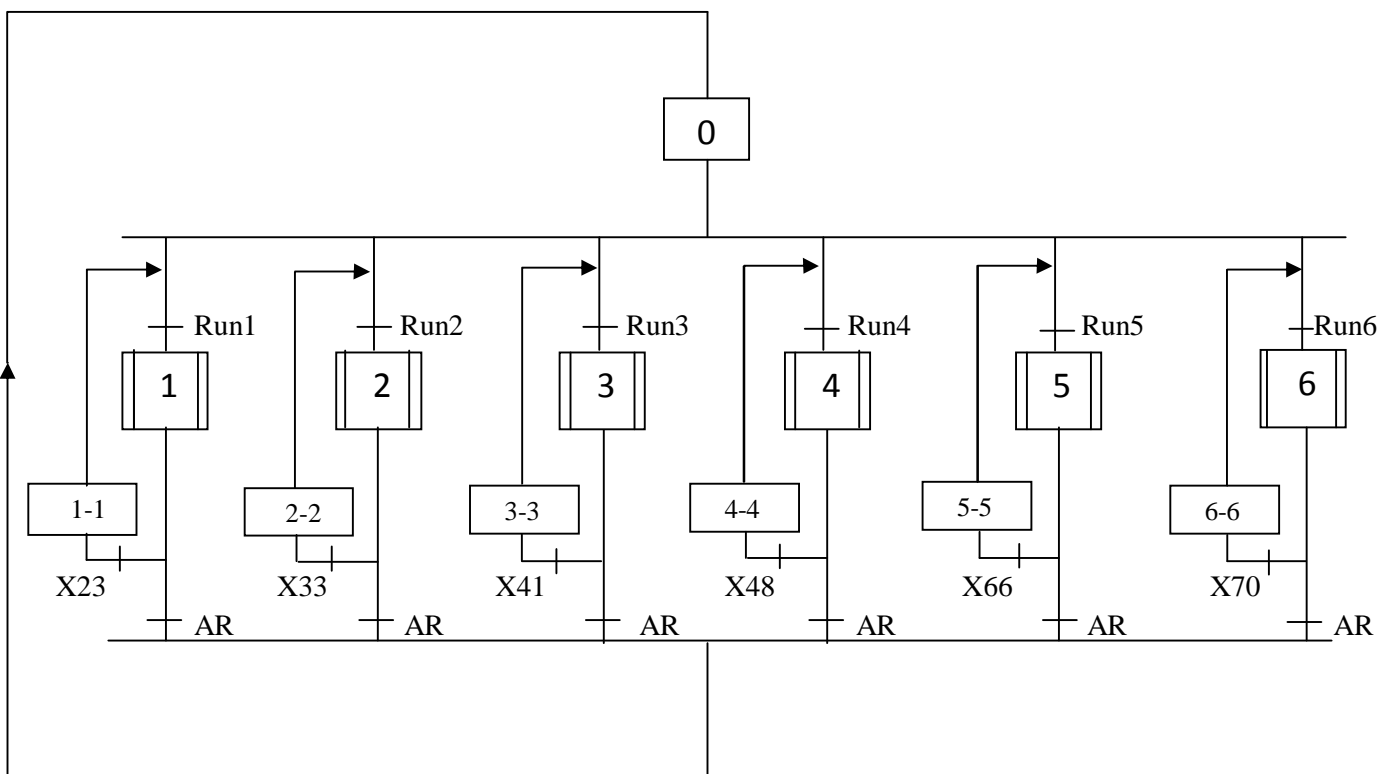
Figure IV.24 : GRAFCET de Conduit et Initialisation.

**IV.7.4 GRAFCET de coordination des tâches**

Nôtre système de production se compose de six tâches:

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

Le GRAFCET de coordination des tâches est donné par la figure (IV.25 ).



**Figure IV.25 :** GRAFCET de coordination de tâche.

IV.7.5 GRAFCET de production normale (GPN)

Il comprend le GRAFCET des différents Tâche de chaîne de production .

IV.7.5.1 Tâche de préparation

La figure (IV.26), représente la tâche de préparation avec la solution en cas d' arrêt de puis toutes Les étapes (Run1).

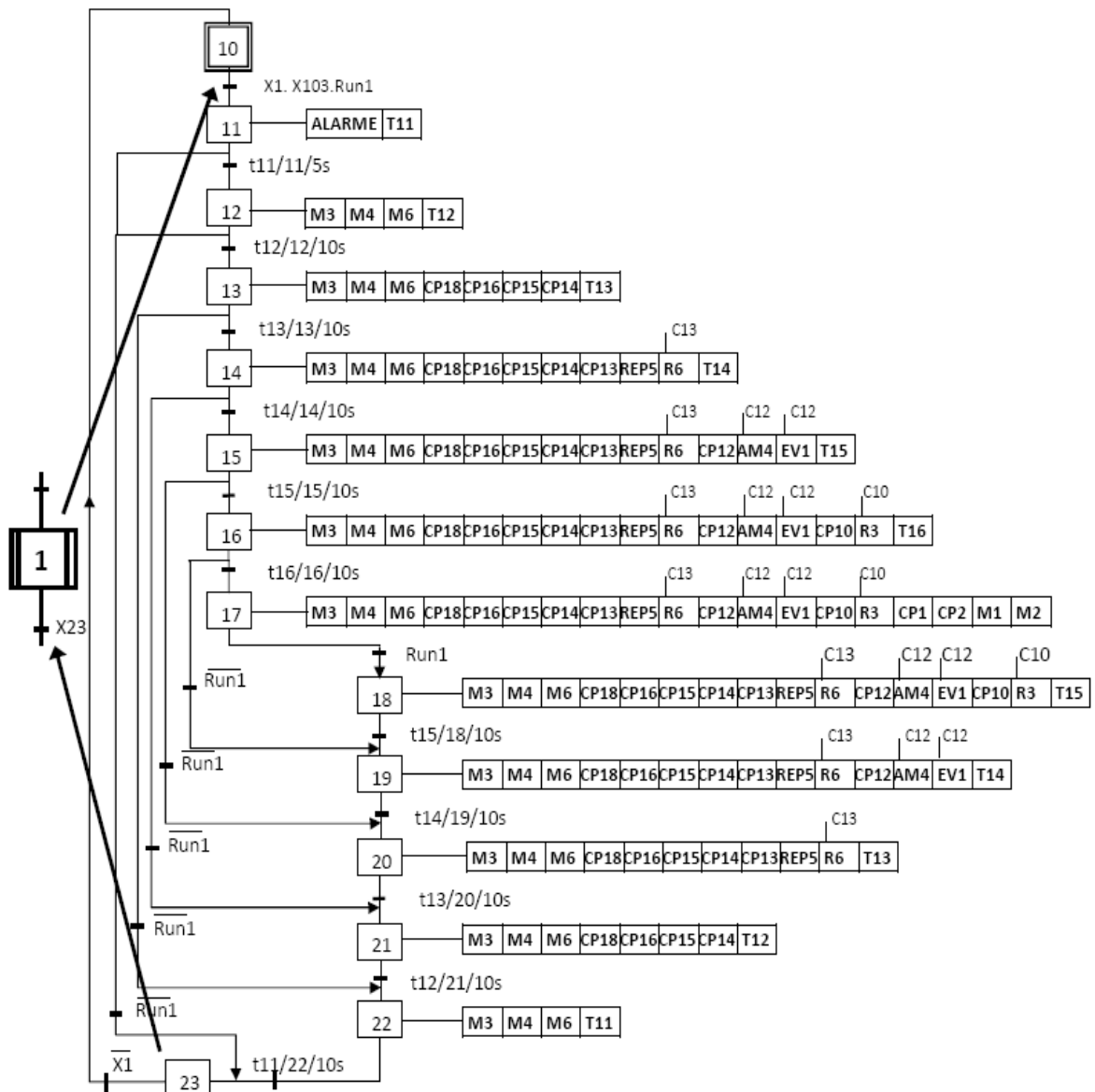


Figure IV.26 : GRAFCET de la tâche de préparation .

IV.7.5.2 Tâche de fabrication

Le GRAFCET de la tâche de fabrication est illustré par la figure (IV.27).

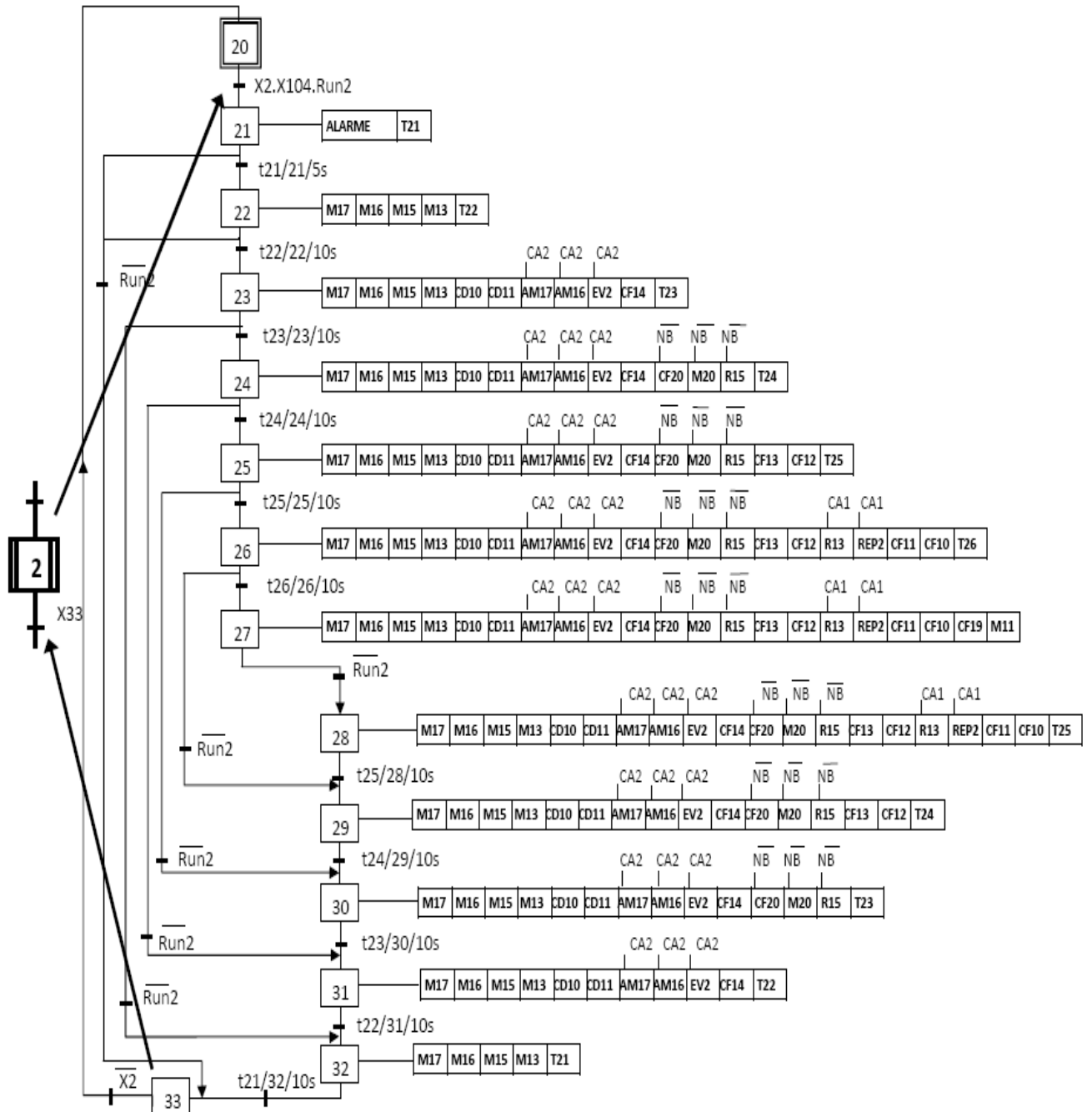


Figure IV.27 : GRAFCET de la tâche de fabrication .

