

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

Université Mouloud MAMMERI de Tizi-Ouzou
Faculté de Génie Electrique et d'Informatique.
Département Automatique



Mémoire de fin d'études

Présenté en vue de l'obtention
Du diplôme de Master en Automatique

Option :automatique et informatique industrielle

Thème

Automatisation d'une machine pour obtenir les contours
de grille de table par un automate programmable S7-300
à l'ENIEM

Promoteur : M MELLAH.R

Encadreur : MBOUGUEDOUR.B

Présenté par :

M BELLEBIA ALI

Promotion :2014-2015

REMERCIEMENTS

A l'issue de ce modeste travail, je tiens à exprimer ma reconnaissance, ma gratitude et mes vifs remerciements à Mr MELLAH.R, à Mr BOUGUEDOUR.B et Mr CHARIF.M pour m'avoir apporté leur connaissance et un soutien permanent pendant mon stage. Leur soutien moral et leur aide précieuse m'a permis d'effectuer ce travail dans les meilleures conditions.

Je tiens également à remercier tout ceux qui m'ont aidé de prêt ou de loin pour l'élaboration de ce projet, les enseignants du département automatique et électronique de l'UMMTO, ainsi que tous les travailleurs de l'unité cuisson de l'ENIEM.

Dédicaces

Je dédie ce modeste travail :

A la mémoire de ma chère mère.

A mon grand-père.

A ma sœur.

A mes oncles.

A mes tantes.

A tout mes amis.

Ali

ENIEM (l'entreprise Nationale des Industries de l'Électroménager) est une entreprise publique économique, constitué le 2 janvier 1983 à partir de fonctions déjà existantes au sein de l'entreprise SONELEC (Société nationale de fabrication et de montage de matériel électrique et électronique). Son siège social se situe au centre de la zone industrielle «Oued Aissi à une dizaines de kilomètres de la ville de Tizi-Ouzou.

L'ENIEM est une société par actions depuis le 8 octobre 1989 avec un capital de 10.279.800.000 DA. Elle emploie actuellement 2 300 travailleurs au niveau de ses unités.

L'Entreprise nationale des industries de l'électroménager (ENIEM) ne cesse d'innover, elle possède de grandes capacités de production et une expérience de plus de 30 ans dans la fabrication et le développement dans les différentes branches de l'électroménager.

L'ENIEM a pour fonction la recherche et le développement dans le domaine de l'électroménager, la satisfaction du consommateur par la mise sur le marché d'un produit de qualité répondant à ses besoins. Elle a aussi une autre activité qui consiste. En le montage de petits appareils électroménager domestique (robots de cuisine, moulins à café, mixeurs, etc.), elle produit tous types de cuisinières, de réfrigérateurs, des congélateurs et des climatiseurs, des appareils de cuisson, de lavage, ainsi que les lampes à incandescence.

L'Entreprise s'est organisée par centres d'activités stratégiques qui se composent de 03 unités de production (cuisson, froid et climatisation sont implantées dans la zones industrielle de Tizi-Ouzou), d'une (01) unité commerciale et d'une (01) unité de prestations techniques ainsi que deux (02) filiales dont le capital est à 100% ENIEM, La filiale sanitaire est installée à Miliana, wilaya de Ain Defla, et la filiale lampe à Mohammadia, wilaya de Mascara.

-Unité cuisson : qui produit des cuisinières, 4 et 5 feux.

- Unité climatisation: spécialisée dans les climatiseurs, chauffe-eau/bain et radiateurs gaz butane.

- Unité froid : qui fabrique des réfrigérateurs petit et grand modèle dotés d'une ou deux portes, combinés réfrigérateurs congélateurs ainsi que des congélateurs verticaux et horizontaux.

-Une unité de prestation technique (UPT) : avec une ligne d'usinage mécanique, une ligne d'usinage par électroérosion ainsi qu'une ligne de traitement thermique. Cette unité est également chargée de produire toutes les utilités nécessaires aux centres de production dont elle assure la maintenance centrale.

-Une unité de prestation de services (UPS) : qui assure la gestion des moyens généraux de toutes les unités dont elle assure également la sécurité.

Le marché national de l' électroménager a connu une forte concurrence, notamment le réfrigérateur et la cuisinière. Pour faire face a cette dernière ENIEM a était obligé de suivre le marché, et à chercher des partenaires pour développer de nouveaux produits, d'automatiser ses différentes installations, tel que l'entreprise à signé des contrats de travail avec des firmes internationales en vue d'améliorer la qualité du produit. C'est en 1987 que l'ENIEM a commencé l'automatisation grâce à Toshiba.

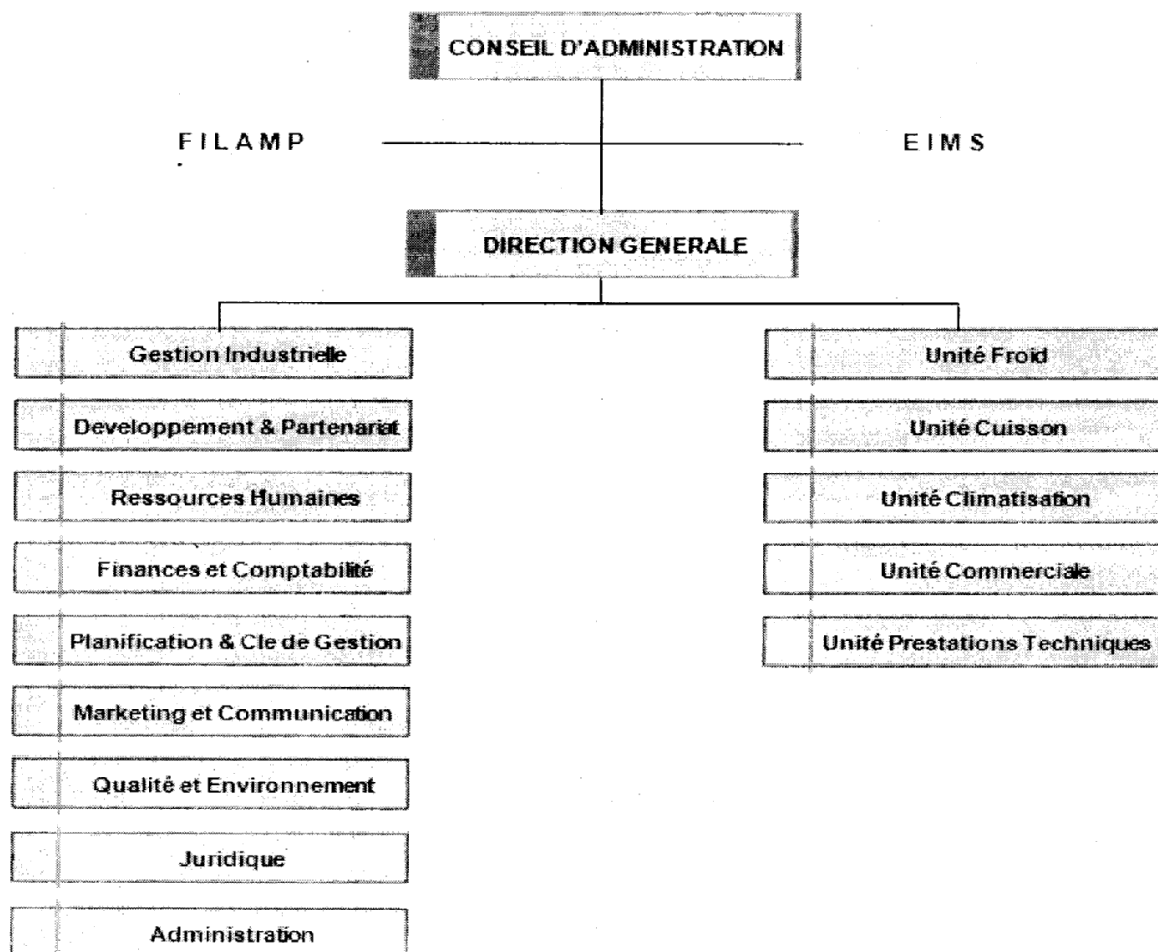


Fig : organisation générale de l'entreprise

II. Présentation de l'unité cuisson

L'unité cuisson est spécialisée dans la production des différents types de cuisinières. Elle est équipée de moyens de production répartis en quatre ateliers

Un atelier mécanique pour la fabrication de composants d'alimentation en gaz, des grilles de cuisinières et des pièces en tôle équipé

- de presses de 30 à 150 T.
- d'une presse de transfert de 630 T.
- de machines automatiques à découper et cintrer les tubes.

Un atelier de traitement de revêtement de surface équipé :

- d'une installation de zingage.
- d'une installation de nickelage et chromage.
- d'une installation de préparation des métaux.
- de deux installations d'émaillage.

Un atelier d'assemblage équipé de:

- une chaîne de montage de portes.
- deux chaînes de montage de cuisinières.
- postes de préparation d'ensembles.

Sommaire

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

Chapitre I : Généralité sur les systèmes automatisés

Préambule.....	2
I. Définition et notions fondamentales d'un système automatisé.....	2
I.1. Définition D'un système automatisé de production	2
I.2. Objectifs de l'automatisation	3
I.3. Structure d'un système automatisé de production	3
I.4. Cahier des charges	4
II. Définition d'un automate programmable industriel API.....	4
III. Architecture d'un API	5
III.1. L'alimentation	6
III.2.1. Mémoire	6
III.3. Interfaces d'entrées	7
III.4. Interfaces de sorties.....	8
III.5. Module logique et analogiques des entrées sorties	9
III.6. Interface de communication	10
IV. Principe de fonctionnement d'API.....	11
V. Choix d'un automate programmable industriel.....	12
V.1. Situation de l'unité de traitement	13
V.2. Choix de l'unité de traitement	13
Discussion	13

Chapitre II : Description du système à automatiser

Préambule.....	14
I.Présentation de la machine.....	14
I.1. La première partie	15
I.1.1. Groupe d'avancement	15
I.1.2. Groupe à redresser le fil.....	16
I.2. La deuxième partie.....	16

I.2.1. Groupe presses	17
I.2.2. Groupe coupe ruban	17
I.2.3. Groupe transport pièce	17
I.3. La troisième partie	18
I.3.1. Groupe de cintrage	18
I.3.2. Groupe de soudure	19
I.4. La quatrième partie	19
I.5. Dérouleur	19
II Composante de la machine	20
II.1. Système d’approvisionnement en pression pneumatique	20
II.1.1. Caractéristique de la source d’énergie	20
II.1.2. Structure d’une installation pneumatique	20
II.1.2.1. Traitement de l’air	21
II.2. Système d’approvisionnement en pression hydraulique	23
II.3. Définition et choix des composants utilisés	27
II.3.1. Partie opérative	27
II.3.1.1. Pré-actionneur	27
II.3.1.1.1. Electrovanne	27
II.3.1.1.2. Distributeurs	28
II.3.1.1.3. Les relais	28
II.3.1.1.4. Contacteurs	29
II.3.1.2. Actionneurs	29
II.3.1.2.1. Vérins	29
II.3.1.2.2. Les moteurs	30
II.3.1.2.2.1. Moteur asynchrone	30
II.3.1.2.2.2. Moteur hydraulique	32
II.3.1.2.2.3. Servovalve	33
II.3.1.3. Les capteurs	33
II.3.1.3.1 Les pressostats	34

II.3.1.3.2. Le détecteur de proximité	34
II.3.1.3.3. Codeur incrémental.....	34
II.3.1.3.4. Flotteurs	35
II.3.1.3.5. Plongeur.....	35
II.3.2. La partie liaison	36
II.3.3. La partie commande	37
II.3.3.1. Programmeur à cames.....	38
III. Le cycle de la machine	40
III.1. Etampage bosses	40
III.2. Avancement et redressage du ruban.....	41
III.3. Coupage ruban.....	41
III.4. Placement du ruban dans la zone de cintrage.....	41
III.5. Cintrage	42
III.6. Soudure.....	42
III.7. Extraction	43
Discussion	43

