

*Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université de Mouloud Mammeri, Tizi-Ouzou  
Faculté de Génie Electrique et d'Informatique  
Département Automatique*



## *Projet*

*De fin d'études  
en vue de l'obtention du diplôme d'ingénieur d'état  
En Automatique*

## *THEME*

*Etude de l'automatisation par API S7-300  
de la machine de découpage et poinçonnage  
des contre portes frigorifique à l'ENIEM.*

*Proposé par :*

*Mr LAHIANI*

*Dirigé par :*

*Mr DIRAMI*

*Présenté par :*

*Mr KAHIL OMAR*

*Mr FRAOUCENE REZKI*

*Promotion 2007-2008*

# Remerciements

Au terme de notre travail nous tenons à remercier tous ceux qui ont contribué de près ou de loin à sa réalisation.

Nous tenons à exprimer nos sincères remerciements à notre co-promoteur, Monsieur LIHIANI Mouhend Ouchaabane qui nous a proposé ce sujet et pour l'aide précieuse qu'il nous a apportée, ainsi que toute l'équipe de maintenance ; ALI, SAID, REZKI, D'ABELKACEM.

Nos remerciements les plus spéciaux à notre promoteur, Monsieur DIRAMI qui a accepté de nous encadrer et pour sa disponibilité et ses orientations pour mener à bien notre travail.

Nous remercions également Mr. AMAROUCHE ALI, Mr. HOUACINE et toute leur équipe pour leur accueil chaleureux et leur disponibilité toute au long de notre projet.

Nous sommes aussi reconnaissants à tous les enseignants qui ont contribué à la réussite de ce travail en particulier à Monsieur CHARIF M.

Nous remercions toutes les personnes qui nous ont encouragés pour aller au bout de notre travail, en particulier nos familles qui nous ont soutenus.

Nos derniers remerciements vont aux membres de jury qui nous feront l'honneur de juger notre travail.

# Dédicaces

Je dédie ce travail :

A la meilleure maman au monde pour son soutien et son amour ;

A mon cher père qui m'a offert toute son aide ;

A mes très chers frères : farid, fouad, djamel ;

A ma très chère sœur fazia ;

A toute la famille KAHIL ;

A tous mes amis ;

A mon binôme Rezki avec lequel j'ai partagé ce travail ;

A tous mes collègues d'études ;

A tous ceux qui me sont cher(e)s.

Omar.

<b>Introduction générale :</b> .....	1
--------------------------------------	---

## **Chapitre I : Description générale de la machine**

I .Introduction :	3
II . Description de la machine :	4
II .1. Partie opérative :	5
II .1.1. Structure mécanique :	5
II.1.2. Prés-actionneur :	6
II.1.2.1. Electrovanne :	6
II.1.2.2. Contacteur :	7
II.1.3. Actionneur :	8
II .1.3.1. Vérin :	8
II.1.3.1.1. Vérin double effet :	9
II.1.3.2. Les moteurs :	9
II.1.3.2.1. Moteur asynchrone :	9
II.1.3.2.1. Moteur hydraulique :	10
II.1.4. Capteur :	10
II.1.4.1 . Capteur de position à action mécanique :	11
II.1.4.2. Capteurs de temporisation (minuterie ) :	12
II.1.4.3. Capteurs de niveaux :	12
II.1.5. composants du circuit hydraulique :	12
II.2. Partie commande :	15
II.3 : Fonctionnement de la machine :	15
II.4. Circuit électrique :	16
II.5. Circuit hydraulique :	16
III. conclusion :	16

## **Chapitre II : Modification apportée à la machine**

I. Introduction :	17
II. Description du nouveau système conçu :	17
II.1. Partie opérative :	17
II.1.1. Structure mécanique :	17
II.1.1.1. Bras chargeur de feuille :	18
II.1.1.2. Tapis roulant 1 et 2 :	18
II.1.1.3. Bras déchargeur de feuille :	18
II.2. Prés-actionneurs :	18
II.3. Les actionneurs :	19
II.3.1. Critères de choix d'un vérin :	19
II.3.2. Les ventouses :	24
II.3.2.1. Domaines d'application :	25
II.3.2.2. Générateur de vide ou « Venturi » :	25
II.3.3. Types de moteurs utilisés :	26
II.3.3.1. Principe de fonctionnement du moteur asynchrone à fréquence fixe :	26
II.3.3.1. Démarrage direct d'un moteur asynchrone :	28
II.3.3.2. Démarrage étoile triangle :	29
II.3.3.3. Electro frein :	29
II.4. Capteurs :	30
II.4.1. Facteurs qui interviennent dans le choix d'un capteur :	30

II.4.3. Détecteur de position :	30
II.4.4. Interrupteurs à lame souple :	30
II.4.5. Détecteurs de proximité photoélectrique :	31
II.4.6 : Détecteur de niveau :	32
II.5. Equipements du chargeur et du déchargeur :	32
III. Poste opérateur :	33
III.1. Tableau de commande :	33
III.2. Mode manuel :	33
III.3. Mode automatique :	33
IV. Partie puissance de la machine :	33
IV.1. Bloc d'alimentation électrique :	34
IV.2. Source et production de l'énergie pneumatique :	34
IV.2.1. Conditionnement de l'air :	34
V. Mise en marche du chargeur :	35
VI. Mise en marche du déchargeur :	35
VII. Conclusion :	35

### **Chapitre III : Modélisation de la machine modifiée à l'aide du GRAFCET**

I. Introduction :	36
II. Généralité sur le GRAFCET :	36
II.1. Définition du GRAFCET :	36
II.2. Aspect structurel du GRAFCET :	36
II.2.1. l'étape :	36
II.2.2. L'étape initiale :	37
II.2.3. Transition :	37
II.2.4. Liaison :	37
II.2.5. Action :	38
II.2.6. Réceptivité :	38
II.3. Règle d'évolution du GRAFCET :	38
II.4. Règles de construction d'un grafcet :	39
III. Niveau d'un Grafcet :	40
III.1. Mise en équation du Grafcet :	41
IV. Grafcet du fonctionnement de la machine modifiée (niveau 1 et niveau 2) :	42
V : Conclusion :	59

### **Chapitre IV: Présentation et programmation de l'API S7-300**

I. Introduction aux automates programmables :	60
I.1. Automates programmables industriels (API) :	60
I.2. Architecture et fonctionnement d'un automate :	60
I.3. Choix d'un automate :	61
I.4. Constitution de l'automate S7-300 :	62
I.4.1. Module d'alimentation (PS) :	63
I.4.2. Description de la CPU :	63
I.4.3. Modules de coupleurs (IM) :	64
I.4.4. Module de signaux (SM) :	64
I.4.5. Modules d'entrées et de sorties tout ou rien (TOR) :	64
I.4.6. Modules d'entrée et de sortie analogique :	65
I.4.7. Module de fonction (FM) :	65

I.4.8. Modules de communication (CP) : .....	65
I.4.9. Châssis d’extension (UR) : .....	65
I.5. Fonctionnement de base d’un API : .....	65
I.5.1. Module central CPU : .....	65
I.5.2. Réception des informations sur les états du système : .....	66
I.5.3. Exécution du programme utilisateur : .....	66
I.5.4. Commande du processus : .....	66
II. Programmation de l’automate S7-300: .....	67
II.1. Création d’un projet STEP7 : .....	68
II.2. Configuration du matériel de l’automate : .....	69
II.3. Création de la table des mnémoniques : .....	70
II.4. Adressage des modules du S7-300 : .....	70
II.4.1. Adressage des modules lié a l’emplacement : .....	70
II.4.2. Adressage absolu des modules de signaux : .....	70
II.5. Mémentos : .....	71
II.6. Traitement du programme par l’automate : .....	71
II.7. Principe de conception d’une structure de programme : .....	72
II.8. Blocs dans le programme utilisateur : .....	72
II.8.1. Différents blocs du programme utilisateur : .....	73
II.9. Bloc fonctionnel système (SFB) : .....	74
II.10. Traitement du programme par la CPU : .....	74
II.10.1. Traitement linéaire du programme : .....	75
II.10.2. Traitement structuré du programme : .....	75
II.10.3. Structure du programme utilisateur : .....	75
II.11. Implantation du Grafset dans le S7-300 : .....	76
II.12. Conclusion : .....	76

**Chapitre V:                                      Simulation et Supervision**

I. Introduction : .....	77
I.1. Présentation du S7 PLCSIM : .....	77
I.2. Mise en route du logiciel S7-PLCSIM : .....	77
I.3. Visualisation de l’état du programme : .....	80
I.4. Simulation du programme de la machine modifiée : .....	80
I.5. Conclusion : .....	82
II. Introduction : .....	84
II.1. Emplacement de la supervision : .....	84
II.2. Constitution d’un système de supervision : .....	84
II.3. Module de visualisation : .....	84
II.4. Module d’archivage : .....	84
II.5. Module de traitement : .....	84
II.6. Module de communication : .....	85
II.7. Apport de la supervision : .....	85
II.8. Application développée sous WinCC : .....	86
II.9. Procédure de programmation avec WinCC : .....	86
II.10. Conclusion : .....	89
<b>Conclusion générale : .....</b>	<b>90</b>

## **Introduction générale :**

La première révolution industrielle a commencé vers 1814 avec l'apparition de la machine à vapeur, puis de la dynamo électrique. Un grand nombre de spécialistes pensent que la seconde révolution industrielle a commencé en 1955 avec la mise au point du premier transistor. Mais, c'est en 1980 que la troisième révolution industrielle a commencé avec l'éclosion de la micro-électronique et de la micro-informatique. Ces nouvelles technologies permettent de réaliser des systèmes complexes dont on ne mesure pas encore les limites. Dès lors, les êtres humains confient à des machines et des robots les tâches pénibles.

L'automatisation est un domaine pluridisciplinaire qui associe les notions de la mécanique, de l'électronique et de l'informatique. Elle permet d'exécuter des tâches industrielles avec une intervention humaine très réduite. Elle redevient ces dernières années le centre de préoccupation d'un bon nombre d'entreprises à cause de son évolution vers l'étude et la maîtrise des systèmes de plus en plus complexes, permettant l'exécution et le contrôle de tâches techniques par des machines fonctionnant sans intervention humaine, ou avec une intervention réduite. Avec le progrès technologique, l'automatisation des installations constitue un des facteurs essentiels contribuant à la croissance de la productivité, et un élément important dans l'amélioration de la sécurité du travail ainsi que la réduction des coûts de production.

De nos jours, grâce au développement de l'informatique et de la microélectronique et à la fabrication de microprocesseurs de plus en plus performants, l'automatisation est assurée par des Automates Programmables Industriels (API) qui intègrent ces nouvelles technologies.

L'Entreprise Nationale des Industries de l'Electroménager (ENIEM), a débuté l'expérience des installations automatisées, en introduisant les Automates Programmables Industriels, suite à l'intervention de la firme BOSCH (Allemagne) en 1977 et TOSHIBA (japon) en 1987. L'ENIEM axe sa politique de qualité sur l'amélioration de ses produits et ses services afin d'obtenir des produits à qualité uniforme et faire face à une concurrence rude des firmes internationales d'électroménager. La compétitive des entreprises exige une automatisation de plus en plus flexible et évolutive des équipements de production.

Ce qui explique la rude concurrence entre les firmes internationales (Schneider, Siemens, Bosch... etc. qui produisent les API.

Actuellement l'ENIEM se voit confronter au manque des cartes d'entrées/sorties des séquenceurs qu'elle utilise. Ces cartes tombent souvent en pannes et ne sont plus disponibles sur le marché. Malgré les efforts des ingénieurs de les réparer, cela reste insuffisant vu le

rythme de la production ce qui oblige l'entreprise à réfléchir d'hors et déjà à remplacer ces séquenceurs.

Dans ce contexte le sujet qui nous a été proposé par le Département Maintenance de l'unité froid, consiste à faire l'étude et la modélisation de la Presse de découpage et de poinçonnage des contre porte de réfrigérateur en utilisant comme outil de modélisation le GRAFCET en vue de proposer une solution à l'automatisation moyennant l'automate programmable Siemens.

Notre mémoire est organisé comme suit :

- ✓ Le premier chapitre est consacré à la présentation et description de la machine.
- ✓ Le deuxième chapitre est consacré à la nouvelle conception de la machine.
- ✓ Le troisième chapitre est réservé à la modélisation de la machine à l'aide du GRAFCET.
- ✓ Le quatrième chapitre est consacré à la présentation de l'automate programmable S7-300 et son langage de programmation ainsi que la programmation de la machine modifiée.
- ✓ Le dernier chapitre est réservé à la simulation du programme avec le logiciel S7-PLCSIM et développement de la plate forme de supervision WinCC.
- ✓ Nous terminons notre travail par une conclusion générale.



## I.Introduction :

La machine étudiée est semi automatique elle sert à découper et à poinçonner des contre portes de réfrigérateurs de format petit modèle selon le moule choisi. Elle occupe une place très importante dans la chaîne de production des réfrigérateurs.

La (Figure I.1) donne la photographie de cette machine. Dans ce chapitre, nous décrivons le fonctionnement de la machine et les éléments qui la composent.

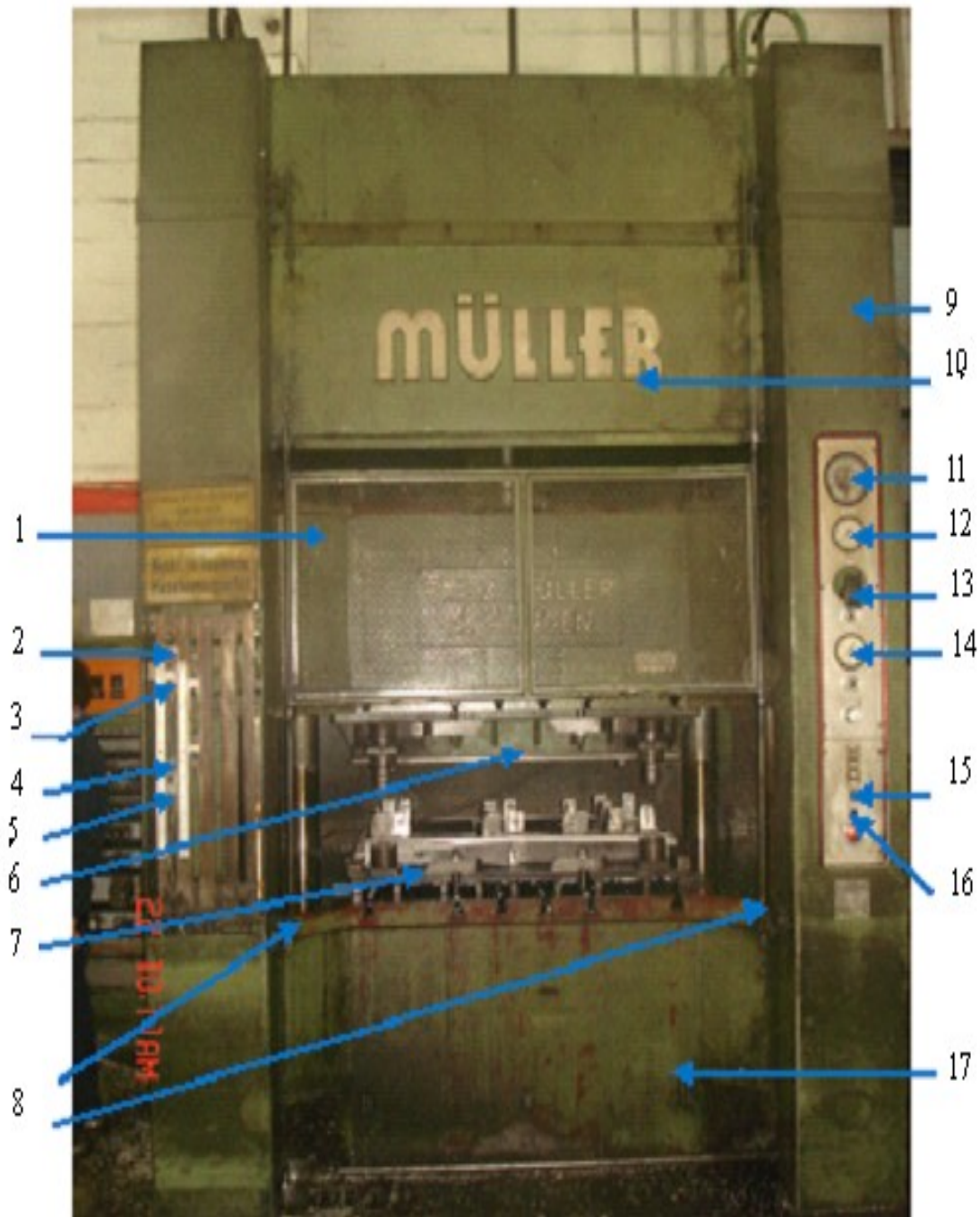


Figure I.1 : Presse FRITZ MÜLLER.

- 1 : Grille de protection.
- 2 : Fin de course coulisseau point mort haut.
- 3 : Fin de course coulisseau point de freinage haut.
- 4 : Fin de course coulisseau point de freinage bas.
- 5 : Fin de course coulisseau point mort bas.
- 6 : Formateur.
- 7 : Table fixe ( moule ).
- 8 : Deux fin de course démarrage cycle.
- 9 : Glissoir ( colonne ).
- 10 : Réservoir.
- 11 : Température de l'huile.
- 12 : Pression de filtre.
- 13 : Pression de commande.
- 14 : Pression de coulisseau.
- 15 : Pompe marche.
- 16 : Arrêt d'urgence (interrupteur de secours ).
- 17 : Bâti.

## **II . Description de la machine :**

La machine est composée de différents modules comme illustré dans la (figure I.2) :

- ❖ La partie opérative composée de la structure mécanique, des pré-actionneurs, des actionneurs, des capteurs et du circuit hydraulique.
- ❖ La partie commande constituée de l'armoire de commande.
- ❖ Les interfaces d'entrées/sorties.
- ❖ Le circuit électrique d'alimentation.

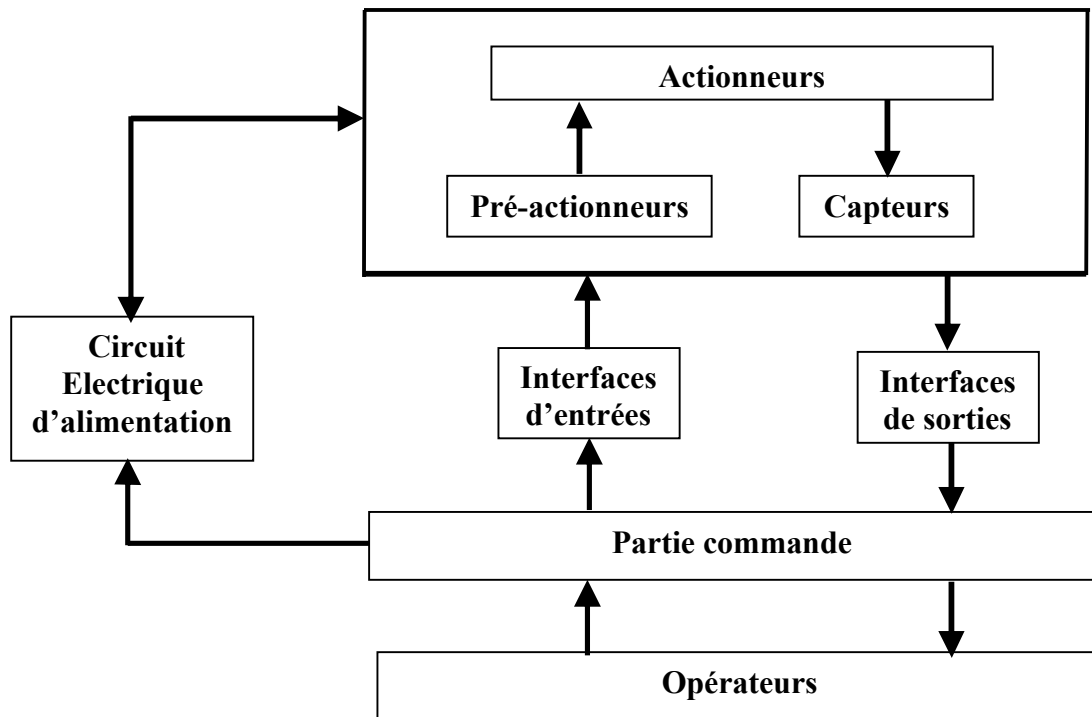


Figure I.2 : Modules composants la machine.

## II .1. Partie opérative :

### II .1.1. Structure mécanique :

Les éléments mécaniques décrits ci-après sont illustrés dans la ( figure I.1).

- ❖ **bâti** : C'est une construction robuste en acier soudé supportant les éléments de la machine et entièrement capotés par des panneaux éclipables.
- ❖ **Piston supérieur** : Le piston supérieur du vérin double effet repose sur une traverse en U à montant particulièrement robuste ; c'est lui qui porte le formateur.
- ❖ **Coulisseau** : C'est l'organe principal de la machine. Toute la pression de coupe est portée par le coulisseau.
- ❖ **Glissoir** : C'est un dispositif de soutien et de guidage du coulisseau.
- ❖ **Grille de protection** : La presse est équipée d'une grille coulissante qui protège l'opérateur, la descente du coulisseau ne peut se faire que si cette grille est bien fermée. La fermeture de la grille actionne deux fin de course.
- ❖ **Réservoir principale** :

Toute installation oléo-hydraulique possède un réservoir ou bêche qui contient le fluide hydraulique, d'une capacité 400L. Il sert principalement :

- Au stockage de la quantité d'huile nécessaire au fonctionnement correct du système.
- A la protection de l'huile contre les éléments extérieurs nuisibles.

-Comme support aux autres composants du groupe hydraulique tels que le moteur qui entraîne la pompe, les filtres ,etc

-Au refroidissement du fluide hydraulique ( la surface totale du réservoir en contact avec l'air ambiant permet un bon échange de chaleur).

Son symbole est :



❖ **Réservoir collecteur :** L'huile de fuite se formant dans chaque opération de pressage hydraulique est collectée dans ce réservoir se trouvant au point de base du système hydraulique.

### **.II.1.2. Prés-actionneur :**

Un pré-actionneur est un composant de gestion de l'énergie de commande d'actionneur, il traduit un signal de commande en signal de puissance. A toute action est associé un pré-actionneur indispensable pour son fonctionnement.

#### **II.1.2.1. Electrovanne :**

C'est un pré-actionneur électro-hydraulique tout ou rien (TOR ) permettant le passage d'huile véhiculé dans le circuit hydraulique.

L'électrovanne est principalement constituée d'un corps de vanne où circule l'huile. Elle est munie d'une bobine alimentée électriquement et engendrant une force magnétique qui déplace le noyau mobile qui agit sur l'orifice de passage. En se déplaçant, le noyau peut permettre ou pas le passage de l'huile.

Le champ de pression dépend directement de la force d'attraction de la bobine. Le bobinage doit être alimenté de façon continue pour maintenir le noyau attiré.

Sur la machine presse muller plusieurs types d'électrovannes sont utilisées selon leur nombre d'orifices, de position et leur sens de passage dans chaque position ; ces dernières sont commandées par une tension 220 V .

#### ❖ **Distributeur :**

C'est un organe dont le rôle est d'établir ou d'interrompre la communication entre le réservoir du fluide et les vérins ; ainsi il est inséré entre la source et les organes moteurs.

Il est caractérisé par :

- Le nombre des orifices : 2,3,4 ou 5.
- Le nombre des modes de distribution ou position : 2 ou 3.
- Le type de commande ou de pilotage assurant le changement de position :

Simple pilotage avec rappel par ressort ou double pilotage, avec rappel au centre par ressort dans le cas des distributeurs à trois positions.

- La technologie de pilotage : hydraulique ou électro-hydraulique.

Dans chaque case ou position, les voies sont figurées par des flèches indiquant le sens de circulation du fluide (figure I.3).

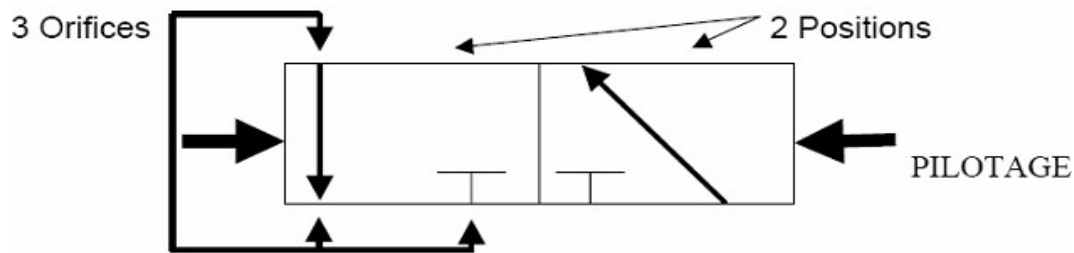


Figure I.3 : Distributeur .

Les vérins à simple effet sont alimentés par des distributeurs 3/2.

Les vérins à double effet sont alimentés soit par des distributeurs 4/2 soit par des distributeurs 5/2.

La commande du changement de position est obtenue par déplacement du tiroir ou des clapets qui sont les éléments mobiles essentiels des distributeurs.

#### II.1.2.2. Contacteur :

Un contacteur est un relais de haute puissance modulaire comportant des contacts à double rupture pour s'assurer de pouvoir couper des tensions et des courants élevés.

Les contacteurs sont utilisés pour commander de moyennes ou grosses charges électriques. Dès que l'on envisage de commander un moteur, quelque soit sa puissance, on devrait utiliser un contacteur.

Il se compose d'une bobine qui est l'organe de commande, des contacts principaux et de contacts auxiliaires.

Lorsque la bobine est alimentée en courant, l'armature est attirée et ferme les contacts.

Lorsque la tension entre A1 et A2 est supprimée les contacts reviennent à la position initiale par l'action d'un ressort de rappel (figure I.4).

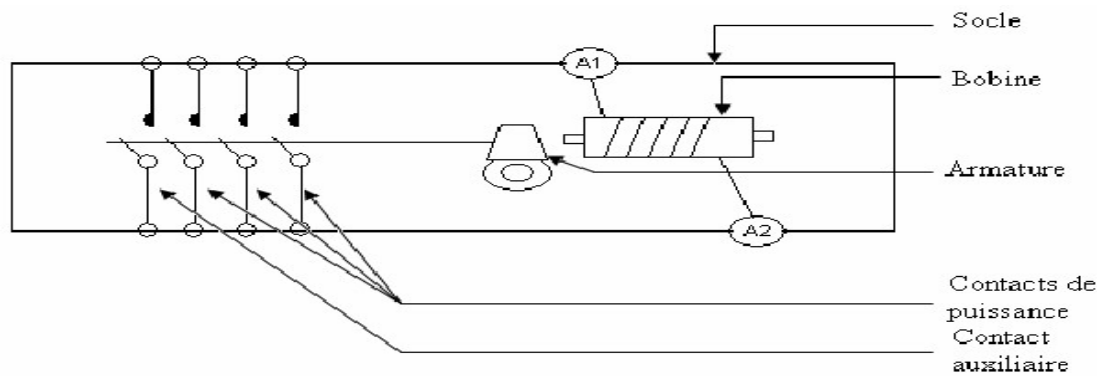


Figure I.4 : Schéma du contacteur.

### II.1.3. Actionneur :

Ceux sont des composants qui transforment une énergie prélevée sur une source extérieure en une action physique sur la matière d'œuvre.

#### II .1.3.1. Vérin :

C'est un actionneur linéaire qui transforme une énergie pneumatique ou hydraulique en un travail mécanique.

Cet actionneur de conception robuste et de simplicité de mise en œuvre est utilisé dans toutes les industries manufacturières.

Il permet de reproduire les actions manuelles d'un opérateur telles que : soulever, pousser, tirer, plier, serrer,... etc

Le vérin est constitué d'un piston muni d'une tige qui se déplace librement à l'intérieur d'un tube. Pour faire sortir la tige on applique une pression sur la face avant du piston, et sur la face arrière pour faire entrer la tige (figure I.5).

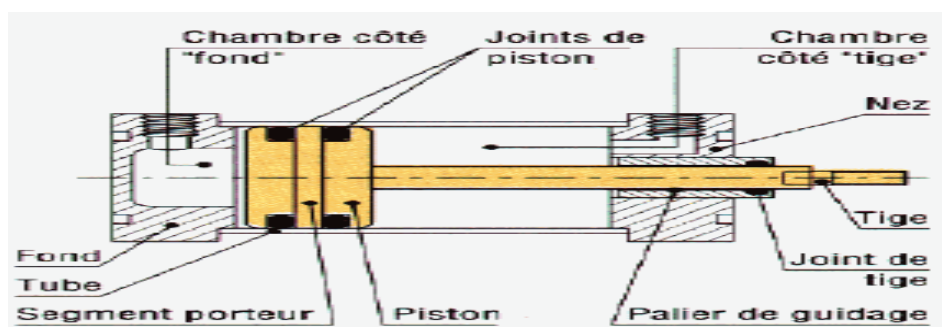


Figure I.5 : Vue en coupe d'un vérin.

Certains vérins disposent d'amortisseurs afin d'obtenir un ralentissement en fin de mouvement de façon à éviter un choc du piston sur le nez ou le fond du vérin.

Selon la manière d'admission de l'air comprimé(ou l'huile), on distingue deux types de vérins :

- si l'air comprimé (ou l'huile) est admissible sur une seule face du piston, le vérin est dit « vérin simple effet ».
- si l'air comprimé (ou l'huile) est admissible sur deux faces du piston, le vérin est dit « vérin à double effet ».

Dans la presse Müller les actionneurs utilisés sont : vérin hydraulique double effet et des moteurs.

#### II.1.3.1.1. Verin double effet :

Ce type de vérin développe une force disponible à l'aller comme au retour pour produire un travail. Il a deux directions de travail et comporte deux orifices d'alimentation.

A tout vérin (Actionneur) est associé un distributeur (pré-actionneur) indispensable pour son fonctionnement. La pression est distribuée alternativement de chaque côté du piston, afin d'assurer son déplacement dans un sens, puis dans l'autre (figure I.6).

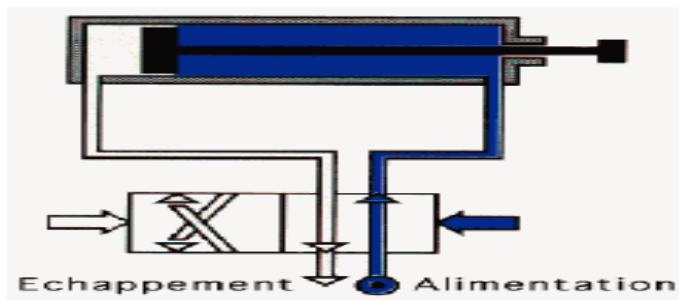


Figure I.6 : Vérin à double effet avec son distributeur.

#### II.1.3.2. Les moteurs :

Nous intéressons à deux types de moteurs :

##### II.1.3.2.1. Moteur asynchrone :

###### a) Présentation :

Ce moteur se caractérise par le fait qu'il est constitué d'un stator (inducteur) alimenté en courant alternatif et d'un rotor (induit) qui est soit en court-circuit, soit bobiné, aboutissant à des bagues dans lesquelles le courant est créé par induction.

On distingue deux catégories motrices asynchrones en fonction du type de rotor :

- les moteurs asynchrones à rotor en court-circuit (à cage) : ils sont en général de faible puissance.
- Les moteurs asynchrones à rotor bobiné à bagues dans lesquelles l'enroulement du rotor aboutit à des bagues par l'intermédiaire desquelles on peut insérer des résistances : ils sont de grande puissance.