IV.7.5.3 Tâche de pré coupeur

Les étapes de premier coupe de la brique sont associe a la tâche 3, figure (IV.28 ).

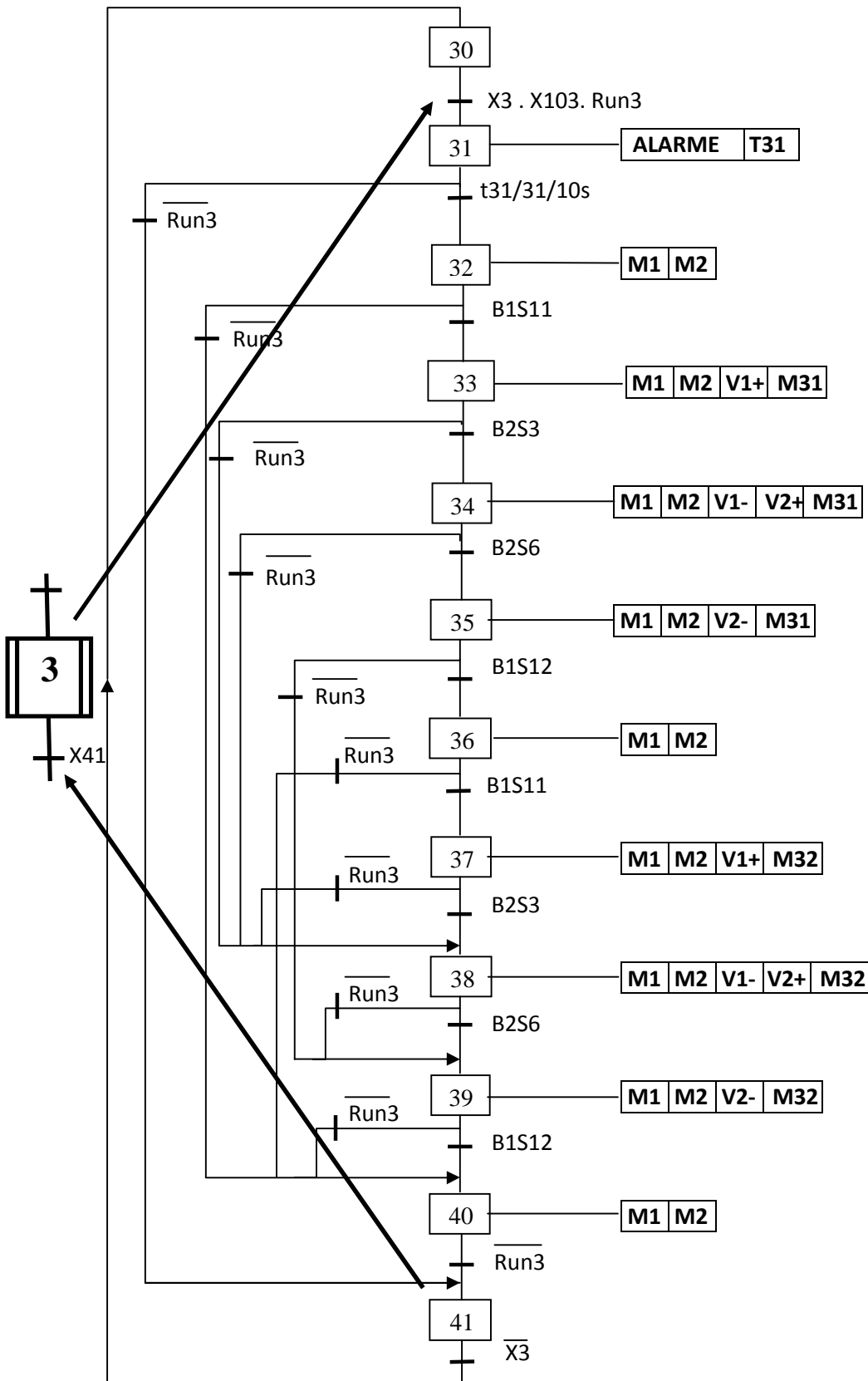


Figure IV.28 : GRAFCET de la tâche de pré coupeur.

IV.7.5.4 Tâche de coupe finale

Les étapes de coupe finale de la brique sont donné par la figure (IV.29 ), ci-dessous.

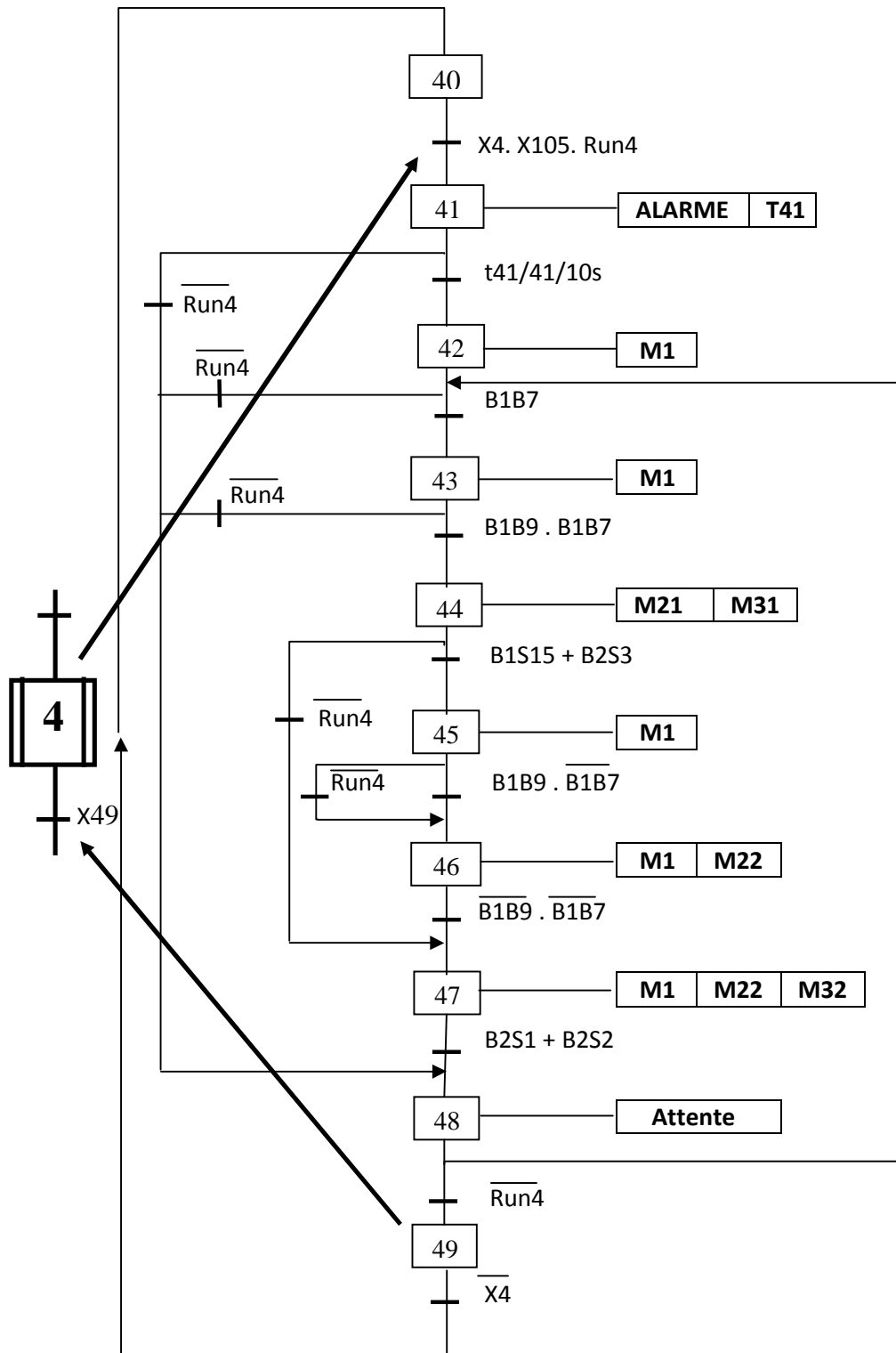


Figure IV.29 : GRAFCET de la tâche de coupe finale.

IV.7.5.5 Tâche de séchage

La solution proposé par GRAFCET de la tâche de séchage est schématisé par la figure (IV.30 ).