Chapitre III : Modélisation par le grafcet

Préambule.....	44
I. Les différents langages	44
I.1. Instruction LIST (IL)	44
I.2. Structured Text (ST)	45
I.3. Function Block Diagram (FBD)	45
I.4. Ladder Diagram (LD)	45
I.5. Sequential Function Charts (SFC)	46
II. Définition d'un GRAFCET	47
III. Niveau d'un GRAFCET	47
III.1. GRAFCET du niveau 1	47
III.2. GRAFCET du niveau 2	47

IV. Eléments de base d'un GRAFCET	47
V. Syntaxe et règles d'évolution	49
V.1. Syntaxe	49
V.2. Règles d'évolution.....	49
VI. Structure de base	50
VI.1. Divergence et Convergence en ET	50
VI.2. Divergence et Convergence en OU.....	51
VI.3. Saut en avant (saut d'étape)	51
VI.4. Saut en arrière (reprise de séquence)	52
VII. Mise en équation d'un GRAFCET	53
VIII. Le passage du GRAFCET en langage Ladder.....	53
IX. Les modes de marche.....	54
X. Application du GRAFCET pour modéliser le système	54
X.1. GRAFCET niveau 1	54
Discussion	62

Chapitre IV : Description et choix de l'automate

Préambule.....	63
I. Les critères de choix de l'automate S7-300	63
II. Présentation de l'automate programmable S7-300.....	63
II.1. Caractéristique de l'automate S7-300.....	63
II.2. Constitution de l'automate S7-300	64
II.2.1. Mode d'alimentation (PS)	64
II.2.2. L'unité centrale (CPU)	65
II.2.3. Coupleur (IM).....	66
II.2.4. Module de communication (CP).....	67
II.2.5. Module des signaux (MS).....	67
II.2.6. Châssis d'extension (UR)	67
II.2.7. Module de simulation (S7-300).....	67

II.3. Console de programmation PG ou PC avec le logiciel de simulation.....	67
III. Mode de fonctionnement de l'automate S7-300	67
III.1. Réception des informations sur l'état du système	67
III.2. Exécution du programme utilisateur	68
III.3. La commande de processus	68
IV. Programmation de l'automate S7-300	68
IV.1. Création d'un objet STEP7	68
V. Traitement du programme par l'automate	71
VI. Le rôle des systèmes d'exploitation d'une structure de programme.....	72
VI.1. Le rôle du système d'exploitation	72
VI.2. Le rôle du programme utilisateur	72
VI.3. Le bloc dans le programme utilisateur	72
VI.4. Les différents blocs du programme utilisateur.....	73
VI.4.1. Blocs d'organisations	73
VI.4.2. Le bloc fonctionnel (FB)	73
VI.4.3. Fonction (FC)	73
VI.4.4. Bloc de données (DB)	73
Discussion	74

Chapitre V : Simulation avec le logiciel S7-PLCSIM

I. Logiciel de simulation S7-PLCSIM.....	75
II. Mise en route du logiciel S7-PLCSIM	75
III. Visualisation de l'état du programme	76
Discussion	76
Conclusion générale	77

Introduction générale

Introduction générale

La concurrence croissante à la quelle font face les entreprises d'aujourd'hui, leur exige de fabriquer des produits de meilleure qualité en grande quantité avec le moindre coût. De ce fait, ces entreprises ont recours aux Systèmes Automatisés de Production (SAP) afin d'atteindre les exigences du marché mondial en termes de qualité, de rapidité et de performance.

Dans ce contexte, notre travail, consiste en l'étude et l'automatisation d'une soudeuse à grille de table, installée à l'unité cuisson de l'entreprise ENIEM (Entreprise Nationale des Industries Electroménagers).

Après avoir étudié et analysé cette machine nous avons constaté qu'une automatisation s'avère nécessaire afin de pallier à certains problèmes notamment ceux liés au rendement. Dans ce sens, notre travail, consiste en l'automatisation d'une soudeuse à grille de table à l'aide d'un automate S7-300 de la firme SIEMENS.

Le SIMATIC constitue une vaste plate forme d'automatisation offrant des solutions à des problèmes complexes pour tous les secteurs d'activité. Le logiciel STEP7 a été conçu dans un souci d'homogénéité et de complémentarité avec un système de contrôle et de commande, offrant des fonctions conviviales de conduite et de simulation du processus, ce qui simplifie d'une manière considérable la mise en œuvre de nombreuses caractéristiques du système de commande, notamment la gestion de base de données communes.

A cet effet nous avons organisé notre mémoire comme suit:

Le **premier chapitre** présente une vue d'ensemble sur les systèmes automatisés, la description et le mode de fonctionnement sont abordés dans le **deuxième chapitre**. Dans le **troisième chapitre** on effectue la modélisation de la machine en faisant appel à l'outil de modélisation qui est le **GRAFCET** (Graphe fonctionnel de Commande Etape-Transition), on utilise pour la programmation l'automate programmable S7-300 décrit dans le **quatrième chapitre**, ainsi on procède à la simulation à l'aide du logiciel S7-PLCSIM présenté dans le **cinquième chapitre**. Enfin nous terminons par une conclusion.

-

Chapitre I
Généralité sur les
systèmes automatisés

Préambule :

Un système automatisé de production (SAP) est un système de production qui reçoit un flux de matière ou de produits génère un flux de produits plus élaborés (moulés, usinés, assemblés, testés...etc.). Il doit aussi gérer l'alimentation en énergie, ainsi que des flux auxiliaires tels les consommables, les déchets. Tout cela, ajouté à des exigences sans cesse accrues de qualité, sécurité, fiabilité qui entraîne un accroissement des besoins, en particulier la manipulation d'un grand nombre de variables et la gestion de véritables flux de communication. Cela explique que les systèmes câblés deviennent trop volumineux et trop rigides pour de telles applications, et que l'on se tourne donc vers des solutions utilisant les techniques de traitement de l'information par processeurs programmables.

I. Définition et notions fondamentales d'un système automatisé**I.1 Définition d'un système automatisé de production**

L'automatisation consiste à « rendre automatique » les opérations qui exigeaient auparavant l'intervention humaine

L'automatisation est considérée comme l'étape d'un progrès technique où apparaissent des dispositifs techniques susceptibles de seconder l'homme.

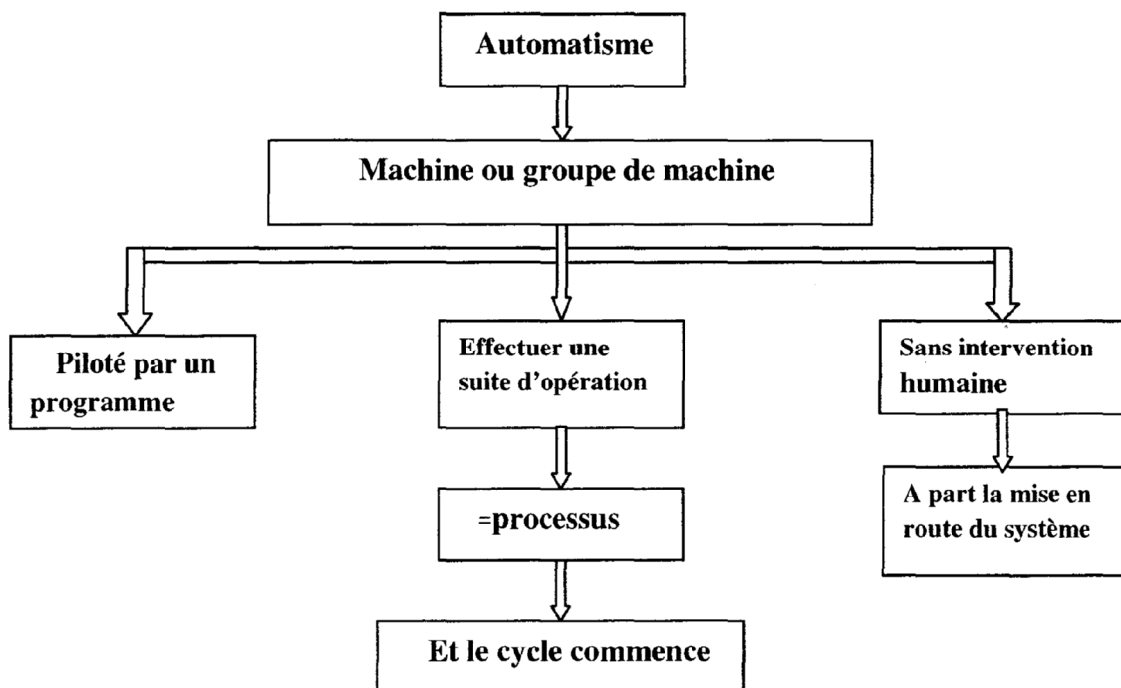


Figure I.1 : Le système automatisé

I.2 Objectifs de l'automatisation

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

I.3 Structure d'un système automatisé de production

Un système automatisé peut se décomposer en deux parties principales:

❖ La partie opérative:

- C'est le processus physique à automatiser.
- C'est elle qui agit sur la matière d'œuvre ou le produit.
- Elle comporte en général des actionneurs, des capteurs, outillages mécaniques permettent le processus d'élaboration.
- **Actionneurs:** ils exécutent les ordres reçus, ils agissent sur le système ou son environnement.
- **Capteurs:** sont capable de détecter un phénomène physique dans son environnement (déplacement, pression, chaleur, lumière....).

❖ La partie commande:

C'est l'automatisme qui élabore en sortie des ordres destinés au processus et en reçoit des informations en retour, afin de coordonner ses actions.

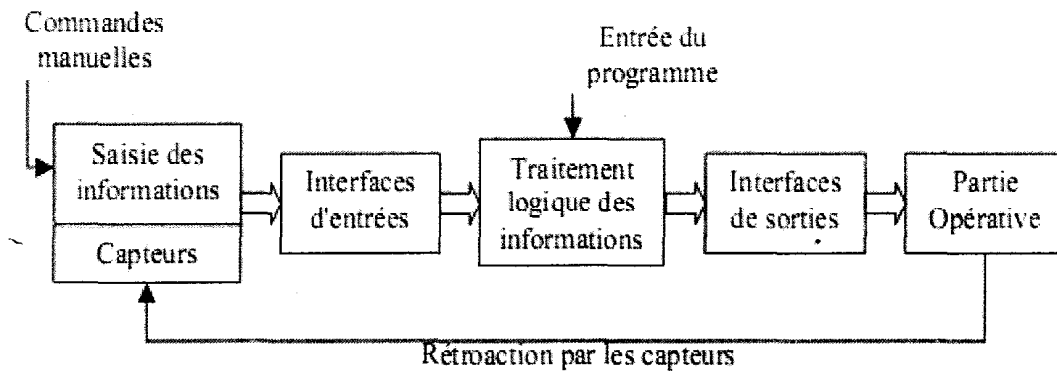


Figure I.2: structure d'un S.A.P

I.4 Cahier des charges

Le cahier des charges d'un système automatisé est la description de son comportement en fonction de l'évolution de son environnement (capteur, actionneurs....).

Il décrit:

- Ce qui se passe entre la partie opérative et la partie commande du système automatisé.
- Les conditions générales d'utilisation de l'automatisme dans l'environnement du système automatisé et dans toutes les phases de sa vie.

II. Définition d'un automate programmable industriel API

L'automate programmable est un dispositif électronique de traitement logique d'information dont le programme de fonctionnement est effectué à partir d'inductions établies en fonction du processus à réaliser.

L'API est destiné industriellement à la gestion de processus séquentiels ou combinatoires.