### II.1.3.2.2. Moteur hydraulique :

#### a) présentation :

Les pompes et les moteurs sont des appareils qui convertissent l'énergie hydraulique en énergie mécanique (moteur) ou inversement (pompe).

Dans le cas de la pompe hydraulique, celle-ci puise généralement le fluide dans un réservoir approprié, par le côté aspiration et elle débite ce fluide par son côté refoulement.

Pour les moteurs hydrauliques, ceux-ci transforment à nouveau l'énergie produite par les pompes en énergie mécanique nécessaire à un récepteur à mouvement de rotation. Les moteurs hydrauliques ont, en général, la même constitution que les pompes hydrauliques de même type. Modèle fonctionnel : Pompe hydraulique (figure I.7).

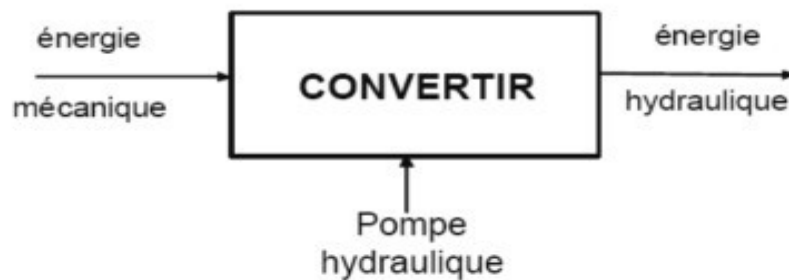


Figure I.7 : Modèle fonctionnel d'une pompe hydraulique.

#### Caractéristiques :

vitesse de rotation en tours/minute ; couple en Newton x mètre ; le débit en L/min.

#### b) Pompes et moteurs à pistons axiaux :

C'est une unité dont les pistons sont disposés parallèlement à l'axe. Cette unité est adaptée, de part sa technologie, à des vitesses relativement élevées. Elle peut être à cylindrée fixe ou variable. La gamme de pression peut aller jusqu'à 450 bars. La course des pistons est provoquée par l'inclinaison d'un plateau par rapport au barillet contenant les pistons.

Le nombre de pistons détermine la stabilité du débit aux orifices ; en effet chaque piston est soit à l'aspiration, soit au refoulement, le débit présente donc des irrégularités d'autant plus grandes que le nombre de pistons est faible ou que celui-ci est pair.

### II.1.4. Capteur :

Un capteur est un objet technique de prélèvement d'information sur un processus. Il réalise la conversion d'une grandeur physique mesurée (grandeur d'entrée) en une autre grandeur



physique accessible, c'est-à-dire exploitable par un composant de traitement (grandeur de sortie).

Il prélève une information sur le comportement de la partie opérative et la transforme en une information exploitable par la partie commande.

L'information délivrée par un capteur peut être logique (deux états 1 ou 0), numérique (valeur discrète), analogique (dans ce cas il faudrait adjoindre à la partie commande un module de conversion analogique numérique).

Les capteurs peuvent être caractérisés selon deux critères :

- La grandeur mesurée : on parle alors de capteurs de position, de température, de vitesse, de force, etc
- Le caractère de l'information délivrée : on parle alors de capteurs logiques appelés aussi capteurs Tout ou Rien (TOR), de capteurs analogiques ou numériques.

#### II.1.4.1 . Capteur de position à action mécanique :

Dans le déroulement d'un cycle automatisé, il est important de connaître la position exacte des vérins (tige sortie ou tige entrée) afin de faire évoluer la partie commande dans le cahier des charges. Les ordres d'évolution sont donnés par des éléments de détection (capteur de position) placés sur la machine ou implémentés directement sur le vérin lorsque la zone de travail ne le permet pas.

Ces capteurs sont à contacts et peuvent être équipés d'un galet, d'une tige souple ou d'une bille. L'information donnée est de type (TOR).

Le capteur de position utilisé dans la machine presse MULLER est de type Tout Ou Rien (figure I.8) :

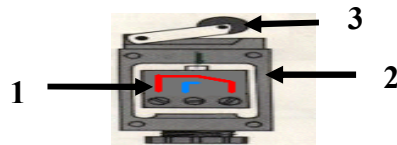


Figure I.8 : Interrupteur de fin de course.

On distingue sur la machine les fins de course suivants :

- Deux capteurs de position pour la descente du coulisseau.
- Capteur de position coulisseau en haut.
- Capteur de position coulisseau frein en haut.
- Capteur de position coulisseau frein en bas.
- Capteur de position coulisseau en bas.

#### ➤ Principe de fonctionnement :

Les interrupteurs de position sont constitués des trois éléments de base suivants (Figure I.7) :

- Un contact électrique (1) ;
- Un corps (2) ;
- Une tête de commande avec son dispositif d'attaque (3).

La détection de présence est réalisée lorsque l'objet à détecter entre en contact avec la tête de commande au niveau de son dispositif d'attaque. Le mouvement engendré sur la tête d'attaque provoque la fermeture du contact électrique situé dans le corps du capteur.

#### **II.1.4.2. Capteurs de temporisation (minuteries) :**

Les minuteries sont réglées par l'opérateur selon la durée voulue de chaque action. Dès que l'action est activée, la minuterie se déclenche. Après écoulement du temps de la minuterie, l'action associée s'arrête.

#### **II.1.4.3. Capteurs de niveaux :**

La presse MULLER possède deux détecteurs de niveau sur le réservoir collecteur.

La commande de l'huile de fuite s'exerce par le commutateur de niveau qui est équipé de deux disjoncteurs. Un flotteur coulissant sur le tuyau de fixation provoque l'enclenchement de la pompe de rappel, dans le cas où le niveau de l'huile monte considérablement, et dépasse le niveau moyen, le flotteur se trouve poussé vers le haut et commute par ce fait la pompe de rappel.

#### **II.1.5. composants du circuit hydraulique :**

La partie hydraulique est constituée de plusieurs organes qui sont :

##### **❖ Réservoir d'huile :**

D'une capacité de 400 l, il sert principalement :

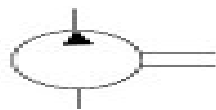
Au stockage de la quantité d'huile nécessaire au fonctionnement du système.

##### **❖ Pompe :**

Les pompes sont des appareils qui convertissent l'énergie mécanique en énergie hydraulique.

La pompe hydraulique puise généralement le fluide dans un réservoir approprié, par le côté aspiration et elle débite ce fluide par son côté refoulement.

Le symbole de la pompe :



La presse possède trois pompes

M 1 : Pompe principale de puissance 15 KW.

M 2 : Pompe pression de commande de puissance 2.2 KW.

M 3 : Pompe de retour d'huile de puissance 0.25 KW.

**Remarque :**

La presse est équipée d'une pompe à main de lubrification, qui est une pompe de construction simple que l'on manœuvre à la main et d'un excellent rendement. Elle permet de procéder dans de bonnes conditions à la lubrification du coulisseau et des colonnes.

❖ **Accumulateur oléo-pneumatique :**

Il sert à emmagasiner de l'énergie hydraulique et de la restituer en cas de besoin au système hydraulique .

Un accumulateur comporte une partie occupée par un fluide sous pression et une autre occupée par un gaz, parties séparées par une paroi élastique (vessie).

Cette énergie résulte de la compression du volume occupé par le gaz, le gaz le plus souvent utilisé est l'azote.

La partie de l'accumulateur occupée par le fluide est en communication avec le circuit hydraulique de l'installation .

L'installation d'un accumulateur dans un circuit oléo-hydraulique est nécessaire pour trois raisons principales :

- L'accumulateur constitue une source de puissance auxiliaire dans le cas de brusques variations de charges (économie d'énergie).
- L'énergie stockée dans l'accumulateur constitue une réserve d'énergie de secours, lors de panne du groupe motopompe (sécurité).
- Limiter le taux d'irrégularité du débit des pompes, il s'ensuit un meilleur fonctionnement de l'installation.

Son symbole est :



- 1 : Corps.
- 2 : Vessie
- 3 : Valve de gonflage.

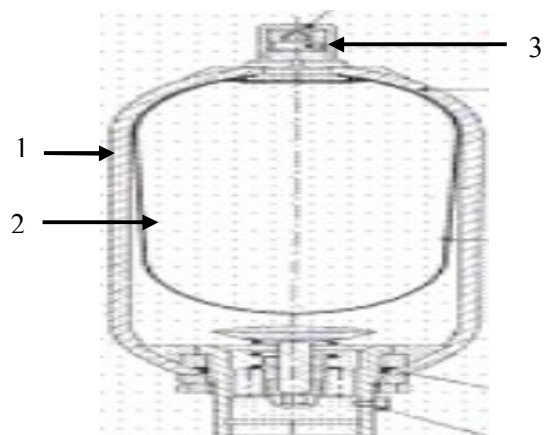


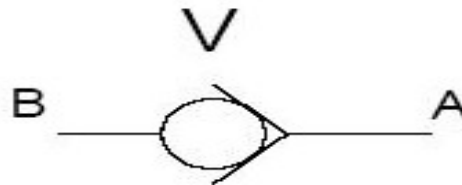
Figure I.9 : accumulateur hydraulique.

### ❖ Soupape de retenue ou clapet anti-retour :

Ce sont les appareils à la fois les plus simples et les plus utilisés dans les systèmes oléo-hydrauliques.

Un tel dispositif permet de contrôler le sens de circulation d'un fluide quelconque. Il permet le passage d'un liquide dans un sens et bloque le flux si celui-ci venait à s'inverser comme la diode dans un circuit électrique.

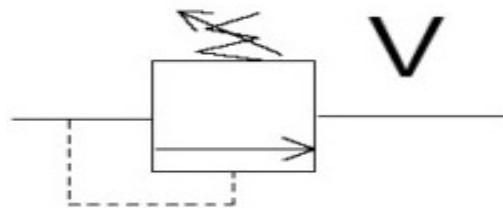
Son symbole est :



### ❖ Régulateur de pression :

Il a pour fonction de limiter la pression dans un circuit et de faire retourner au bac le débit excédentaire. Il met un système de commande oléo-hydraulique à l'abri d'une élévation de pression dangereuse en acheminant le fluide vers la bêche chaque fois que la pression de consigne est dépassée.

Son symbole est:



### ❖ Manomètre ( indicateur de pression d'huile ) :

Il indique la pression de l'huile de service dans les différents organes hydraulique. dès que le moteur est mis en marche, le manomètre doit donner une indication prédéfinie (dans notre cas 30 bars). La pression de l'huile est mesurée par rapport à une pression de référence qui doit être réglée.

#### • Définition de la grandeur pression :

La pression est la force appliquée à une surface ou répartie sur celle-ci. Elle se définit comme suit :  $P = F/S$

P : pression en  $N/m^2$  ( 1 Pa = 1  $N/m^2$ ).

F : force en Newton.

S : surface en  $m^2$ .

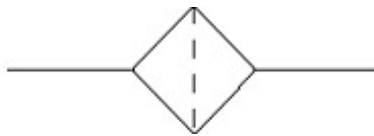
La pression est souvent exprimée en bar ( 1 bar =  $10^5$  Pa).

### ❖ **Filtre :**

❖ D'une manière générale, 60 à 70 % des pannes de circuits oléo-hydrauliques sont dues à un mauvais état du fluide. Une filtration efficace réduit les temps d'arrêt de production, évite le remplacement prématuré des composants.

L'huile circulant dans l'installation véhicule toute sorte d'impuretés se trouvant à l'intérieur du circuit hydraulique. Ces impuretés sont éliminées par filtrage. Le filtre est monté sur la conduite de retour vers le réservoir dont le rôle est de maintenir les particules indésirables pour éviter l'encrassement du système.

Son symbole est :



### ❖ **Tuyauteries :**

La jonction des différents appareils oléo-hydrauliques peut se faire au moyen de tuyauteries rigides ou flexibles.

## **II.2. Partie commande :**

### ❖ **Dispositif de commande :**

Ce dispositif commande le déroulement du cycle de travail. Tous les déplacements sont commandés hydrauliquement par vérins en relation avec des boutons poussoirs (cycle coup par coup) et des électrovannes (cycle automatique programmé).

Toutes les fonctions incluses dans un cycle de découpage peuvent être programmées et commandées automatiquement par relais temporisés. Par ailleurs, chaque fonction peut être commandée manuellement en coup par coup.

#### ▪ **Parcours en marche par inertie :**

Le parcours en marche par inertie est le parcours effectué par le coulisseau jusqu'à son arrêt depuis le moment où l'organe de commande a été relâché.

Le parcours en marche par inertie est à mesurer à des intervalles réguliers, ces contrôles permettent de s'assurer que la valeur limite indiquée par le constructeur n'est pas dépassée.

## **II.3 : Fonctionnement de la machine :**

### ❖ **Préparatifs :**

- Le montage du moule est supposé être effectué correctement.
- Choisir un mode de fonctionnement.
- Mettre la machine sous tension ( les deux pompes sont en marche ).
- Contrôler la pression de service.

### ❖ Cycle de découpage et de poinçonnage :

L'opérateur prend la feuille en plastique et la pose sur le moule ; il l'arrange de façon à ce qu'il y ait une étanchéité, puis descend la grille de protection . En position de protection, la grille doit interdire totalement l'accès de la zone dangereuse. La presse ne peut être commandée qu'une fois la grille en position de protection. La grille demeure en position de protection jusqu'à la fin du mouvement de fermeture de la presse ; si elle quitte temporairement cette position, le mouvement de fermeture de la presse est immédiatement interrompu.

- Les mouvements du coulisseau sont commandés par un distributeur à commande hydraulique (distributeur principal ).

Lorsque les valves S1 et S2 ne sont pas alimentées, le distributeur est ouvert en position milieu ; le débit d'huile que fournit la pompe est alors renvoyé, sans pression au réservoir.

- La valve S1 vient à être alimentée, le débit de la pompe est aiguillé vers le vérin de presse qui provoque la descente du coulisseau. Pour accélérer cette descente, la presse est équipée d'une soupape de mouvement rapide.

- Le freinage du mouvement rapide est réglé de telle façon que l'action de freinage intervienne peu de temps avant la fermeture de l'outillage. Le freinage du coulisseau augmente la durée de vie des outils et permet au processus d'emboutissage de se dérouler de façon optimale.

- Pour éviter un arrêt brusque du coulisseau en fin de course ascendante, il importe de freiner la dernière partie de la course. Elle est réalisée par modification du débit fourni par la pompe. A cet effet, le régulateur de puissance de la pompe modifie l'angle d'inclinaison du corps de pompe, faisant passer le débit de sa valeur maximale à sa valeur minimale. Un freinage progressif en résulte ; ( ce freinage est commandé par le fin de course coulisseau haut).

### **II.4. Circuit électrique :**

Le schéma électrique ainsi que la signification des différents éléments le composant sont donnés dans l'annexe A.

### **II.5. Circuit hydraulique :**

Le circuit hydraulique ainsi que la signification des différents éléments le composants sont donnés dans l'annexe B.

### **III. conclusion :**

La machine de fabrication des contres portes de réfrigérateurs reste semi automatique et ne peut se passer de l'intervention d'un opérateur humain pour l'introduction de la feuille et l'extraction de la pièce finie, malgré l'industrie de nos jours qui se tourne vers une production automatisée, passant de la production manuelle à faible rendement à une production robotisée et simplifiée.

Dans la suite du travail, nous allons présenté la nouvelle conception du système d'introduction de pièces.

## **I. Introduction :**

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. Donc 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. Afin d'atteindre cet objectif, des modifications (améliorations) des systèmes existants s'avèrent souvent nécessaires. Dans notre cas d'étude, l'objectif est d'automatiser une machine qui fait le découpage et le poinçonnage des contre-portes de réfrigérateurs. Pour cela, nous avons choisi de concevoir un « bras » à l'aide d'un automatisme pneumatique. Les utilisations de l'air comprimé sont très nombreuses. Ainsi les automatismes pneumatiques se révèlent fort utiles partout où il existe des atmosphères déflagrantes, comme dans les raffineries de pétrole, etc. D'une manière générale les déplacements très rapides, et la grande fiabilité de fonctionnement sont les avantages connus et incontestés des systèmes automatisés pneumatiques.

Dans ce chapitre, nous allons décrire les modifications à apporter pour rendre la machine complètement automatique, sans intervention de l'opérateur.

## **II. Description du nouveau système conçu :**

La machine presse MÜLLER de contre-portes de réfrigérateurs fonctionne actuellement en semi-automatique. C'est-à-dire qu'un opérateur introduit manuellement une feuille plastique à l'intérieur de la machine, puis le cycle est mis en route après avoir appuyé sur deux boutons de démarrage de cycle.

Le travail demandé est de concevoir un système d'introduction de la feuille plastique, et extraction de la pièce finie à la fin du cycle de découpage, où l'opérateur n'aura plus à intervenir sur la machine, sauf pour les réglages.

### **II.1. Partie opérative :**

#### **II.1.1. Structure mécanique :**

La nouvelle conception est représentée en figure II.1.

**Bloc1 :** Dispositif de transport de feuille (tapi roulant) et bras chargeur.



**Bloc 2 :** Matrice hydraulique (presse).

**Bloc 3 :** Bras déchargeur, transporteur (tapis roulant d'évacuation).

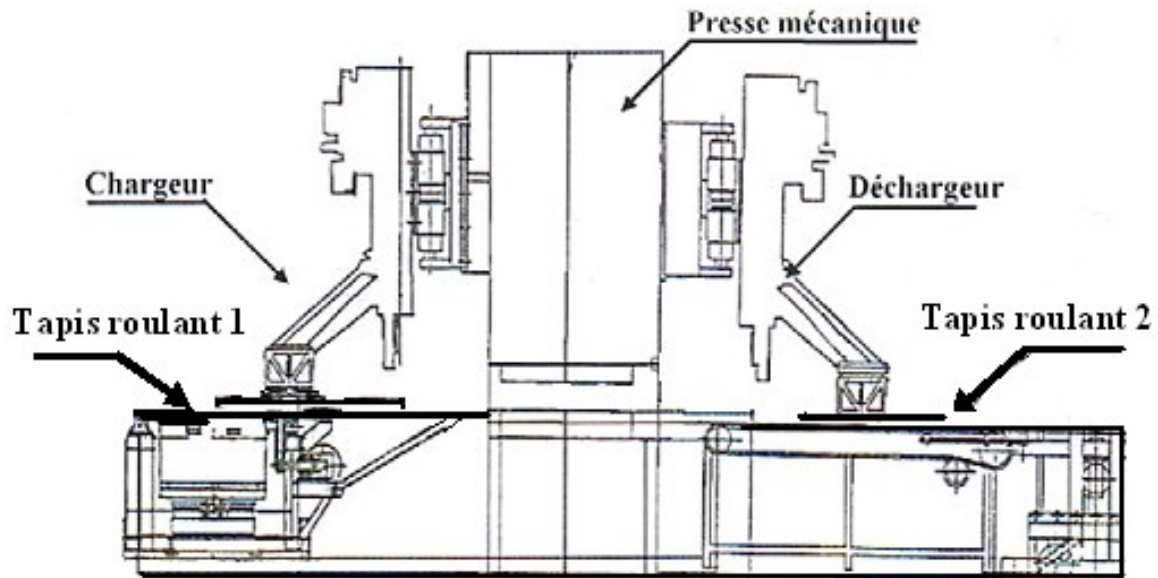


Figure II.1 : Schéma de la machine réadaptée de découpage et poinçonnage de contre porte.

#### **II.1.1.1. Bras chargeur de feuille :**

Ce dispositif a pour fonction d'alimenter en feuille plastique la presse hydraulique, Il s'agit de déplacer les feuilles du tapis roulant vers la presse.

L'entraînement se fait par un moteur à changement de pôles embrayage et frein.

La fixation se fait par support à charnière sur le montant gauche de la presse, orientable de 180°.

#### **II.1.1.2. Tapis roulant 1 et 2 :**

C'est un dispositif de transport muni d'une surface plane entraînée par un moteur électrique triphasé (entraînement par intermittence) et permettant le déplacement des contre-portes de réfrigérateurs. Son travail commence lorsqu'une pièce est détectée à l'entrée et s'arrête quand celle-ci arrive en fin de course.

#### **II.1.1.3. Bras déchargeur de feuille :**

Ce dispositif a pour fonction de faire évacuer les pièces ayant été façonnées dans la presse de découpage et les déposer sur le transporteur.

## **II.2. Près-actionneurs :**

### **a) Choix d'un distributeur pneumatique :**

Le choix d'un distributeur dépend de sa fonction (nombre d'orifices), de la position de repos, du type de pilotage et de sa taille.

**b) Choix de la fonction (nombre d'orifices) :** Ce choix dépend naturellement de l'actionneur à alimenter : 2/2, blocage ou ventouse ; 3/2, pour vérin simple effet, ventouse.

### **c) Choix de position de repos :**

De ce choix dépend le comportement de l'actionneur alimenté lorsque l'énergie de commande est coupée.

#### **○ Distributeur monostable :**

Si le distributeur possède un pilotage par ressort, il est monostable. C'est-à-dire que seule la position obtenue grâce au ressort est stable : absence de signal de commande extérieur, le tiroir se place automatiquement dans la position ressort.

#### **○ Distributeur bistable :**

Si le distributeur possède deux pilotages de même nature, c'est-à-dire que les deux positions sont des positions stables : en absence de signal de commande extérieur, le tiroir ne bouge pas et reste dans la position qu'il occupe.

Deux électrovannes sont utilisées pour commander la descente et la montée des vérins portes ventouses de chargeur et du déchargeur de type 5/2 (vérin double effet).

## **II.3. Les actionneurs :**

### **II.3.1. Critères de choix d'un vérin :**

Un certain nombre de critères doit être pris en compte pour déterminer le vérin à utiliser. Il faut d'abord connaître l'effort de déplacement de la charge et son sens pour définir les deux caractéristiques dimensionnelles du vérin, son diamètre ( $\Phi$ ) et sa course. Il sera ensuite

nécessaire de déterminer la vitesse de la tige afin de déterminer l'énergie cinétique et l'amortissement de l'ensemble mobile (piston+tige+charge).

La figure (II.2) illustre les caractéristiques dimensionnelles d'un vérin.

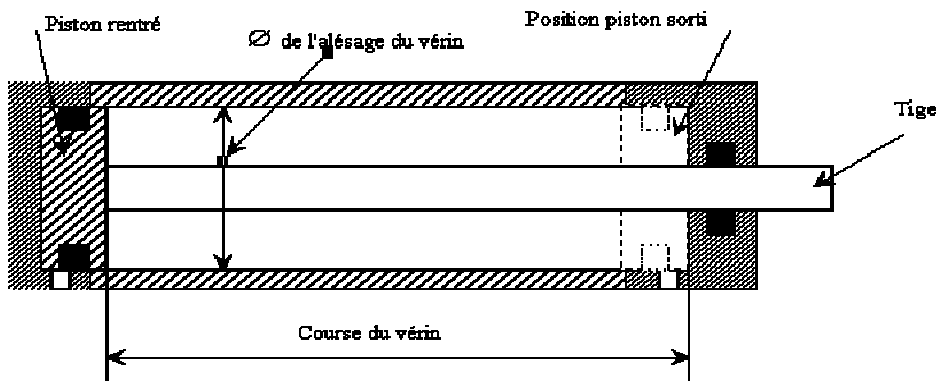


Figure II.2 : Caractéristiques dimensionnelles du vérin.

Pour le calcul des efforts de poussée et de rentrée du vérin, on choisit d'utiliser un vérin P avec un piston de diamètre D cm et une tige de diamètre d cm sous une pression « p » en bars.

Pour cela nous aurons deux cas.

1<sup>er</sup> cas- Calcul d'effort d'un vérin poussant une charge :

Efforts exercés (newton) = pression \* section du vérin.

$$= P * \pi * R^2 \quad \text{Avec } R = \frac{D}{2}$$

2<sup>ème</sup> cas- Calcul d'effort d'un vérin double effet tirant une charge :

Dans ce cas la pression ne s'exerce plus sur la totalité du diamètre du piston mais sur une surface égale à la section du piston moins la section de la tige.

Efforts exercés (Newton) = pression\* section du vérin.

$$= P * \pi * (R^2 - r^2) \quad \text{Avec } r = \frac{d}{2}$$

Pour calculer le diamètre D de l'alésage, il faut d'abord calculer la section S, avec  $F_{nécessaire}$  et la pression P de l'air comprimé.

$$S = \frac{F_{nécessaire}}{P}$$

La section  $S$  s'écrit en fonction du diamètre  $D$  :  $S = \pi * \frac{D^2}{4}$ , on en déduit le diamètre  $D$  :

$$D(mm) = \sqrt{\frac{4 * S(mm^2)}{\pi}}$$

Il faut ensuite choisir le diamètre parmi les diamètres normalisés. Deux solutions sont possibles :

\_Soit, on choisit un diamètre légèrement inférieur, et le taux de charge sera plus grand.

\_Soit, on choisit un diamètre légèrement supérieur, et le taux de charge sera plus petit.

Une autre méthode pour déterminer le diamètre d'un vérin consiste à utiliser les abaques du constructeur (figure II.3.a et b).

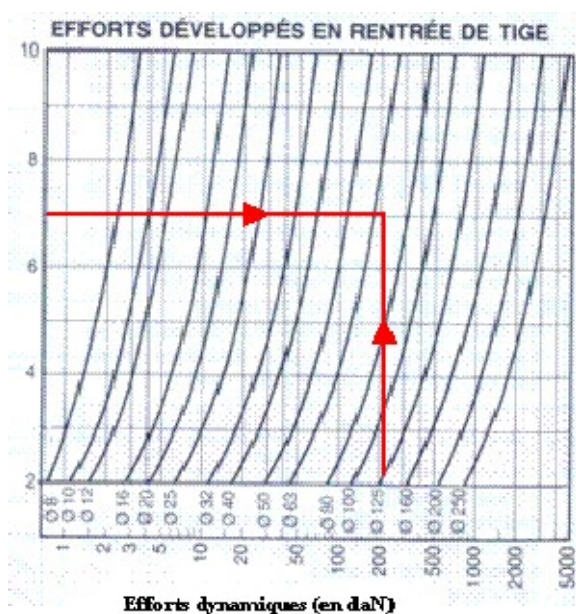


Figure II.3.a : Efforts en rentrée de tige.

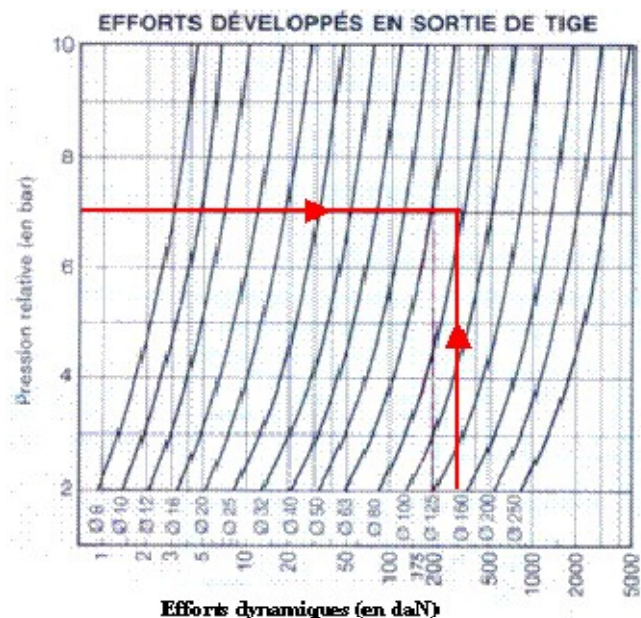


Figure II.3.b : Efforts en sortie de tige.

Pour utiliser ces abaques représentées en (figure II.3.a et b), il faut choisir si le vérin travaille en « rentrée de tige » ou en « sortie de tige », et prendre l'abaque correspondante. Il faut définir le point de rencontre entre l'effort dynamique calculé et la pression d'alimentation. Le diamètre sera celui dont la courbe passe par ce point. Si le point est entre deux courbes, il faudra faire un choix comme précédemment entre un vérin plus petit, moins cher et dont le taux de charge sera supérieur à 0.5, et un vérin plus grand, plus cher, dont le taux de charge sera inférieur à 0.5.

Avec l'air comprimé, on dispose d'une énergie potentielle exploitable sous forme statique ou sous forme dynamique par transformation en énergie cinétique.

- **Force statique :**

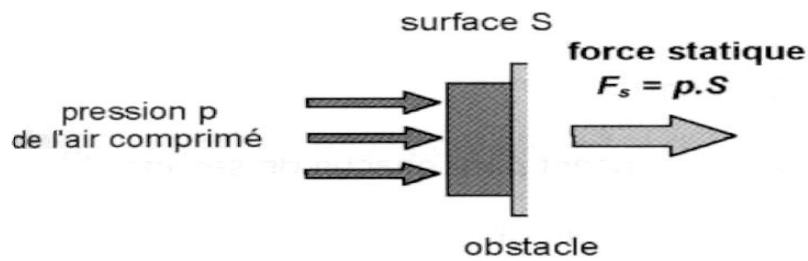


Figure II.4 : force statique dans un

Force vérin.

En faisant

agir l'air

comprimé sur une face immobile, on obtient une force statique  $F_s$  proportionnelle à la pression  $p$  et à sa surface d'action  $S$ .

- **Force dynamique :**

Si la face est mobile en translation, la force dynamique  $F_d$  obtenue pendant le mouvement est plus faible car elle dépend des forces qui s'opposent à son déplacement : force liée à la pression opposée (dite contre-pression), force de frottement, force d'inertie.

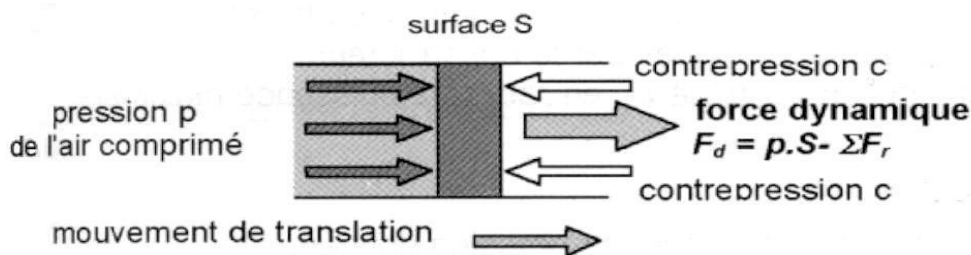


Figure II.5 : force dynamique dans un vérin.

La force statique  $F_s$  ne pose pas de problème de calcul puisque toutes les variables sont connues, du moins pour le vérin à double effet. Pour que le vérin soit exploitable, il suffit que sa force statique  $F_s$  soit supérieure à la charge statique  $C_s$  opposée (force de blocage ou de serrage) :

$$\text{Force statique } F_s > \text{Charge statique } C_s$$

Il n'en est pas de même de la force dynamique. A défaut de connaître les forces de frottement et d'inertie propres au vérin, on définit son rendement  $\eta$  comme le rapport de la force dynamique sur la force statique. Les mesures montrent que  $\eta$  est compris entre 0,8 et 0,95 suivant le type de vérin, ses dimensions, la pression et le fonctionnement à sec ou lubrifié. On

peut donc, faute de connaître le rendement exact du vérin, estimer la force dynamique en prenant pour  $\eta$  la valeur minimum de 0,8.