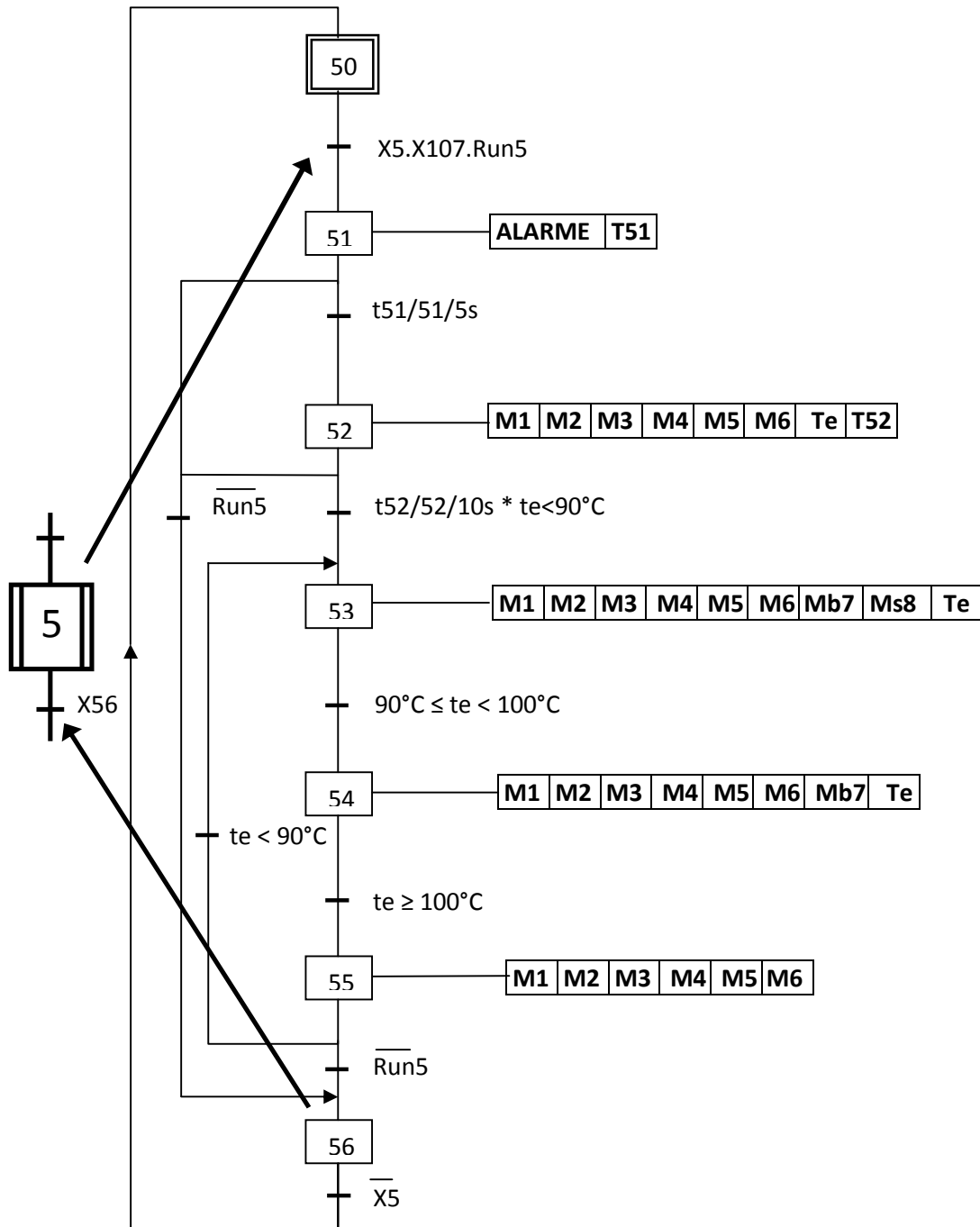


Figure IV.30 : GRAFCET de la tâche de séchage.

IV.7.5.6 Tâche cuisson de brique

Les étapes de cuisson de la brique sont donné par la figure (IV.31 ), ci-dessous.

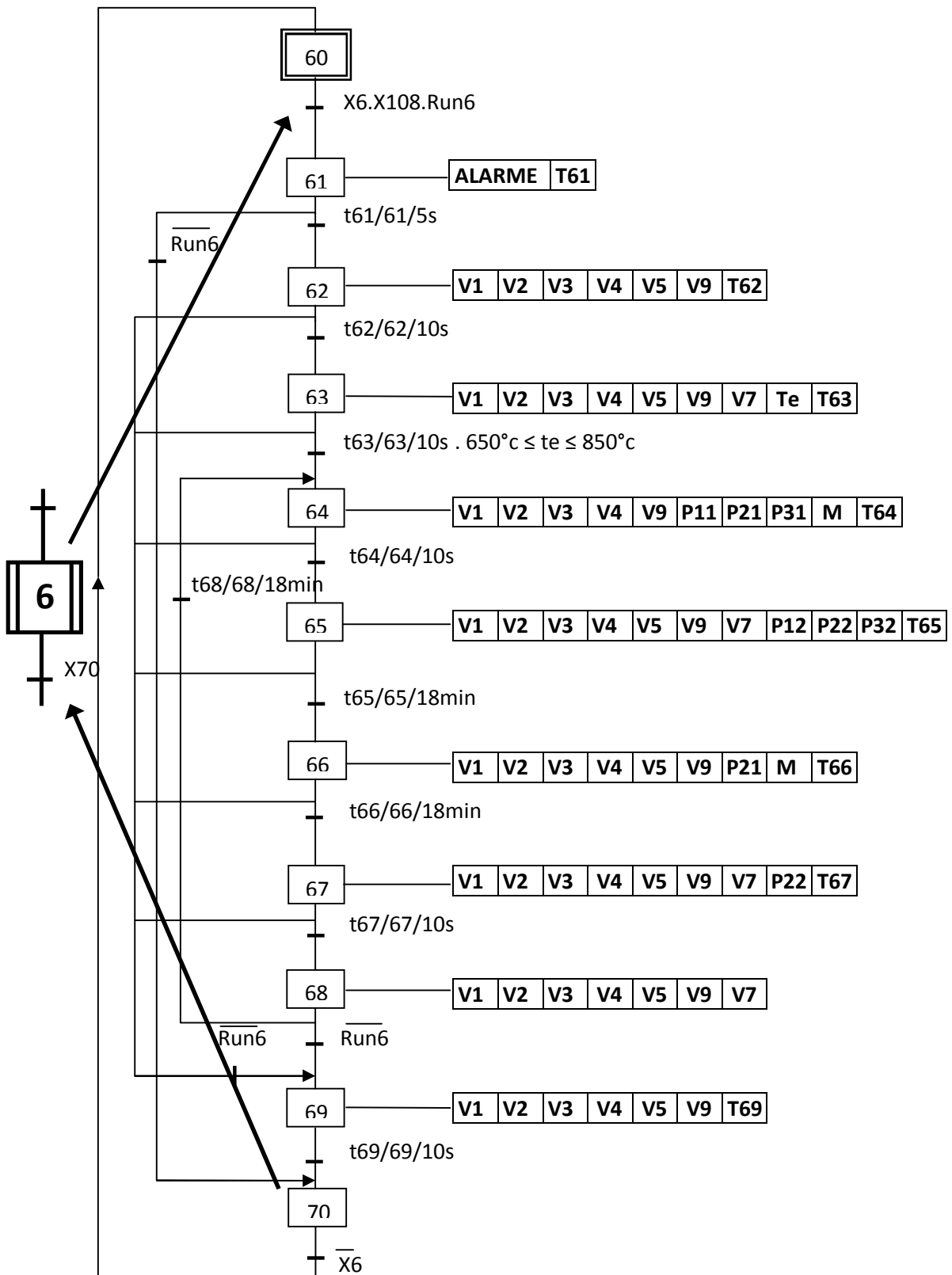


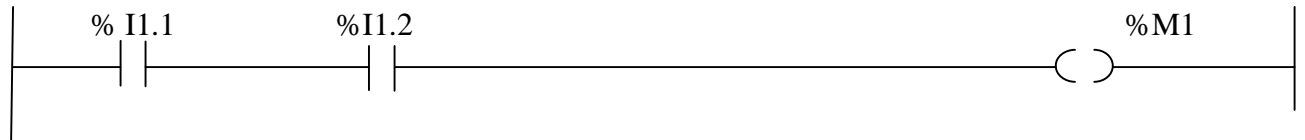
Figure IV.31 : GRAFCET de la tâche de cuisson.

### IV.8 Conversion écriture Grafcet en langage LD

Le langage LD(ladder) (schéma à relais) peut être utilisé pour décrire une condition associée à une transition. Le schéma est alors composé d'un seul échelon contenant un seul relais dont la valeur représente la valeur de la condition. On a choisi le tâche de préparation, comme exemple pour la programmation.