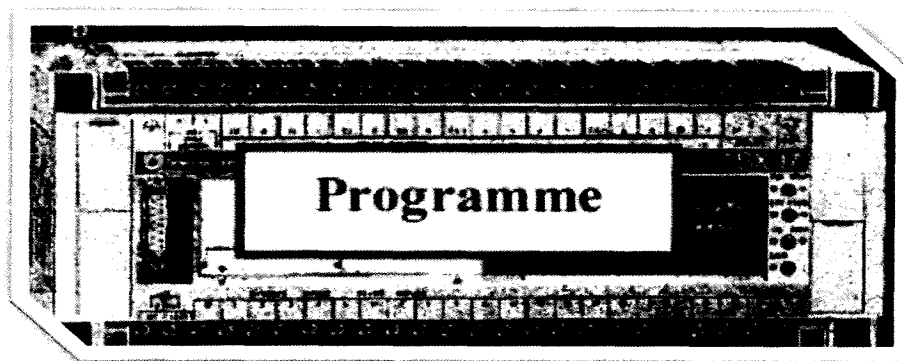


Figure I.3: un automate programmable industriel

❖ Caractéristiques techniques:

Les caractéristiques principales d'un API sont:

- Compact ou modulaire;
- Tension d'alimentation;
- Taille mémoire;
- Temps de scrutation;
- Sauvegarde (EPROM, EEPROM, Pile);
- Nombre d'entrées /sorties;
- Module complémentaire (analogiques, communication);
- Langage;

III. Architecture d'un API

À partir d'information d'entrées (signaux issus de capteur, bouton poussoir, clavier, signaux numériques de communication) ; l'API assure la commande d'actionneur selon un programme de séquence défini par l'utilisateur.

Pour remplir correctement la fonction d'un API, l'automate intègre des fonctions de base incontournables.

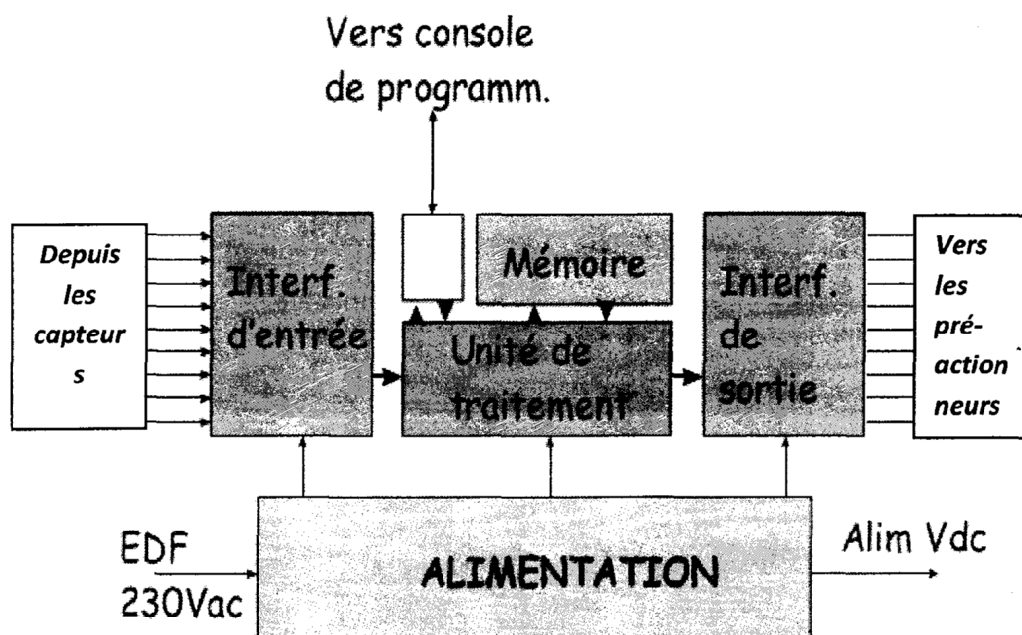


Figure I.4 Architecture matérielle d'un API.

III.1 L'alimentation

Fournit en énergie électrique les différents organes de l'automate et accessoirement les capteurs externes. Elle 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 transformateurs supplémentaires.

III.2 L'unité centrale CPU

L'unité centrale est constituée du processeur et de la mémoire centrale.

La tension provenant des capteurs est appliquée aux bornes du module d'entrée. Le processeur de l'unité centrale traite le programme qui se trouve dans la mémoire et interroge l'état des entrées pour s'avoir si la tension est présente ou non, et en fonction de l'état de ces entrées et du programme en mémoire, le processeur instruit le module de sortie afin qu'il applique la tension aux connecteurs correspondants.

❖ Caractéristiques principales

-Vitesse de traitement : c'est la vitesse de l'UC pour exécuter 1K-instructions logiques (10 à 20 ms/K mots).

-Temps de réponse : scrutation des entrées, vitesse de traitement et affectation des sorties

III.2.1 Mémoire

Un circuit électronique complexe qui n'enregistre que des données binaires (données élémentaires appelées « bits »).

La mémoire centrale est l'élément fonctionnel qui peut recevoir, conserver et restituer. Elle est découpée en zones où l'on trouve:

- La zone mémoire programme (programme à exécuter);
- La zone mémoire des données (état des entrées et des sorties, valeurs des compteurs, temporisations)
- Une zone où sont stockés des résultats de calcul utilisés ultérieurement dans le programme;
- Une zone pour les variables internes.

❖ **Durant la phase d'étude et de mise au point du programme:**

- Des mémoires vives RAM (Random Access Memory) volatiles
- Des mémoires EAROM (Electrically Alterable Read Only Memory) non volatiles et effaçables partiellement par voie électrique.

❖ **Durant la phase d'exploitation:**

- Des mémoires vives RAM imposent un dispositif de sauvegarde par batterie rechargeable pour éviter la volatilité de leurs contenus en cas de coupure de courant.
- Des mémoires mortes ROM à lecture seulement ou PROM programmables à lecture seulement.
- Des mémoires reprogrammables EPROM (Erasable PROM) effaçables par un rayonnement ultra-violet et EEPROM (Electric Erasable PROM) effaçables électriquement.
- **Attribution des zones mémoire travail en RAM**

Nature des Inform.	Désignations	Exploitation	Zones Mémoires
Etats des Capteurs	Variable d'entrée	Evolution de leur valeur en fonction du déroulement du cycle	Zone mémoire des Données
Ordres aux préactionneurs	Variable de sortie		
Résultats de fonctions comptage, tempo...	Variable Interne et / ou Variable mot		
Résultats intermédiaires			
Instructions du cycle dans l'API	Programme	Ecrit 1 fois et lu à chaque scrutation	Zone mémoire PROGRAMME

• **Sauvegarde :**

Sauvegarde de la RAM (programmes, configuration, données)		Sauvegarde Externe (programme, configuration)
1 heure minimum par pile interne	1 an par pile externe	permanente par EPROM (effaçable par ultraviolet), EEPROM (effaçable par courant électrique)....

III.3 Interfaces d'entrées

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

Elles sont destinées à

- Recevoir l'information en provenance des capteurs
- Traiter le signal en éliminant les parasites et en isolant électriquement l'unité de commande de la partie opérative.

- transformer les signaux logiques ou analogiques provenant des capteurs en informations numériques exploitables par l'unité de traitement

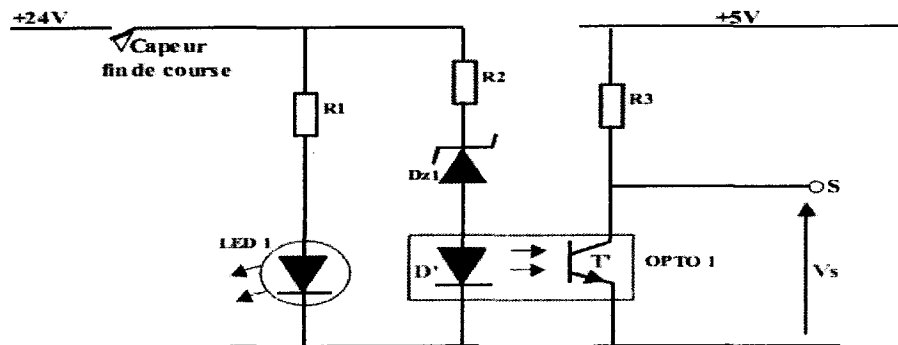


Figure I.5 : Interface d'entrée

❖ Fonctionnement de l'interface d'entrée

Lors de la fermeture du capteur:

- LED 1 signale que l'entrée automate est actionnée;
- La led D' de l'optocoupleur s'éclaire;
- Le photo transistor T' de l'optocoupleur devient passant;
- La tension $V_s = 0V$;

Donc lors de l'activation d'une entrée automate, l'interface d'entrée envoie un 0 logique à l'unité de traitement et un 1 logique lors de l'ouverture du contact du capteur (entrée non actionnée).

III.4 Interfaces de sorties

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

Elles sont destinées à:

- Commander les pré-actionneurs et les éléments de signalisation du système.
- Adapter les niveaux de tensions de l'unité de commande à celle de la partie opérative du système en garantissant une isolation galvanique entre ces dernières.
- Transformer les informations numériques en signaux logiques ou analogiques.

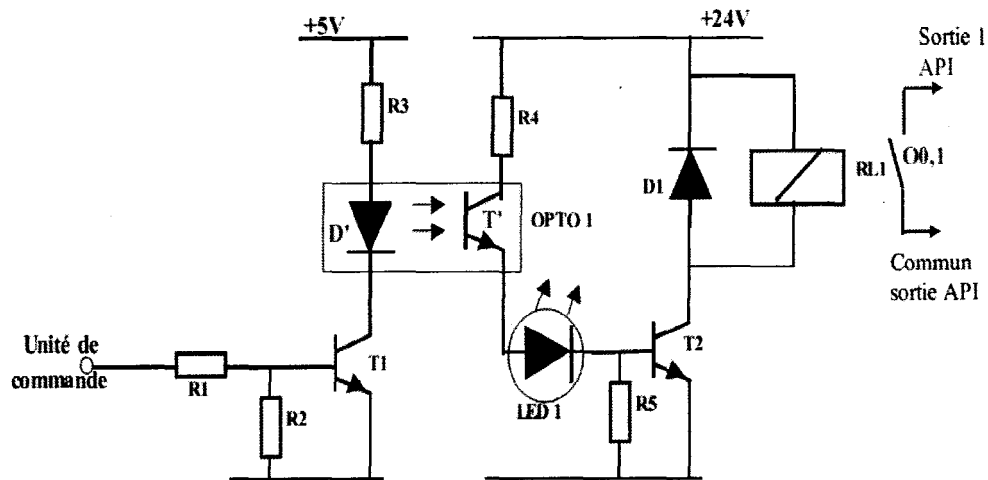


Figure I.6 : Interface de sortie

❖ Fonctionnement de l'interface de sortie

Lors de la commande d'une sortie automate:

- L'unité de commande envoie un 1 logique (5V);
- T1 devient passant, donc D' s'éclaire;
- Le photo transistor T' de l'optocoupleur devient passant;
- LED 1 s'éclaire et nous informe de la commande de la sortie 00, 1;
- T2 devient passant;
- La bobine RL1 devient sous tension et commande la fermeture du contact de la sortie 00, 1;

Donc, pour commander une sortie automate l'unité de commande doit envoyer:

- Un 1 logique pour actionner une sortie API;
- Un0 logique pour stopper la commande d'une sortie API;

III.5 Modules logiques et analogiques des entrées et sorties

➤ Module tout ou rien TOR

Le module TOR occupe 8 bits d'entrée et 8 bits de sortie dans l'espace d'adressage des entrées et sorties

TOR. Le module TOR assure la coordination entre la CPU S7—2Q0 et le CP242—8. Les bits de sélection de banc permettent de sélectionner à partir du programme utilisateur la zone de données à adresser dans le module analogique.

➤ **Module analogique**

Le module analogique occupe 16 octets d'entrée et 16 octets de sortie de l'espace d'adressage des entrées et sorties analogiques. Le module analogique assure les échanges de données aussi bien avec le maître DP PROFIBUS qu'avec les esclaves AS—i.

Le mécanisme de sélection permet d'adresser sur le module analogique au total une zone de données plus grande que celle adressable sur la CPU S7—200 du module d'extension.

III.6 Interface de communication

Permet de connecter la CPU à une console de programmation ou à d'autres appareils. Elle comprend: les consoles, les boîtiers de tests et les unités de dialogue en ligne.