D'où : Force dynamique  $F_d = \text{Force statique } F_s \times 0,8$ .

Pour que le vérin ait un comportement acceptable, il faut que sa force dynamique  $F_d$  soit supérieure à la charge dynamique  $C_d$  opposée (force dynamique résistante) :

Force dynamique  $F_d > \text{Charge dynamique } C_d$ .

- **Taux de charge  $t$  :**

Pour être certain d'utiliser le vérin dans de bonnes conditions, on définit le taux de charge  $t$ . C'est un paramètre qui tient compte à la fois des effets de la contre-pression et des frottements internes ; son emploi élimine les risques de broutements.

$$\text{Taux de charge } t = \frac{F_{charge}}{F_s}$$

Avec  $F_{charge}$  : effort à vaincre pour déplacer la charge ;

et  $F_s$  : poussée théorique (P.S.)

En pratique :  $0,5 \leq \text{taux de charge } t \leq 0,75$ .

- **Course du vérin :**

La course est déterminée en fonction de la distance de déplacement du mobile. Il convient toutefois de tenir compte des deux paramètres que sont la flexion et le flambage.

✚ **Flexion de la tige du piston** : La flexion est provoquée par l'application d'une charge  $W_s$  perpendiculaire à la tige,

✚ **Flambage** : Il est provoqué par l'application d'une charge dans l'axe de la tige, il est déterminé en fonction :

- De l'effort exercé sur la tige ;
- Du diamètre de la tige ;
- De la course.

- **Consommation du vérin :**

Pour remplir la chambre du vérin d'air comprimé, le compresseur doit aspirer un volume déterminé d'air à la pression atmosphérique. Cette quantification du débit est utilisée pour déterminer la taille du distributeur et de l'unité de traitement d'air.

La relation à appliquer est : **Débit aspiré = volume de la chambre \* pression \* nombre de cycle**

Avec :

Débit aspiré en  $\text{cm}^3$  (ANR : atmosphère normale de référence) par unité de temps (min ou h)

Volume de la chambre en  $\text{cm}^3$  ;

Pression en bars ;

Nombre de cycles allé et retour par unité de temps (min ou h).

- **Dispositif d'amortissement :**

Une masse M en mouvement à une vitesse V possède une énergie cinétique

$$Ec = \frac{1}{2} * M * V^2, \text{ qu'il faut dissiper en fin de course.}$$

Les vérins non amortis doivent être réservés aux faibles courses, ou associés à des amortisseurs extérieurs. Si le vérin arrive en fin de course, il convient de vérifier qu'il peut absorber l'énergie cinétique des masses en mouvement, pour cela il faut utiliser les abaques constructeurs (figure II.6) : On définit le point de rencontre entre la vitesse de déplacement et la masse à déplacer. Pour amortir cette charge, il faudra utiliser le vérin dont la courbe capacité d'amortissement est immédiatement supérieure à celle nécessaire.

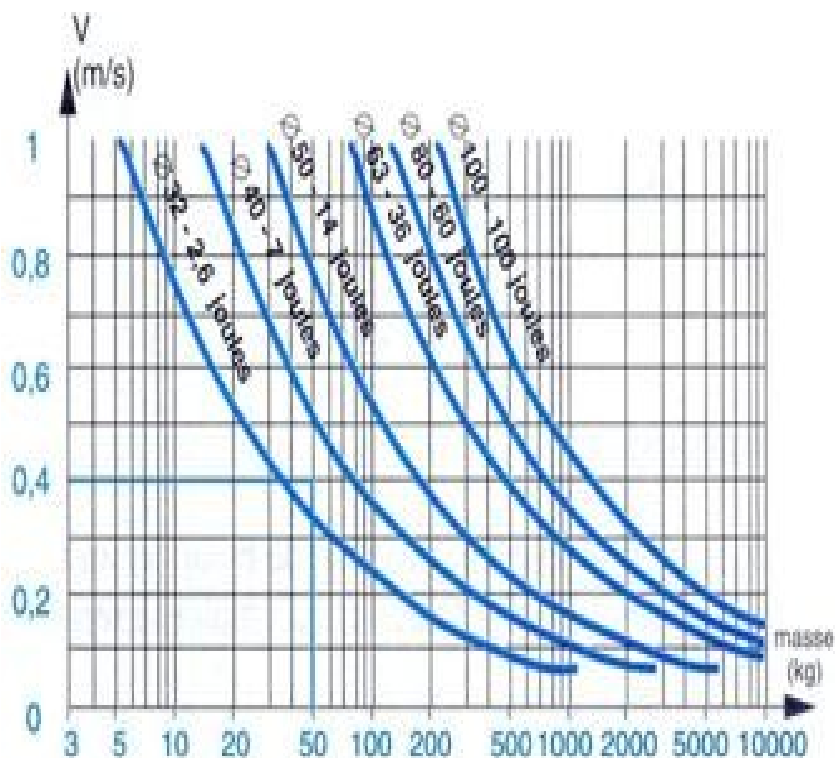


Figure II.6 : Capacités d'amortissement des vérins (d'après Schneider Télémécanique).

Le vérin double effet est utilisé pour porté, le bloc ventouse (figure II.7).

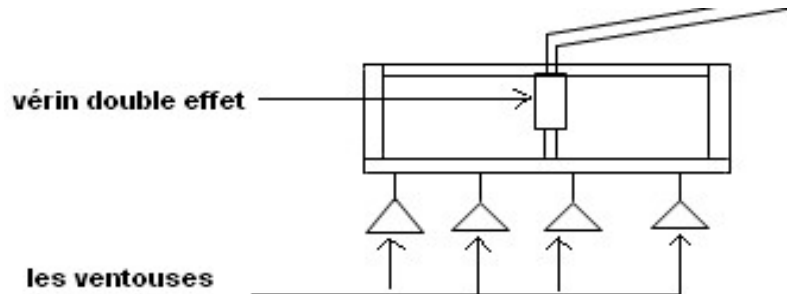


Figure II.7: Vérin porte ventouses.

### II.3.2. Les ventouses :

Ce sont des éléments de préhension souples destinés à être utilisés avec un générateur de vide. Elles permettent de répondre pratiquement à tous les cas d'applications de manutention. Le symbole d'une ventouse est illustré dans la figure suivante (Figure II.8):

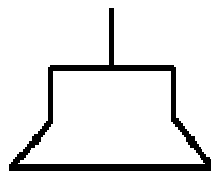


Figure II.8 : Symbole d'une ventouse.

#### II.3.2.1. Domaines d'application :

Les ventouses sont fort utilisées en milieu industriel, nous les trouvons dans les industries de l'agro-alimentaire, du verre, du bois pour la prise de pièces en sortie de presse à injecter, à l'imprimerie pour le brochage et le retournement de feuilles,...

#### II.3.2.2. Générateur de vide ou « Venturi » :

Avec le développement de l'automatisation, saisir une pièce devient un problème courant. La préhension par le vide est souvent utilisée. Comme illustré en (Figure II.9), un étranglement prévu à l'intérieur de l'éjecteur provoque une accélération du flux d'air (P) vers l'orifice R qui entraîne l'air ambiant de l'orifice A et provoque ainsi une dépression.

Basés sur le principe de l'effet venturi ces appareils permettent d'obtenir à partir d'une source d'air comprimé à 5 bars, un vide correspondant à 87 % de la pression atmosphérique.



On utilise souvent un silencieux avec un venturi.

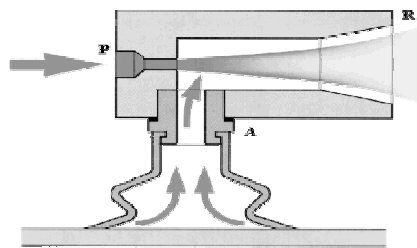


Figure II.9 : Venturi associé à une ventouse.

La figure suivante symbolise un venturi associé à une ventouse (figure II.10) :

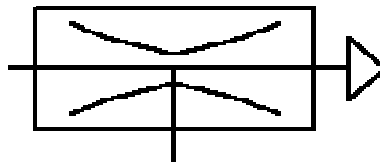


Figure II.10 : Schéma d'un venturi associé à une ventouse.

Le nombre de ventouses nécessaires à une application dépend de la charge à déplacer et de la position de déplacement de la pièce.

Pour un déplacement horizontal avec centrage des éléments de préhension, le nombre de ventouses peut être calculé de la manière suivante :

$$\text{Nombre de ventouses} = \frac{\text{charge ( masse réelle à soulever en Kg)}}{\text{Force de préhension d'une ventouse (en daN)}}$$

Pour un déplacement vertical avec centrage des éléments de préhension, on utilise la formule précédente mais compte tenu de l'effet de glissement, il convient de déclasser dans un rapport de  $\frac{3}{5}$  les forces de préhensions. Une ventouse développe un effort  $F = P_r * S$ , avec :

S : surface de contact avec la pièce saisie ;

$P_r$  : pression relative ( $P_r = P_{atm} - P_i$ ), et  $P_i$  est la pression interne = « depression créée ».

(pour notre système le nombre de ventouse est de quatre).

### II.3.3. Types de moteurs utilisés :

Un moteur est une machine électrique servant à transformer l'énergie électrique en énergie mécanique. L'entraînement du bras chargeur et déchargeur est assuré par des moteurs asynchrones triphasés à cage.

Le changement de vitesse (deux vitesses fixes), est fait par commutation d'enroulement statorique (changement de pôles).

Les avantages qu'il présente sont :

- ✚ Coût de revient bas.
- ✚ De construction simple.
- ✚ Peu d'entretien et de surveillance.
- ✚ Encombrement réduit et simple à commander au démarrage.
- ✚ Il est performant en mode de démarrage direct.
- ✚ Moteur de petites puissances, faible par rapport au réseau.
- ✚ Machine ne nécessitant pas une mise en vitesse progressive.
- ✚ Couple de démarrage est faible.

### **II.3.3.1. Principe de fonctionnement du moteur asynchrone à fréquence fixe :**

Le principe de fonctionnement est le suivant :

- Considérons un moteur asynchrone à l'arrêt, et connectons son stator à une tension alternative triphasée. Des courants alternatifs circulent alors dans ses enroulements et crée un champ tournant à vitesse synchrone.
- Le rotor étant encore à l'arrêt, il est balayé par ce champ variable. Ses spires interceptent un flux variable et sont donc le siège de tensions induites. Comme elles sont court-circuitées sur elles mêmes, les tensions induites créent des courants induits.
- L'interaction de ces courants avec le champ tournant provoque l'apparition d'un couple mécanique. Le rotor démarre et se met à tourner dans le sens du champ tournant.
- Ceci peut s'expliquer par la loi de Lenz, qui dit que tout phénomène induit cherche à s'opposer à la cause qui l'a induit. Dans le cas présents, la cause de l'apparition de courants induits au rotor est la différence de vitesse entre le champ tournant et le rotor.
- A la fin du démarrage, la vitesse du rotor se stabilise à une valeur telle que le couple mécanique développé par les courants induits contrebalance exactement le couple résistant

de la charge. Cette vitesse finale reste donc légèrement inférieure à la vitesse du champ tournant statorique. En effet, si elle était égale, il n'y aurait plus de différence de vitesse, donc plus de courants induits et plus de couple mécanique (figure II.11).

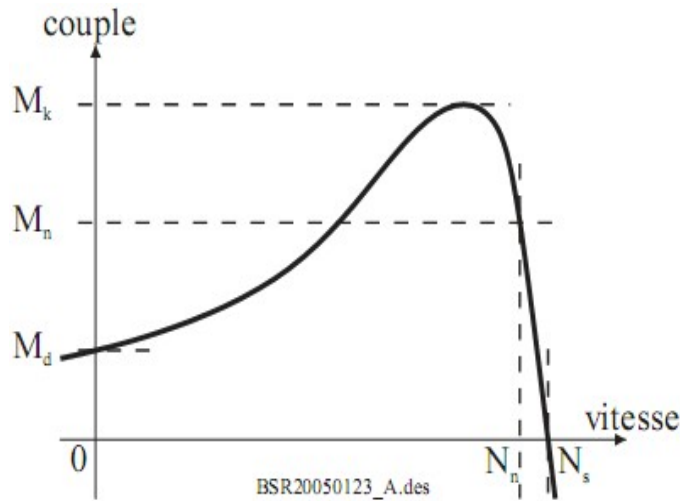


Figure II.11 : Allure du couple d'un moteur asynchrone en fonction de sa vitesse.

La vitesse du champ tournant à fréquence constante est :

$$N_s = 60 * \frac{f}{P} \text{ (tr/mn)} \quad \text{Ou } f = \text{fréquence en [HZ]} ; P = \text{nombre de paires de pôles.}$$

Pour la petite vitesse :  $p = 6$  ; la grande vitesse :  $p = 2$ .

Le glissement ( $g$ ) mesure l'écart entre la vitesse de rotation de la machine et la vitesse de synchronisme.

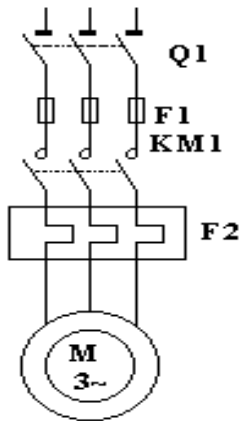
$$g = \frac{N_s - N}{N_s}$$

La vitesse de rotation du moteur est donc liée à la fréquence de l'alimentation électrique  $f$  et au nombre de paires de pôles, comme suit :  $\omega = (1 - g) * \frac{2\pi f}{p} \text{ (rad/s)}$

Le rendement ( $\eta$ ) du moteur, tenant compte des pertes ohmiques et des pertes interne de frottement :  $\eta = \frac{P_{arbre}}{P_{électrique}}$

### II.3.3.1. Démarrage direct d'un moteur asynchrone :

C'est le mode de démarrage le plus simple. Le moteur démarre sur ses caractéristiques "naturelles". Au démarrage, le moteur se comporte comme un transformateur dont le secondaire (rotor) est presque en court-circuit, d'où la pointe de courant au démarrage. Ce type de démarrage est réservé aux moteurs de faible puissance devant celle du réseau, nécessitant pas une mise en vitesse progressive. Le couple est énergique, l'appel de courant est important (5 à 8 fois le



courant nominal).

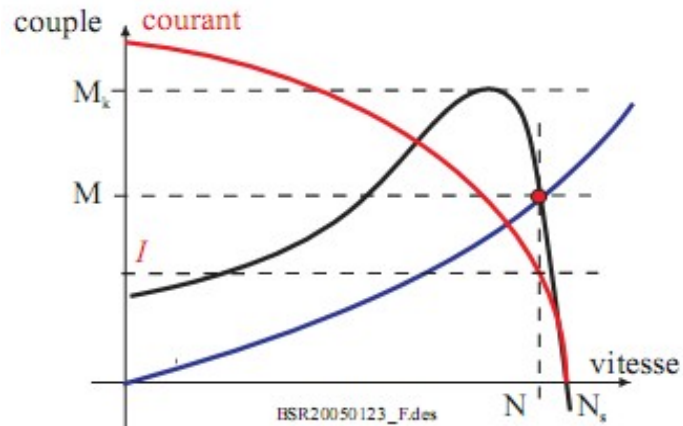


Figure II.12 : Démarrage direct d'un moteur asynchrone.

Malgré les avantages qu'il présente (simplicité de l'appareillage, démarrage rapide, coût faible), le démarrage direct ne convient dans les cas où :

- La puissance du moteur est faible par rapport à la puissance du réseau (dimension du câble)
- La machine à entraîner ne nécessite pas de mise en rotation progressive et peut accepter une mise en rotation rapide
- Le couple de démarrage est élevé

Ce démarrage ne convient pas si

- Le réseau ne peut accepter de chute de tension
- La machine entraînée ne peut accepter les à-coups mécaniques brutaux
- Le confort et la sécurité des usagers sont mis en cause.

### II.3.3.2. Démarrage étoile triangle :

Une solution consiste à profiter du fait que, si l'on couple les phases d'une charge triphasée en étoile plutôt qu'en triangle, on divise par 3 le courant qui les traverse.

En effet, les moteurs asynchrones sont généralement prévus pour fonctionner avec leurs 3 phases en triangle. Il est alors possible de modifier le couplage pour les mettre en étoile. Le couple de démarrage est bien sûr 3 fois plus faible, comme les courants, mais si le moteur n'est pas trop chargé (pompe fonctionnant à vide, ventilateur), il pourra atteindre et dépasser le seuil de décrochage.

A ce moment, il suffit de commuter le moteur à nouveau en triangle.

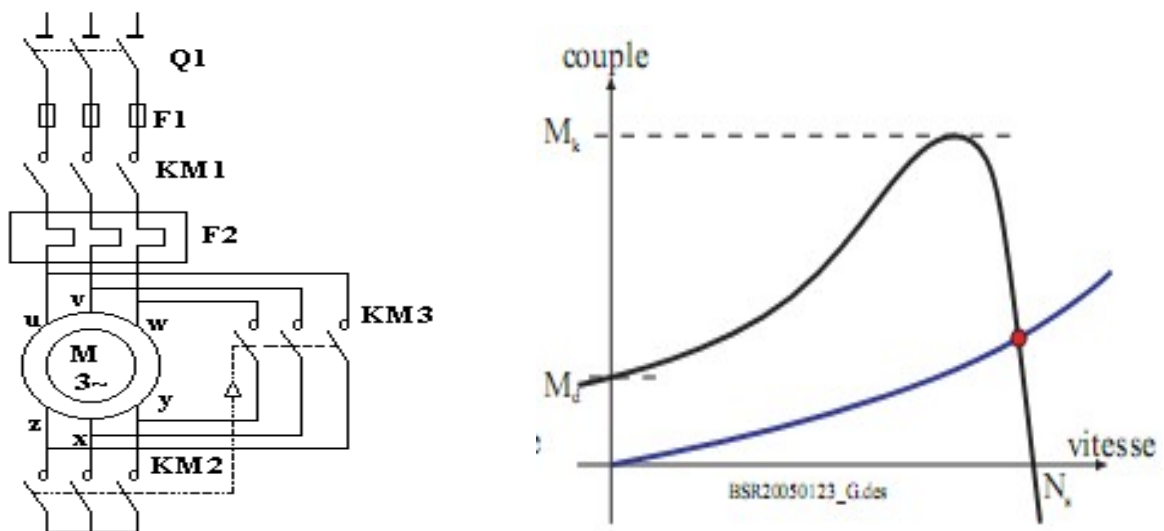


Figure II.13 : Démarrage en étoile triangle d'un moteur asynchrone.

### II.3.3.3. Electro frein :

C'est un organe électromécanique, composé d'une bobine et d'un dispositif de freinage. Lorsque le moteur est en marche, la bobine est désexcitée, c'est-à-dire que l'électrofrein est en repos, lorsqu'on coupe l'alimentation du moteur la bobine excitatrice est alimentée automatiquement et actionne le dispositif de freinage.

## II.4. Capteurs :

### II.4.1. Facteurs qui interviennent dans le choix d'un capteur :

Parmi les principaux et nombreux facteurs qui interviennent dans le choix d'un détecteur, nous citons :

Les conditions d'exploitation, caractérisées par la fréquence de manœuvre, la nature, la masse et la vitesse du mobile à contrôler.

\_L'effort nécessaire pour actionner le contact.

\_La nature de l'ambiance, humide, poussiéreuses, corrosive, ainsi que la température.

\_La place disponible pour loger, fixer et régler l'appareil.

Les capteurs utilisés dans le nouveau système sont : (détecteur de position, interrupteur à lame souple, détecteur de proximité photoélectrique et les détecteurs de niveaux).

#### II.4.3. Détecteur de position : (voir chapitre I).

Pour synchroniser les mouvements du bras chargeur et déchargeur on a utilisé un programmeur à cames.

Les programmeurs à cames utilisés doivent pouvoir s'adapter au cycle de la machine, de ce fait la position angulaire de leurs disques de cames est réglable de 0° à 360°.

Les cames sont en fait constituées de deux demi-disques. Des repères (graduations) facilitent le réglage.

#### II.4.4. Interrupteurs à lame souple :

Un interrupteur à lame souple (I.L.S) est constitué d'un boîtier à l'intérieur duquel est placé un contact électrique métallique souple sensible aux champs magnétiques.

Lorsque le champ est dirigé vers la face sensible du capteur le contact se ferme. La figure suivante représente le symbole d'un I.L.S (Figure II.14):

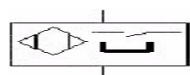


Figure II.14 : Symbole d'un interrupteur à lame souple.

La figure suivante illustre un exemple d'I.L.S sur vérin (figure II.15):

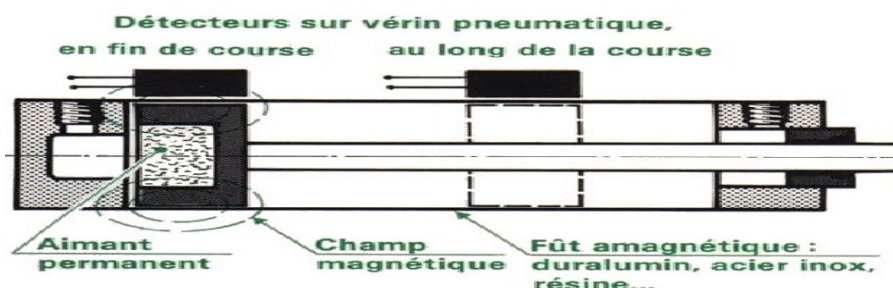


Figure II.15 : Exemple d'I.L.S sur vérin.

On a utilisé quatre ILS au début et à la fin de course des deux vérins porte ventouse.

#### II.4.5. Détecteurs de proximité photoélectrique :

Le capteur de proximité infrarouge (capteur photoélectrique) se compose d'un émetteur de lumière associé à un récepteur. La détection d'un objet se fait par coupure ou variation d'un faisceau lumineux. Le signal est amplifié pour être exploité par la partie commande.

Les récepteurs ont comme élément de base des dispositifs sensibles au rayonnement infrarouge.

La détection est réalisée selon deux procédés :

- Blocage du faisceau par la cible
- Renvoi du faisceau sur le récepteur par la cible

Leur utilisation offre les avantages suivants :

\_Pas de contact physique avec l'objet détecté.

\_Détection d'objets de toutes formes et de matériaux de toutes natures.

Pour réaliser la détection d'objets, 3 systèmes de base sont proposés :

- Le système barrage comporte deux boîtiers, il a une portée de 30m, et ne détecte pas les objets transparents (Figure II.16).

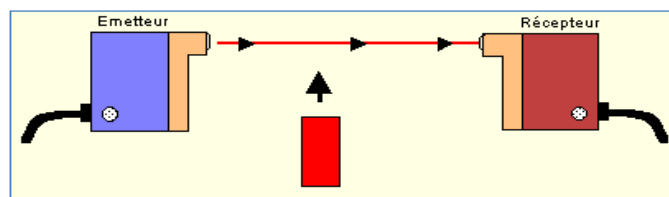


Figure II.16 : Système barrage.

- Le système réflexe ne comporte qu'un seul boîtier, il a une portée de 15m, et ne détecte pas les objets transparents réfléchissants.
- Le système proximité comporte un seul boîtier, sa portée dépend de la couleur de l'objet (clair mieux détectée) et ne détecte pas les objets transparents.

Pour notre système on a utilisé le système barrage :

- Une photocellule pour la détection de pièces à l'entrée du tapis 1;
- Une photocellule pour la détection de pièces à la fin du tapis 1 ;
- Une photocellule pour la détection de pièces à l'entrée du tapis 2;
- Une photocellule pour la détection de pièces à la fin du tapis 2 ;
- Une photocellule pour la détection de pièce dans la matrice ;
- Une photocellule de sécurité du chargement ;
- Une photocellule de sécurité du déchargement.

#### **II.4.6 : Détecteur de niveau :**

Pour la détection de niveau on a utilisé des sondes capacitives : Elles fonctionnent à l'aide d'une électrode plongeante dans le réservoir.

Pour les produits isolants (huile, pétrole...), la sonde est constituée d'une tige métallique isolée du réservoir. Quand la sonde est découverte, le diélectrique est alors l'air ambiant (constante diélectrique = 1). En présence d'un produit isolant, la capacité du condensateur augmente sous l'effet de produits qui possèdent une constante diélectrique supérieure à 1. Cette variation de capacité est traitée pour actionner un relais ou fournir un signal de sortie proportionnel au niveau du produit.

Pour la protection de la pompe principale, de commande et la pompe de rappel, on a équipé le réservoir principal et de fuite d'une sonde capacitive pour la détection du niveau bas (protection des pompes d'un fonctionnement à vide).

#### **II.5. Equipements du chargeur et du déchargeur :**

##### **❖ Fin de courses et capteurs :**

- \_Une fin de course de fin de réglage de chargeur.
- \_Indicateur de position de départ de chargeur.
- \_Indicateur de confirmation de chargement.
- \_Indicateur d'ordre de mise en marche du chargeur et aspiration par ventouse.
- \_Indicateur de petite et grande vitesse de moteur d'entraînement de chargeur.
- \_Interrupteur de confirmation d'adhésion de tôles.



### ❖ Actionneurs et pré-actionneurs :

\_Bobine magnétique pour le désengagement de frein de sécurité du chargeur.

\_Bobine magnétique pour l'embrayage du chargeur.

\_Bobine magnétique pour le freinage du chargeur.

\_Electrovanne pour la mise en service de la ventouse du chargeur.

\_Electrovanne pour la coupure d'air de la ventouse du chargeur.

\_Electrovanne pour le soufflage d'air de la ventouse du chargeur.

\_Deux valves et indicateurs de pression, ils sont manuellement réglables.

\_Un moteur d'entraînement à changement de pôles embrayage/frein.

### III. Poste opérateur :

Tout processus nécessite un système de contrôle de commande permettant à l'homme d'intervenir dans le processus.

Dans le système, chaque élément démarre ou s'arrête par l'intermédiaire d'un commutateur situé sur le poste d'opération. Ce poste comporte des indicateurs montrant l'état de fonctionnement, et un bouton d'arrêt d'urgence arrêtant immédiatement le processus. Ce poste comporte une armoire dite de commande. Pour sélectionner le mode de fonctionnement, manuel ou automatique on doit agir sur le bouton de sélection de mode, qui se trouve sur l'armoire de commande.

#### III.1. Tableau de commande :

Le tableau de commande est un ensemble, comprenant l'appareillage de commande, de contrôle, de réglage et de sécurité du dispositif électrique.

- **Bouton poussoir (BP) :** Arrêt d'urgence, mise en marche et arrêt.
- **Voyants lumineux :** Indiquant l'état de fonctionnement du système.
- **Sélecteur de mode de fonctionnement :** Soit en mode automatique ou manuel.

#### III.2. Mode manuel :

Pour pouvoir commander le système manuellement, on utilise l'armoire de commande qui comporte toutes les commandes (boutons) nécessaires pour un cycle de fonctionnement complet (Descente et montée du coulisseau, avance et recul du chargeur,...).

Elle est utilisée dans le cas d'un essai à vide, dans la phase de test et de simulation des actionneurs ainsi que la remise du système dans les conditions initiales après un dysfonctionnement d'un composant de la partie opérative ou pour travailler entièrement en manuel.

### **III.3. Mode automatique :**

Après avoir sélectionné le mode automatique, l'opérateur doit appuyer sur le bouton poussoir (START) de l'armoire de commande.

## **IV. Partie puissance de la machine :**

Le système sera alimenté par deux réseaux d'énergie pneumatique et électrique.

### **IV.1. Bloc d'alimentation électrique :**

Il reçoit à l'entrée une tension de **380 volts** triphasée alternative, et délivre en sortie trois tensions différentes :

- Tension **380v** alternative triphasé pour alimenter les moteurs asynchrones.
- Tension **30v** continue pour alimenter les bobines d'électro-freins.
- Tension de **220v** alternative pour le circuit de commande (Pré-actionneurs).

### **IV.2. Source et production de l'énergie pneumatique :**

L'air atmosphérique est le gaz généralement utilisé comme fluide moteur dans les équipements pneumatiques par rapport à tout autre gaz.

L'air atmosphérique est à priori doublement économique, d'une part son prélèvement dans l'atmosphère est libre et gratuit et d'autre part après usage, il n'est pas nécessaire de le canaliser et de le ramener dans le réservoir. Il est rejeté à l'atmosphère dès qu'il n'est pas utilisé, sous réserve que sa pollution le permette. La plupart des ateliers industriels sont équipés d'un réseau de distribution d'air comprimé par un compresseur. La pression d'utilisation est souvent de 6 bars environs ( $1 \text{ bars} = 10^5 \text{ Pascals} = 1 \text{ daN/cm}^2$ ).

Avantage de l'air comprimé : énergie propre est facile à mettre en œuvre, puissance développée élevée, grande vitesse, sécurité pour les interventions humaines, simplicité et fiabilité des composants.

#### IV.2.1. Conditionnement de l'air :

##### ▪ Unité FRL (Filtre, Régulateur et Lubrificateur) :

Avant d'utiliser l'air, il faut le filtrer, l'assécher, le graisser et réguler sa pression. Ainsi, avant chaque système automatisé de production, on place une unité de conditionnement FRL qui adapte l'énergie pneumatique au système. Cette unité FRL est constituée d'un Filtre, d'un **mano Régulateur** et d'un **Lubrificateur** (figure II.17).

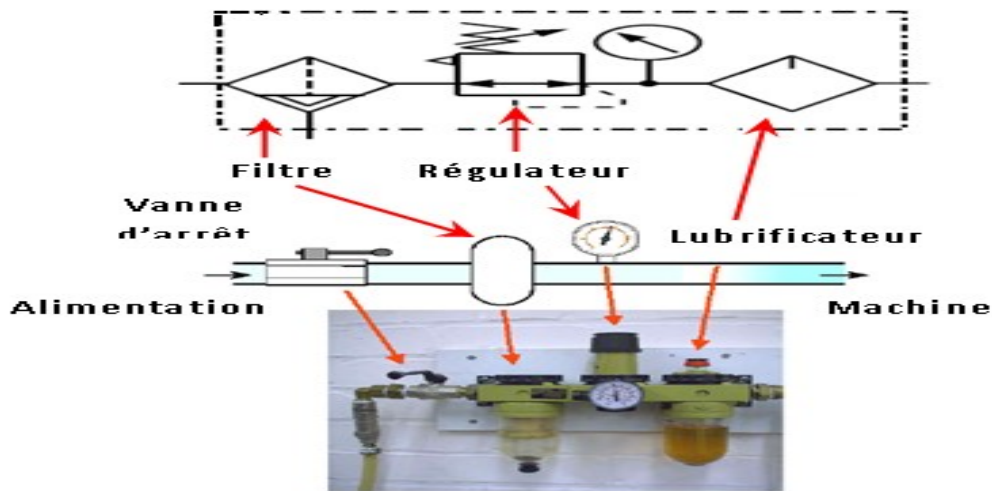


Figure II.17 : Unité de conditionnement et ses composants.

- **Le filtre** : Sert à assécher l'air et filtrer les poussières.
- **Le mano régulateur** : Sert à régler et réguler la pression de l'air.
- **Le lubrificateur** : Sert à éviter la corrosion et à améliorer le glissement.

#### V. Mise en marche du chargeur :

Il faut d'abord faire les vérifications préliminaires à la mise en marche :

- 1- S'assurer que le chargeur est placé en face de la presse et qu'il est bien immobilisé au niveau de sa charnière de fixation sur le montant de la presse.
- 2- Vérifier la pression du régulateur d'alimentation en air du circuit pneumatique.
- 3- La presse doit se trouver en position initiale de repos.
- 4- Le déchargeur doit se trouver en position initiale de repos.