#### IV.8.1 Programme de GRAFCET de sécurité

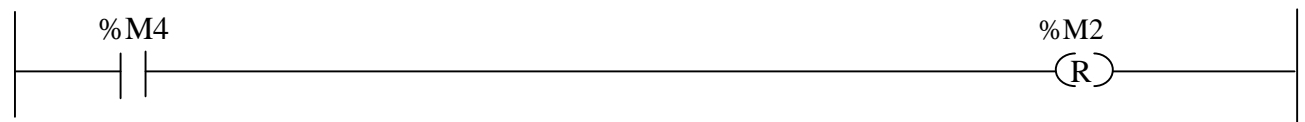
Transition d' étape 200



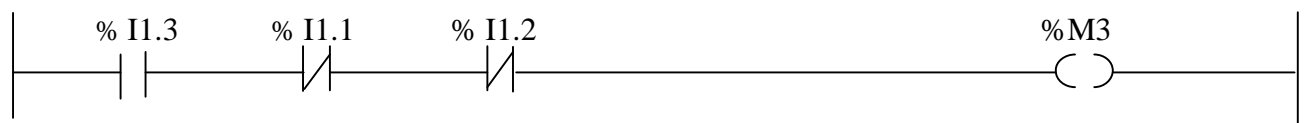
L' activation d' étape 200



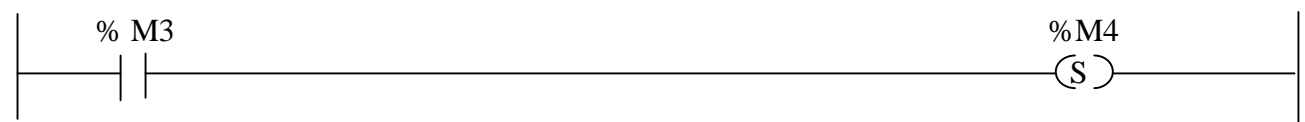
désactivation d' étape 200



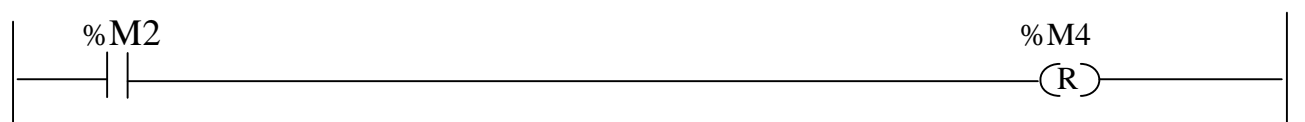
Transition d' étape 201



L' activation d' étape 201

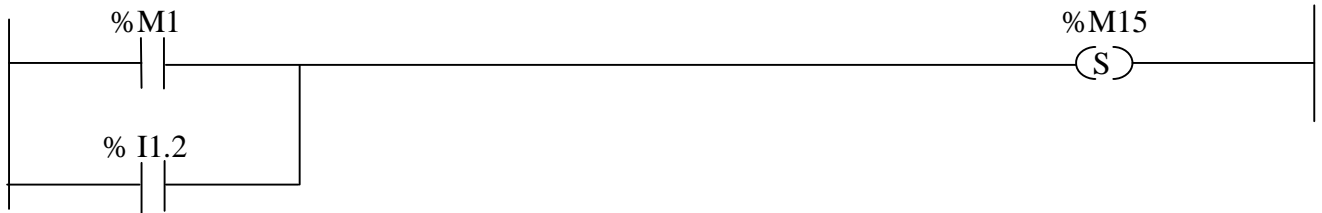


désactivation d' étape 201

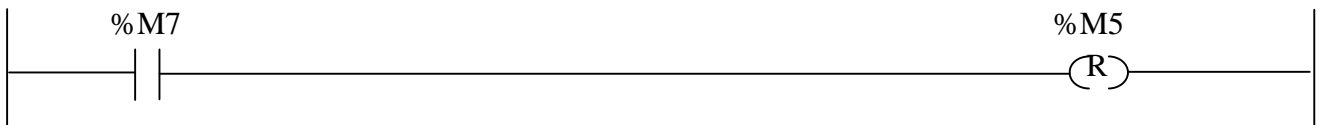


IV.8.2 Programme de GRAFCET de conduite et initialisation

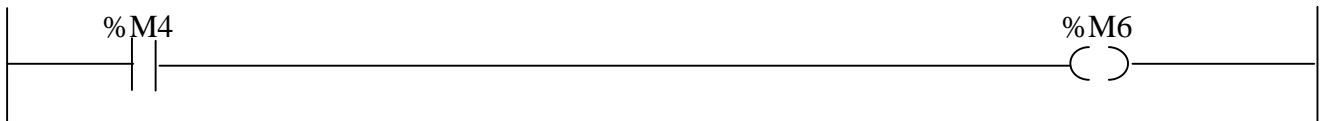
L'activation d'étape 100



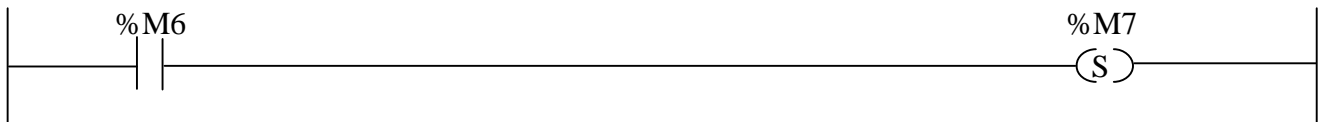
désactivation d'étape 100



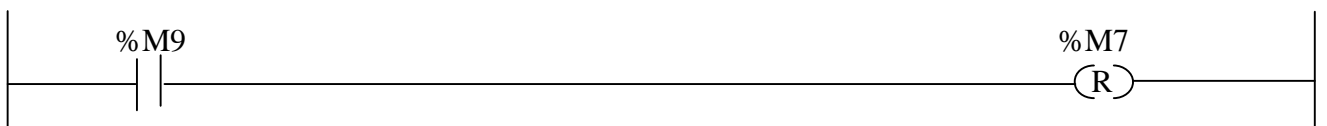
Transition d'étape 101



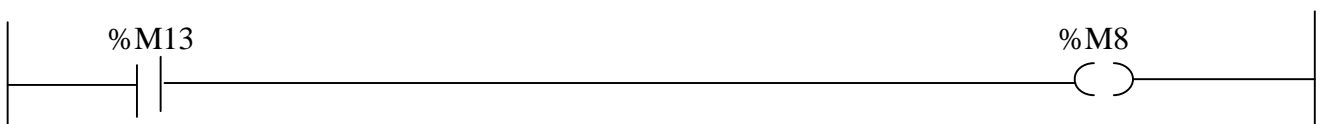
L'activation d'étape 101



désactivation d'étape 101



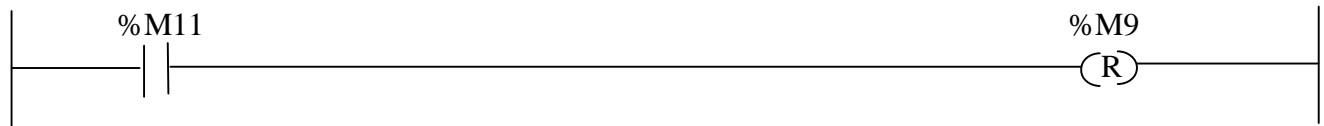
Transition d'étape 101



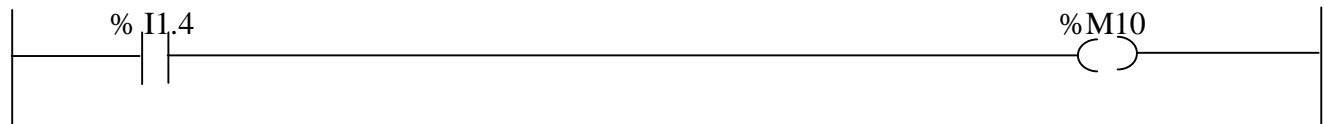
L'activation d'étape 101



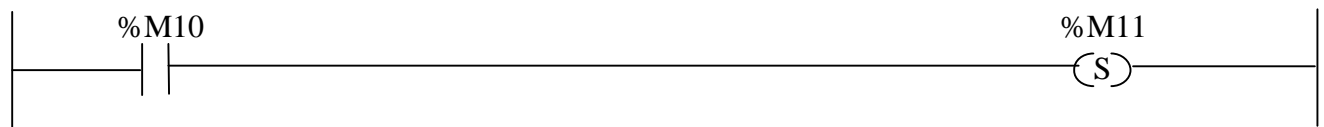
désactivation d'étape 102



Transition d' étape 103



L' activation d' étape 103



désactivation d'étape 103

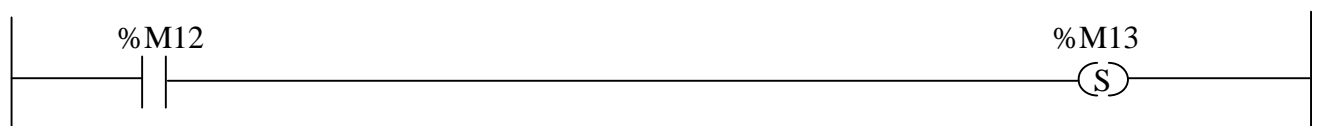


### IV.8.3 Programme de GRAFCET de coordination de tâche préparation

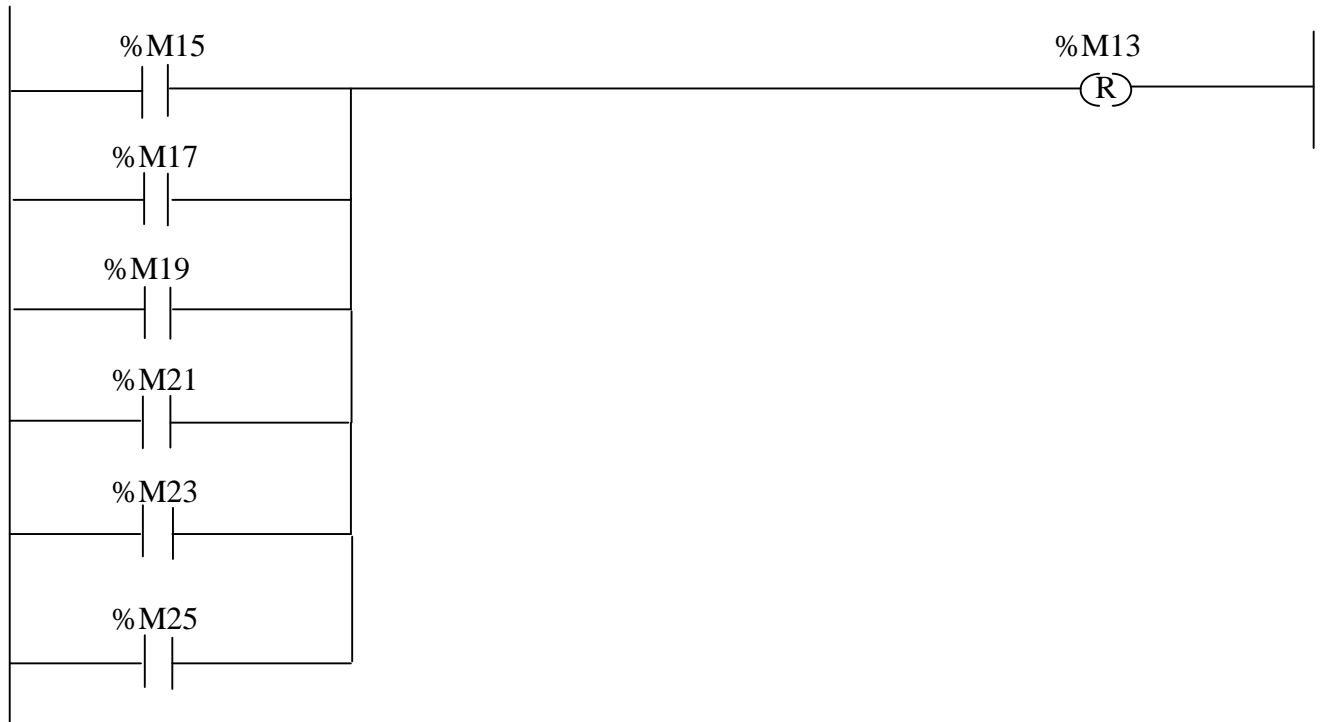
Transition d' étape 0



L' activation d' étape 0



désactivation d'étape 0



Transition tâche 1



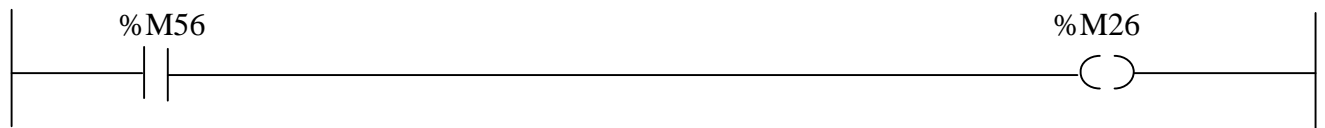
L'activation de tâche 1



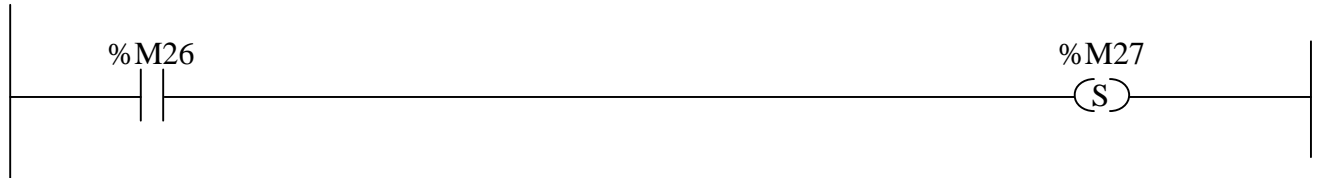
désactivation de tâche 1



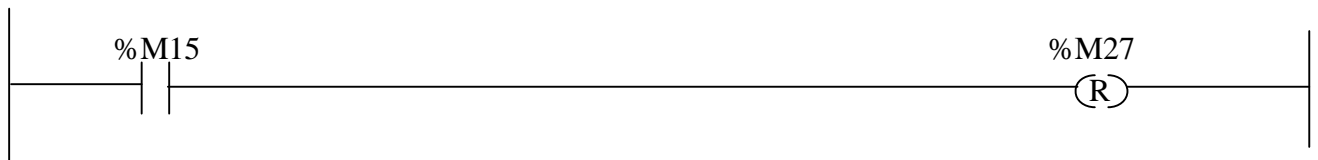
Transition d'étape 1-1



L'activation d'étape 1-1



désactivation d'étape 1-1



#### IV.8.4 Programme de tâche de préparation

Transition d'étape 11



L'activation d'étape 11



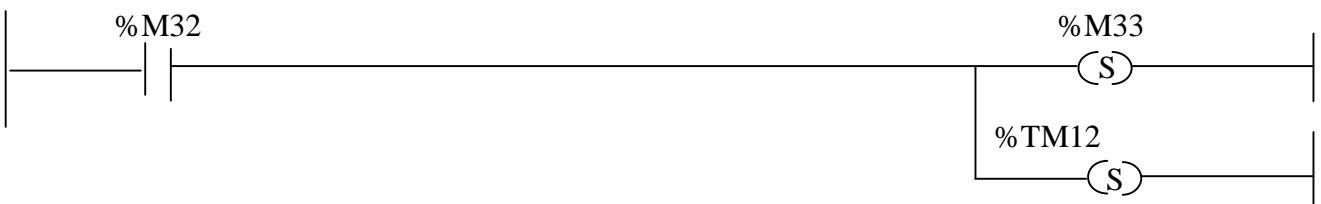
désactivation d'étape 11



Transition d' étape 12



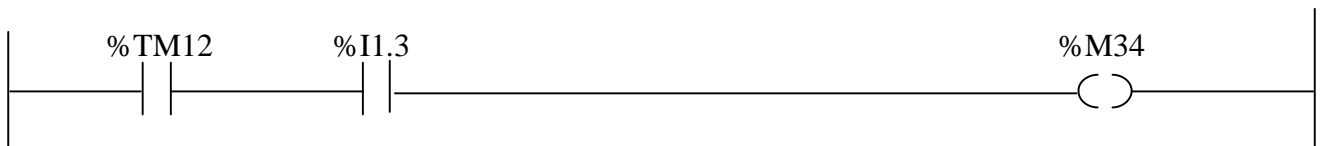
L'activation d' étape 12



désactivation d' étape 12



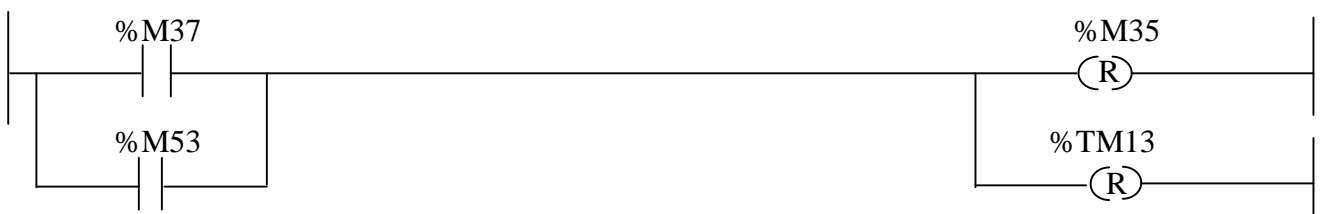
Transition d' étape 13



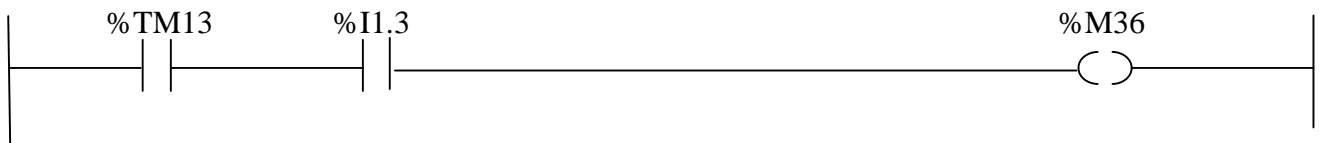
L' activation d' étape 13



désactivation d' étape 13



Transition d' étape 14



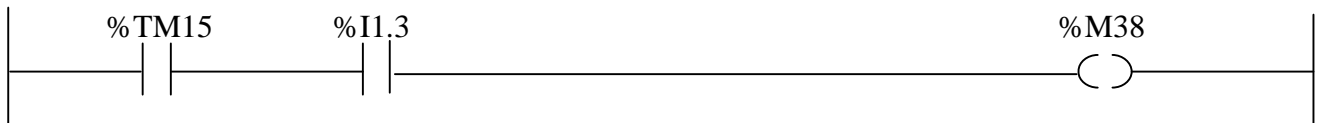
L' activation d' étape 14



désactivation d' étape 14



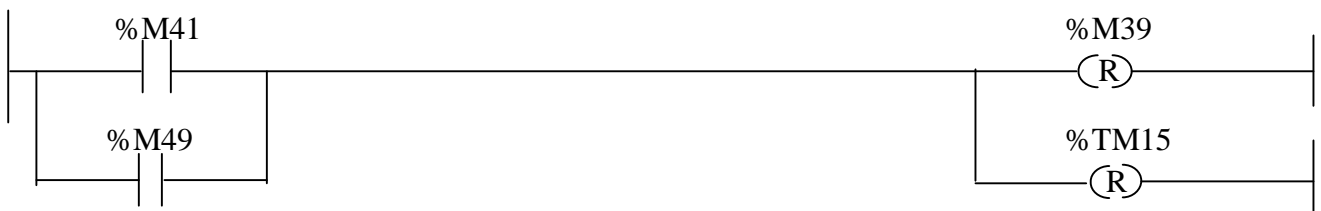
Transition d' étape 15