❖ **Les consoles**

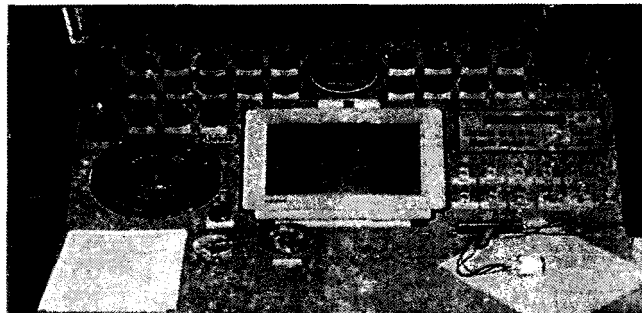


Figure I.7 la console de programmation (PG 670)

Il existe deux types de consoles. L'une permet le paramétrage et les relevés d'informations (modification des valeurs, et visualisation), l'autre permet en plus la programmation, le réglage et l'exploitation. Cette dernière dans la phase de programmation effectue:

- L'écriture
- La modification
- L'effacement

Le transfert d'un programme dans la mémoire de l'automate ou dans une mémoire REPRON.

Dans la phase de réglage et d'exploitation elle permet:

- D'exécuter le programme pas à pas
- De le visualiser
- De forcer ou de modifier des données telles que les entrées, les sorties, les bits internes, les registres de temporisation, les compteurs. . . etc.

➤ **Les unités de dialogue en ligne**

Elles sont destinées aux personnels spécialistes du procédé et non de l'automate programmable, et leur permet d'agir sur certains paramètres:

- Modification des constantes, compteurs, temporisations
- Forçage des entrées/sorties
- Exécution de parties de programme
- Chargement de programmes en mémoire à partir de cassettes.

Ces boîtiers se présentent sous la forme enfichable dans l'unité centrale ou séparés de celle-ci. Ils comportent des touches de fonctions, numériques, une visualisation, un dispositif de sécurité, l'ensemble est piloté par micro-processeur.

Quelque soit le logiciel utilisé, le mode de fonctionnement de l'APT est le même. A partir d'un schéma représentant une commande, le programmeur traduit ce schéma en un programme stocké dans les mémoires. Ce programme lisible par un microprocesseur contient différents éléments

- ✓ Un pas de programme (ligne numérotée).
- ✓ Une instruction ou code (exemple lire l'état d'une entrée . . . etc.)
- ✓ Une référence (exemple numéro d'une sortie.)

IV. Principe de fonctionnement d'un API

Le traitement s'effectue en quatre phases:

- **Phase 1** : Gestion du système Autocontrôle de l'automate
- **Phase 2** : Acquisition des entrées Prise en compte des informations du module d'entrées et écriture de leurs valeurs dans la RAM (zone DONNEE).
- **Phase 3** : Traitement des données

Lecture du programme (située dans la RAM programme) par l'unité de traitement, lecture des variables (RAM données), traitement et écriture des variables dans la RAM données.

•Phase 4 : Emissions des ordres

Lecture des variables de sorties dans la RAM données et transfert vers le module de sorties.

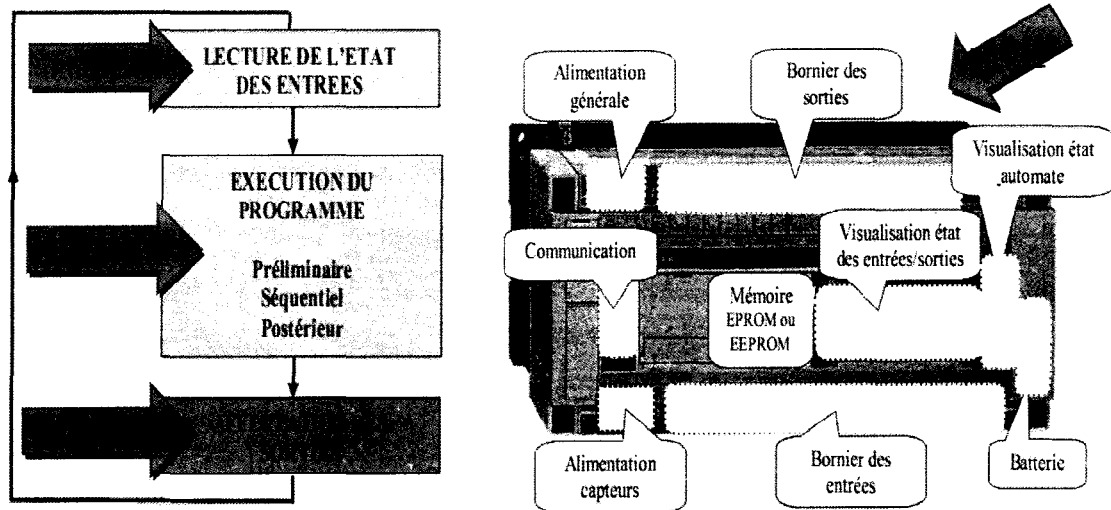


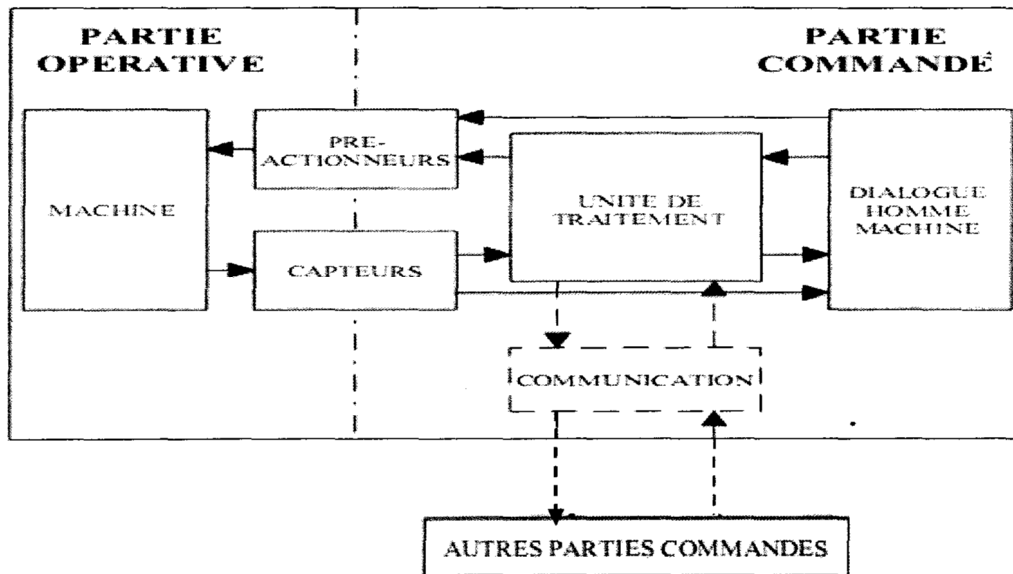
Figure I.8 : le principe de fonctionnement d'un API.

V. Choix d'un automate programmable industriel

Le choix de l'API se fait après avoir établi le cahier des charges du système à automatiser cela en considérant un certain nombre de critères importants

- Le nombre d'entrées/sorties nécessaire;
- La nature des entrées /sorties (numérique ou analogique);
- La fiabilité et la robustesse;
- L'immunité aux parasites et aux bruits;
- La nature de traitement (temporisation, comptage . . . etc.);

V.1 Situation de l'unité de traitement



V.2 Choix de l'unité de traitement

Le choix d'une technologie d'unité de traitement dépend de nombreux paramètres. Les critères essentiels permettant un choix sont définis ci-après :

- Temps de développement court;
- Nombre d'entrées/sorties illimité;
- Capacité de mise en réseau et de traitement;
- Vitesse de traitement;

Discussion

Dans ce chapitre on s'intéresse aux constituants d'un API et à leur organisation interne et à son fonctionnement.

Les automates programmables accomplissent des tâches d'automatisations traduites sous forme de programme d'application que l'utilisateur définit dans une suite d'instructions.

Chapitre II
Description du système à
automatiser

Préambule

Avant de procéder à l'automatisation de la machine pour obtenir les périmètres des grilles de table, aussi appelée GE48, nous allons présenter les différentes parties qui la composent ainsi que son cycle de fonctionnement.

I. Présentation de la machine

La machine est composée de quatre parties principales:

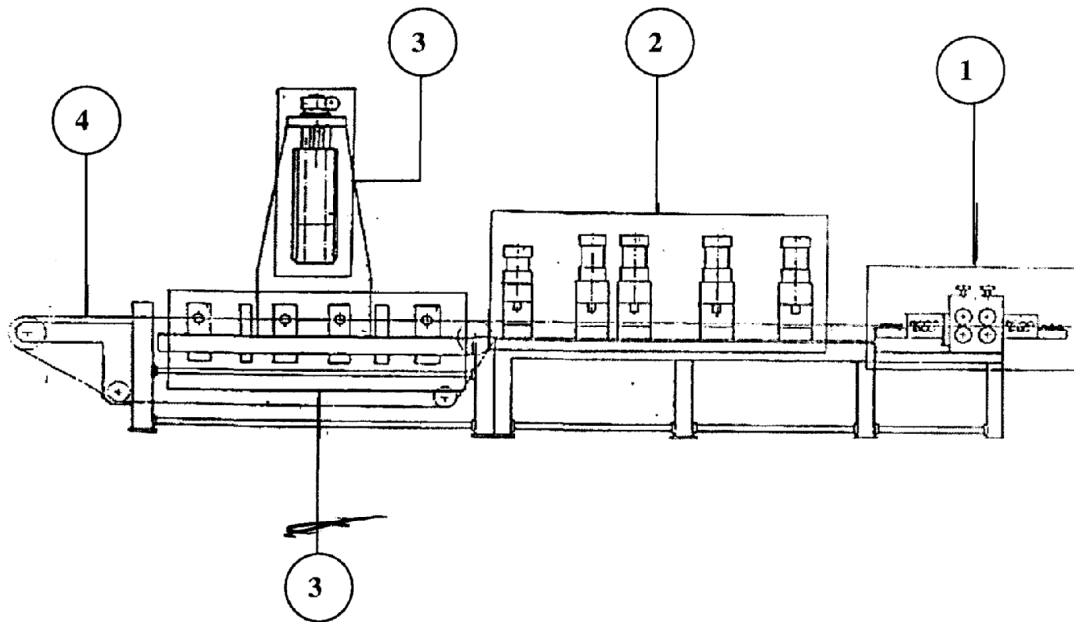


Figure II.1 : Machine pour obtenir les périmètres des grilles de table.

Partie 1 : Groupe d'avancement et groupe à redresser le fil

Partie 2: Groupe presse, groupe coupe ruban et le groupe transport pièce.

Partie 3 : Groupe de cintrage et groupe de soudure.

Partie 4: Groupe d'extraction plus un dérouleur.

- **Dimension**

Longueur de la machine : **8.20m.**

Largeur de la machine : **2.7m.**

Hauteur de la machine : **3.2m.**

- **Caractéristiques techniques**

Production: 10+15 pcs/min.

Puissance totale installée: 53.5kW

Poids total: 60quintaux

I.1 La première partie

Cette partie est constituée de deux groupes, groupe d'avancement et le groupe à redresser le fil.

I.1.1 Groupe d'avancement

a) Description

Le groupe d'avancement sert à tirer le fil métallique. Il est constitué d'un moteur hydraulique qui effectue l'avancement du fil, commandé par une station hydraulique lui appartenant. Un encodeur de mesure est fixé sur le groupe d'avancement, son disque est placé sur le fil, relevant donc la mesure directement à son passage.