Une fois que la vérification est faite, une feuille est présentée sur le tapis 1.

-Le vérin porte-ventouse descend, la feuille est aspirée. Après écoulement du temps d'aspiration le vérin monte, l'ordre de marche est donné au chargeur. La feuille est libérée lorsque le bras chargeur atteint la fin de course de recul. Ce dernier s'arrêtera de lui-même après avoir rejoint sa position initiale.

#### **VI. Mise en marche du déchargeur :**

-Le bras du chargeur est à sa position initiale de repos.

-A la fin du cycle de la presse, le coulisseau se trouve au point mort haut.

-Le bras du déchargeur avance dans le sens de la presse. La feuille est aspirée lorsque le bras déchargeur se trouve à la fin de course d'avance et elle sera libérée lorsque le bras déchargeur atteint la fin de course recul. Ce dernier s'arrêtera de lui-même après avoir rejoint sa position initiale. Quand la feuille est libérée par le bras déchargeur, elle sera transportée par le tapis d'évacuation 2.

#### **VII. Conclusion :**

Dans ce deuxième chapitre, nous avons mené une étude approfondie sur la machine réadaptée de découpage et poinçonnage des contre portes frigorifiques. Pour comprendre plus précisément le fonctionnement automatique du processus, c'est-à-dire l'interaction entre la partie commande et la partie opérative et pour développer une solution de conduite programmable, la modélisation de ce cycle s'avère nécessaire.

En effet après l'élaboration d'un cahier des charges qui comprend tous les aspects fonctionnels du processus, nous allons passer au prochain chapitre à la modélisation du cahier des charges à l'aide du GRAFCET.

## **I. Introduction :**

Une conception d'un système automatisé industriel comporte un cahier des charges, établi en collaboration avec les différents services utilisant ce système. Outre les contraintes techniques, il comporte des instructions impératives reliant la partie commande à la partie opérative, ainsi que le dialogue avec l'opérateur.

Avec l'arrivée des nouvelles technologies et l'accroissement de la complexité des systèmes industriels automatisés, une définition précise des spécifications fonctionnelles qui régissent le comportement du système s'impose et ceci indépendamment de la matérialisation technologique retenue : mécanique, pneumatique et électronique câblée ou programmée.

Les automaticiens utilisent plusieurs outils de description, ceux établis par les chercheurs « réseaux de Pétri » s'appuient sur d'importants travaux théoriques. D'autres, mis en œuvre par des industriels « GRAFCET,... ».

## **II. Généralité sur le GRAFCET :**

### **II.1. Définition du GRAFCET :**

Le GRAFCET (graphe de commande étapes transitions) est un outil graphique de représentation du cahier des charges d'un automatisme séquentiel. Il est à la fois simple à utiliser et rigoureux sur le plan formel. Il est basé sur les notions d'**étapes** auxquelles sont associées des **actions** et des **transitions** auxquelles sont associées des **réceptivités**. Il décrit les ordres émis par la partie commande vers la partie opérative en mettant en évidence les actions engendrées et les événements qui les déclenchent. Cette représentation est étroitement liée à la notion d'évolution du processus.

Le GRAFCET permet de construire des modèles ayant une structure graphique à laquelle on associe une interprétation (elle correspond à l'aspect fonctionnel du grafcet).

### **II.2. Aspect structurel du GRAFCET :**

#### **II.2.1. l'étape :**

Une étape correspond à une phase durant laquelle on effectue une action pendant une certaine durée (même faible mais jamais nulle). L'action doit être stable, c'est-à-dire que l'on fait la même chose pendant toute la durée de l'étape.

On représente chaque étape par un carré, l'action est représentée dans un rectangle à droite. L'entrée se fait par le haut et la sortie par le bas. On numérote chaque étape par un entier positif, mais pas nécessairement croissant par pas de 1, il faut simplement que jamais deux étapes différentes aient le même numéro (figure III.1).

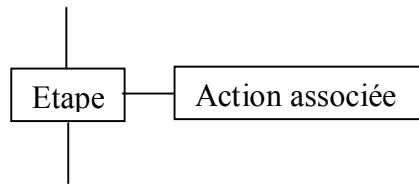


Figure III.1 : Une étape du grafcet et son action.

Une étape peut être active ou inactive (un point à l'intérieur d'une étape signifie qu'elle est active).

### II.2.2. L'étape initiale :

Représente une étape qui est activée au moment de la mise en énergie de la partie commande.

Elle est représentée par un double carré (figure III.2).



Figure III.2 : Etape initiale.

### II.2.3. Transition :

Une transition est représentée par un trait horizontal placée entre une étape d'entrée, située en amont, et une étape de sortie située en aval.

Le passage du système d'un événement au suivant, respectivement d'une étape à l'étape suivante, correspond au franchissement de la transition.

Une transition représente une, et une seule, possibilité de dévolution.

Une transition est validée lorsque toutes les étapes à partir desquelles la transition peut s'effectuer sont actives (figure III.3).

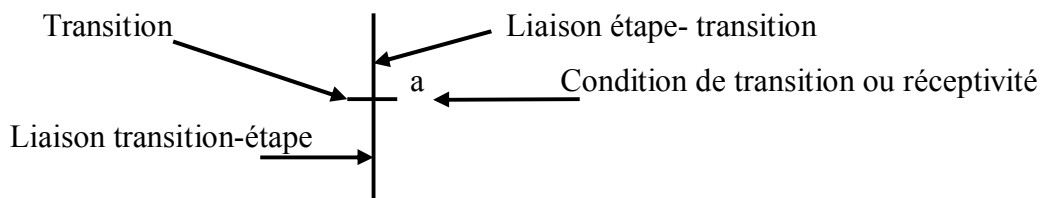


Figure III.3 : Transition.

### II.2.4. Liaison :

Les liaisons orientées relient les étapes aux transitions et les transitions aux étapes.

Elles indiquent les voies suivant lesquelles se font les évolutions.

Elles sont représentées par des lignes verticales et horizontales.

### II.2.5. Action :

Une ou plusieurs actions élémentaires ou complexes peuvent être associées à une étape. Les actions traduisent ce qui doit être fait chaque fois que l'étape à laquelle elles sont associées et moyennant ou non certaines conditions supplémentaires.

Les actions associées à une étape sont inscrites dans un rectangle d'action de façon à mettre en évidence ce qui s'exécute lorsque cette étape est active.

### II.2.6. Réceptivité :

Une réceptivité est associée à chaque transition (l'absence de réceptivité est en fait la réceptivité toujours vrai). C'est une condition qui détermine la possibilité ou non d'évolution du système par cette transition.

Une réceptivité s'exprime comme étant une expression booléenne écrite à l'aide des variables d'entrées  $E_i$ , des variables d'étapes  $X_i$ , des opérateurs logiques et, ou, non ainsi que de l'opérateur à retard " $t1/Xn/t2$ ", auquel on peut rajouter les opérateurs front montant et front descendant (notés respectivement  $\uparrow$  et  $\downarrow$ ). Ces opérateurs permettent d'introduire le concept d'événement. Ils expriment le changement d'état d'une variable booléenne (figure III.4).

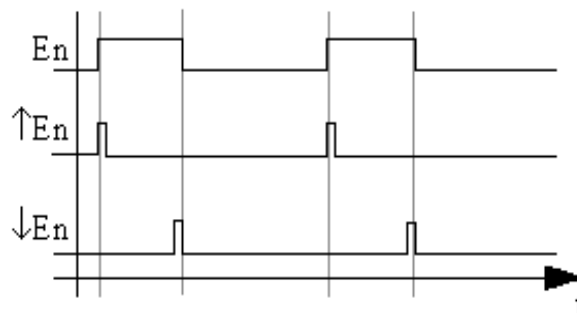


Figure III.4 : Opérateurs Fronts montants et descendants.

### II.3. Règle d'évolution du GRAFCET :

Un grafcet possède un comportement dynamique dirigé par cinq règles, elles précisent les causes et les effets du franchissement des transitions.

La modification de l'état de l'automatisme est appelée évolution, elle est régie par ces 5 règles :

#### ➤ Règle 1 relative à la situation initiale de la partie commande :

La situation initiale d'un grafcet caractérise le comportement initial de la partie commande vis-à-vis de la partie opérative, de l'opérateur et/ou des éléments extérieurs. Elle correspond aux étapes actives au début du fonctionnement.

On appelle début du fonctionnement le moment où le système n'a pas besoin de souvenir de ce qui c'est passé auparavant.

➤ **Règle 2 relative au franchissement d'une transition :**

Une transition est dite validée lorsque toutes les étapes amont c'est-à-dire immédiatement précédentes reliées à cette transition sont actives.

Le franchissement d'une transition se produit :

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

➤ **Règle 3 relative à l'évolution de la partie commande :**

Le franchissement d'une transition entraîne simultanément l'activation de toutes les étapes immédiatement suivantes et la désactivation de toutes les étapes immédiatement précédentes.

➤ **Règle 4 relative à l'évolution de la partie commande :**

Plusieurs transitions simultanément franchissables sont simultanément franchies (ou du moins toutes franchies dans un laps de temps négligeable pour le fonctionnement).

➤ **Règle 5 relative à l'activation et à la désactivation simultanée d'une même étape :**

Si au cours du fonctionnement la même étape est simultanément activée et désactivée elle reste active.

#### II.4. Règles de construction d'un grafcet :

❖ On relie étapes et transitions, qui doivent strictement alterner, grâce à des arcs orientés.

Par convention, étapes et transitions sont placées suivant un axe vertical. Les arcs orientés sont de simples traits verticaux lorsque la liaison est orientée de haut en bas, et sont munis d'une flèche vers le haut lorsque la liaison est orientée vers le haut.

❖ Si Plusieurs étapes doivent être reliées vers une même transition, alors on regroupe les arcs issues de ces étapes à l'aide d'une double barre horizontale appelée convergence « en et » (synchronisation) (figure III.5).

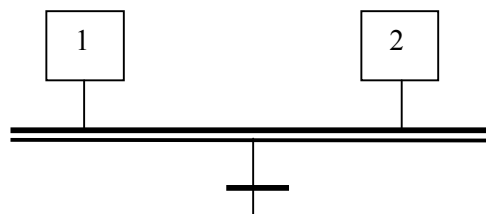


Figure III.5 : convergence en ET.



❖ 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 « en ET » (ou également synchronisation) (figure III.6)

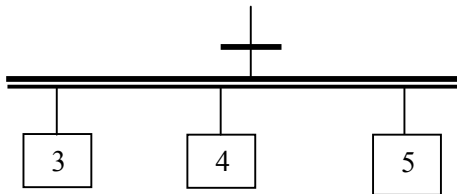


Figure III.6 : divergence en ET.

❖ 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 « en ou » (figure III.7) respectivement divergence « en ou » (figure III.8).

On parle d'étape avale (respectivement d'étape amont) 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 avale à une étape.

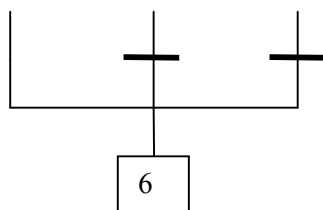


Figure III.7 : convergence en OU.

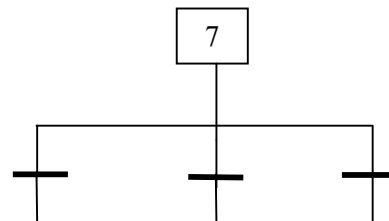


Figure III.8 : divergence en OU.

### III. Niveau d'un Grafcet :

#### ❖ Niveau 1 :

Appelé aussi le niveau de la partie commande, il décrit l'aspect fonctionnel du système et des actions à faire par la partie commande en réaction aux informations provenant de la partie opérative indépendamment de la technologie utilisée, les réceptivités sont décrites en mots et non en abréviations. Nous associons le verbe à l'infinitif pour les actions.

#### ❖ Niveau 2 :

Appelé aussi le niveau la partie opérative, il tient compte de plus de détail de la technologie actionneurs, des pré-actionneurs et des capteurs.

La présentation des actions et des réceptivités est écrite en abrégiation, nous associons une lettre majuscule à l'action et une lettre minuscule à la réceptivité.

### III.1. Mise en équation du Grafcet :

Soit la partie du GRAFCET représentée par la figure (III.9) pour décrire l'activité de l'étape n, On utilise la notation suivante :  $X_n=1$  si l'étape n est active et  $X_n=0$  si l'étape n est inactive. La réceptivité «  $t_n$  », étant une variable binaire, ayant pour valeur :  $t_n=1$  si la réceptivité associée à la transition (2) est vraie et  $t_n=0$  si la réceptivité associée à la transition (2) est fausse.

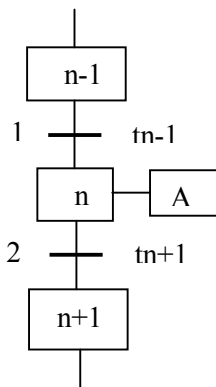


Figure III.9 : modèle simple du GRAFCET.

Soient les variables d'arrêt d'urgence Dur (AUD) et d'arrêt d'urgence doux (AUD) tel

Que :

$AUD = 1$  désactivation de toutes les étapes.

$AUd = 1$  désactivation des actions, les étapes restent actives.

Pour une étape initiale, on définit aussi la variable Init comme suit :

$Init = 1$  initialisation de Grafcet (mode arrêt).

$Init = 0$  déroulement de cycle (mode marche).

La 2ème et la 3ème règle du grafcet permet de déduire les variables qui interviennent dans les équations d'activation et de désactivation de chaque étape. Ces mêmes règles permettent d'écrire :

Pour une étape initiale n :  $X_n = (CAX_n + X_n * \overline{CDX_n} + Init) * \overline{AUD}$ .

$CAX_n = (X_{n-1} * t_{n-1} + Init) * \overline{AUD}$ .

$$CDX_n = X_{n+1} * \overline{Init} + AUD$$

Avec :  $CAX_n$  est la condition d'activation de l'étape n et  $CDX_n$  la condition de désactivation de l'étape n.

Pour une étape non initiale n :

$$X_n = (CAX_n + X_n * \overline{CDX_n}) * \overline{Init} * \overline{AUD}$$

$$CAX_n = X_{n-1} * t_{n-1} * \overline{Init} * \overline{AUD}$$

$$CDX_n = X_{n+1} + Init + AUD$$

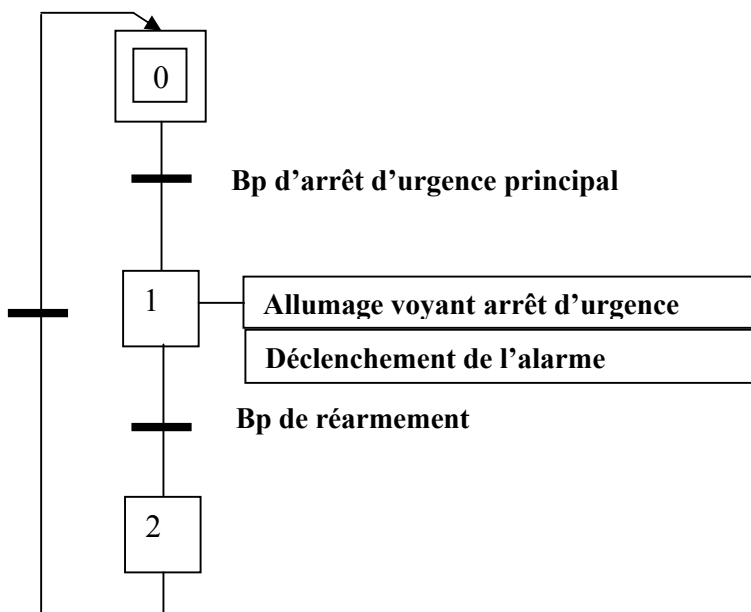
Pour une action :  $A = X_n * AUD$ .

Après l'étude du système à automatiser et avoir identifié les différentes réceptivités ainsi que les différentes actions, le cahier des charges décrit au premier chapitre est modélisé sous forme de grafcet niveaux 1 et 2.

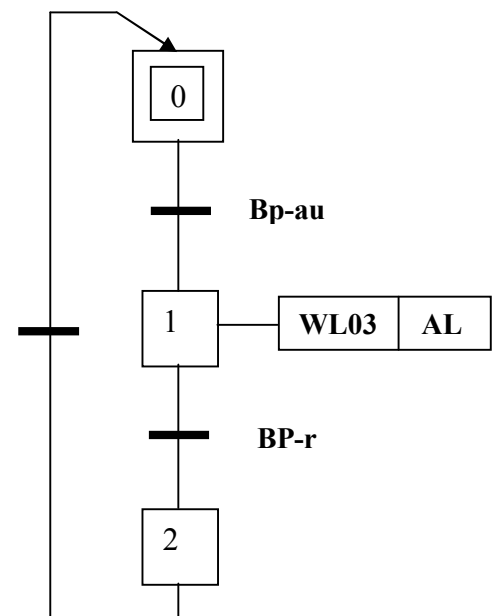
#### IV. Grafcet du fonctionnement de la machine réadaptée (niveau 1 et niveau 2) :

❖ Grafcet de gestion d'arrêt d'urgence de la machine réadaptée :

Grafcet de niveau 1 :

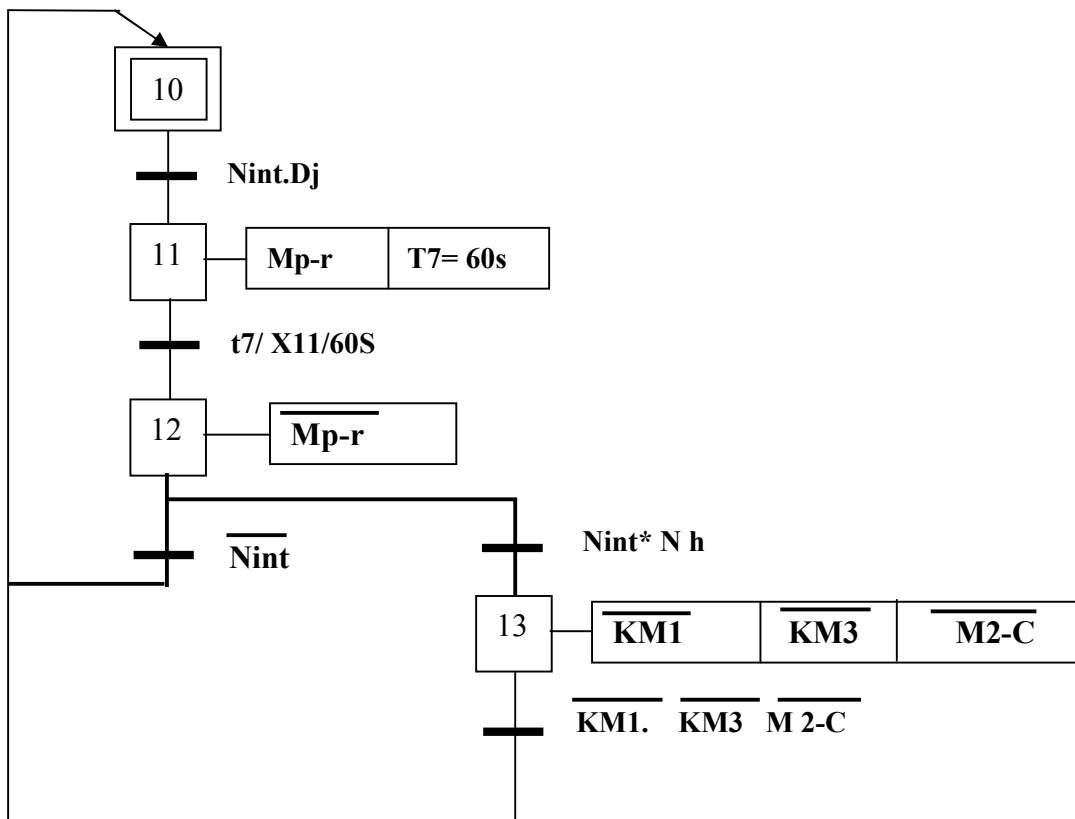
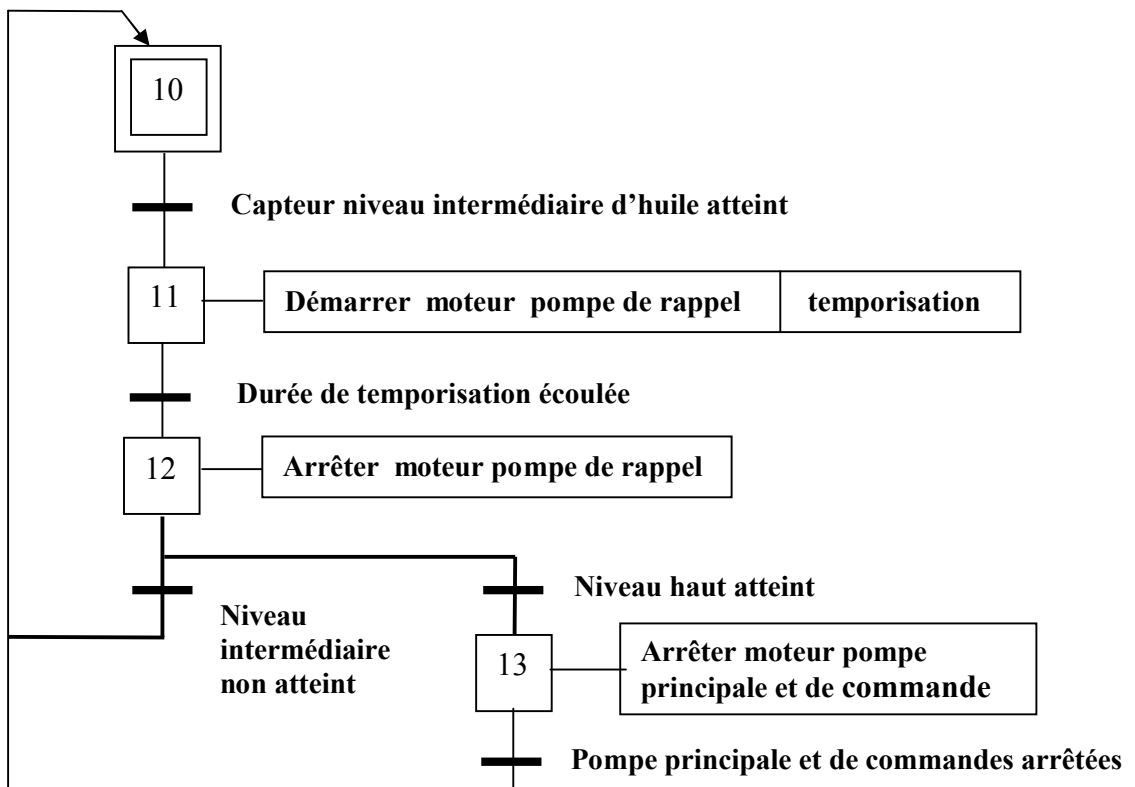


Grafcet de niveau 2 :

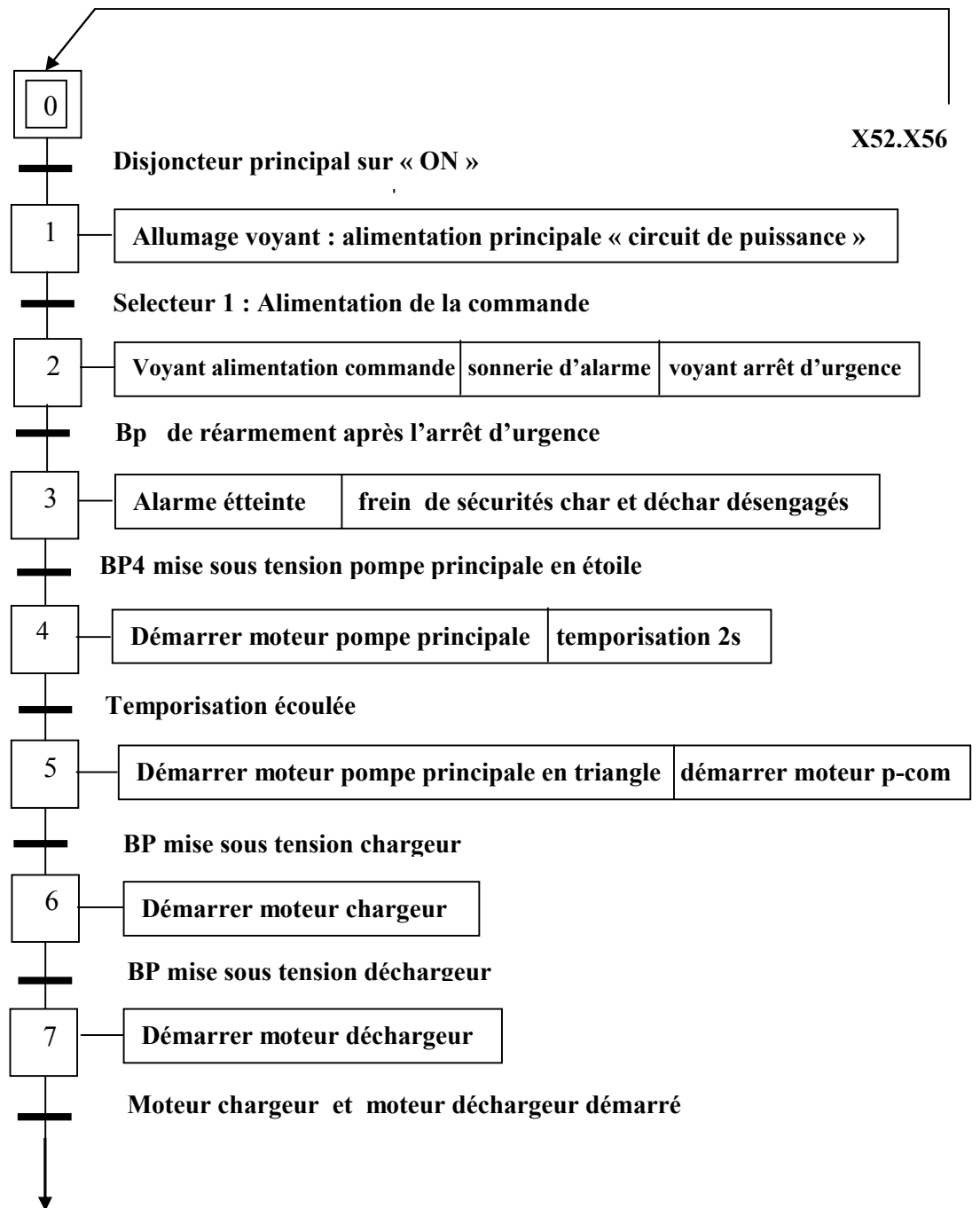


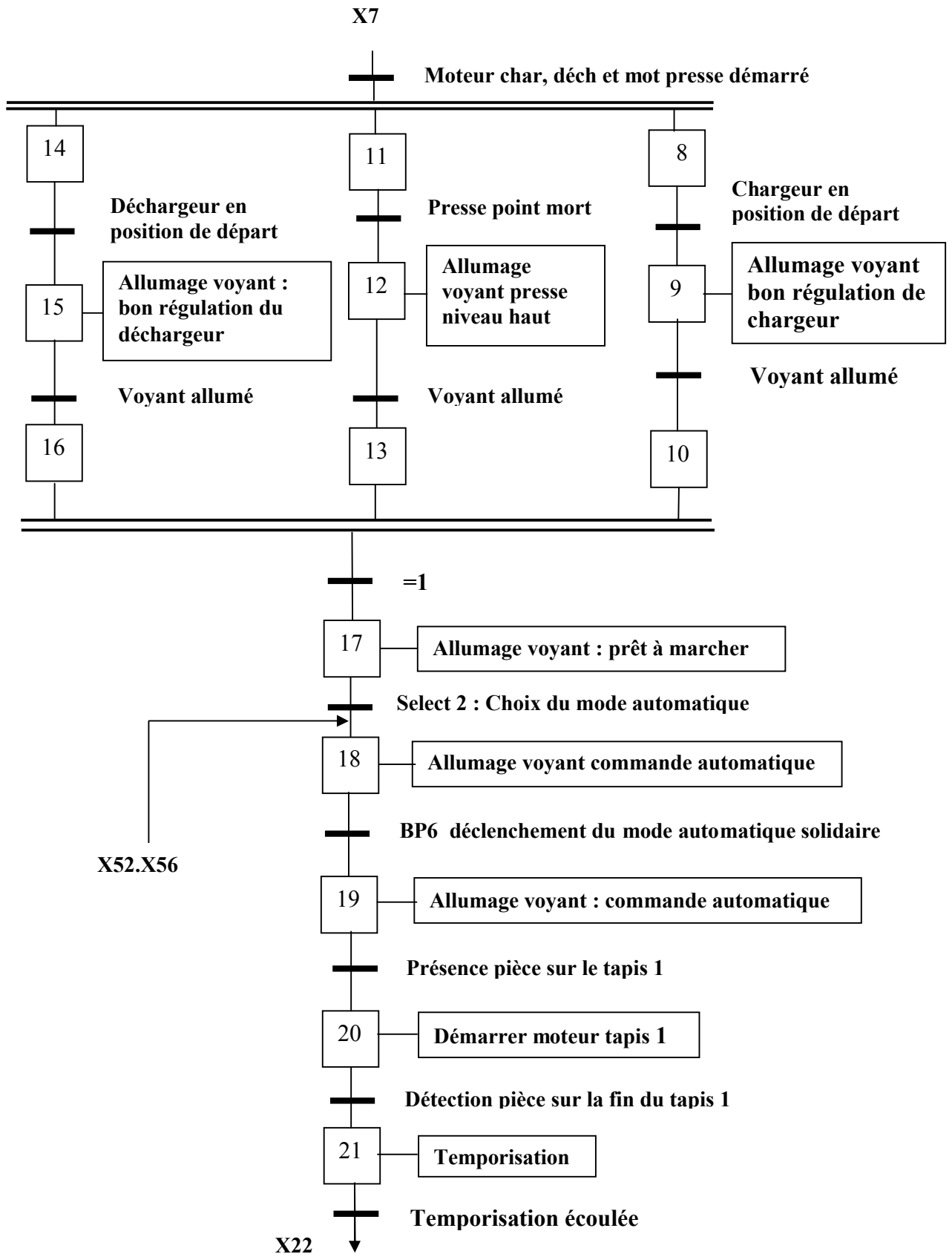
❖ Grafcet de la pompe de rappel d'huile :