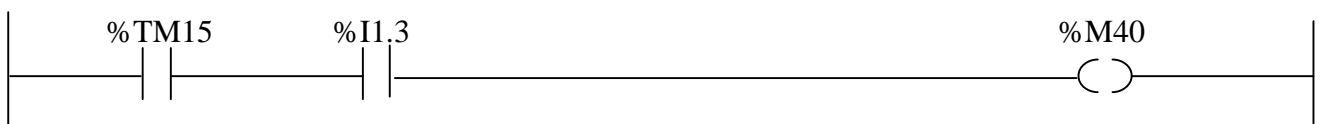
L' activation d' étape 15



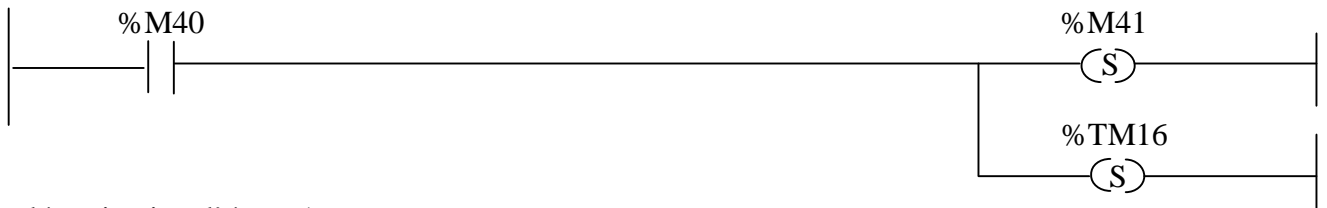
désactivation d' étape 15



Transition d' étape 16



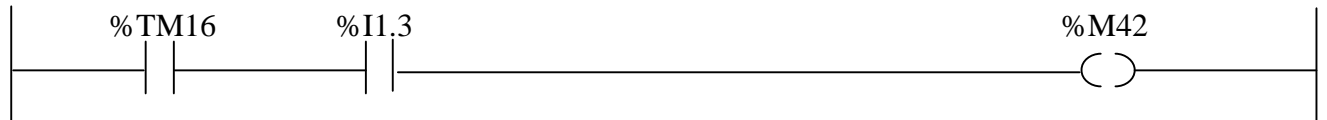
L'activation d'étape 16



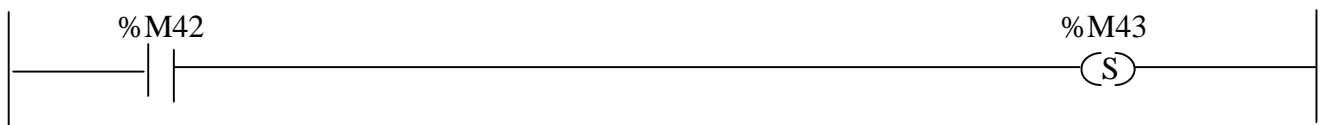
désactivation d'étape 16



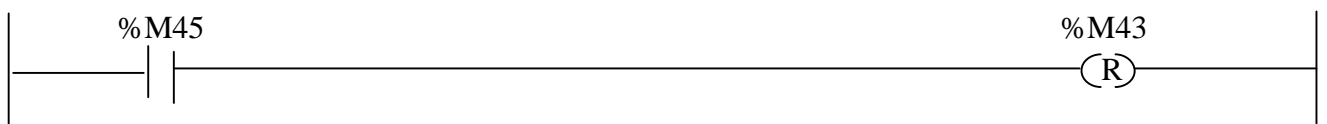
Transition d'étape 17



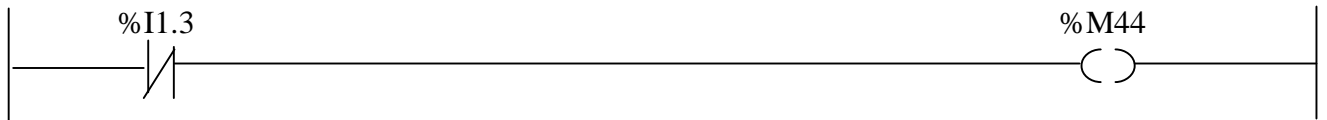
L'activation d'étape 16



désactivation d'étape 17



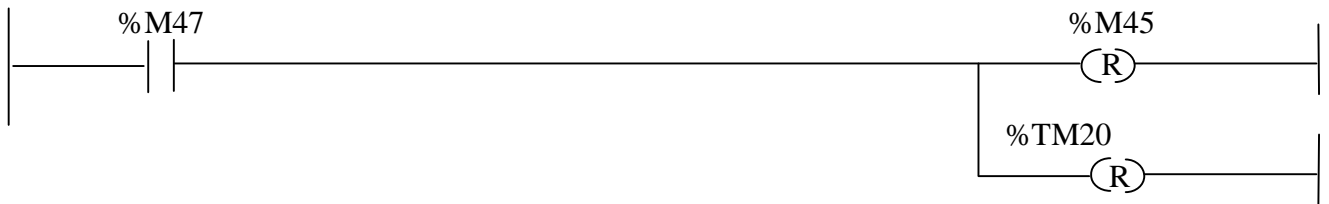
Transition d'étape 18



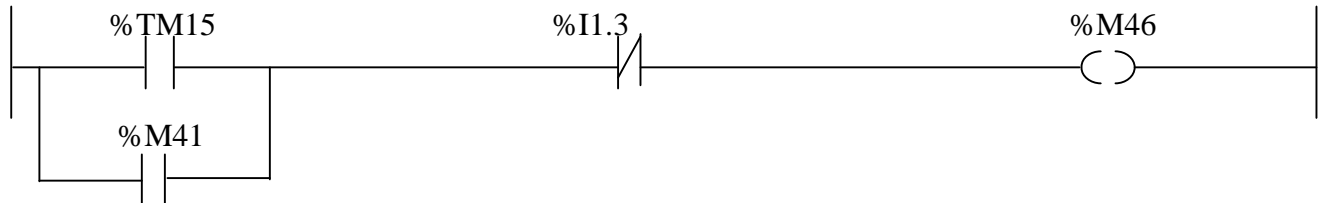
L'activation d'étape 18



désactivation d'étape 18



Transition d'étape 19



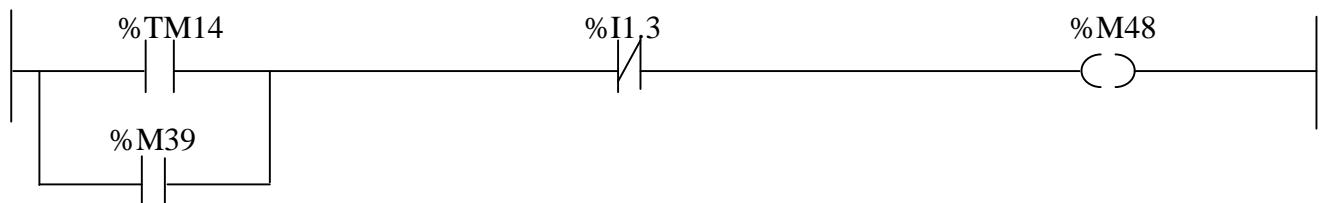
L'activation d'étape 19



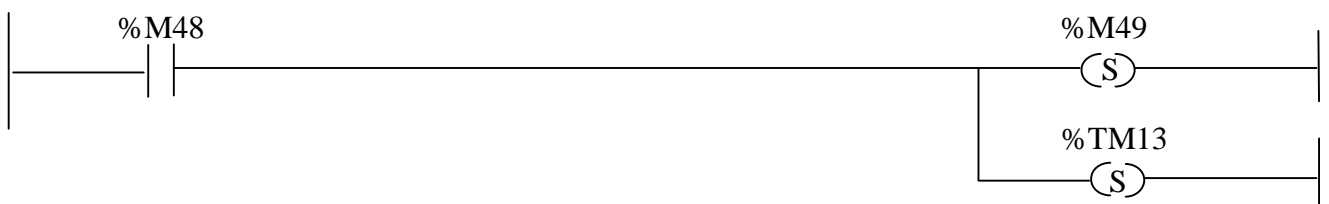
désactivation d'étape 19



Transition d'étape 20



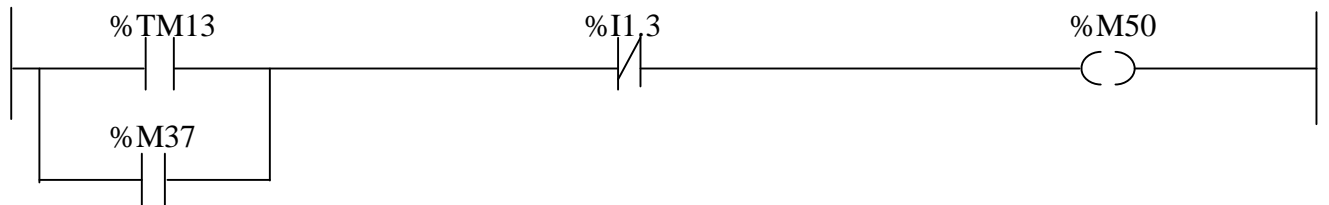
L'activation d'étap 20



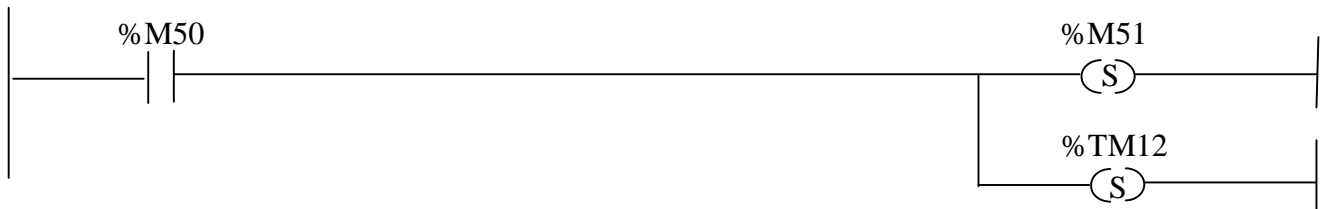
désactivation d'étape 20



Transition d'étape 21



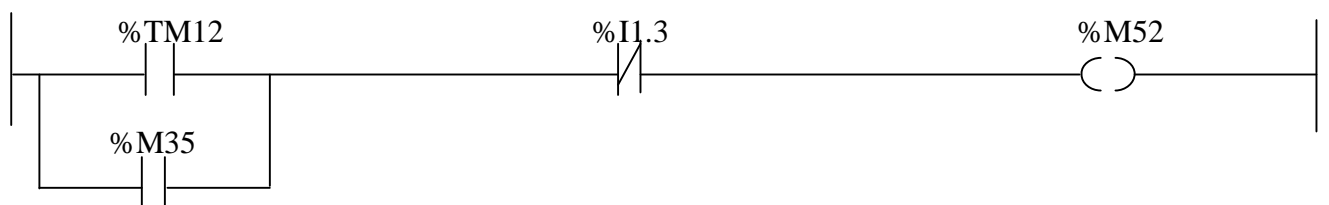
L'activation d'étap 21



désactivation d'étape 21



Transition d'étape 22



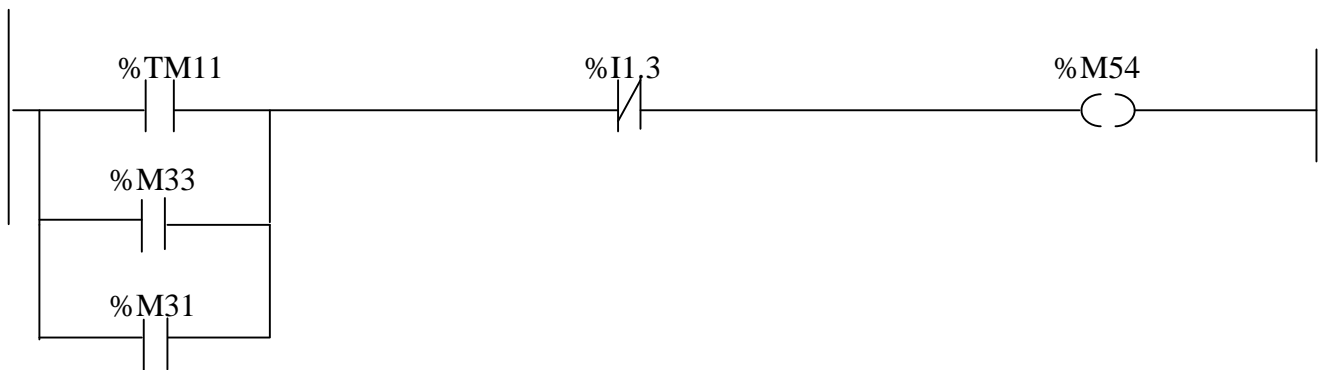
L'activation d'étap 22



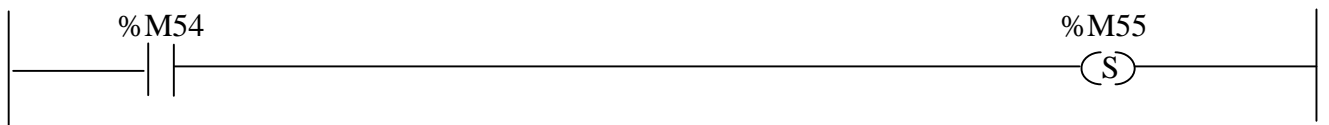
désactivation d'étape 22



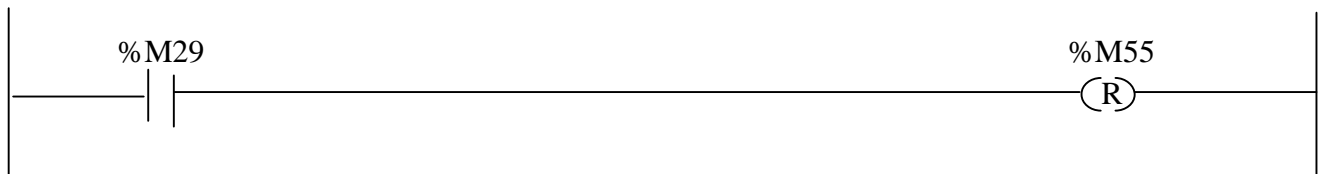
Transition d'étape 23



L'activation d'étape 23



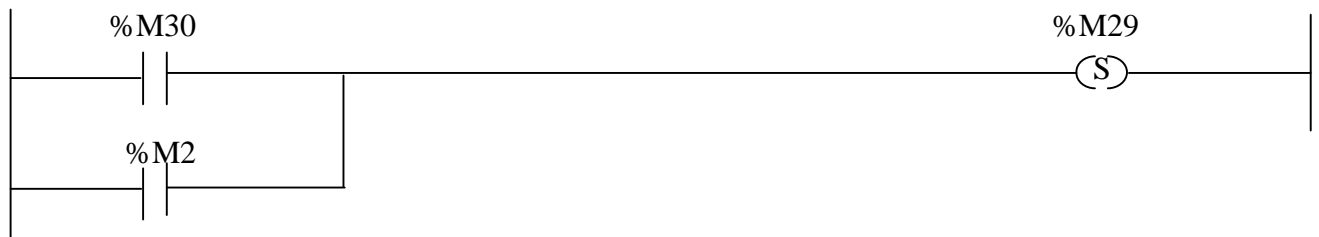
désactivation d'étape 23



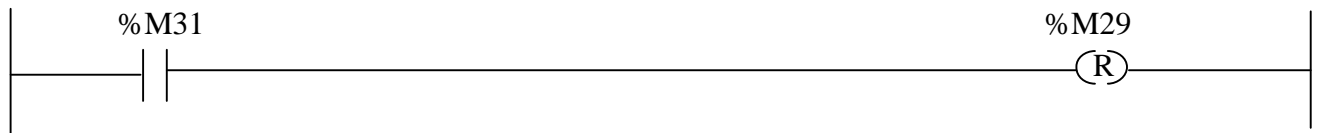
Transition d'étape 10



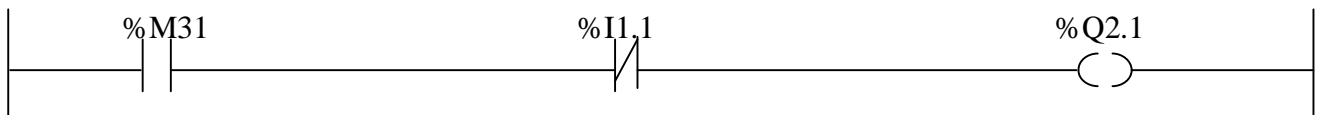
L'activation d'étap 10



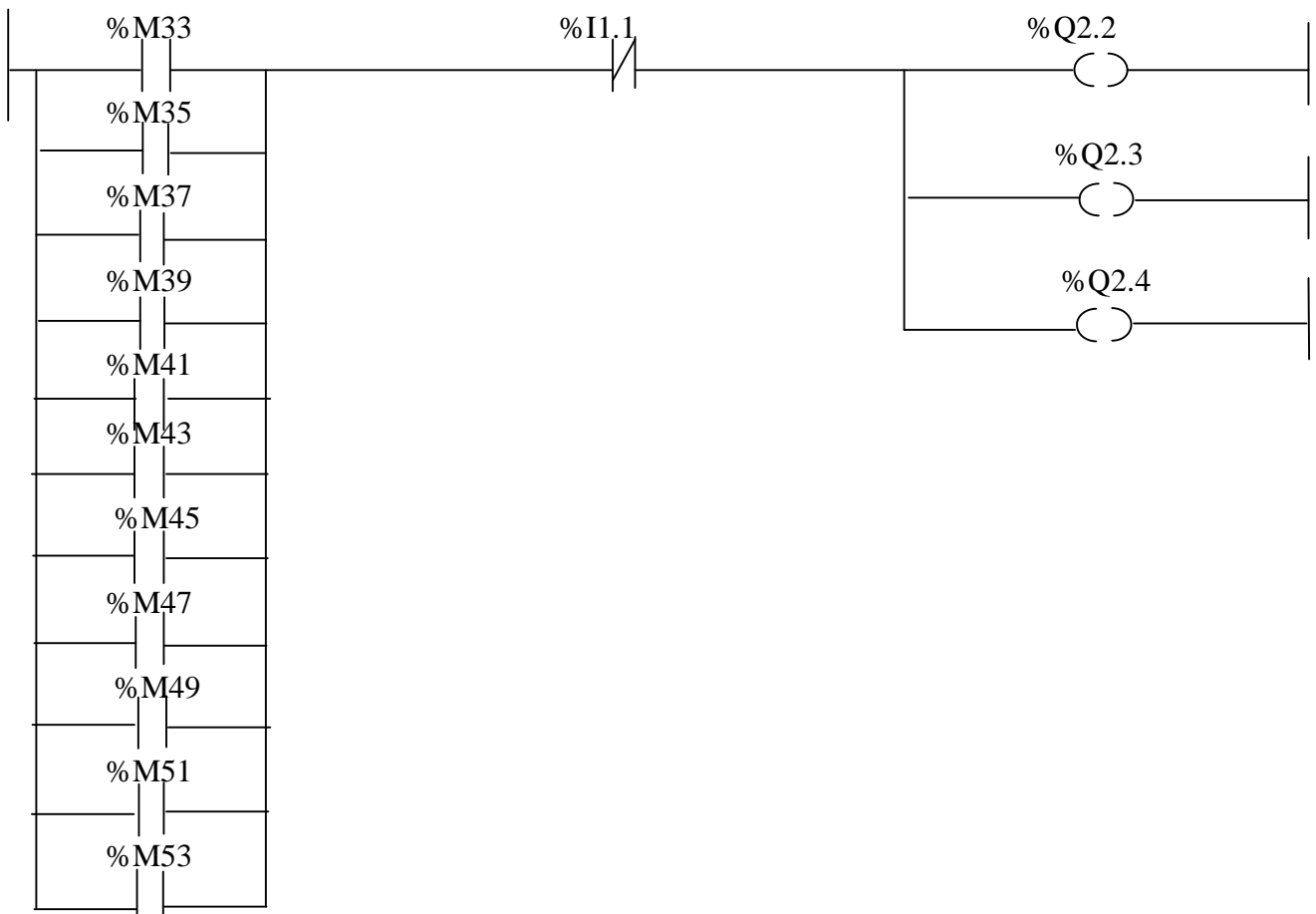
désactivation d'étape 10



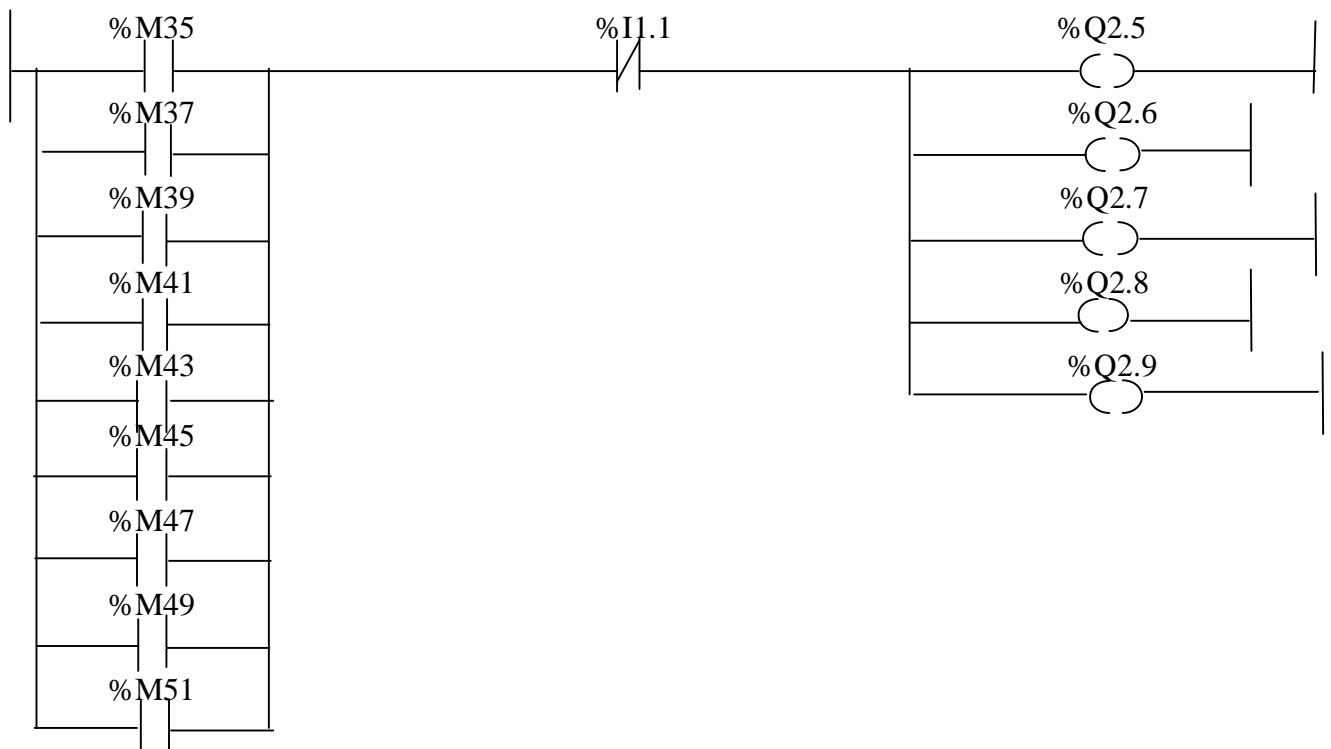
Alarme



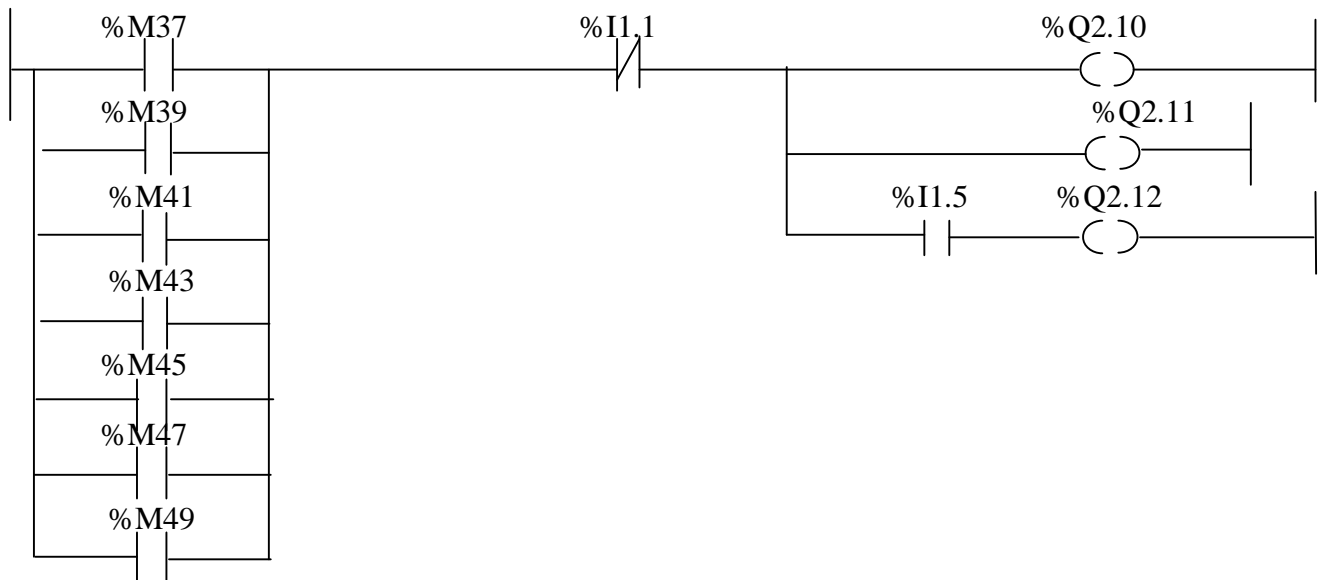
Action de M3, M4 et M6



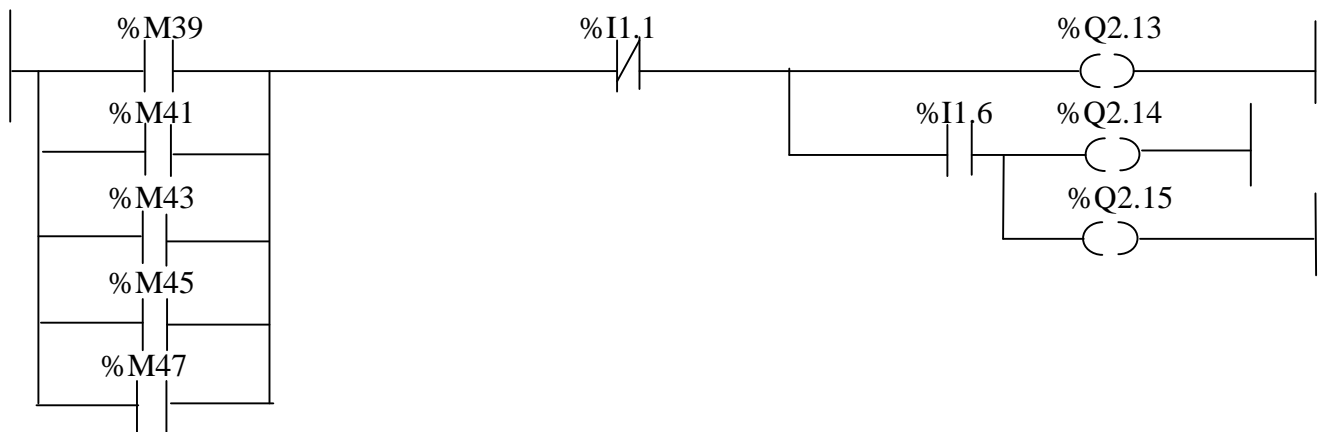
Action de CP18, CP17, CP16, CP15, CP14



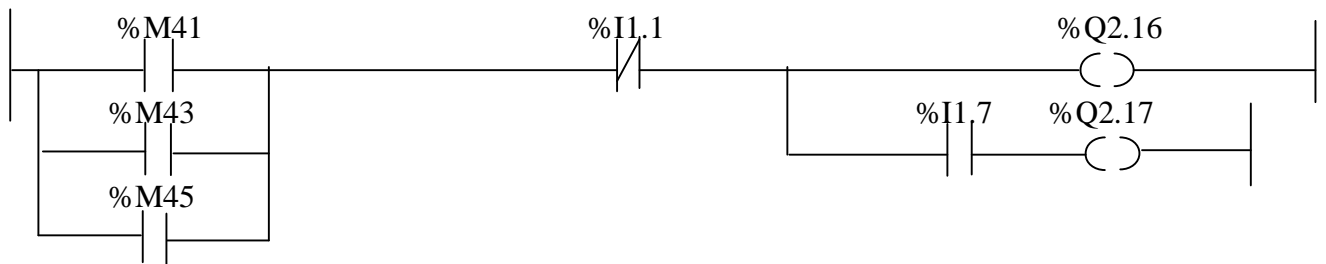
Action de CP13, Rep5, et R6



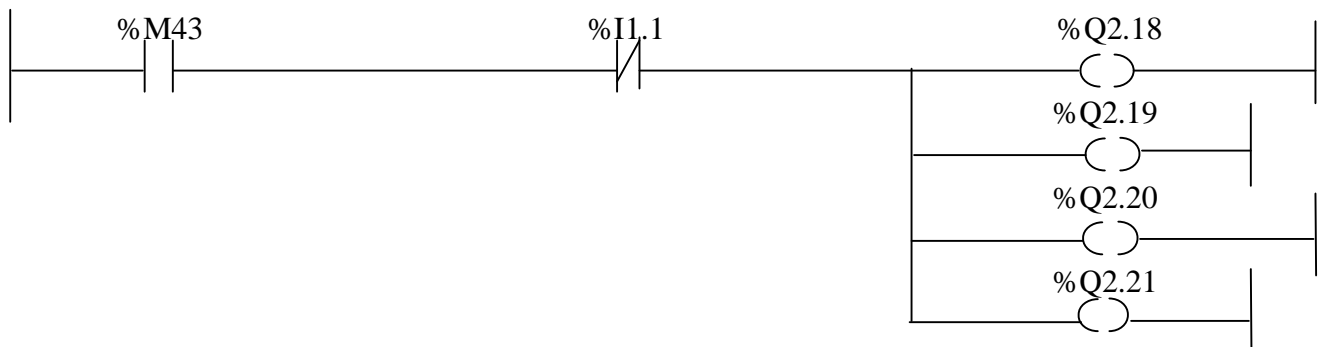
Action de CP12, AM4, EV1



Action de CP10 et R3



Action de CP1, CP2, M1 et M2



**Recensement des entrées et des sorties****- Les entrées**

N°	Mnémonique	Commentaire
1	%I1.1	Arrêt d'urgence
2	%I1.2	Défaut
3	%I1.3	Marche automatique
4	%I1.4	Initialisation de système
5	%I1.5	Capteur de présence d'argile C13
6	%I1.6	Capteur de présence d'argile C12
7	%I1.7	Capteur de présence d'argile C10