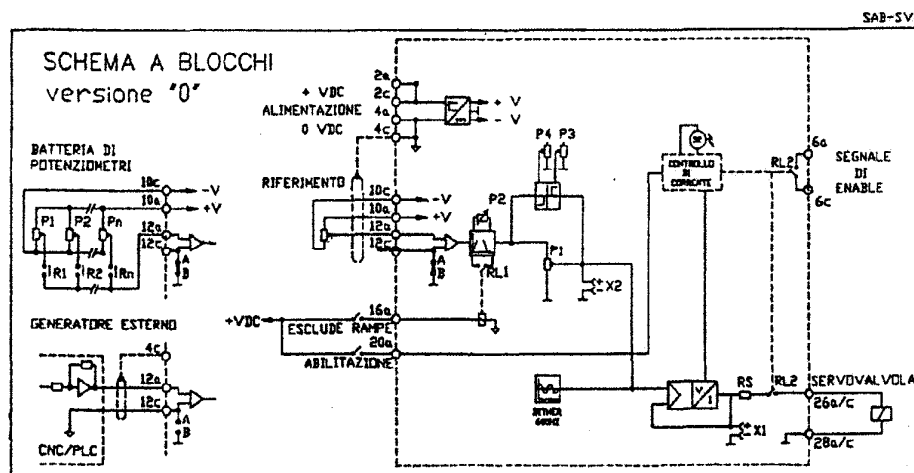


Figure II.2 Schéma de la carte électronique

Une servovalve assure l'asservissement en position du moteur hydraulique en prenant comme consigne le signal fourni par l'encodeur.

La figure suivante montre le schéma de l'asservissement du moteur hydraulique:

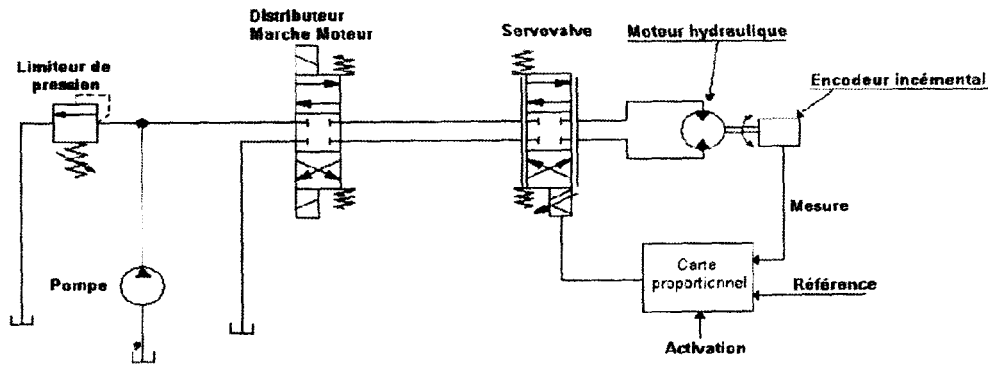


Figure II.3 : Schéma de l'asservissement du moteur hydraulique

Tous les paramètres sont ultérieurement vérifiés par un circuit électronique câblé à l'intérieur de l'appareillage électronique.

b) Caractéristiques techniques

- Vitesse max. d'avancement : 70m/min.
- Puissance moteur station hydraulique: 5.5kW.
- Débit pompe hydraulique: 24l/min.
- Précision d'avancement ruban : ± 0.1 mm

1.1.2 Groupe à redresser le fil

a) Description

Le groupe à redresser le fil sert à redresser le fil métallique tiré par le groupe d'avancement, il est formé de deux blocs distincts: le premier contient 14molettes, il est positionné avant le groupe d'avancement, le deuxième contient 18molettes, il est positionné après le groupe d'avancement. Le réglage de chaque molette est possible à l'aide de grains de registre. Tous ces facteurs permettent d'obtenir un redressement toujours constant.

b) Caractéristiques techniques

- Développement max ruban: 2500mm.
- Dimension max du ruban: 12x6mm.

1.2 La deuxième partie

Cette partie est constituée de trois groupes: groupe presses, groupe coupe ruban et le groupe de transport pièce.

1.2.1 Groupe presses**a) Description**

Le groupe presses sert à l'étampage des bosses sur le fil métallique qui servent de points d'appui pour la grille.

Le groupe est formé de quatre presses hydrauliques (vérins hydrauliques).

Le réglage horizontal des espacements entre les presses se fait simplement en desserrant les brides de blocage, permettant ainsi aux presses de glisser sur un guide. Le positionnement est facilité par la présence de traits millimétrés.

b) Caractéristiques techniques

-force d'étampage à 100kg/cm^2 .

-entraxe min d'étampage: 120mm

-puissance moteur station hydraulique : 5.5kW.

1.2.2 Groupe coupe ruban**a) description**

Le groupe coupe ruban sert à couper le fil métallique, c'est en fait un vérin hydraulique. Il partage la même station hydraulique que le groupe presse.

b) caractéristiques techniques

-Force de coupage à 100kg/cm^2 .

1.2.3 Groupe transport pièce**a) Description**

Le groupe transport pièce permet de placer avec extrême précision la barre de ruban coupé dans la zone de cintrage.

Ce groupe est réalisé deux petits vérins pneumatiques fonctionnent en même temps et font déplacer une barrière verticalement, cette dernière descend pour permettre au ruban de se positionner et monte quand le ruban est maintenu par deux pinces, pour libérer le chemin au ruban et le placer dans le groupe de cintrage. Les pinces sont actionnées par deux petits vérins pneumatiques.

1.3 La troisième partie

Cette partie comprend le corps principal de la machine et elle est constituée de deux groupes : groupe de cintrage et groupe de soudure.

1.3.1 Groupe de cintrage

a) Description

Ce groupe est formé de quatre têtes de cintrage, où sont assemblés tous les composants qui permettent de réaliser la cinématique pour effectuer la rotation d'où naît le cintrage.

La tête de cintrage est actionnée par des vérins hydrauliques convenablement dimensionnés.

Le groupe a été projeté de façon à rationaliser et à rendre moins pénibles les tâches de l'opération en cas de fabrication de châssis différent.

Le déplacement des têtes de cintrage se fait à l'aide de moteur asynchrone, en agissant sur des boutons poussoirs de commande placés sur la console.

Le déplacement devra se faire individuellement, en vérifiant chaque cote sur la règle millimétrée.

Le blocage de la pièce est garanti par deux vérins hydrauliques, ainsi que tous les mouvements de cintrage se produisent de façon hydraulique.

b) Caractéristiques techniques

- Châssis carré mm: 200x200mm. (Int.)
- Châssis carré max: 625x625mm. (Ext.)
- Cote min châssis: 200mm
- Cote max châssis: 1050mm
- Angle de cintrage : 90°.
- Vitesse de translation tête de cintrage : 0.004 m/sec.
- Dimensions max ruban: 12x6 mm.
- Puissance moteur station hydraulique: 5 .5kw+5 .5kw.
- Puissance moteur translation têtes : 0.25kw.

1.3.2 Groupe soudure**a) Description**

Le groupe de soudure spécialement conçu par OMAS a été réalisé en se servant d'un transformateur à 35kw.

Ce groupe glisse pour prendre la pièce cintrée grâce à deux vérins pneumatiques, il saisit cette dernière grâce à deux pinces actionnées par deux petits vérins pneumatiques, et il réunit les deux bouts de la pièce avec un vérin pneumatique pour le souder.

b) Caractéristique techniques

-Puissance du moteur: 0.37kw.

-Vitesse ruban transporteur: 30m/min.

1.4 La quatrième partie

Cette partie est constituée d'un seul groupe qui est le groupe d'extraction ;

a)Description

Ce groupe est formé d'une courroie en mouvement permanent, actionnée par un moteur asynchrone, faisant fonction de bande transporteuse.

b) Caractéristiques techniques

-Puissance du moteur: 0.37kw.

-Vitesse ruban transporteur: 30m/min.

1.5 Dérouleur

Le dérouleur motorisé n'est pas un groupe implanté dans la machine GE48, c'est une petite machine auxiliaire qui aide le groupe d'avancement dans la fourniture ruban.

a)Description

Le dérouleur est constitué d'une partie pivotante supportée par un soutien approprié, convenablement dimensionné, monté sur des roulements à rouleaux coniques et actionné par un moteur électrique.

Au moyen d'une moto variateur hydraulique de vitesse lié à un réducteur, la vitesse de rotation est instantanément proportionnée selon la demande du ruban de la part de la machine.

Le dérouleur peut dérouler le ruban soit dans le sens des aiguilles d'une montre soit dans le sens inverse.

Le dérouleur est en effet doté d'un système de sécurité qui intervient en cas de secours, en bloquant instantanément soit le touret soit la machine.

b) Caractéristiques techniques

-Portée: 1500/2500 kg.

-Puissance moteur: 1,1/1,5 kW.

-Poids: 350 kg.

-Vitesse de rotation : de 0 à 70 tours/min.

II. Composants de la machine

La machine pour l'obtention des périmètres des grilles de table utilise pour son fonctionnement: de la force pneumatique qui est produite par une station de compression d'air centrale de l'usine, de la force hydraulique produite par la station hydraulique de la machine, ces forces assurent le fonctionnement normal de son système automatisé de production.

II.1 Système d'approvisionnement en pression pneumatique

II.1.1 Caractéristiques de la source d'énergie

Dans les systèmes pneumatiques, l'air comprimé est utilisé comme source d'énergie. L'air comprimé utilisé dans les systèmes pneumatiques est au départ de l'air à la pression atmosphérique porté artificiellement à une pression élevée appelée pression d'utilisation.

- **Avantages de l'air comprimé:** il se transporte facilement dans des conduites bon marché. Il est propre et les composants fonctionnant sous cette énergie sont peu coûteux. Il est également insensible aux variations de température.
- **Inconvénients de l'air comprimé:** cette source d'énergie nécessite un excellent conditionnement (filtration). Aucune impureté, poussière . . . etc. ne doit pénétrer dans le système. Il est difficile d'obtenir des vitesses régulières du fait de la compressibilité de l'air.

II.1.2 Structure d'une installation pneumatique

Toute installation pneumatique assurant une production et une distribution d'air comprend:

- Un compresseur avec un réservoir d'air

- Un système de traitement de l'air
- Un dispositif de sécurité et de régulation
- Un ensemble de circuits de distribution généralement réalisé en tube acier ou cuivre

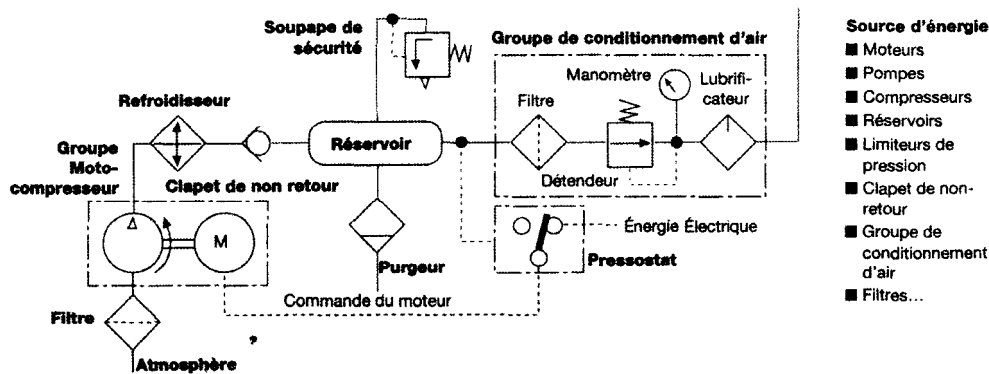


Figure II.4: structure générale d'une ligne d'alimentation pneumatique

II.1.2.1 Traitement de l'aire

a) Structure d'un groupe de conditionnement

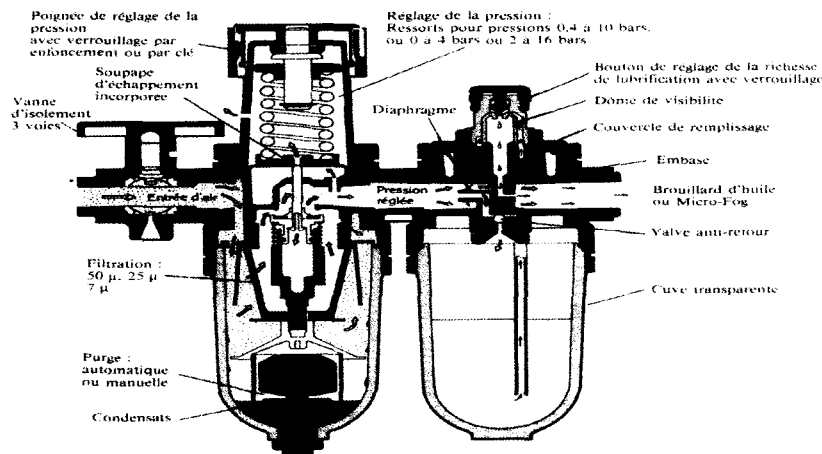


Figure II.5 : Unité de conditionnement FRL

Pour le traitement de l'air, le matériel utilisé est une unité de conditionnement d'air comprimé appelée FRL (Filtre — Régulateur — Lubrificateur), qui est destinée à préparer l'air en vue de son utilisation dans les systèmes en le débarrassant des poussières, d'eau et autres particules nuisibles qui risqueraient de provoquer des pannes dans l'installation.