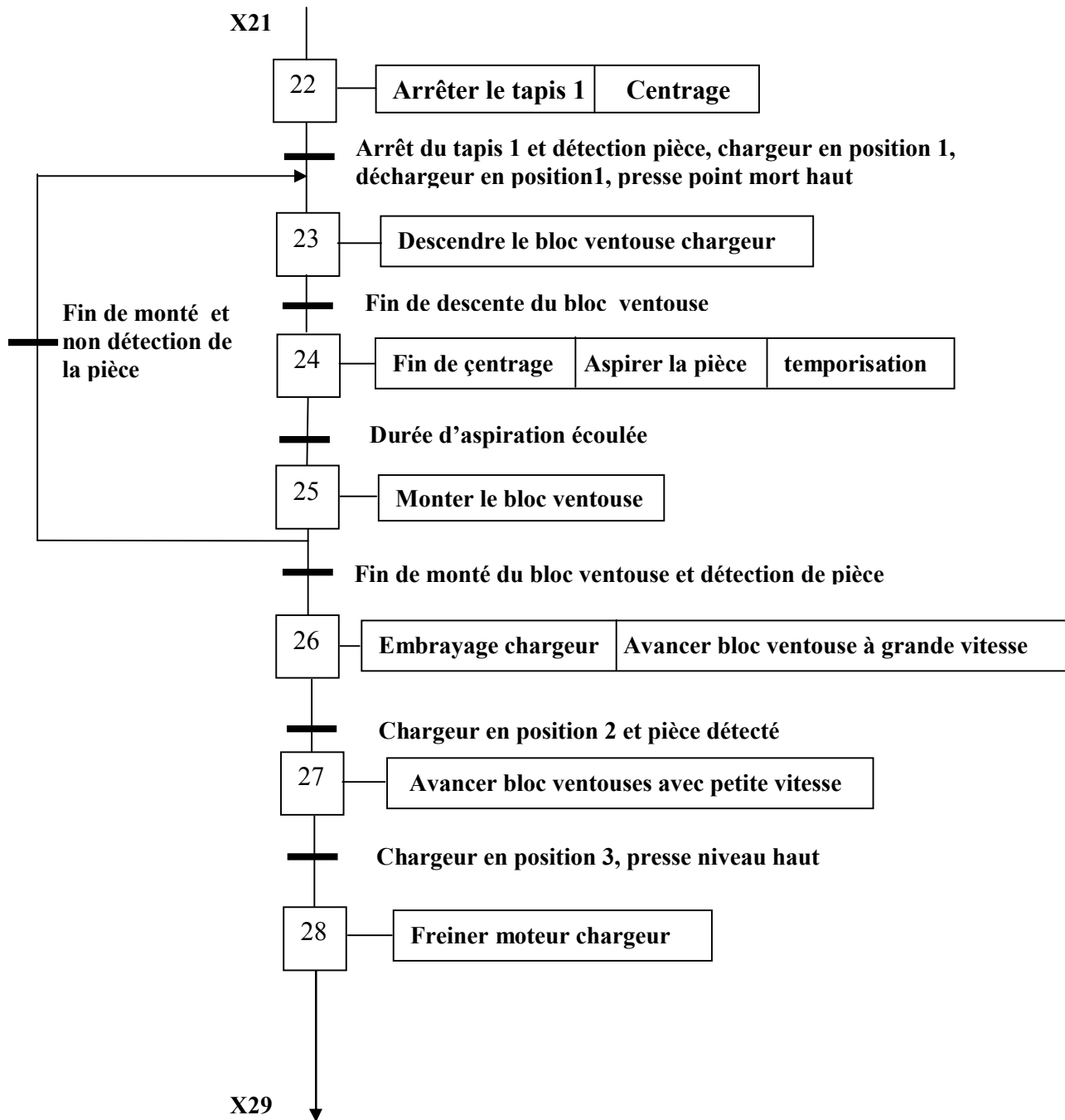
Grafcet niveau 1 et 2 :

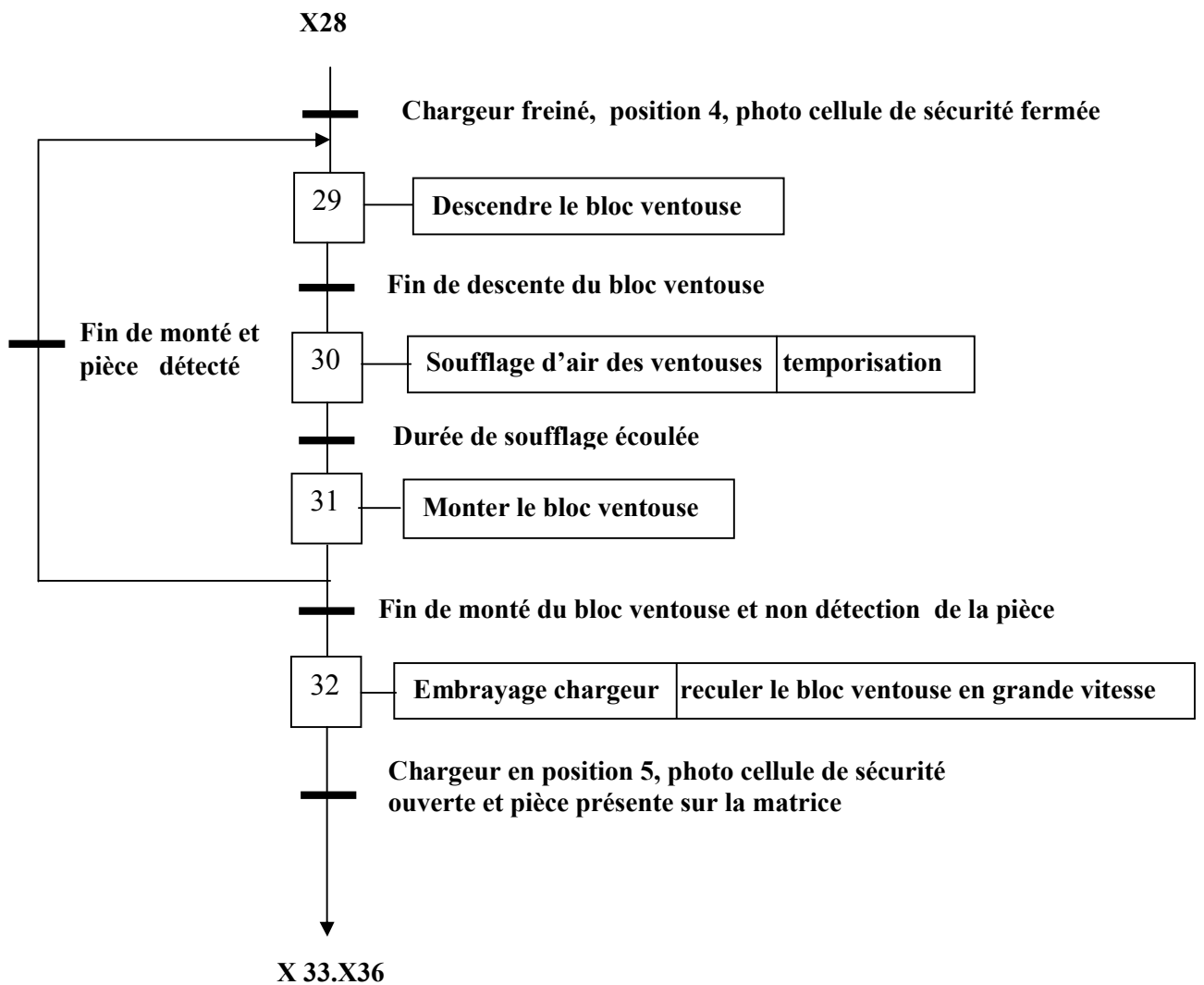


❖ **Grafcet de la machine réadaptée :**

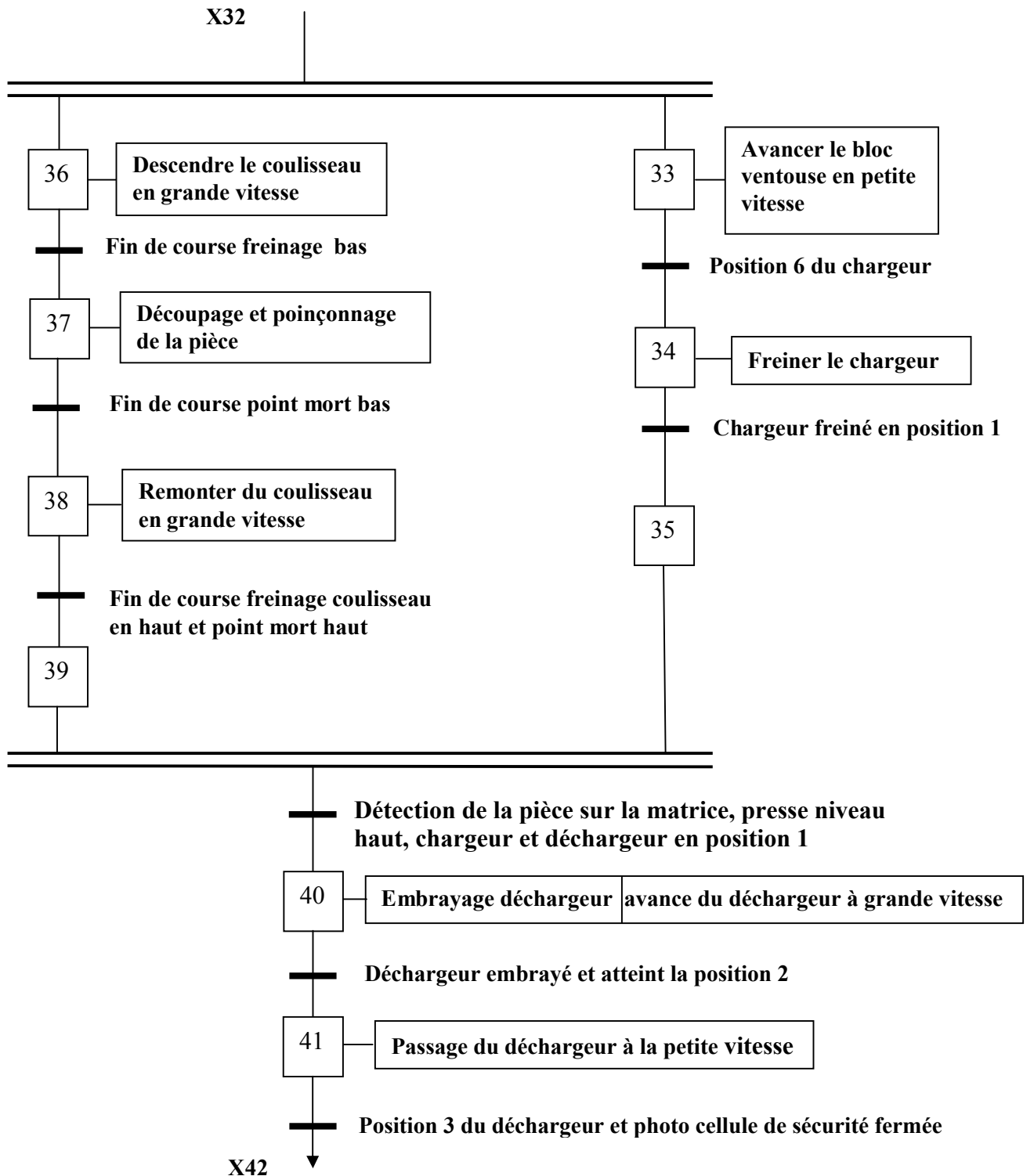


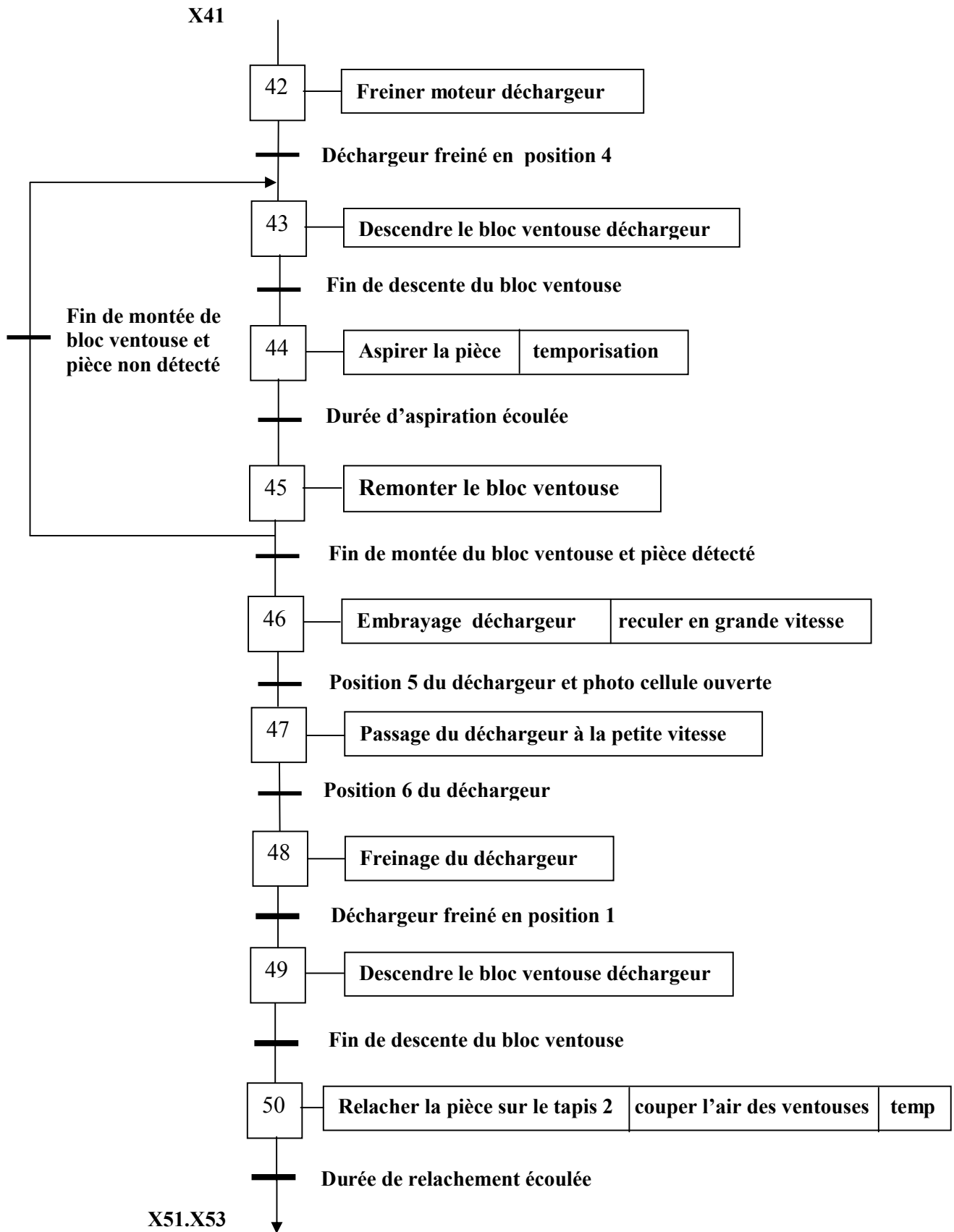


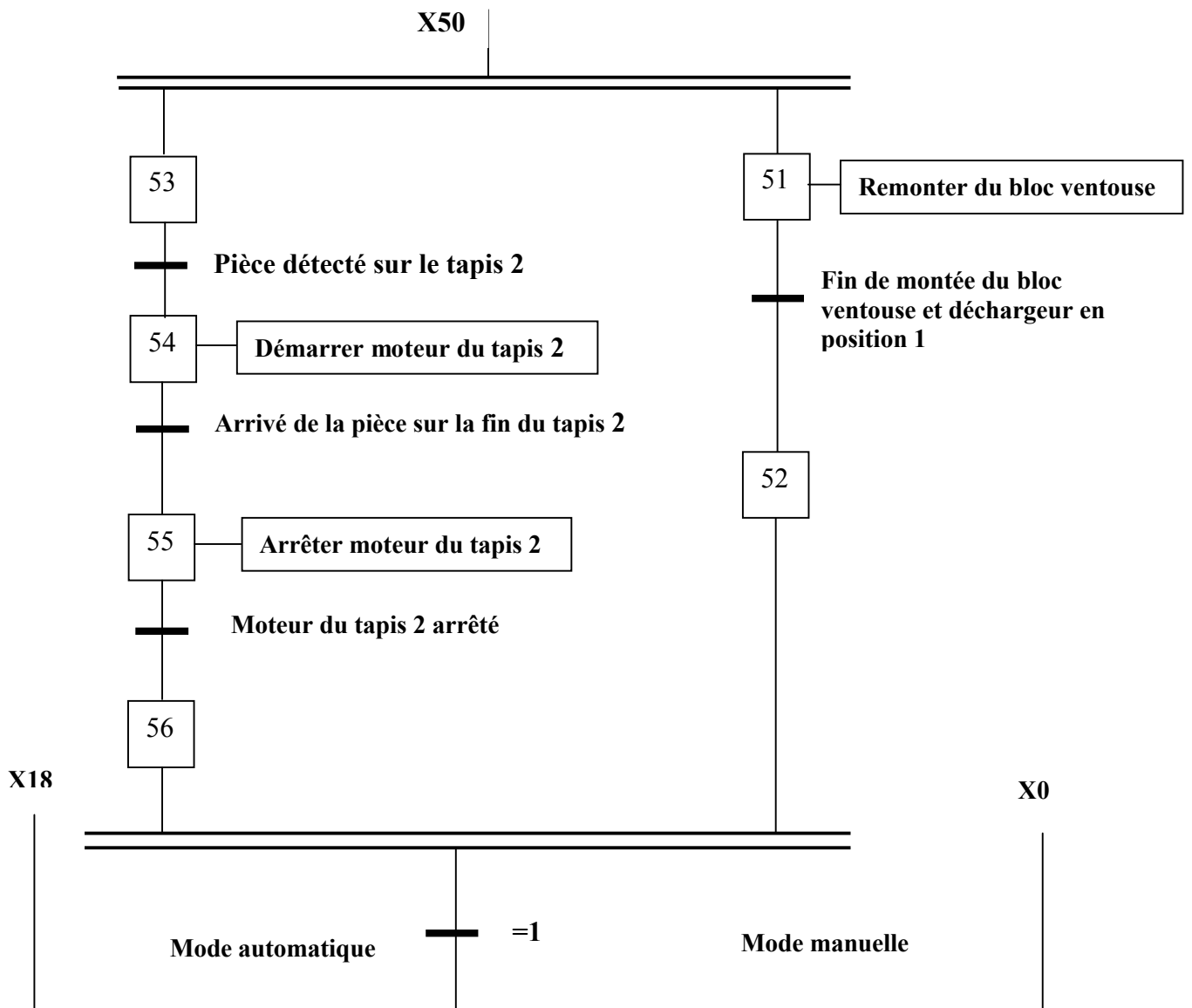




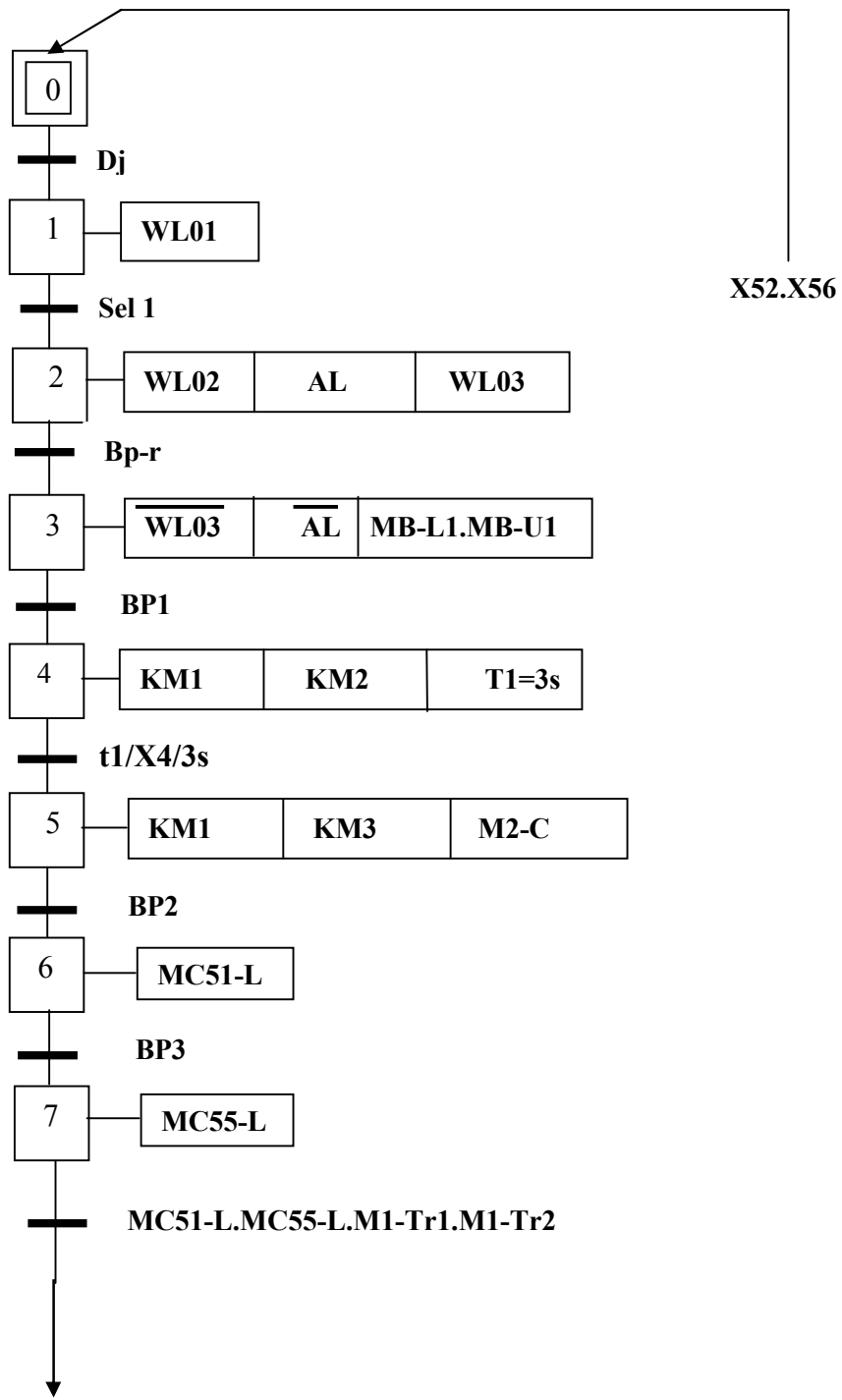




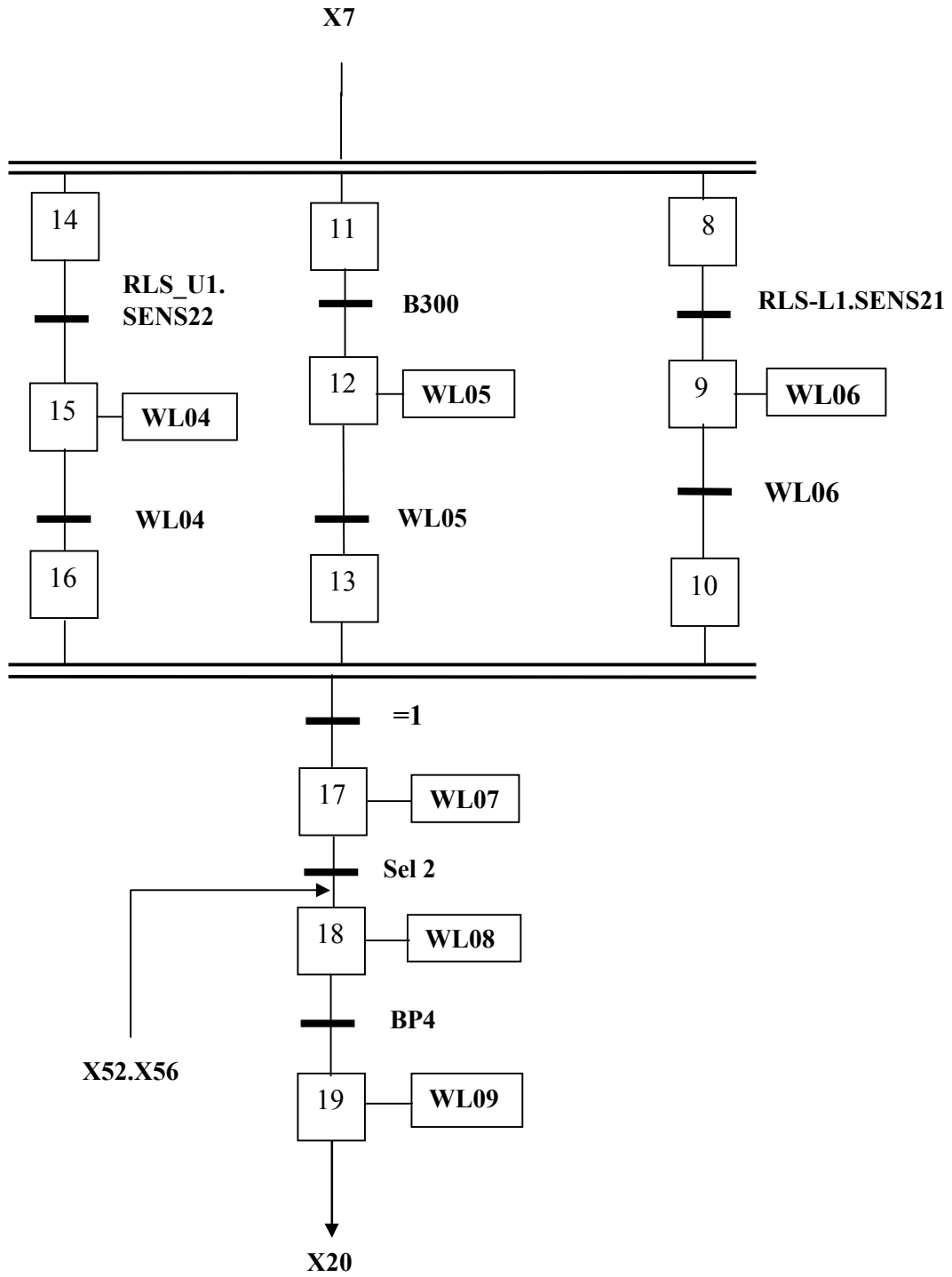


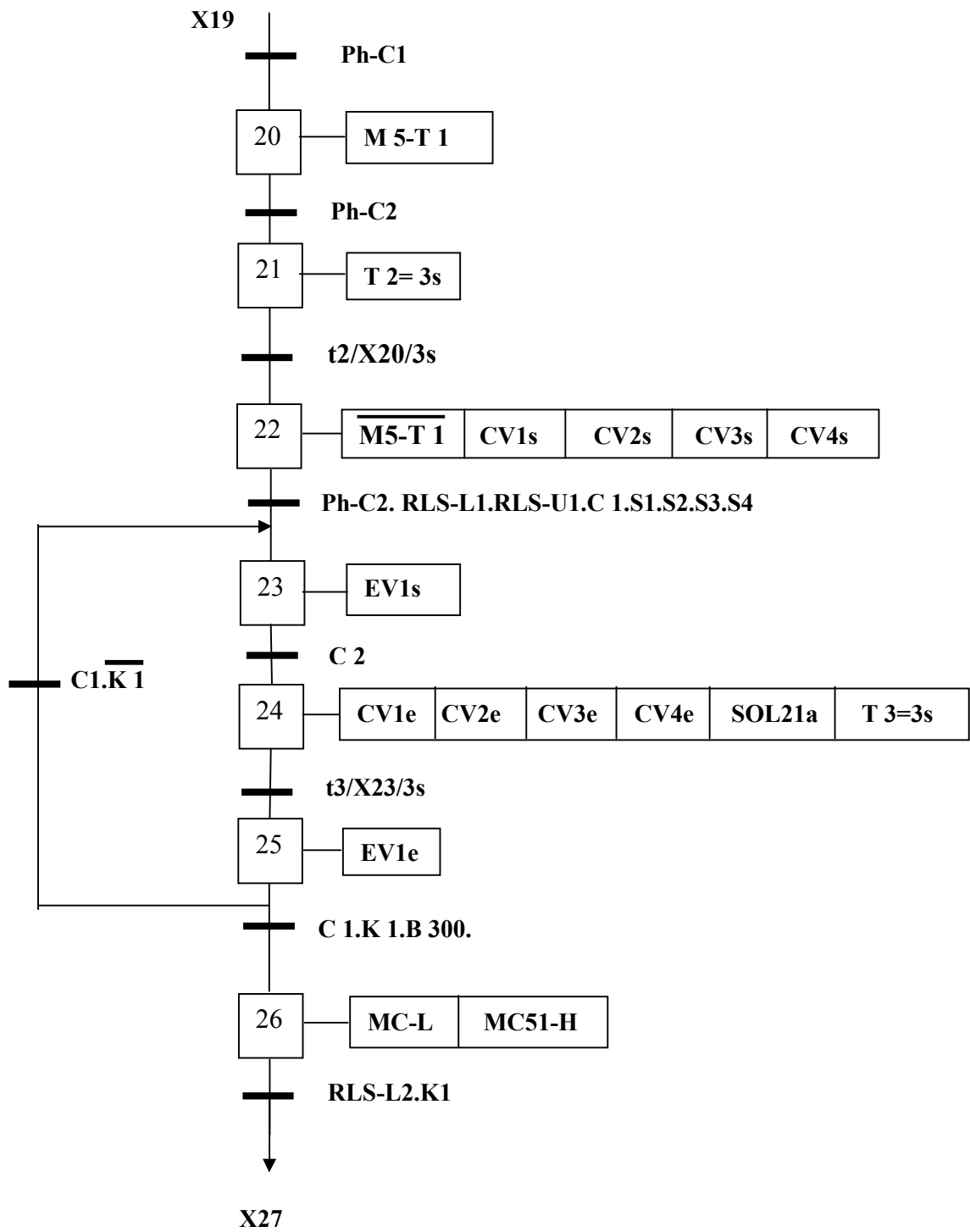


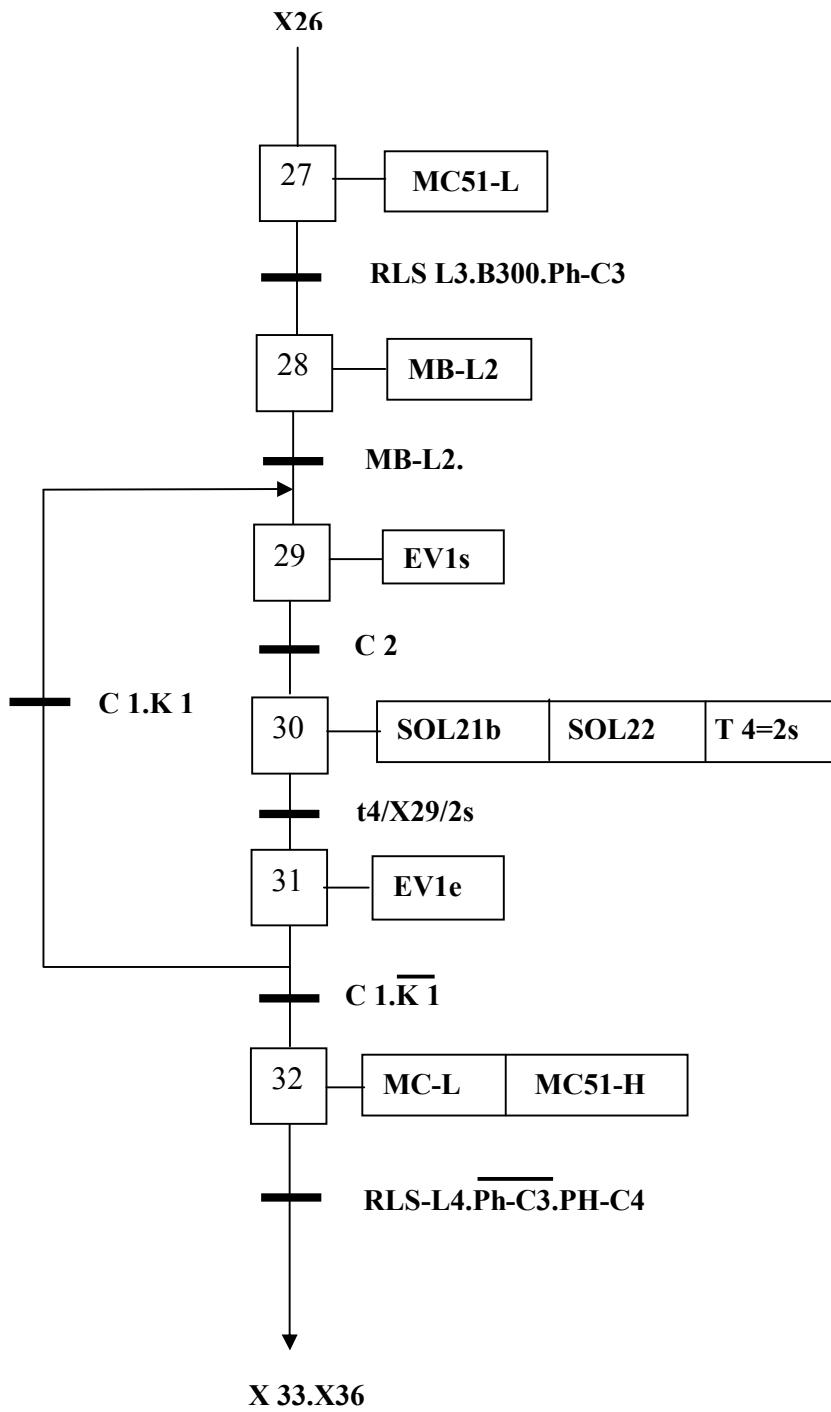
Grafcet niveau 2 :