**- Les sorties**

N°	Mnémonique	Commentaire
1	%Q2.1	Alarme
2	%Q2.2	moteur M3
3	%Q2.3	moteur M4
4	%Q2.4	moteur M6
5	%Q2.5	convoyeur CP18
6	%Q2.6	convoyeur CP16
7	%Q2.7	convoyeur CP15
8	%Q2.8	convoyeur CP14
9	%Q2.9	convoyeur CP13
10	%Q2.10	racleur R6
11	%Q2.11	répartiteur Rep5
12	%Q2.12	convoyeur CP12
13	%Q2.13	Traction de embrayage AM4
14	%Q2.14	Ouverture de électrovanne d'eau EV1
15	%Q2.15	convoyeur CP10
16	%Q2.16	racleur R3
17	%Q2.17	convoyeur CP01
18	%Q2.18	convoyeur CP02
19	%Q2.19	moteur piocheur M01
20	%Q2.20	moteur piocheur M02

### **IV.1 Conclusion**

*Après avoir examiné et étudié, durant une courte durée au sein de l'entreprise de fabrication de la brique, le cycle de production, on est arrivé à décomposer se dernier en six tâches :*

- Tâche 1 : préparation de l'argile,*
- Tâche 2 : fabrication de la brique,*
- Tâche 3 : pré-coupeur,*
- Tâche 4 : coupeur,*
- Tâche 5 : séchage de la brique,*
- Tâche 6 : la cuisson.*

*Etant donné que les automates programmables sont devenus actuellement indispensables pour la commande des systèmes automatisés industriels, nous avons passé en revue les automates disponibles au niveau de l'entreprise ; puis on a développé une approche de programmation en langage contact de la tâche de préparation, tout en introduisant dans ce programme les consignes des GRAFCET sécurité, initialisation et coordination des tâches.*

## **Conclusion générale**

*Avant tout nous pouvons que cette expérience fut enrichissante à tous les niveaux ; elle nous a permis d'explorer un domaine incontournable de l'industrie, c'est l'automatisme industriel.*

*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.*

*Partant du fait que le GRAFCET est l'élément de base pour la résolution des systèmes automatisés, la deuxième partie a été utilisée pour la mise en évidence des éléments de bases du GRAFCET, les différentes structures, règles d'évolutions et GRAFCET hiérarchisé*

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

*Quant à la quatrième partie, elle a été consacrée, aux solutions GRAFCET proposées des différentes tâches tout en introduisant le principe des GRAFCET hiérarchisé. Enfin, une approche de programmation par automate programmable en utilisant le langage contact pour la tâche de préparation.*

*Ce mémoire conduit tout naturellement à envisager un certain nombre de prolongements :*