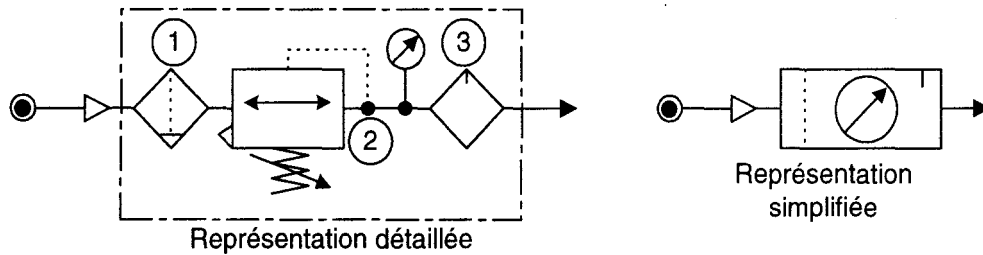


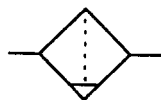
Figure II.6: représentation détaillée et simplifiée de FRL

Cet ensemble est constitué de 3 ou 4 appareils montés en série dans un ordre déterminé.

❖ **Filtre avec séparateur d'eau .**

Le rôle du filtre est de soustraire du système tous les éléments nuisibles au bon fonctionnement comme l'eau, l'huile et les poussières dégagées par les transformateurs.

• **Symbole**



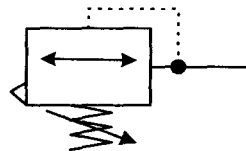
Position du filtre
dans l'appareil



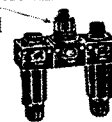
❖ **Le régulateur de pression**

Le rôle de cet appareil est de maintenir l'air comprimé à une pression constante, quelles que soient les fluctuations en air du réseau. Il doit réguler la pression en fonction de la demande. Il est souvent associé à un manomètre qui permet de contrôler la pression.

• **Symbole**



Position du régulateur
dans l'appareil



❖ **Un manomètre**

Il indique la pression de l'air du service dans les différents organes pneumatiques. Dès que le moteur est mis en marche le manomètre donne une indication précise.

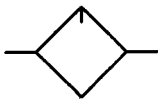


Figure II.7: un manomètre

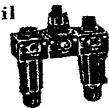
❖ Le lubrificateur

La 3^{ème} étape du traitement de l'air comprimé consiste à injecter dans l'air une quantité d'huile afin de permettre la lubrification des parties mobiles des composants entrant dans la constitution des systèmes pneumatiques pour éviter la corrosion et à améliorer le glissement des organes pneumatiques de l'équipement.

- **Symbole**



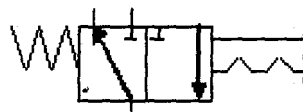
Position du lubrificateur
dans l'appareil



b) Sectionneur

C'est une vanne de type 3/2, qui est manœuvrée manuellement ou électriquement. Son rôle couper l'alimentation en air comprimé et le mettre à l'échappement afin de mettre les systèmes en ou hors énergie.

- **symbole**



c) Démarreur progressif

Il assure une montée progressive de la pression dans l'installation en agissant sur la vitesse du remplissage du circuit. Monté en sortie du FRL et avant le sectionneur général, il protège les personnes d'une brusque remise en service des actionneurs.

II.2 Système d'approvisionnement en pression hydraulique

La pression est assurée par une station hydraulique dont les composants sont:

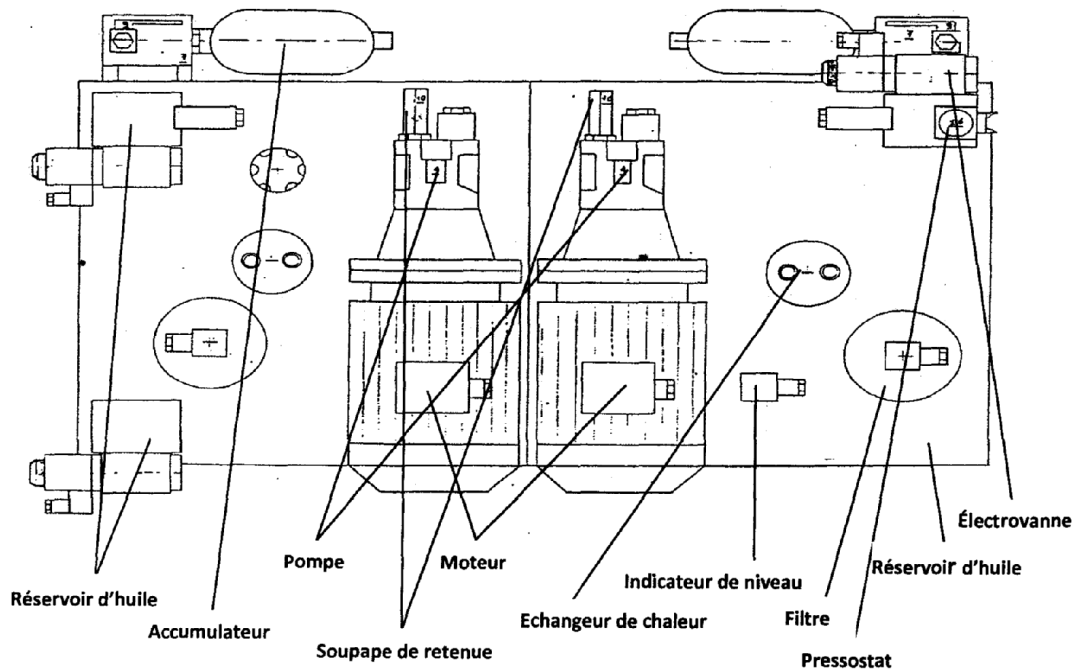


Figure II.8: Vue de dessus de la station de production de l'énergie hydraulique

❖ Centrale hydraulique

Elle est composée de trois parties principales

➤ Moteur asynchrone (voir III.3.1.2.2.1)

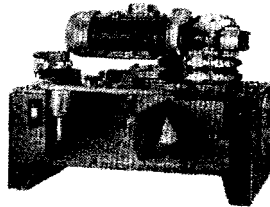


Figure II.9 : Moteur asynchrone

➤ Pompe

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

• Symbole:



Figure II.10: Une pompe hydraulique

➤ **Réservoir d'huile**

D'une capacité de 75 litre,

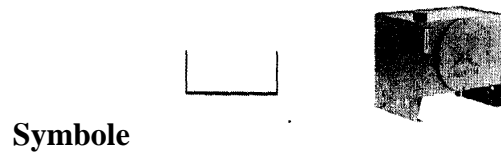


Figure II.11 : un réservoir d'huile

La machine GE48 possède trois stations hydrauliques suivantes:

- La première qui actionne la servovalve.
- La deuxième responsable sur le poinçonnage.
- La troisième pour le groupe pliage.

➤ **Echangeur de chaleur**

C'est un circuit ou circule l'eau froide pour refroidir l'huile dans le réservoir.

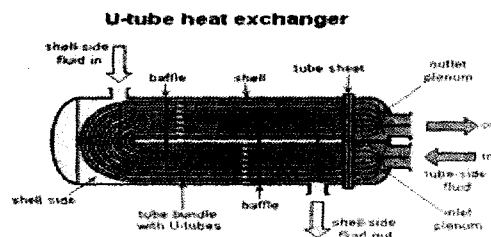


Figure II.12 : Un échangeur de chaleur

➤ **Accumulateur**

Il sert à emmagasiner une réserve d'énergie, et la restituer en cas de besoin il est monté en dérivation avec le circuit principal

Un accumulateur hydropneumatique est un accumulateur à gaz avec élément de séparation entre le gaz et le fluide. Le gaz le plus souvent utilisé est l'azote inerte qui est de bonne compressibilité.

- symbole:

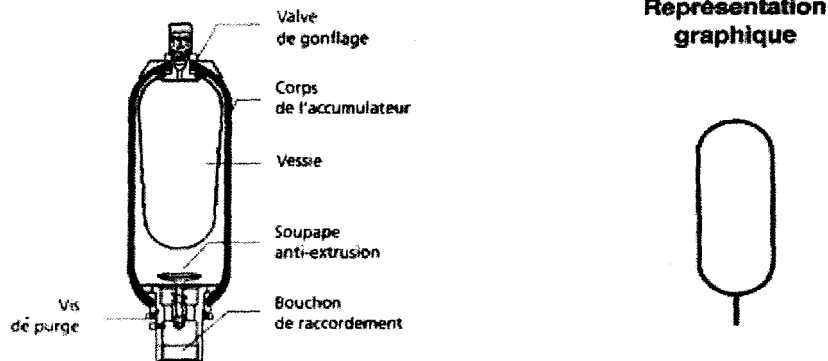


Figure II.13: accumulateur hydropneumatique

➤ Clapet anti retour

Ce composant permet au fluide sous pression de circuler dans un sens déterminé. Il lui interdit le passage dans l'autre sens. Il assure le verrouillage étanche d'un ou deux orifices.

- symbole:

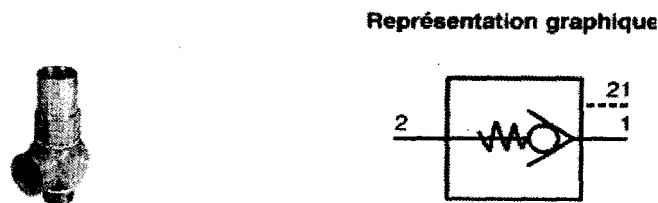


Figure II.14: clapet anti retour

La figure(II.5) montre un simple exemple de raccordement du clapet anti retour:

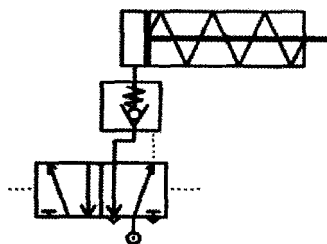


Figure II.15 : Exemple de raccordement du clapet anti retour

➤ Régulateur .depression

Les régulateurs de pression sont destinés à maintenir la pression de sortie constante indépendamment de la pression d'entrée variable et de la consommation d'air.

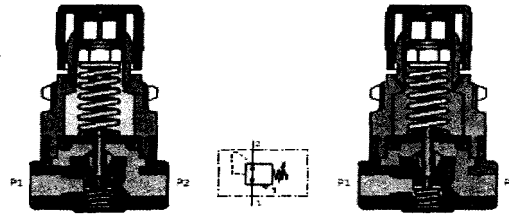


Figure II.16 : Régulateur de pression

II.3 Définition et choix des composants utilisés

II.3.1 Partie opérative

C'est la partie qui assure les transformations des matières d'œuvre permettant d'élaborer la valeur ajoutée recherchée (produit fini).

Elle est constituée de:

II.3.1.1 Pré actionneur

Les pré- actionneurs servent à mettre en service ou hors service un actionneur, une machine ou une installation, c'est un constituant de gestion d'énergie fournie à l'actionneur.