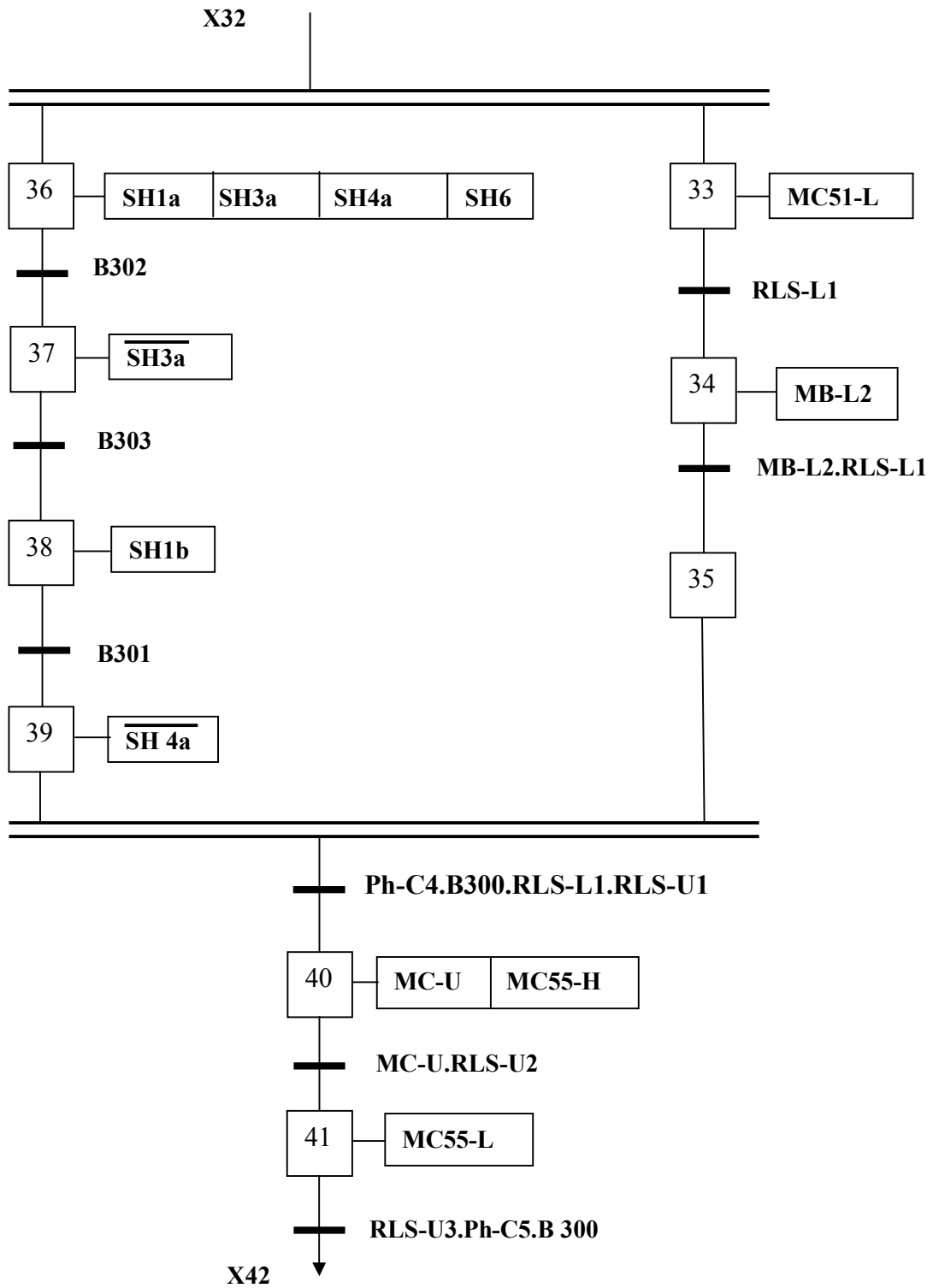


X8.X11.X14

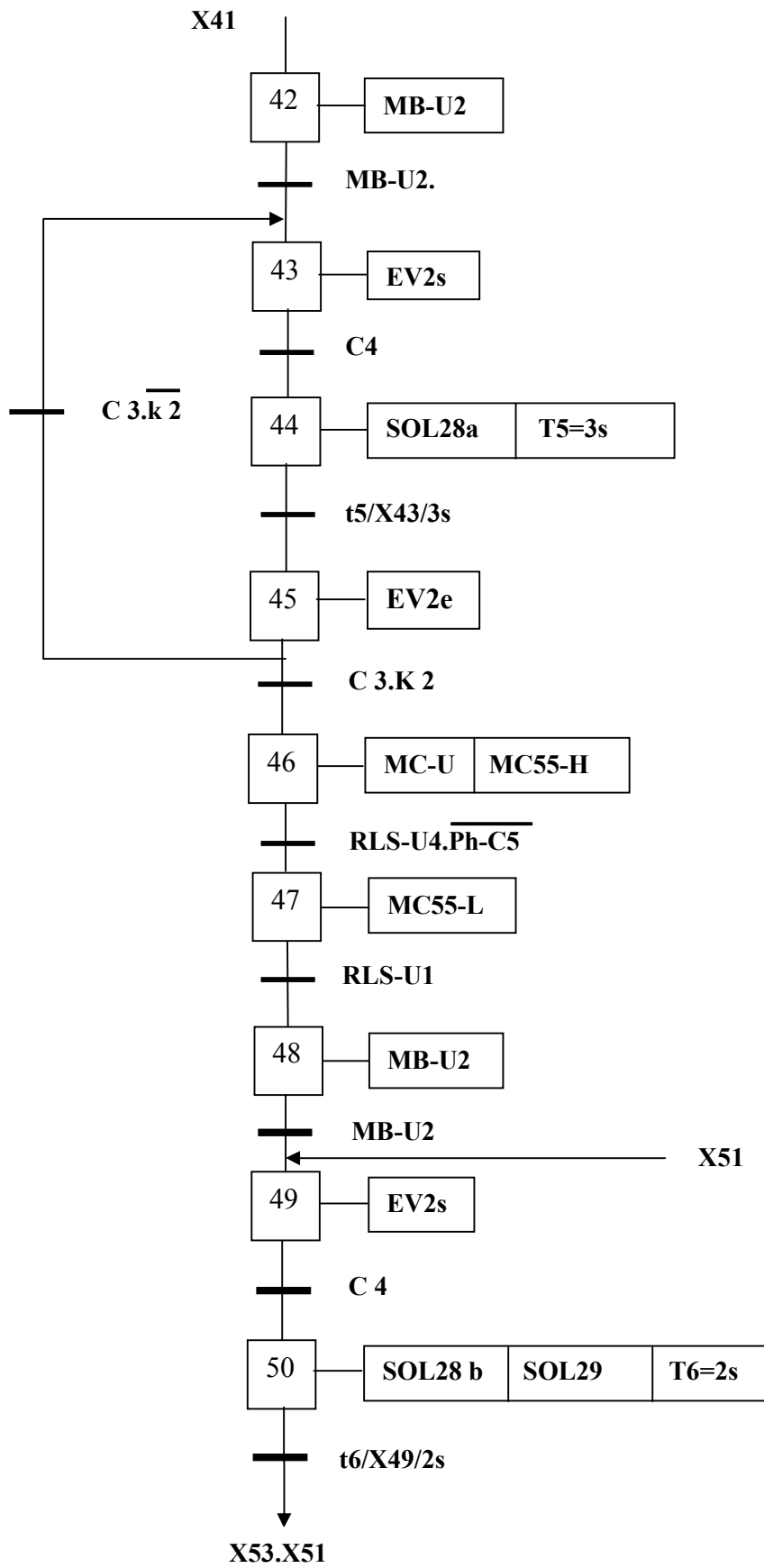


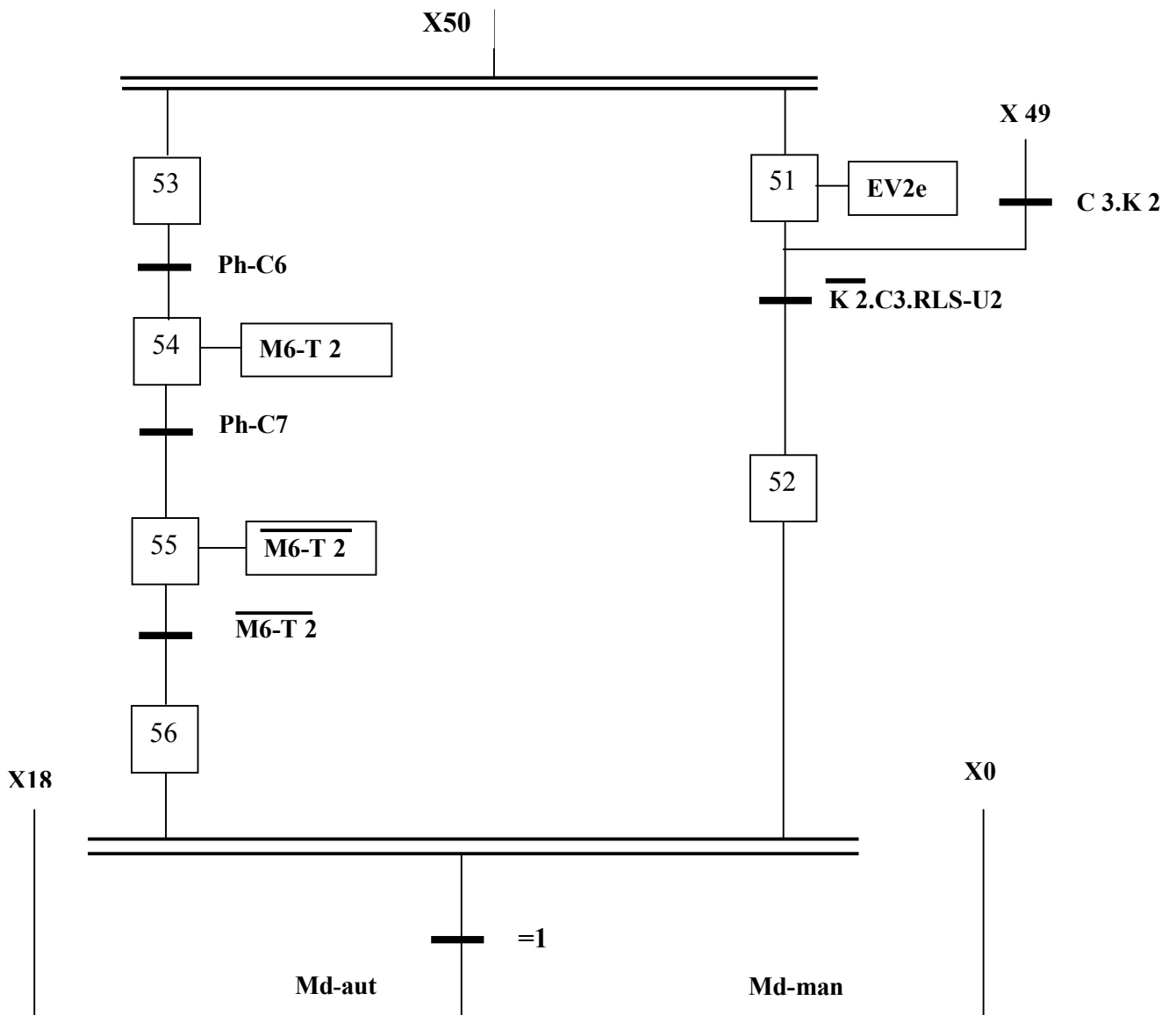








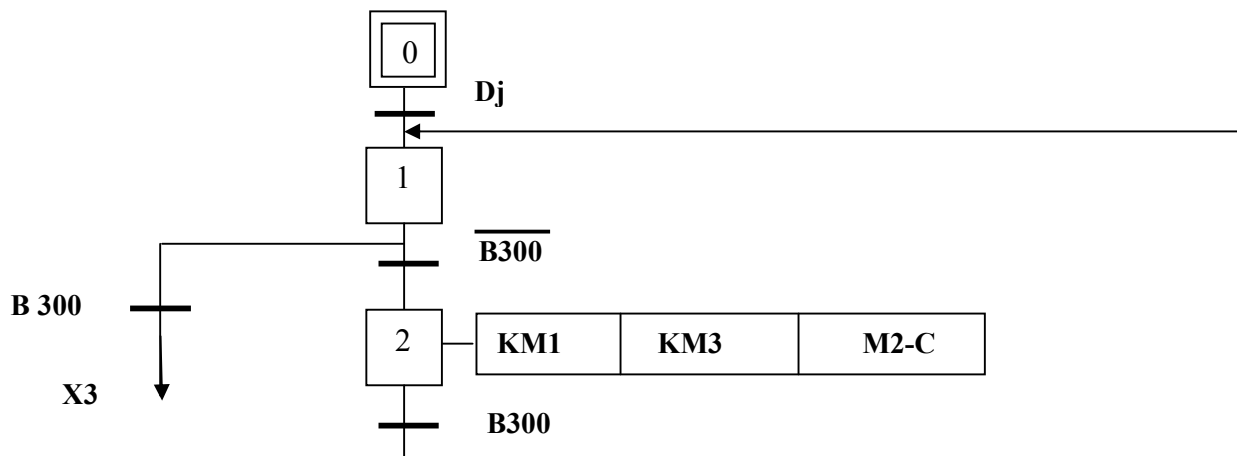
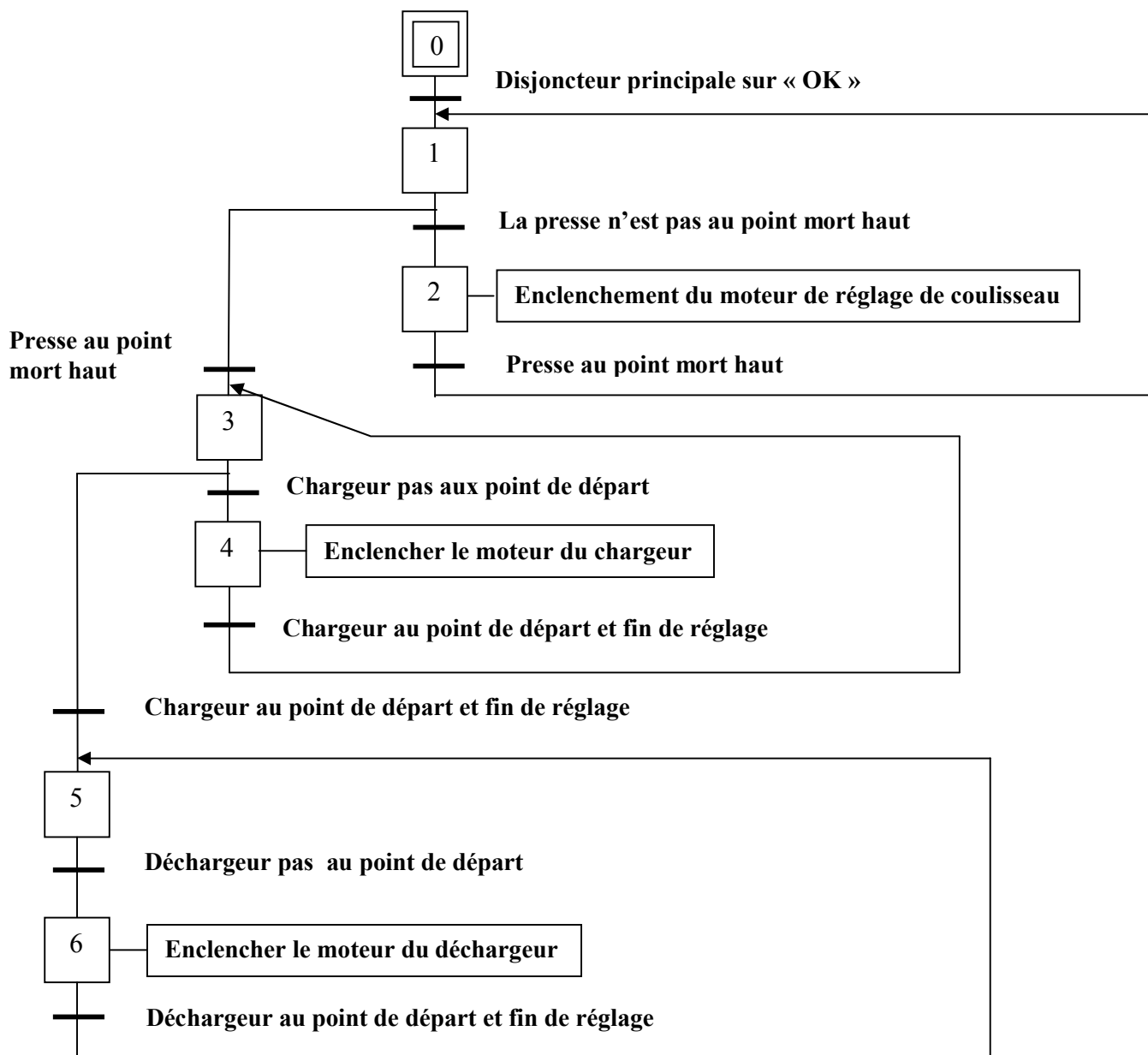


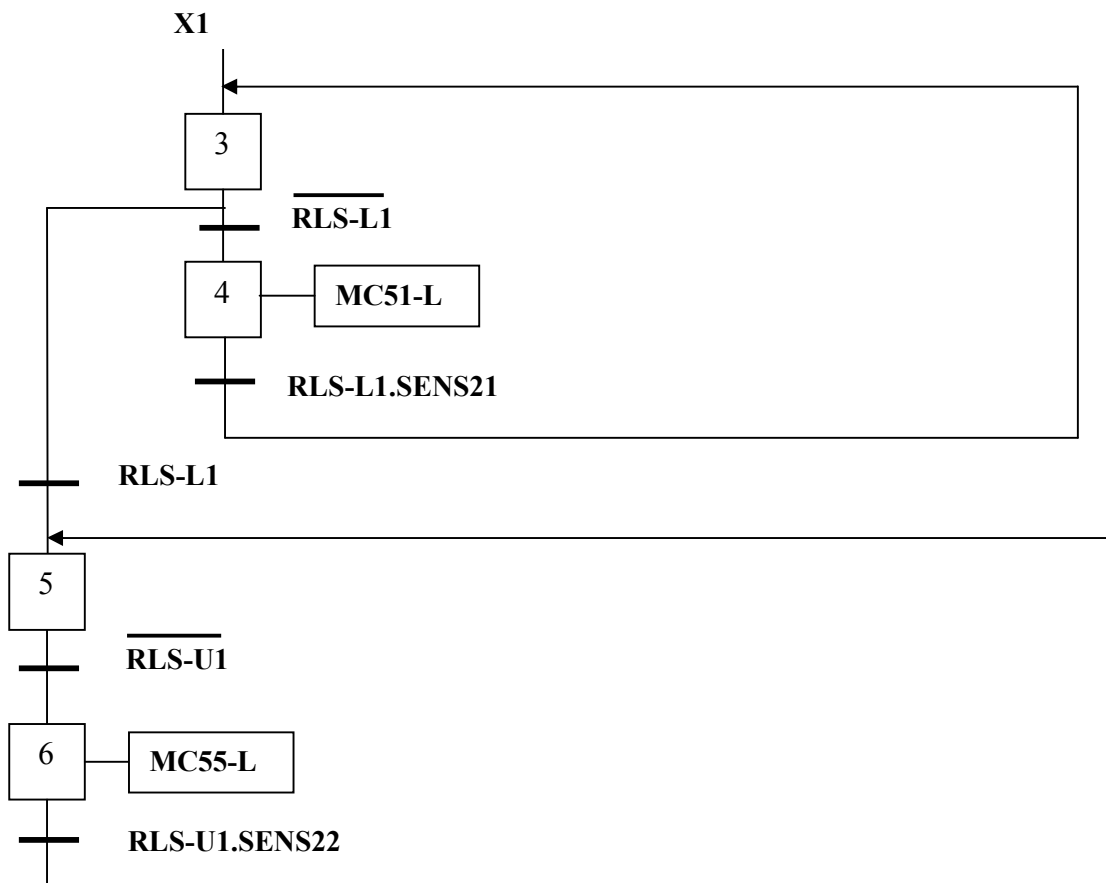


❖ **Grafcet de mise en position initiale de la machine :**

Après vérification étape par étape des conditions nécessaire pour le bon fonctionnement, la machine est prête à fonctionner en mode manuel au moins pour un cycle.

Pour s'assurer du fonctionnement global et après le mode automatique est exécuté.





### V : Conclusion :

En tenant compte de la complexité et la difficulté du processus ainsi que des contraintes imposées par l'entreprise, nous avons modélisé le procédé de commande à l'aide du GRAFCET. Nous avons élaboré en premier lieu un grafcet de niveau 1 pour expliquer le système, puis le grafcet niveaux 2 qui met en œuvre et décrit la partie opérative. Ce grafcet niveau 2 est utilisé pour la réalisation ou le dépannage des systèmes automatisés.

Au terme de ce chapitre nous concluons que le GRAFCET est un outil de modélisation qui permet facilement le passage d'un cahier des charges fonctionnel à un langage d'implantation optionnel, il permet la description du comportement attendu de la partie commande d'un système automatisé, comme il permet de créer un lien entre la partie commande et la partie opérative.

Ainsi, le grafcet a facilité considérablement le passage de la description à la modélisation et nous permettra au chapitre suivant d'aborder la programmation de la partie opérative qui pilotera le procédé et ce à l'aide du STEP7.

## **I. Introduction aux automates programmables :**

Les Automates programmables industriels (API), sont apparus vers 1969 aux Etats unis pour répondre aux besoins de l'industrie automobile. Ils ont été conçus pour l'automatisation des chaînes de fabrication et réaliser des fonctions logiques combinatoires et séquentielles en remplacement des armoires à relais trop volumineuses.

Depuis, leur utilisation s'est largement répandue dans l'industrie, où ils représentent l'outil de base de l'automatisation des systèmes de production. Les API ont trouvé leur place dans les domaines les plus variés comme dans les chaînes de fabrication (usinage, montage,...etc.), dans les opérations de manutention (stockage, chargement,...etc.), ou encore dans les systèmes de contrôle (installation de climatisation, frigorifique, de chauffage, détection des incendies, industrie nucléaire,...etc.)

Avec le temps, on a trouvé sur le marché différentes variétés ; ceci est dû à la diversité des Constructeurs. Pour la réalisation de notre travail on utilise l'automate SIMATIC S7-300 de SIEMENS.

### **I.1. Automates programmables industriels (API) :**

L'automate programmable industriel API (programmable logic controller : PLC) est un dispositif électrique de traitement logique d'informations dont le programme de fonctionnement est effectué à partir d'instructions établies en fonction du processus à réaliser. Il est adapté à l'environnement industriel ; il génère des ordres vers les pré-actionneurs de la partie opérative à partir de données d'entrées (capteurs) et d'un programme. Il est généralement relié à un pupitre (ou console).

A partir de cette définition, nous distinguons dans les fonctions que l'automate doit remplir :

-Un rôle de **commande** où il est un composant d'automatisme, élaborant des actions, suivant une algorithmique approprié, à partir des informations que lui fournissent des détecteurs (Tout ou Rien) ou des capteurs (analogiques ou numériques).

-Un rôle de **communication** dans le cadre de la production :

- Avec des opérateurs humains : c'est le dialogue d'exploitation,
- Avec d'autres processeurs, hiérarchiquement supérieurs (calculateur de gestion de production), égaux (autres automates intervenant dans la même chaîne) ou inférieurs.

## I.2. Architecture et fonctionnement d'un automate :

La structure interne d'un API peut se présenter comme représenté en figure (IV.1) :

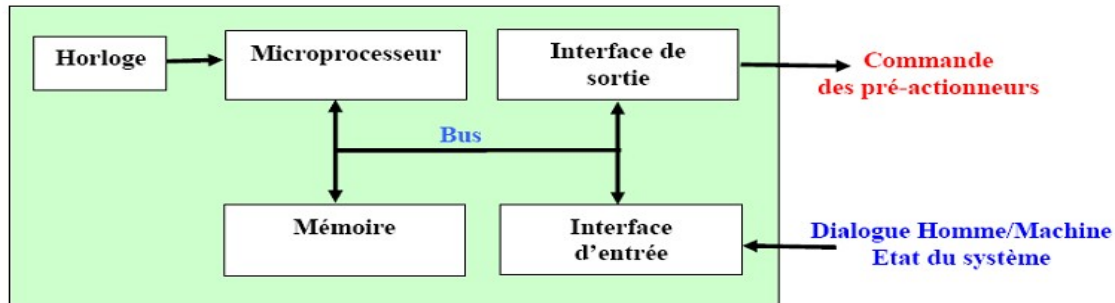


Figure IV.1 : Structure interne d'un API.

L'API se compose de trois grandes parties :

### a) Le microprocesseur :

Le microprocesseur réalise toutes les fonctions logiques (ET, OU), les fonctions de temporisation, de comptage, de calcul... à partir d'un programme contenu dans sa mémoire. Il est connecté aux autres éléments (mémoire et interface E/S) par BUS.

### b) La zone mémoire :

Elle permet de :

- Recevoir des informations issues des capteurs d'entrées.
- De recevoir les informations générées par le processeur et destinées à la commande des sorties (valeur des compteurs, des temporisations,...).
- De recevoir et conserver le programme du processus.

### c) Les interfaces d'entrées/sorties :

Les entrées reçoivent des informations en provenance des éléments de détection (capteurs) et du pupitre opérateur.

Les sorties transmettent des informations aux pré-actionneurs (relais, électrovannes...) et aux éléments de signalisations (voyants) du pupitre.

## I.3. Choix d'un automate :

Le choix d'un automate se fait après avoir établi le cahier des charges du système à automatiser, cela en considérant un certain nombre de critères :

- Le nombre d'entrées et de sorties.
- La nature des entrées et des sorties (numérique, analogique).

- La nature de traitement (temporisation, comptage,...).
- La fiabilité et la robustesse.
- L'immunité aux parasites et aux bruits.
- Le service après vente et la durée de garantie.
- La formation et la documentation.

Il est primordial de connaître le nombre d'entrées, et de sorties du système (machine de découpage et de poçonnage des contre portes frigorifique) à fin d'adapter l'automate. Pour les entrées, tout ce qui est capteurs, interrupteurs, boutons poussoirs... etc. Pour les sorties, tout ce qui est actionneurs, comme les moteurs, les vérins...etc.

- Nombre d'entrées : 62.
- Nombre de sorties : 50.

Le choix d'un automate performant, intégrant plus de modules d'entrées / sorties impose de choisir l'automate S7-300.

Les caractéristiques de l'API S7-300 convient parfaitement à ces exigences car il peut gérer sans extension 256 entrées/sorties et avec extension jusqu'à 1024 entrées /sorties (numérique, logique ou analogique).

#### I.4. Constitution de l'automate S7-300 :

L'automate programmable S7-300 de la famille SIMATIC est un système d'automatisation modulaire (figure IV.2).

Il est d'une forme compacte et permet un vaste choix de gamme de modules suivants :

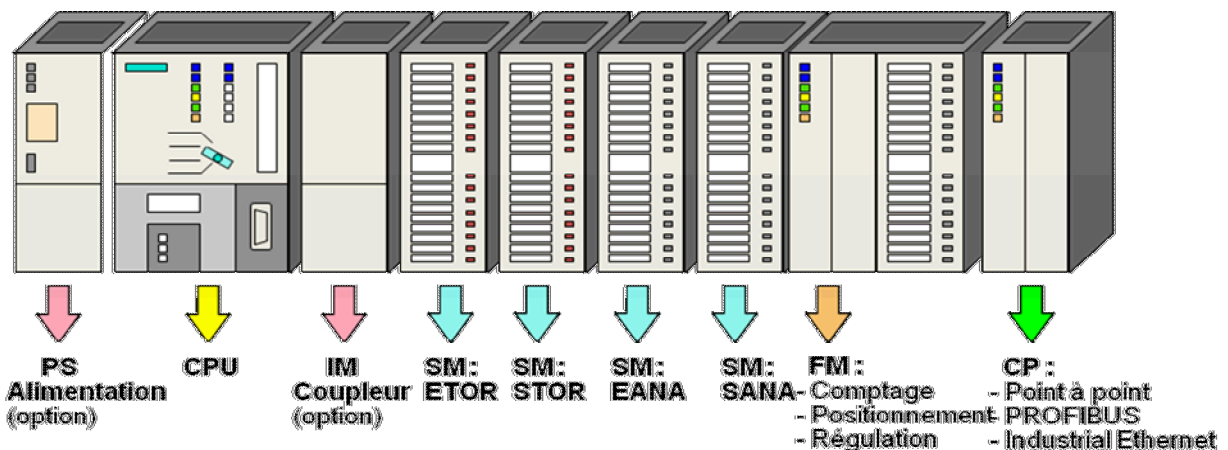


Figure IV.2 : Constitution d'un API S 7-300.

- ✚ Module d'alimentation (PS) 2A ,5A, 10A.
- ✚ Unité centrale CPU 314 travaillant avec une mémoire de 48 KO, sa vitesse d'exécution est de 0.3ms/1Ko instructions.
- ✚ Modules d'extension (IM) pour configuration multi rangée de S7-300.
- ✚ Modules de signaux (SM) pour entrées et sorties TOR et analogiques.
- ✚ Module de fonction (FM) pour les fonctions spéciales (par exemple activation d'un moteur asynchrone).
- ✚ Processus de communication (CP) pour la connexion au réseau.
- ✚ Châssis d'extension (UR).

#### **I.4.1. Module d'alimentation (PS) :**

Il transforme la tension secteur en tension d'alimentation pour les modules électriques de l'automate programmable. Cette tension s'élève à 24 volts.

Les tensions qui dépassent 24 volts comme pour les capteurs, actionneurs et voyants lumineux sont fournies par des blocs d'alimentations ou transformateur supplémentaires.