- Passage de la programmation par contacts aux autres langages pour plus de maîtrise de la programmation (automates programmable) ;*
- Conception de prototype de systèmes automatisés miniatures ;*
- Amélioration des résultats obtenus par programmation de toutes les tâches du cycle de production ;*
- Amélioration des connaissances théoriques par des stages de courte et moyenne durée dans les unités de productions commandées par automates programmable.*

# Annexes

TOR	: Tout ou – rien
%M	: Adresse de Mémentos
%I1.2	: Adresse d'entrée
%Q2.1	: Adresse de sortie
%TM1	: Adresse de Temporisateur

## Partie de fabrication

CD10,CD11	: Les Convoyeurs de récupération (déchet de brique)
CF14,CF13,CF12,CF11,CF10,CF19,CF20	: Les Convoyeurs
M20	: Moteur de piocheur
M16	: Moteur de Mouleuse
M17	: Moteur de Malaxeur
CA1,CA2	: Capteurs présence d'argile

## Partie de pré-coupeur

M1,M2	: Moteurs des rouleaux
M3	: Moteur de bras ( Monte et descente )
V1,V2	: Les vérins

## Partie de séchoir

M1,M2,M3,M4,M5,M6	: Moteurs de extraction
Mb7	: Moteur de bruleur
Ms8	: Cerveau moteur (récupération four )

## Partie de four

V1,V2,V3,V4,V5,V6,V7,V9	: Les ventilateur de four
P1,P2,P3	: Les portes
M	: Moteur marche des wagons
P11,P21,P31	: Les portes ouverts
P12,P22,P32	: Les portes fermées

# ***Bibliographie***

**[1] ATLAOUI NOUREDDINE . FRAHTIA RACHID**

« automatisation d'une petite unité de production par la résolution grafcet », mémoire d'ingénieur, université de M'sila 2008.

**[2] BENARIBAMMAR. BELMILOUD ABD ELAZIZ**

« Étude de réalisation d'un API a bas d'un microcontrôleur » Mémoire d'ingénieur, université de M'sila 2006.

**[3] 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.

**[4] C.BOURBOUNNE. J.COJEAN**

« Les systèmes automatisés de la connaissance à la conception », Tome 1, les éditions Foucher, 1984.

**[5] HAMDY Hocine**

« Automatismes logiques », Les éditions de l'université Mentouri Constantine, 1999.

**[6] LES AUTOMATE PROGRAMABLE**

« CEPADUES EDITIONS ».

**[7] MICHEL BERTRAND**

«Automate programmable industrielle » Docteur-Ingénieur. École Nationale Supérieure d'Arts et Métiers ENSAM, Centre d'Enseignement et de Recherche de Lille.

**[8] D. Dubois, 1999**

« Automate programmable industrielle ». « Le Grafcet et Les systèmes automatisés ».