II.3.1.1.1 Électrovannes

Il existe deux types d'électrovannes: « tout ou rien » et « proportionnelle ».

-Les électrovannes dites de « tout ou rien» sont des électrovannes qui ne peuvent s'ouvrir qu'en entier ou pas du tout.

-Les électrovannes proportionnelles sont celles qui peuvent être ouvertes avec plus ou moins d'amplitude en fonction du besoin. Elles sont généralement utilisées grâce à une commande.

L'ouverture et la fermeture de l'électrovanne est liée à la position du noyau mobile qui se déplace sous l'effet du champ magnétique engendré par la mise sous tension de la bobine.

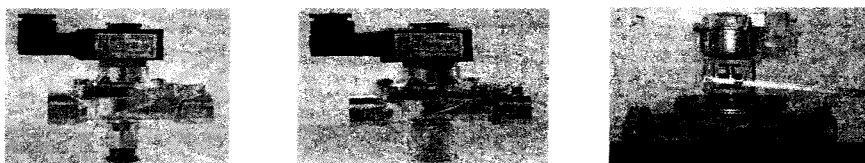


Figure II.17: Quelques types d'électrovannes

II.3.1.1.2 Distributeurs

Leurs fonction est de commander le départ, l'arrêt, la direction d'un débit et la réception d'un signal de commande.

➤ **Sa taille et son type sont en fonction du vérin**

Si le vérin est à simple effet et ne comporte donc qu'un seul orifice à alimenter, on utilise un distributeur ne comportant qu'un seul orifice de sortie : distributeur 3/2 à trois orifices (pression, sortie, échappement) et à deux positions;

Si le vérin est à double effet et comporte donc deux orifices sur lesquels il faut alterner les états de pression et d'échappement, on utilise un distributeur comportant deux orifices de sortie. (Voir Figure II.18)

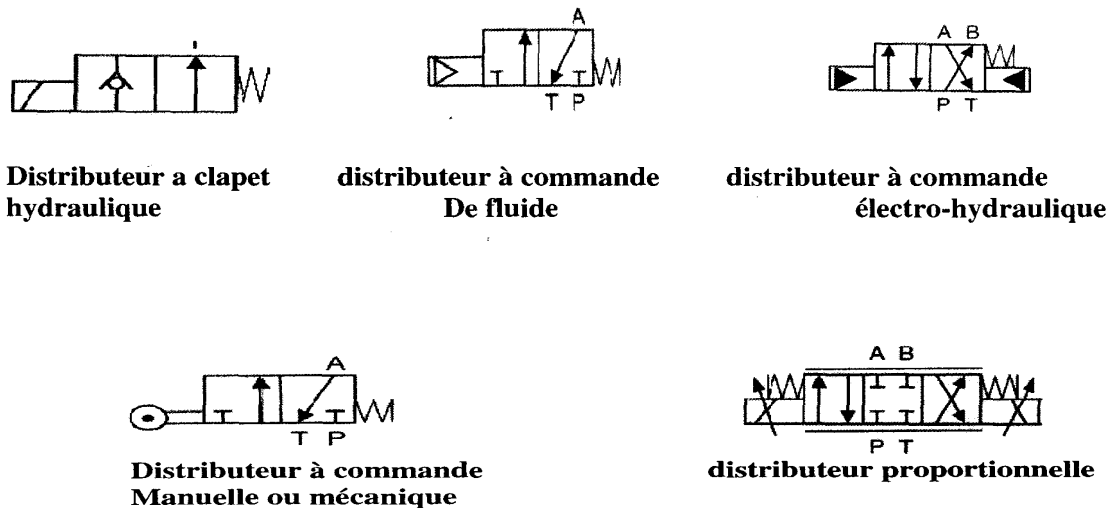


Figure II.18: les différents types de distributeurs

II.3.1.1.3 Les relais

Les relais sont des dispositifs électromécanique qui ferment un interrupteur sur commande, lorsqu'un courant électrique leurs parvient. Pour cela, les relais disposent de deux parties principales: une lame métallique et un électro-aimant.

➤ il existe plusieurs types de relais dont on cite:

- **Relais thermique** : permet de protéger un récepteur contre surcharge faible et prolongées.
- **Relais instantanés**: ne peuvent être placés que deux étapes appelées «open / close». On peut citer les relais élémentaires, à contact de passage, et les relais bistable.

- **Relais temporisés:** ce sont des relais dont on a retardé volontairement un changement d'état.

II.3.1.1.4 Contacteurs

Son rôle est de mettre en fonctionnement ou arrêter un actionneur. ses contacts de puissance sont prévus pour supporter les arcs électrique (pouvoir de coupure) qui se créent lors des arrêts.

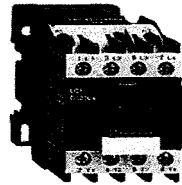
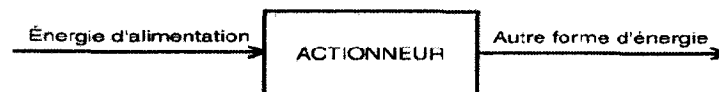


Figure II.19: un contacteur

II.3.1.2 Actionneurs

Un actionneur est l'organe fournissant la force nécessaire à l'exécution d'un travail ordonné par une unité de commande distante.



II.3.1.2.1 Vérins

- **Le vérin pneumatique :** est utilisé avec de l'air comprimé entre 2 et 10 bars dans un usage courant. Simple à mettre en œuvre, il est très fréquent dans les systèmes automatisés industriels.

- **Vérin simple effet:**

Un vérin simple effet produit un effort significatif dans un seul sens, le rappel de la tige vers la position de repos étant assuré par un ressort ou d'une force extérieure.

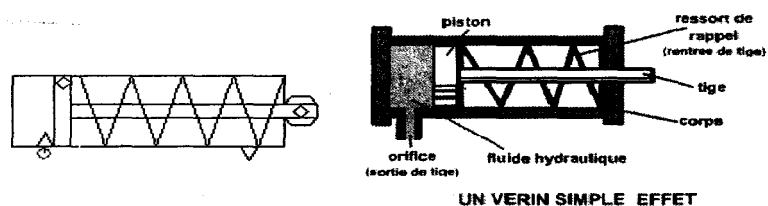


Figure II.20: Schématisation d'un vérin double effet

- **Vérin double effet:**

Le vérin double effet est un composant bistable (Stable dans deux positions). Ce type de vérin peut produire un effort significatif dans les deux sens, le rappel de tige est obtenu par inversion de l'alimentation des deux chambres.

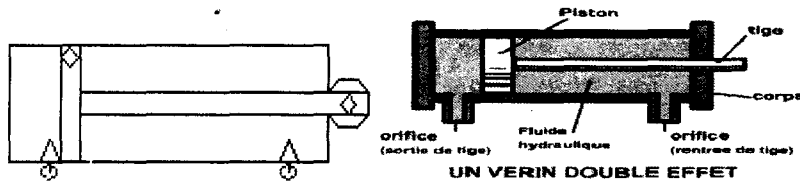


Figure II.21: Schématisation d'un vérin double effet

Dans notre cas on a 4 vérins pneumatiques.

- **Le vérin hydraulique** : est utilisé avec de l'huile sous pression, jusqu'à 350 bars dans un usage courant. Plus coûteux, il est utilisé pour les efforts plus importants et les vitesses plus précises (et plus facilement réglable)

Notre machine contient 9 vérins hydrauliques.

II.3.1.2.2 Les moteurs

Nous allons nous intéresser à deux types de moteurs:

- moteur asynchrone
- moteur hydraulique

II.3.1.2.2.1 Moteurs asynchrones

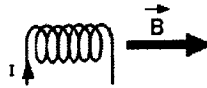
a) Présentation

Le moteur asynchrone triphasé est largement utilisé dans l'industrie, il est constitué d'une partie fixe, le stator qui comporte le bobinage, et d'une partie rotative, le rotor qui est bobiné en cage d'écureuil. Les circuits magnétiques du rotor et du stator ont constitués d'un empilage de fine tôles métalliques pour éviter la circulation de courant de Foucault.

b) Fonctionnement

Le principe de fonctionnement des moteur à courant alternatifs réside dans l'utilisation d'un champ magnétique tournant produit par de tentions alternatives.

La circulation d'un courant dans une bobine crée un champ magnétique B dans l'axe de la bobine, sa direction et son intensité sont en fonction du courant I .



Les 3 enroulements du moteur asynchrone triphasé créent un champ magnétique tournant, dont la fréquence de rotation dite fréquence de synchronisme. Donc, si on place une boussole au centre, elle va tourner à cette vite de synchronisme.

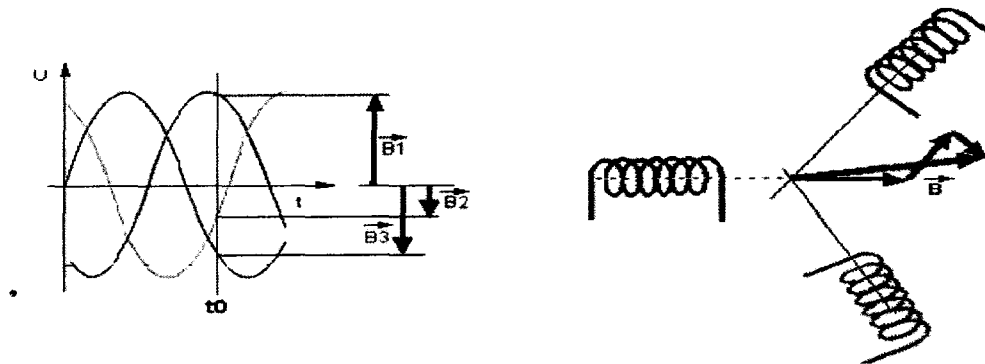


Figure II.22: le champ résultant

Le stator est constitué d'une cage d'écureuil, elle est en fait un bobinage à grosse section et très faible résistance et balayée par le champ magnétique tournant.

c) Liaison avec réseau triphasé

Le moteur est relié au réseau par un certain nombre de dispositifs de sécurité et de commande que nous avons vus dans la section précédente (sectionneur, contacteur, relais)

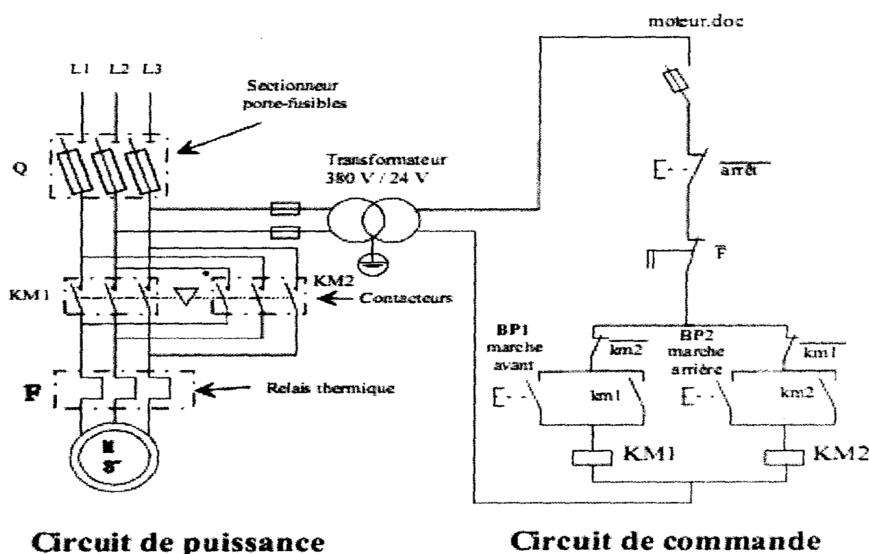


Figure II.23: schéma de commande d'un *moteur* triphasé asynchrone à deux sens de rotation

Le circuit de puissance est alimenté en triphasé, alors que le circuit de commande est alimenté par l'intermédiaire d'un transformateur de sécurité.