#### **I.4.2. Description de la CPU :**

La CPU (Central Processing unit) est le cerveau de l'automate, elle lit les états des entrées, ensuite, elle exécute le programme utilisateur en mémoire et enfin, elle commande les sorties. Elle comporte une unité de commande et de calcul, des mémoires, un programme système et des interfaces vers les modules de signaux.

La CPU est constituée de :

##### Interfaces MPI :

Chaque CPU est équipée d'une interface MPI pour la connexion de la console de programmation (PG) ou un autre appareil (par exemple adaptateur PC).

##### Commutateur de mode de fonctionnement :

Le commutateur de mode de fonctionnement permet de changer le mode de fonctionnement.

Les modes de fonctionnement suivant sont possibles :

- RUN-P : exécution de programme, accès en écriture et en lecture avec la PG.
- RUN : exécution de programme, accès en lecture seule avec la PG.
- STOP : le programme n'est pas exécuté, toutes les fonctions avec la PG sont autorisées.

##### Signalisation des états :

Certains états de l'automate sont signalés par des leds sur la face avant de la CPU tel que :



**SF** : signalisation groupée des défauts, défauts de la CPU ou d'un module avec fonction diagnostique.

**BATF** : défaut de pile, pile à plat ou absente.

**DC5v** : signalisation de tension d'alimentation 5v (allumé : les 5v sont présente, clignote : surcharge courant).

**FCRE** : forçage signalisation qu'au moins une entrée ou une sortie est forcée de manière permanente.

**RUN** : clignotement lors de la mise en route de la CPU, allumage continue en mode RUN.

**STOP** : allumage continue en mode STOP, clignotement long lorsqu'un effacement général est requis, clignotement rapide lorsqu'un effacement général est en cours.

#### La carte mémoire :

Une carte mémoire peut être montée à la CPU, elle conserve le contenu de programme en cas de coupure de courant.

#### La pile :

Elle permet de sauvegarder le contenu de la RAM en cas de coupure de courant.

La gamme S7-300 offre une grande variété de CPU, chaque CPU possède certaines caractéristiques différentes des autres et par conséquent le choix de la CPU est conditionné par les caractéristiques offertes par la CPU choisie.

#### **I.4.3. Modules de coupleurs (IM) :**

Les coupleurs peuvent être utilisés pour un couplage sur de courtes distances. Pour un couplage sur de longues distances, il est recommandé d'émettre les signaux via le bus Profibus.

Les coupleurs IM 360/IM 361 ou Im365 permettent de réaliser des configurations à plusieurs châssis.

#### **I.4.4. Module de signaux (SM) :**

Il comporte plusieurs signaux tels que : STOR ; ETOR ; SANA ; EANA ; ou E/SANA, et E/STOR, ils ont comme fonction l'adaptation des niveaux de signaux entre le processus et le S7-300.

#### **I.4.5. Modules d'entrées et de sorties tout ou rien (TOR) :**

Les API offrent une grande variété d'entrées/sorties TOR adaptées aux milieux auxquels ils sont soumis. Ces entrées/sorties peuvent accepter des informations en courant ou en tension, alternatifs ou continus.

#### **a) Modules d'entrées TOR :**

Un module d'entrée doit permettre à l'unité centrale de l'automate, d'effectuer une lecture de l'état logique des capteurs qui lui sont associés (modules 4, 8, 16 ou 32 entrées).

A chaque entrée correspond une voie qui traite le signal électrique pour élaborer une information binaire.

Le processeur de l'automate vient questionner le module ; le contenu du mot d'entrées du module est alors recopié dans la mémoire de données de l'automate programmable.

Les modules d'entrées « Tout Ou Rien » permettent de raccorder à l'automate différents capteurs logiques tel que les boutons poussoirs, les fins de course, ...etc.

#### **b) Modules de sorties TOR :**

Un module de sortie permet à l'automate programmable d'agir sur les actionneurs. IL réalise la correspondance : état logique → signal électrique.

Les modules de sorties Tout Ou Rien permettent de raccorder à l'automate différents pré-actionneurs tels que : les électrovannes, les contacteurs, ...etc.

#### **I.4.6. Modules d'entrée et de sortie analogique :**

Les modules d'entrées/sorties analogiques réalisent la conversion des signaux analogiques issus de processus en signaux numériques pour le traitement interne dans le S7-300 et des signaux numériques du S 7-300 en signaux analogiques destinés au processus.

#### **I.4.7. Module de fonction (FM) :**

Il a pour rôle l'exécution de tâche du traitement des signaux du processus à temps critique et nécessitant une importante capacité mémoire comme le comptage, le positionnement et la régulation.

#### **I.4.8. Modules de communication (CP) :**

Ils permettent d'établir des liaisons homme-machine ou machine-machine ; ces liaisons sont effectuées à l'aide des interfaces de communication :

Point à point.

Profibus.

Industriel Ethernet.

#### **I.4.9. Châssis d'extension (UR) :**

Il est constitué d'un profilé support en aluminium et bus de fond de panier avec connecteur. Il permet le montage et le raccordement électrique de divers modules tels que : les modules d'entrées/sorties et l'alimentation. Il est possible d'utiliser plusieurs racks en fonction du nombre d'entrées/sorties.

## **I.5. Fonctionnement de base d'un API :**

### **I.5.1. Module central CPU :**

Dans la CPU (Module central), le processeur traite le programme se trouvant dans la mémoire et interroge les entrées de l'appareil pour savoir si elles délivrent de la tension ou pas.

En fonction de cet état des entrées et du programme se trouvant en mémoire, le processeur ordonne au module de sortie de commuter sur le connecteur de la barrette de connexion correspondante. En fonction de l'état de tension sur les connecteurs des modules de sortie, les appareils à positionner et les lampes indicatrices sont connecté ou déconnecté.

### **I.5.2. Réception des informations sur les états du système :**

Le S7-300 reçoit des informations sur l'état du processus via les capteurs de signaux reliés aux entrées.

Le S7-300 met à jour la mémoire image des entrées au début de chaque cycle de programme en transférant le nouvel état des signaux d'entrées des modules vers la mémoire image des entrées ce qui permet à la CPU de savoir l'état du processus.

### **I.5.3. Exécution du programme utilisateur :**

Après avoir acquis les informations d'entrées et exécuter le système d'exploitation, la CPU passe à l'exécution du programme utilisateur, qui contient la liste d'instruction à exécuter pour faire fonctionner le procédé.

### **I.5.4. Commande du processus :**

Pour commander le processus, on doit agir sur les actionneurs. Ces derniers reçoivent l'ordre via le module de sortie du S7-300. Donc l'état des sorties est connu après l'exécution du programme utilisateur par la CPU, puis elle effectue la mise à jour de la mémoire image des sorties pour communiquer au processus le nouvel état.

## II. Programmation de l'automate S7-300:

La programmation des automates de la famille S7 se fait par la console de programmation ou par PC et sous un environnement WINDOWS, via le langage de programmation STEP7. La programmation en STEP7 présente trois modes de représentation qui peuvent être combinés dans une même application :

🚧 Schéma à contacts « CONT ».

🚧 Logigramme « LOG ».

🚧 Liste d'instruction « LIST ».

Dans la mémoire de programmation de l'automate, le programme est toujours stocké en LIST (plus exactement en langage machine).

### a) Schéma à contacts (CONT) :

Le schéma à contacts est une représentation graphique de la tâche d'automatisation ayant recours aux symboles. C'est un langage des habitués des schémas électriques.

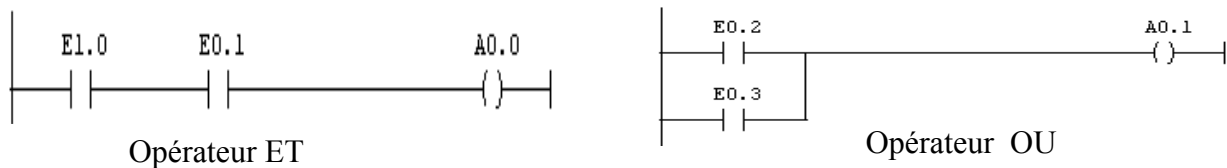


Figure IV.3 : Représentation des opérations ET, OU par CONT.

### b) Logigramme (LOG) :

Le logigramme est une représentation graphique ayant recours aux symboles de la logique.

Les différentes fonctions sont représentées par un symbole avec indicateur de fonction.

Les entrées sont disposées à gauche du symbole, et les sorties à sa droite (figure IV.4).



Figure IV.4 : Représentation des opérations ET, OU par LOG.

### c) Liste d'instructions (LIST) :

La tâche d'automatisation est écrite dans ce cas à l'aide des différentes instructions. C'est un langage qui s'apparente au langage machine.

Les programmes en CONT et en LOG sont en principe toujours traduisibles en LIST.

## II.1. Création d'un projet STEP7 :

Pour créer un projet STEP7 on dispose d'une certaine liberté d'action ; en effet nous avons deux solutions possibles :

- Solution 1 : commencer par la configuration matérielle.
- Solution 2 : commencer par la création de programme.

Le schéma suivant (figure IV.5) illustre les deux solutions possibles lors de la conception d'une solution d'automatisation :

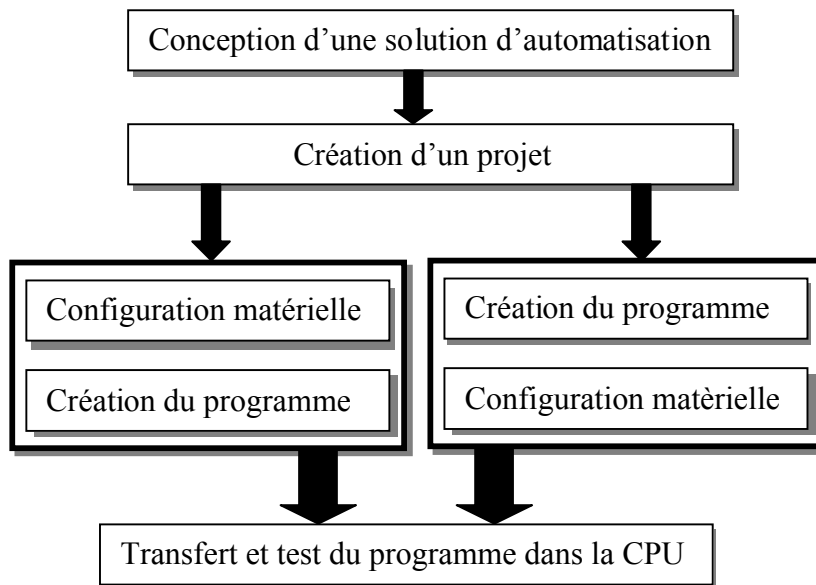


Figure IV.5 : Les deux solutions possibles pour la programmation.

L'application de la configuration matérielle de STEP7 présente l'avantage de la sélection automatique des adresses.

### ▪ Lancement du logiciel :

Double clique sur l'icône SIMATIC Manager sur le bureau Windows ; ceci lance l'assistant de STEP7.

### ▪ Création du projet :

La fenêtre illustrée en figure (IV.6) apparaît, elle permet la création d'un nouveau projet.



Figure IV.6 : Assistant de STEP7 ' nouveau projet'.

En cliquant sur l'icône suivant, la fenêtre suivante apparaît, elle nous permet de choisir la CPU. Pour notre projet nous avons choisi la CPU 314.

Après validation de la CPU, la Fenêtre qui apparaît permet de choisir les blocs à insérer, et choisir le langage de programmation (LIST, CONT, LOG).

Pour notre projet nous avons choisi l'OB1 et le langage à contact.

En cliquant sur suivant, la création de projet apparaît pour le nommer.

## II.2. Configuration du matériel de l'automate :

La configuration matérielle consiste en la disposition de profilés support ou châssis (racks), de modules et d'appareils de la périphérie centralisée. Les châssis sont représentés par une table de configuration dans laquelle on peut enficher un nombre défini de module, comme dans les châssis réels.

STEP 7 affecte automatiquement une adresse à chaque module dans la table de configuration.

Pour notre système, nous avons choisi une configuration dans laquelle nous avons (figure IV.7).

- Un rack
- Le module d'alimentation PS 307 5A ;
- La CPU 314 ;
- Deux (02) modules d'entrées logiques de 32 bits ;
- Deux (02) modules de sortie logique de 32 bits.

(0) UR						
Emplacement	Module	...	Référence	Adresse MPI	Adresse d'entrée	Adresse de sortie
1	PS 307 5A		6ES7 307-1EA00-0AA0			
2	CPU 314(1)		6ES7 314-1AE01-0AB0	2		
3						
4	DI32xDC24V		6ES7 321-1BL80-0AA0		0...3	
5	DI32xDC24V		6ES7 321-1BL80-0AA0		4...7	
6	DO32xDC24V/0.5A		6ES7 322-1BL00-0AA0			8...11
7	DO32xDC24V/0.5A		6ES7 322-1BL00-0AA0			12...15
8						

Figure IV.7 : Configuration matérielle de l'automate.

Sauvegardons cette configuration en cliquant sur l'item Enregistrer du menu déroulant Fichier. Fermons la fenêtre.

### II.3. Création de la table des mnémoniques :

Les mnémoniques permettent de déclarer les différents entrées/ sorties de la machine ainsi que les mementos utilisés à fin de mieux les distingués et faciliter la simulation du programme, rendre le programme utilisateur très lisible et aide donc à gérer facilement les grands nombres de variables couramment rencontrées dans ce genre de programme.

### II.4. Adressage des modules du S7-300 :

On a deux types d'adressage :

- Adressage lié à l'emplacement : il s'agit du mode d'adressage par défaut, c'est-à-dire que le STEP7 effectue à chaque numéro d'emplacement une adresse de défaut de modèle fixé à l'avance.
- Adressage libre : dans ce mode d'adressage, il faut effectuer à chaque mode une adresse de votre choix, pourvu qu'elle soit continue dans la plage d'adresses possible de la CPU.

#### II.4.1. Adressage des modules lié a l'emplacement :

Dans le cas de cet adressage, (adressage par défaut), une adresse de début de module est affectée à chaque numéro d'emplacement sur le profilé support (châssis).

#### II.4.2. Adressage absolu des modules de signaux :

Ce type d'adressage est nécessaire pour adresser les voies des modules de signaux dans le programme utilisateur. Chaque sortie possède une adresse absolue déterminée par la configuration matérielle.

##### a) Adressage des modules TOR :

L'adressage d'une entrée ou d'une sortie est constitué d'une adresse d'octet et d'une adresse de bit.

- L'adressage d'octet dépend de l'adresse de début de module.

- L'adressage de bit est indiqué sur le module.

La figure (IV.8) illustre un exemple d'adressage absolu.

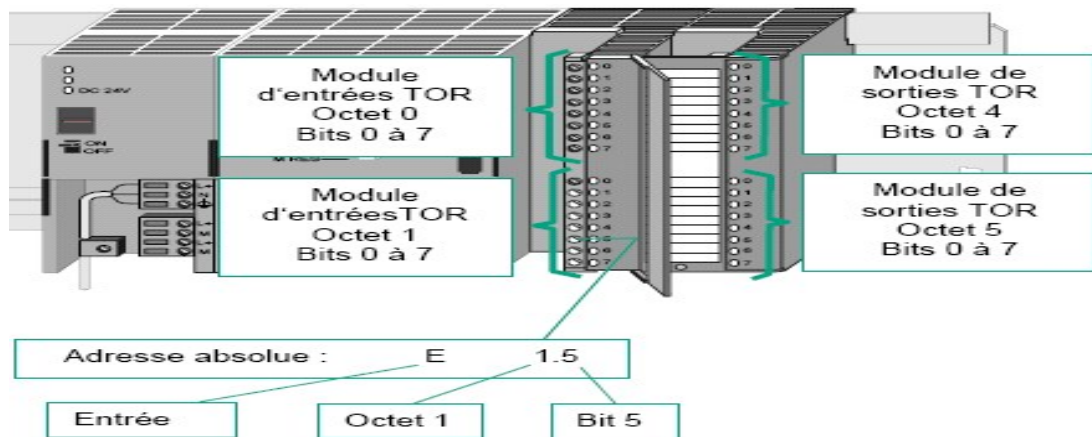


Figure IV.8 : Exemple d'adressage absolu d'un module TOR.

## II.5. Mémentos :

Les mémentos sont utilisés pour les opérations internes à l'automate pour lesquelles l'émission d'un signal n'est pas nécessaire. Chaque automate programmable dispose d'un grand nombre de mémentos (S7-300 dispose de 2048 bits de mémentos).

## II.6. Traitement du programme par l'automate :

La CPU traite le programme d'une manière cyclique en plusieurs phases :

- Phase 1 : le système d'exploitation démarre la surveillance de temps de cycle.
- Phase 2 : la CPU lit l'état des entrées dans les modules d'entrées et met à jour la mémoire image des entrées.
- Phase 3 : la CPU exécute les instructions de programme utilisateur.
- phase 4 : la CPU écrit les résultats dans la mémoire image des sorties, puis elle transfère ces dernier, vers les modules de sorties.
- Phase 5 : à la fin de cycle, le système d'exploitation exécute les travaux en attente, tel que le chargement et l'effacement des blocs ou la réception et l'émission des données globales.
- Phase 6 : la CPU revient alors au début du cycle et démarre à nouveau la surveillance du temps du cycle.

## II.7. Principe de conception d'une structure de programme :

Pour chaque solution d'automatisation, la CPU exécute deux types de programmes différents, le système d'exploitation et le programme utilisateur.

### ❖ Rôle du système d'exploitation :



Le système d'exploitation, contenu dans la CPU, organise toutes les fonctions et procédures qui ne sont pas liées à une tâche d'automatisation spécifique. Ces tâches sont les suivantes :

- Le déroulement de démarrage ou de redémarrage.
- L'actualisation de la mémoire image des entrées et l'émission de la mémoire image des sorties.
- L'appel du programme utilisateur.
- L'enregistrement des alarmes et l'appel des blocs d'alarmes.
- La détection et le traitement d'erreurs.
- La gestion des zones de mémoire.
- La communication avec des consoles de programmation et d'autres partenaires de communication.

❖ **Rôle du programme utilisateur :**

Le programme utilisateur est chargé dans la CPU. Il contient toutes les fonctions nécessaires au traitement de la tâche d'automatisation, en plus il doit prendre en charge :

- La détermination des conditions pour le démarrage et le redémarrage de la CPU (exp : initialisation des signaux).
- La réaction aux alarmes.
- Le traitement des données de processus (exp : combiner des signaux binaires, écrire des valeurs analogiques).

**II.8. Blocs dans le programme utilisateur :**

Il faut subdiviser le procédé à automatiser en ces différentes tâches.

Les parties d'un programme utilisateur structuré correspondant à ces différentes tâches sont les blocs de programmes.

Le STEP7 offre la possibilité de structurer notre programme utilisateur, c'est-à-dire le subdiviser en différentes parties autonomes qui donnent les avantages suivants :

- Écriture des programmes importants mais clairs.
- Simplification de l'organisation du programme.
- Modification facile de programme.
- Simplifier le test du programme, car on peut l'exécuter section par section.
- Faciliter la mise en service.

### **II.8.1. Différents blocs du programme utilisateur :**

Le logiciel STEP7 dans ses différents langages de programmation possède un nombre important de blocs utilisateurs, destinés à structurer le programme utilisateur dont on peut citer les blocs importants suivants :

- **Bloc d'organisation (OB) :**

Un OB est une structure importante d'un programme utilisateur, il est appelé cycliquement par le système d'exploitation ; il gère le traitement du programme cyclique et déclenché par alarme ainsi que le comportement à la mise en route de l'automate programmable et le traitement des erreurs.

Nous pouvons programmer les blocs d'organisations et déterminer ainsi le comportement de la CPU. En résumé l'OB constitue l'interface entre le système d'exploitation et le programme utilisateur. Les OB définissent l'ordre dans lequel les différentes parties du programme sont traitées.

- **Bloc fonctionnel (FB) :**

Un bloc fonctionnel contient un programme qui est exécuté, lorsque ce bloc fonctionnel est appelé par un autre bloc de code. Il facilite la programmation de fonctions complexes, comme la commande de moteur. Un bloc de données est associé à chaque FB qui constitue la mémoire, les paramètres transmis au FB ainsi que les variables statiques sont sauvegardés dans le bloc de données locales.


- **Fonction (FC) :**

Les fonctions font partie des opérations que le concepteur du programme utilisateur utilise. Elles ne possèdent pas de mémoire. Les variables temporaires d'une fonction sont sauvegardées dans la pile des données locales. Ces données sont perdues à l'achèvement de la fonction. Les fonctions peuvent faire appel à des blocs de données globaux pour la sauvegarde de données.

Une fonction contient un programme qui est exécuté quand cette fonction est appelée.

- **Bloc de données (DB) :**

Les DB sont utilisés pour la mise à la disposition de l'espace mémoire pour les variables de type données. On a deux types de blocs :

-  **Bloc de données d'instance :** Il est associé à chaque appel de bloc fonctionnel transmettant des paramètres, il contient des paramètres effectifs et des données statiques du FB, on appelle l'instance l'appel d'un FB pour un DB.

- ✚ Bloc des données globales (DB) : Contrairement aux blocs de codes (OB, FC, FB), les DB ne contiennent pas d'instructions step7. Ils servent à l'enregistrement de données utilisateur, ils contiennent des données variables que le programme utilisateur utilise.

Tout FB, FC, OB peut lire les données contenues dans un DB global ou écrire des données dans un DB global. Ces données sont conservées dans le bloc de données même l'orsque on quitte le DB. Le rôle des DB dans un programme utilisateur est illustré en figure (IV.9) :

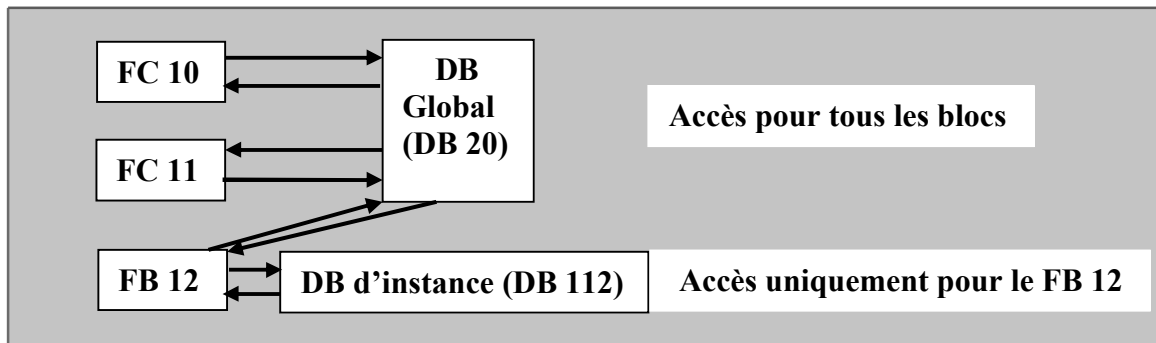


Figure IV.9 : Illustration du rôle des DB dans un Programme utilisateur.

## II.9. Bloc fonctionnel système (SFB) :

Parmi les atouts des CPU S7, on trouve, les fonctions déjà programmées, qui peuvent être utilisées par le programme utilisateur.

- **Bloc fonctionnel système (SFB) :**

Un SFB est un bloc fonctionnel intégré à la CPU S7. Comme les SFB font partie du système d'exploitation, ils ne sont pas chargés en tant que partie de programme. Comme les FB, les SFB constituent des blocs avec mémoire, nous devons donc leur associer des blocs de données d'instance que nous chargeons dans la CPU en tant que partie du programme.

- **Fonction système (SFC) :**

SFC sont des fonctions préprogrammées et testées intégrées dans la CPU S7, elles peuvent être appelées par le programme utilisateur.

- **Bloc de données système (SDB) :**

C'est une zone de mémoire dans le programme, configurée par différentes applications de STEP7 pour le stockage des données dans le système d'automatisation.

## II.10. Traitement du programme par la CPU :

Selon la manière de programmation choisie pour le problème d'automatisation, la CPU peut traiter le programme comme suit :

### II.10.1. Traitement linéaire du programme :

La CPU exécute le cycle habituel, en appelant le bloc OB1 dans le programme principal. Ce type de traitement est requis lors de la commande de procédé simple exigeant de programme de taille et de complexité réduite.

### II.10.2. Traitement structuré du programme :

Dans le cas d'un procédé complexe possédant plusieurs fonctions, il est recommandé de partager le programme utilisateur en fonctions principales que l'on programme à l'aide des blocs de codes (OB, FB, FC), d'une manière à faciliter la programmation et le test des parties du programme.

Dans ce cas la CPU exécute le programme principal dans OB1 et fait appel à et quand il le faut aux autres blocs (FC, FB), à la fin de l'exécution du bloc appel, la CPU revient pour poursuivre l'exécution du programme appelant.

### II.10.3. Structure du programme utilisateur :

Nous avons structuré notre programme utilisateur en différentes parties autonomes comme c'est illustré en figure (IV.10). L'OB 1 est la structure importante du programme, il constitue l'interface entre le programme utilisateur et le système d'exploitation, c'est-à-dire que la CPU exécute uniquement l'instruction qui se trouve sur ce bloc. Nous avons subdivisé le procédé à automatiser en ces différentes tâches. Nous avons choisi d'utiliser les fonctions FC qui contrairement au bloc fonctionnel ne possède pas de zone mémoire, les données locales d'une fonction sont perdues après l'exécution de la fonction. Une fois tous les FC programmés, on les insère dans le bloc d'organisation OB 1 pour la phase de simulation.

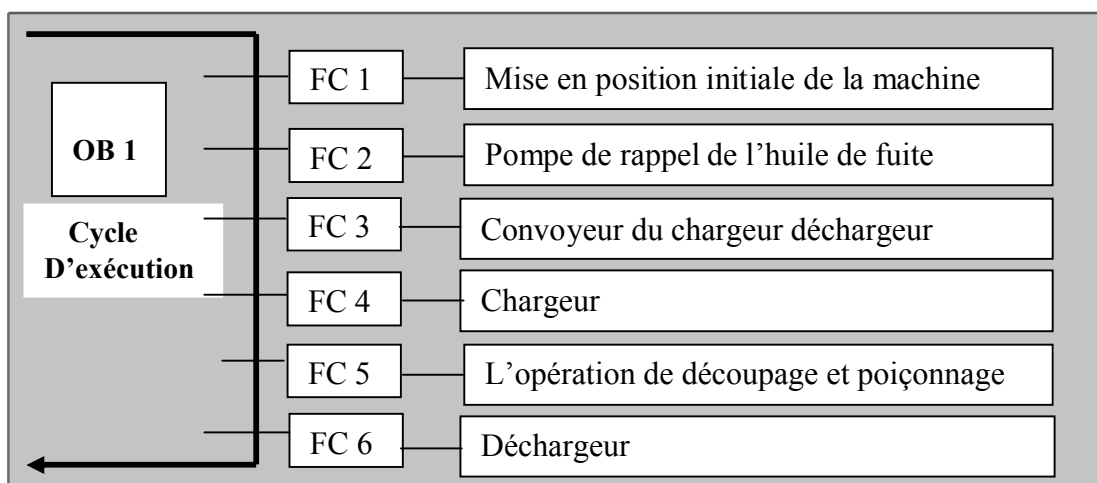


Figure IV.10 : Schéma illustrant la programmation structurée.

## **II.11. Implantation du Grafcet dans le S7-300 :**

Après avoir bien étudié le procédé à automatiser et réaliser son grafcet, on doit implanter ce dernier dans le S7-300 or la version du S7 dont nous disposons ne possède pas de langage GRAFCET qui permet la transition de la modélisation à la commande.

Avec la possibilité offerte par le S7-300 en matière de langage il existe toujours un moyen d'implanter le grafcet dans l'automate en vue de son exécution, pour cela nous devons suivre certaines étapes :

- 1) Analyser et valider le grafcet de la machine (blocage, conflit,...).
- 2) Détermination des conditions d'activation et de désactivation des étapes du grafcet.
- 3) Définition des exigences en matière de sécurité (les alarmes, AUD et AUd...).
- 4) Affectation des mementos aux variables intermédiaires, et des adresses effectives et formelles aux entrées et sorties.
- 5) Ecriture des équations des étapes.
- 6) Traduction de ces équations en programme séquentiel, en utilisant l'un des langages offerts par le STEP7 (List, Cont et Log).
- 7) Raccorder toutes les entrées, sorties et la PG à l'automate.
- 8) Après avoir choisi la configuration matérielle et simuler le programme, on procède au chargement du projet dans la CPU physique.
- 9) On lance la machine en la mettant en mode RUN.
- 10) Après validation du projet, on peut commencer la production.

## **II.12. Conclusion :**

Ce chapitre nous a permis de présenter les différents modules constituant l'ensemble de l'automate S7-300 et son langage de programmation.

La constitution modulaire, la facilité de réalisation d'architecture décentralisée et la facilité d'emploi font du S7-300 la solution économique et conviviale pour les tâches les plus diverses dans les petites et moyennes applications.

Le logiciel de programmation STEP 7 constitue le lien entre l'utilisateur et l'automate programmable S7-300 car ce dernier ne peut gérer ses fonctionnalités sans un programme approprié. Mais avant l'élaboration du programme il faut d'abord modéliser le procédé. Cela a fait l'objet du chapitre précédent.

## **I. Introduction :**

Après l'élaboration du programme de commande de notre système à automatiser, nous arrivons à l'étape décisive du travail effectué. Cette étape est la validation du programme par simulation et vérification de son bon fonctionnement.

Pour cela nous avons utilisé le logiciel S7-PLCSIM qui est un logiciel optionnel de STEP7. L'application de simulation de modules S7-PLCSIM nous permet d'exécuter et de tester notre programme dans un automate. La simulation étant complètement réalisée au sein du logiciel STEP7, il n'est pas nécessaire qu'une liaison soit établie avec un matériel S7 quelconque (CPU ou modules de signaux). L'objectif de ce logiciel est le test des programmes STEP7 pour les automates S7-300 et S7-400 qu'on ne peut pas tester immédiatement sur le matériel et ceci pour différentes raisons, l'application est critique, car elle peut occasionner des dommages matériels ou blessures corporelles en cas d'erreurs de programmation, mais la simulation permet de corriger ces erreurs pendant le test de la simulation.

### **I.1. Présentation du S7 PLCSIM :**

L'utilisation du simulateur de modules physiques S7-PLCSIM nous permet d'exécuter et de tester le programme dans un automate que nous simulons dans un ordinateur ou dans une console de programmation. La simulation est complètement réalisée au sien du logiciel STEP7.


Le S7-PLCSIM dispose d'une interface simple nous permettant de visualiser et de forcer les différents paramètres utilisés par le programme (comme activer ou désactiver des entrées). Tout en exécutant le programme dans l'API de simulation, nous avons également la possibilité de mettre en œuvre les diverses applications du logiciel STEP7 comme, par exemple, le test de blocs afin de visualiser les variables d'entrées et de sorties.

### **I.2. Mise en route du logiciel S7-PLCSIM :**

Le mode de simulation est disponible à partir du gestionnaire de projet SIMATIC à condition qu'aucune liaison à des API réels ne soit établie. On peut suivre la procédure suivante pour la mise en route du logiciel S7-PLCSIM.

La procédure à suivre est :

 Ouvrir le gestionnaire de projet SIMATIC.