**[9] MEZIANI Adel, OULBA Rachid**

« Automatisation de la commande de régénération du sécheur à argan, Mémoire d'ingénieur», université Annaba, juin 1999.

**[10] TIAIBA Oussama .HALILOU Saif eddine**

« Etude simulation des (API) en vue de leurs application dans le domaine de démarrage triphasés avec logiciel en langage GRAFCET», 2008.

**[11] TECHNIQUES DE L'INGENIEUR**

«Les systèmes automatise ».

« Langage de programmation pour API norme IEC 1131-3 ».

**[12] Mourad BRAHIMI. Fathi ALAHOUM**

«Conception et réalisation d'un logiciel de simulation conjointe : partie opérative d'un procédé industriel». Mémoire d'ingénieur, 2005.

**[13] PHILIPPE LE BRUN FLORENCE**

« Programmation des automates » Lycée Louis ARMAND, 173 Bd de Strasbourg, 94736 NOGENT sur Marne.

**[14] DOCUMENTATION D'USINE DE BRIQUE**

« PDF de PL7 PRO », guide de l'utilisateur PL7, « Schneider électrique ».

**MEMOIRE DE FIN D'ETUDES EN VUE DE L'OBTENTION DE DIPÔLME**

**D'INGENIEUR D'ETAT EN GENIE ELECTROTECHNIQUE**

**OPTION : ÉLECTROMECHANIQUE**

**PROPOSE ET DIRIGE PAR : Mr. ABDOU ABDEALHAK**

**PRESENTE PAR : Mr. LASSELAT RACHID**

**Mr. BENIA ALI**

**THÈME :**

**RÉSOLUTION PAR GRAFCET ÉVOLUÉ**

**D'UN SYSTÈME AUTOMATISÉ INDUSTRIEL**

**« BRIQUETERIE D'EL KALAA BENI HAMMAD - M'SILA »**

**RESUME :**

*Pour concevoir, réaliser et exploiter un automatisme, il est indispensable d'en d'écrire le comportement. Les outils ou langages qui permettent cette description peuvent, au choix, être littéraires, symboliques ou graphique. Il est important de connaître ces outils pour permettre une expression claire et concise des problèmes d'automatisation à résoudre*

*Dans ce mémoire, après étude, sur site, du cycle de fonctionnement de la briqueterie « d'EL KALAA BENI HAMMAD », on proposées des solutions par GRAFCET évolué des différentes tâches qui compose le cycle de production, et comme, pour toute entreprise de production, l'utilisation des API est devenue incontournable et synonyme de compétitivité du produit fini sur le marché, on a donné une approche de programmation de la tâche de préparation.*

**MOTS CLES :**

- **Système Automatisé Industriel.**
- **GRAFCET et GRAFCET hiérarchisés.**
- **Tâches, cycle de production.**
- **Automate Programmable Industriel ( A P I ).**