Cliquez sur  ou sélectionnez la commande Outils-simulation de modules,

Cela lance l'application S7-PLCSIM et ouvre une fenêtre CPU (figure V.1) :

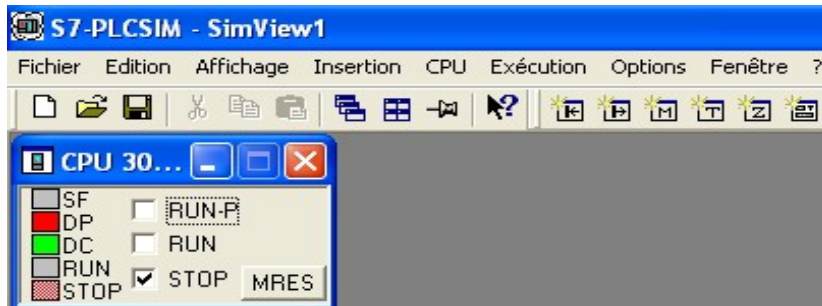






Figure V.1 : Fenêtre du S7-PLCSIM.

- ✚ Dans le gestionnaire de projets SIMATIC, chercher le projet-exemple projet presse Müller.
- ✚ Dans le projet exemple « projet Presse Müller », chercher le dossier blocs.
- ✚ Dans le gestionnaire de projet SIMATIC, cliquez sur  ou choisir la commande Système cible-Charger pour charger le dossier blocs dans l'API de simulation.

Dans l'application S7-PLCSIM, on crée de nouvelles fenêtres pour visualiser les informations provenant de l'API de simulation :

- ✚ Cliquez sur  ou choisissez la commande Insertion-Entrée pour créer une fenêtre dans laquelle vous pouvez visualiser et forcer des variables dans la zone de mémoire des entrées (zone E). Cette fenêtre s'ouvre avec l'adresse de mémoire par défaut EB0. Mais peut modifier l'adresse (EB1, EB2...).
- ✚ Cliquez sur  ou choisissez la commande Insertion-Sortie pour créer une fenêtre dans laquelle vous pouvez visualiser et forcer des variables dans la zone de mémoire des sorties (zone A). Cette fenêtre s'ouvre avec l'adresse de mémoire par défaut Ab0. Mais on peut modifier l'adresse (AB1, AB2...).
- ✚ Cliquez sur  ou choisissez la commande Insertion-Temporisation pour créer une fenêtre dans laquelle vous pouvez visualiser et forcer les temporisations utilisées par le programme. Cette fenêtre s'ouvre avec l'adresse de mémoire par défaut T0.

Choisir le menu CPU dans la fenêtre du S7\_PLCSIM et vérifier que la commande mettre sous tension est activée (figure V.2).

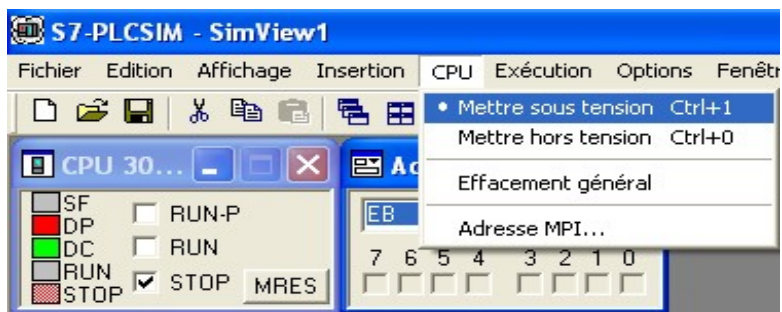


Figure V.2 : Mise sous tension de la CPU.

✚ Choisir la commande Exécution-Mode d'exécution et vérifier que la commande cycle continue est activée (figure V.3).

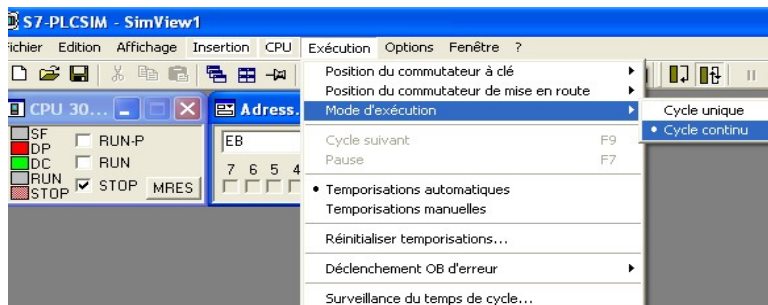


Figure V.3 : Choix du cycle continu.

du cycle continu.

Mettre la CPU de simulation en marche en cliquant sur l'une des cases à cocher RUN ou RUN-P (figure V.4).



Figure V.4 : Mise en marche de la CPU.


Une fois toutes les fenêtres d'entrées et de sorties sont prêtes, nous activons les entrées voulues pour lire l'état des sorties (figure : V.5). Dans notre cas, pour aspirer la pièce par le chargeur, c'est-à-dire descendre le bloc ventouses chargeur et aspirer la pièce.





Figure V.5 : Simulateur du S7-PLCSIM.

### I.3. Visualisation de l'état du programme :

Après le chargement du programme dans la CPU du simulateur et la mise de cette dernière en mode « RUN » le STEP7 nous permet de visualiser l'état du programme soit en cliquant sur l'icône  ou sélectionnant la commande Test-Visualiser (figure V.6).

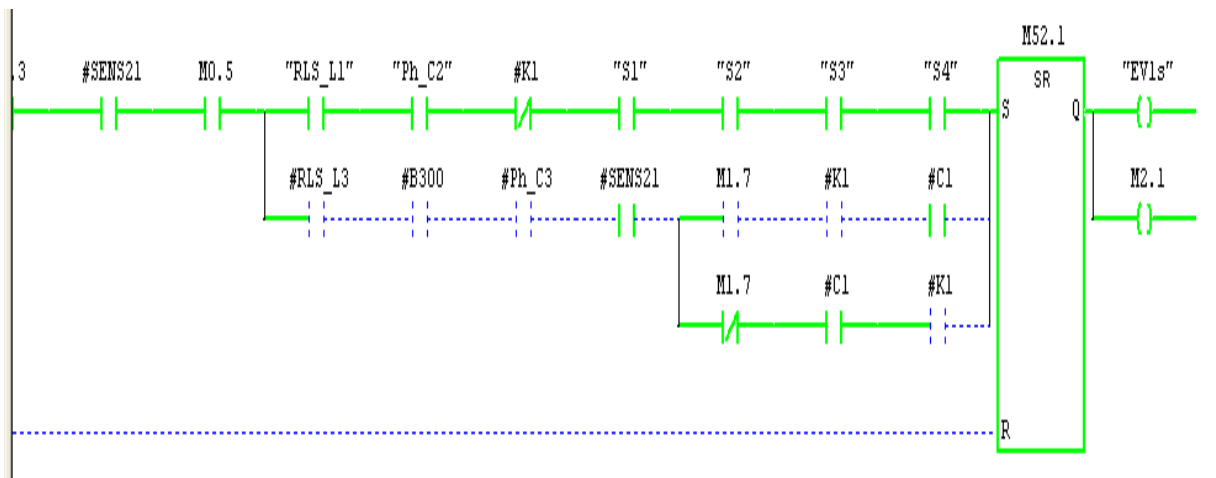


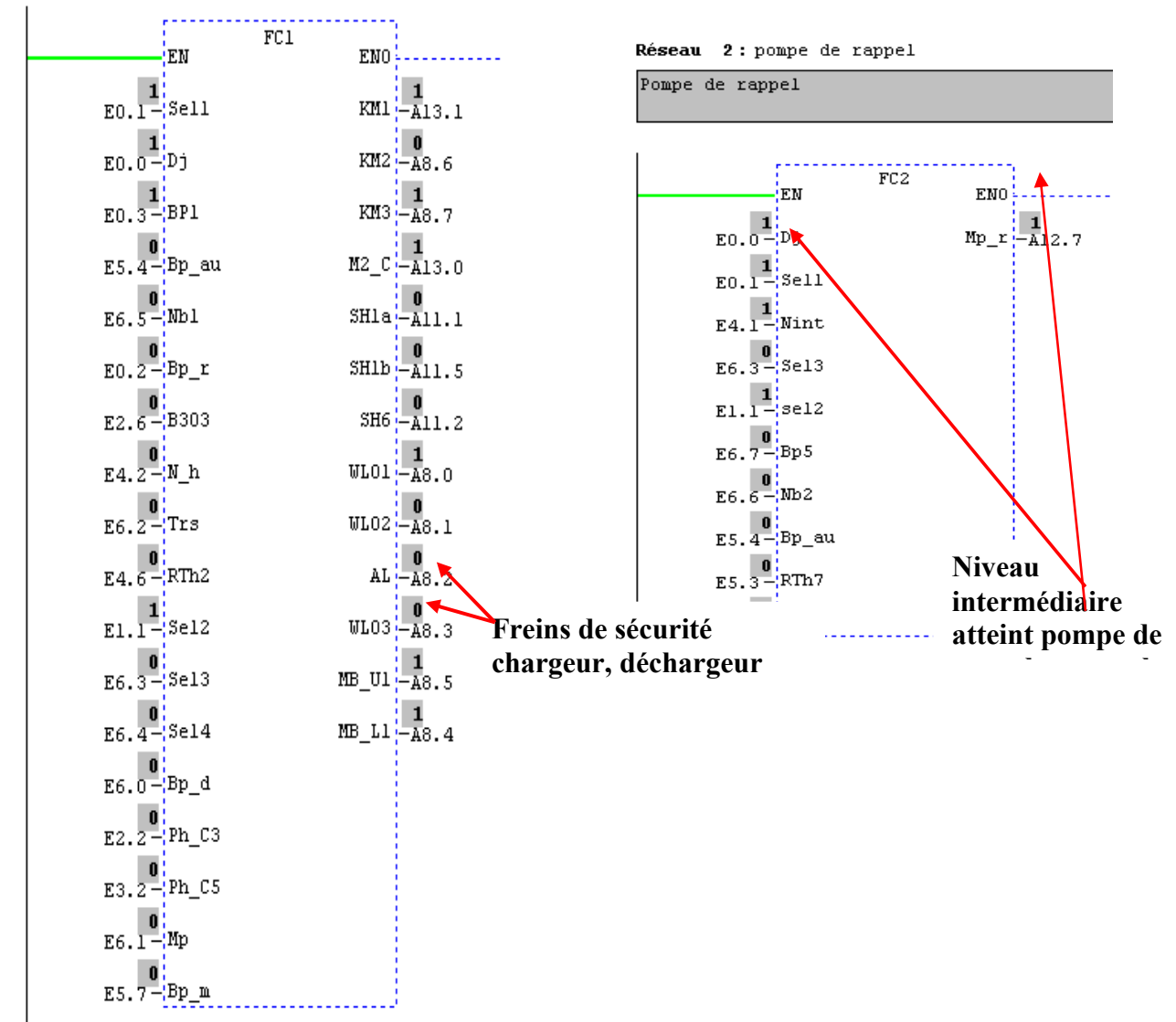
Figure V.6 : Visualisation de l'état du programme.

### I.4. Simulation du programme de la machine réadaptée :

La simulation du programme de la machine est faite en deux étapes :

- Etape 1 : simulation du programme par bloc, c'est-à-dire charger chaque FC tout seul puis effectuer la simulation.
- Etape 2 : simulation du cycle complet, c'est-à-dire charger tous les blocs FC puis effectuer la simulation du cycle.

**Exemple :** L'exemple suivant (figure : V.7) montre la simulation du programme qui commande la machine avec le logiciel de simulation S7-PLCSIM.



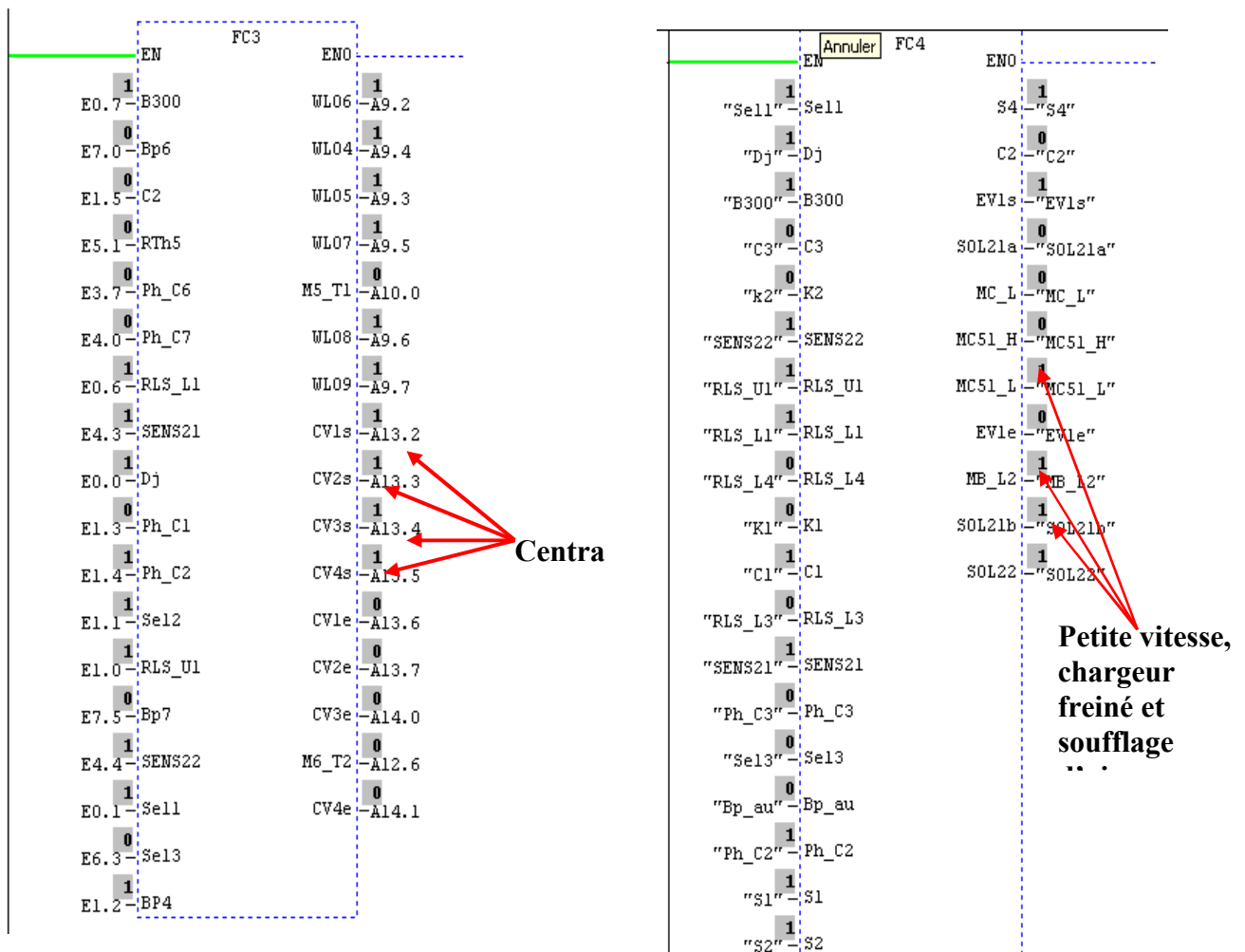


Figure V.7 : Visualisation du programme de la machine au niveau de l'OB1.

### I.5. Conclusion :

L'automatisation de la machine réadaptée de découpage et poinçonnage des contres portes frigorifiques avec l'API S7-300 nous procure plusieurs avantages tels que la flexibilité, la facilité d'extension de ses modules et la possibilité de visualisation et validation du programme établi avant son implantation sur l'automate grâce à son logiciel de simulation S7-PLCSIM.

Les actions de chaque équipement de la machine sont programmées dans un FC dans le but de repérer et rendre facile les modifications à apporter, alors que les FC programmés sont insérés dans le bloc d'organisation OB1.

L'utilisation des bascules SR (S : condition d'activation, R : remise à zéro) nous permet d'activer et de désactiver chaque action et nous permettra de figer l'automate en cas de défaillance puis continuer l'exécution du programme après maintenance.

Le logiciel de simulation S7-PLCSIM nous a permis de tester la solution programmée que nous avons développée pour la commande de la machine, de corriger toutes les erreurs que nous avons commises et apporter les modifications nécessaires, comme il nous a permis de valider et de visualiser le comportement des sorties.

## **II. Introduction :**

Autrefois, pour pouvoir suivre chaque phase du procédé et intervenir dans le cas échéant, il fallait câbler les voyants, interrupteurs et boutons poussoirs. Lorsqu'il s'agissait de procédés complexes, il fallait avoir recours à des synoptiques coûteux. Ces solutions appartiennent maintenant au passé. En effet, avec le développement de l'informatique il est devenu possible de traiter des données dans le domaine industriel, grâce à des vues préalablement créées et configurées, et à l'aide d'un logiciel adéquat.

Le logiciel de supervision est une entité capable de présenter à l'opérateur des informations utiles, afin qu'il prenne à temps les bonnes décisions pour la conduite du procédé. Il a essentiellement pour mission de collecter les données (acquisition et stockage) et les mettre en forme (traitement), afin de les présenter à l'opérateur (supervision).

### **II.1. Emplacement de la supervision :**

La supervision se situe au plus haut niveau dans la hiérarchie des fonctions de production, il est donc essentiel de présenter à l'opérateur sous forme adéquate les informations sur le procédé, indispensables pour une éventuelle prise de décision. Cette présentation passe par les images synthétiques qui représentent un ensemble de vues. Le processus est représenté par une synoptique comprenant des images et objets animés, par l'état des organes de commande et les valeurs transmises par les capteurs. Outre la synoptique, on trouve aussi des vues d'alarme, de statistique, de régulation...etc.

### **II.2. Constitution d'un système de supervision :**

La plus part des systèmes de supervision se composent d'un moteur central (logiciel), auquel se rattachent des données provenant des équipements (automates). Le logiciel de supervision assure l'affichage, le traitement des données, l'archivage et la communication avec d'autres périphériques.

### **II.3. Module de visualisation :**

Il permet d'obtenir et de mettre à la disposition des opérateurs des éléments d'évaluation du procédé par ses volumes de données instantanées.

### **II.4. Module d'archivage :**

Il mémorise des données (alarme et événement) pendant une longue période, et permet l'exploitation des données pour des applications spécifiques à des fins de maintenance ou de gestion de production.

### **II.5. Module de traitement :**

Il permet de mettre en forme les données afin de les présenter via le module de visualisation aux opérateurs sous une forme prédéfinie.

## II.6. Module de communication :

Il assure l'acquisition et le transfert de données et gère la communication avec les automates programmable industriels et autre périphériques (figure V.8).

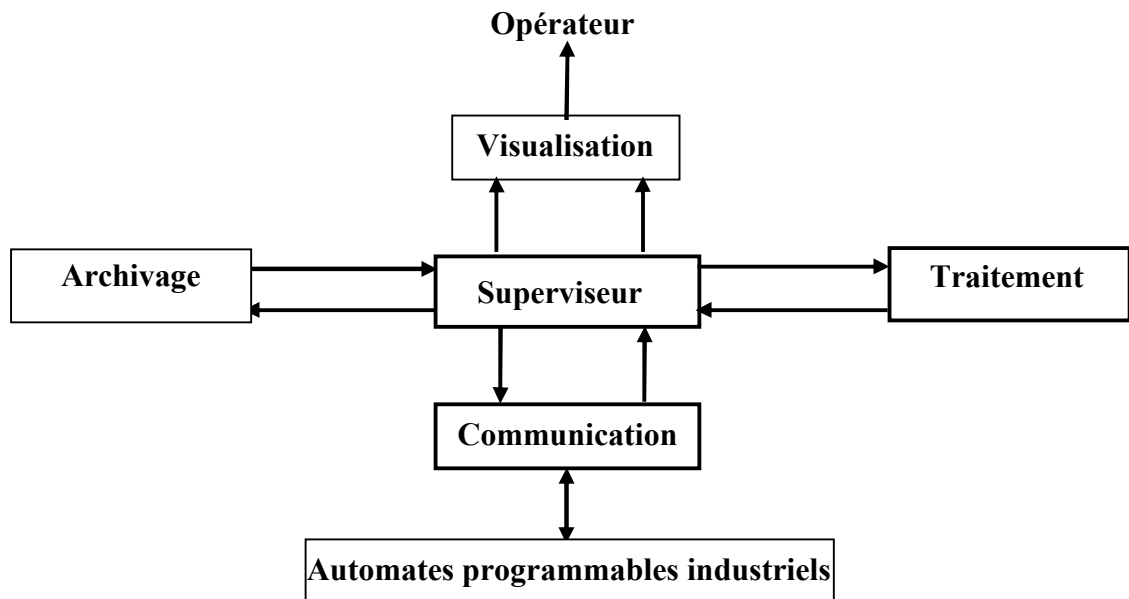


Figure V.8 : Principe de communication.

## II.7. Apport de la supervision :

La supervision a eu un impact considérable sur le monde industriel, tant pour les exploitants que pour les entreprises.

- **Apport pour le personnel :**

-La supervision permet de dégager les exploitants des tâches délicates, surtout dans des milieux hostiles, et de ne les réserver que pour des tâches importantes. Elle permet de rendre le travail moins contraignant pour celui qui l'exécute et améliore les conditions de travail.

-La supervision permet à l'opérateur de suivre le fonctionnement du procédé et d'effectuer des tâches de routine (vérification des paramètres, inspection de l'installation...).

- **Apport pour l'entreprise :**

L'effet de la supervision sur l'entreprise est considérable, elle permet entre autre :

-Respecter les délais en diminuant le nombre de pannes et en réduisant le nombre de dépannages.

-Améliorer et maintenir la qualité, ceci se fait par le maintien des équipements dans un bon état de fonctionnement.

-Réduire les coûts en diminuant les pertes de production liées aux pannes.

## **II.8. Application développée sous WinCC :**

Le programme de supervision que nous avons développé a été élaboré avec le logiciel WinCC (Windows Control Center), développé par SIEMENS.

Il est caractérisé par sa flexibilité c'est-à-dire qu'il peut être utilisé pour les composants hors SIEMENS. Il nous permettra la visualisation du fonctionnement de la machine à tout moment de fonctionnement.

WinCC constitue la solution de conduite et de supervision de procédés sur ordinateur, pour système monoposte et multiposte, et permet le transit des informations sur l'Internet.

Il offre une bonne solution de supervision car il met à la disposition des opérateurs des fonctionnalités adaptées aux exigences courantes des installations industrielles.

## **II.9. Procédure de programmation avec WinCC :**

Les principales étapes suivies pour créer notre application sous WinCC sont :

1. Créer un projet.
2. Sélectionner et installer l'API.
3. Définir les variables dans l'éditeur stock de variable.
4. Créer et éditer les vues (vue d'accueil, vue de tous les blocs) dans l'éditeur Graphics Designer.
5. Paramétrer les propriétés de WinCC runtime.
6. Activer les vues dans le WinCC runtime.
7. Utiliser le simulateur pour tester les vues du process.

On présentera la procédure que nous avons suivie pour réaliser la supervision de la chaîne.

Le projet monoposte créé est appelé ENIEM.

Puis on procède à la configuration du système de supervision pour assurer la communication entre l'API S7-300 avec WinCC. Pour ce faire nous avons sélectionné à partir de l'éditeur stock de variable, le pilote « SIMATIC S7 Protocol suite » et choisi MPI comme réseau de communication.

L'étape suivante est l'introduction des variables de procès. Ces variables correspondent à des variables manipulées par le programme de l'API S7-300.

Une telle variable peut être par exemple la position du coulisseau de la presse, délivrée par les deux capteurs de position inférieur ou supérieur est enregistrée dans l'API S7-300 pour être enfin communiquée à WinCC.

L'étape d'après est la création des vues dans l'éditeur « Graphic designer », ce dernier nous a permis d'insérer les différents types d'objets dont on a besoin, grâce à la palette d'objet et la bibliothèque interne du WinCC.

Pour ce faire on crée la vue d'accueil, qui contient les boutons de navigation à partir desquels on peut choisir la vue à visualiser parmi :

- Le chargeur.
- La presse hydraulique.



Figure V.9 : Vue d'accueil.

Après la création de la vue d'accueil on fera de même pour les autres vues. Ensuite on doit configurer les boutons qui serviront à basculer de la vue d'accueil vers les autres vues, aussi de ces autres vues vers la vue d'accueil (figure V.10).



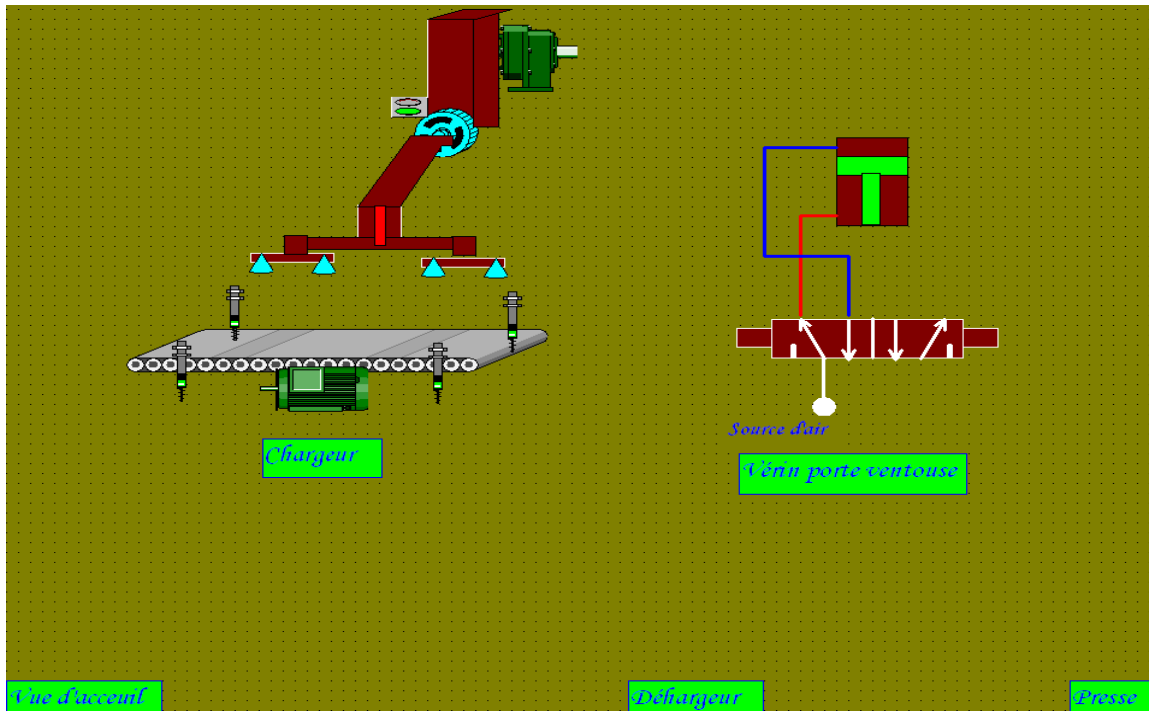


Figure V.10 : Vue du chargeur.

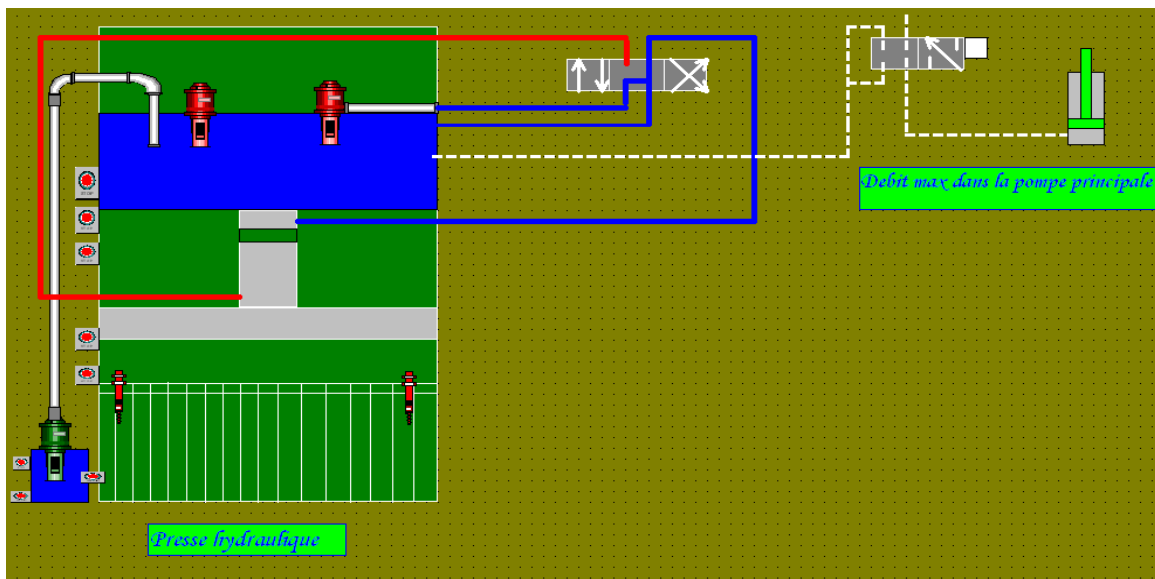


Figure V.11 : Vue de la presse hydraulique.

Une fois les vues réalisées et la configuration des boutons de navigation effectuées, on passera à la dynamisation des objets en leur affectant les variables correspondantes.

## II.10. Conclusion :

Dans ce dernier chapitre nous avons décrit la supervision en précisant sa place dans l'industrie, ou nous avons élaboré sous le logiciel WinCC les vues qui permettent de suivre l'évolution du procédé en fonction du temps.

## **Conclusion générale :**

Notre projet de fin d'étude est effectué en grande partie au sein de l'entreprise d'électroménager (ENIEM) zone industrielle Oued-Aissi dans le but de concevoir une solution programmable pour la presse de fabrication des portes de réfrigérateur. L'automatisation est un domaine pluridisciplinaire qui associe les notions de la mécanique, de l'électronique et de l'informatique ; elle permet d'exécuter des tâches industrielles avec une intervention humaine très réduite. A l'issue de notre travail, nous pouvons conclure que :

Le GRAFCET et les outils qui lui sont associés ont apporté bien des progrès en matière de méthodologie d'élaboration de cahier des charges, de réalisation et de programmation des systèmes automatisés. Pour être à la page avec les exigences introduites par l'évolution des industries, la commande des processus avec un automate programmable industriel est la solution souvent recherchée, vu la justesse des traitements que les API effectuent.

L'évolution des API ne cesse de continuer et notamment leurs logiciels de programmation ; l'API S7-300 procure plusieurs avantages tels que la flexibilité, la facilité d'extension de ses modules et la possibilité de visualisation et validation du programme établi avant son implantation grâce à son logiciel de simulation S7-PLCSIM. Une bonne automatisation d'un procédé doit être performante et d'un coût optimal, cela est obtenu en passant par :

- ✓ L'élaboration d'un cahier des charges qui comprend tous les aspects fonctionnels du processus.
- ✓ La modélisation du cahier des charges par un des outils de modélisation par exemple le GRAFCET.
- ✓ Le choix optimal de la partie commande (API), de la partie opérative (actionneurs) et les moyens de dialogues (capteurs).

Ce que nous retenons du travail que nous avons effectué durant ces trois mois à l'ENIEM est primordial dans notre vie d'ingénieur. Ce stage pratique nous est bénéfique à plus d'un titre compte tenu des nombreux avantages qu'il présente. La découverte du monde industriel, la mise en application de la théorie acquise lors de notre cursus.

Nous espérons que notre travail verra naître sa concrétisation sur le plan pratique et que les promotions à venir puissent en tirer profit.