II.3.1.2.2 Moteur hydraulique

a) Présentation

Dans ce type d'actionneur, l'énergie hydraulique fournie par un fluide sous pression est transformée en énergie mécanique.

Il en résulte un mouvement de rotation sur l'arbre de sortie.

Les moteurs hydrauliques présentent deux caractéristiques: le couple moteur et la vitesse de rotation.

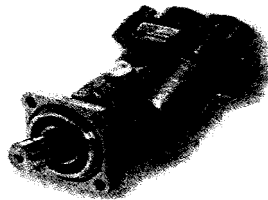
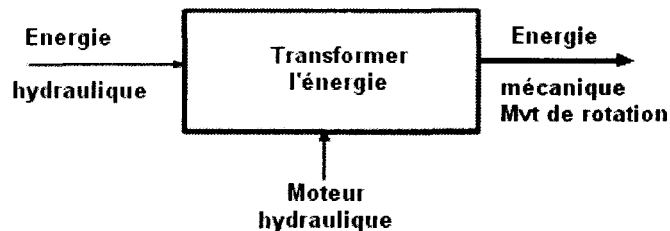


Figure II.24: moteur hydraulique

Modèle fonctionnel : Moteur hydraulique



Ces moteurs entraînent des systèmes mécaniques. Si le couple résistant devient trop important, la pression monte. Quand elle atteint la valeur de réglage du limiteur de pression, le débit retourne au réservoir.

Dans notre machine le « groupe d'avancement » est constitué d'un moteur à pistons axiaux que nous changerons par la suite avec un moteur électrique.

- b) Moteur à pistons axiaux** Les pistons en communication avec la haute pression se déplacent en tournant et par une liaison rotule avec le tourillon obligent ce dernier à tourner. la gamme de pression pouvant aller jusqu'à 450bar.

Avantages : couple très important, possibilité de varier la cylindrée, vitesse importante.

Inconvénient: coûteux.

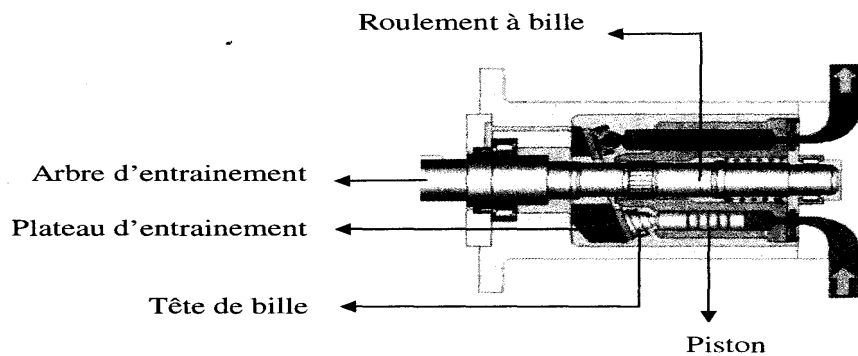


Figure II.25: moteur à piston axial

II.3.1.2.2.3 Servovalve

Est un élément utilisé dans les systèmes hydrauliques qui assure les fonctions de distribution et de régulation de débit. Le déplacement du tiroir de distribution est proportionnel au courant électrique d'entrée. Elle est une valve de contrôle de débit, elle contient étage de puissance et un étage d'amplification

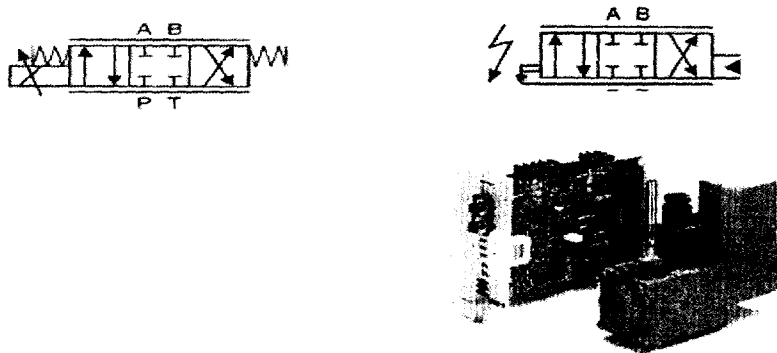


Figure II.26: une servovalve

II.3.1.3 Les capteurs

Les capteurs sont des composants de la chaîne d'acquisition dans une chaîne fonctionnelle. Ils prélèvent une information sur le comportement de la partie opérative et la transforment en une information exploitable par la partie commande, cette dernière qui traite des variables logiques ou numériques. L'information délivrée par un capteur pourra être logique (2 états), numérique (valeur discrète), analogique (dans ce cas il faudra adjoindre à la partie commande un module de conversion analogique numérique).

II.3.1.3.1 Les pressostats

Ils sont destinés pour contrôler la pression d'un circuit et fournir une information lorsque cette pression atteint la valeur de réglage du pressostat

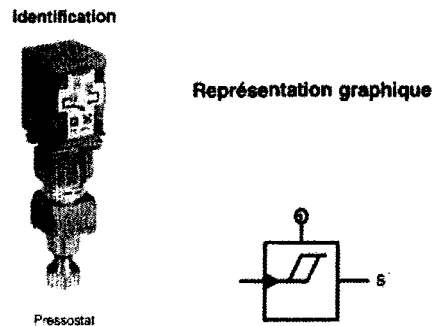


Figure II.27 : pressostat

II.3.1.3.2 Le détecteur de proximité

Les détecteurs sont des capteurs tout ou rien (TOR) .ils possèdent toutes les propriétés des capteurs, excepté la nature du signal de sortie qui est binaire. Détecteur de proximité : délivre une information logique de présence de l'élément à détecter sans contact physique

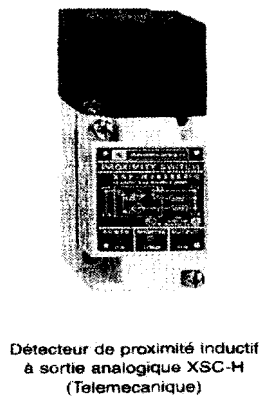


Figure II.28: détecteur de proximité

II.3.1.3.3 Codeur incrémental

Les codeurs incrémentaux sont destinés à des applications de positionnement et de contrôle de déplacement d'un mobile par comptage et décomptage des impulsions qu'ils délivrent

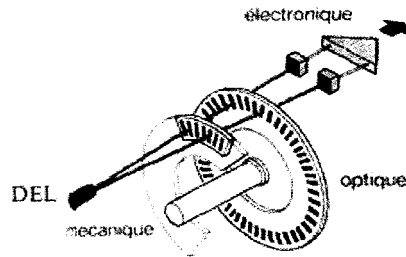


Figure II.29: codeur incrémental

II.3.1.3.4 Flotteur

Le flotteur se maintient à la surface du liquide. Il est solidaire d'un capteur de position qui délivre un signal électrique correspondant au niveau. Sa position est peu dépendante de la masse volumique de liquide.

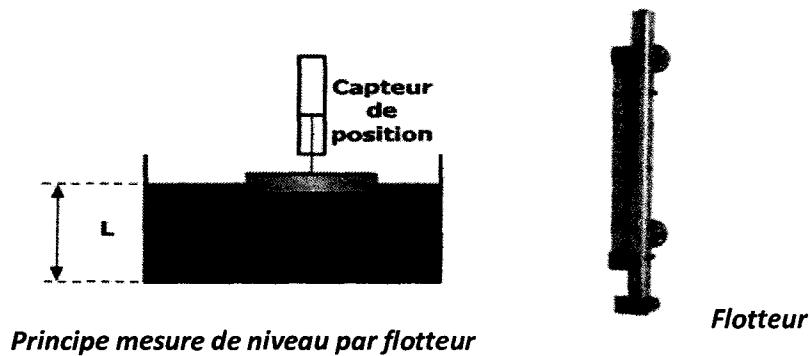


Figure II.30 : principe mesure de niveau par flotteur

II.3.1.3.5 Plongeur

Le plongeur est un cylindre immergé (fig. *plongeur*) dont la hauteur est au moins égale à la hauteur maximale du liquide dans le réservoir. Il est suspendu à un capteur dynamométrique qui se trouve soumis à une force F (le poids apparent), fonction de la hauteur L du liquide:

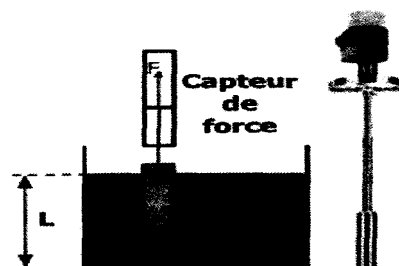


Figure II.31: principe mesure de niveau par plongeur

II.3.2 La partie liaison

Le pupitre de commande est formé de trois parties:

- Une partie comportant les différents accessoires de commande (boutons et sélecteurs).
- Une partie comportant les différents voyants lumineux témoignant des différents états de fonctionnement.
- Un clavier permet l'insertion des paramètres géométriques de la pièce (longueur) et ceux des modes de marche.

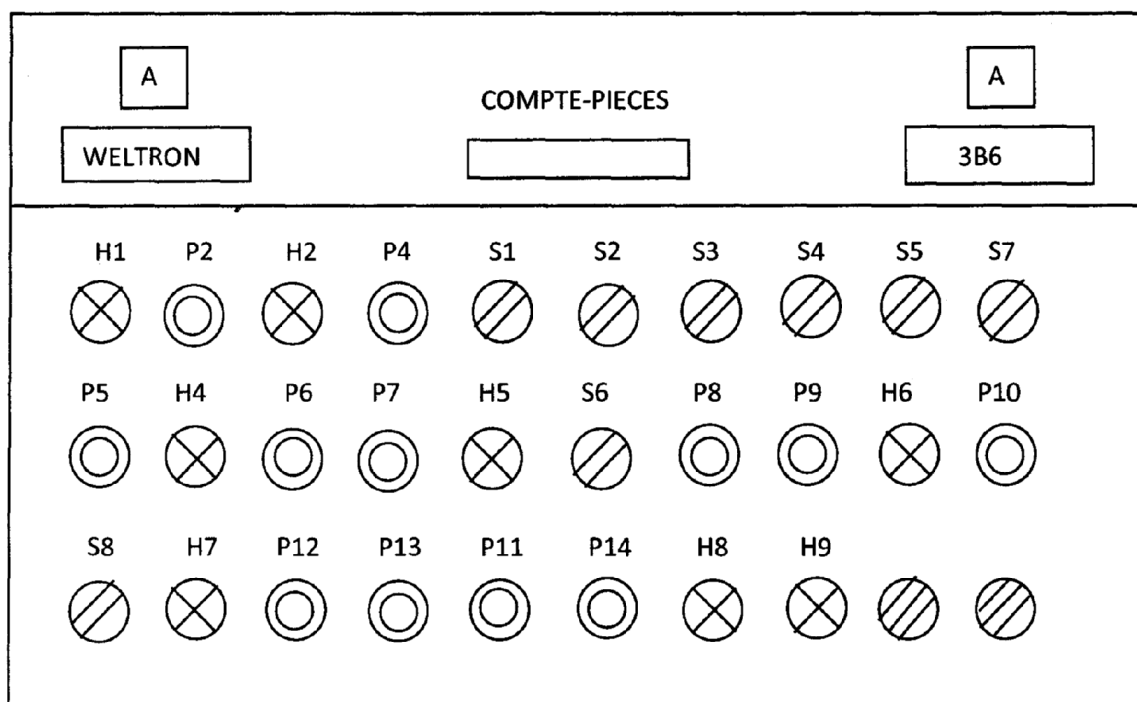


Figure II.32: pupitre de commande

H1: tension de marche.

H2 : voyant marche de commande,

H4 : voyant presse/cyclo distributeur,

H5 : voyant distributeur avance,

H6 : voyant moteur extraction,

H7: voyant intervention protection thermique,

H8, H9 : signaleurs alarmes distributeurs,

S1 :Tnc. /Exc. Commande têtes,

S2:Dr./Ga.Tétel,

S3 :Dr./Ga.Tête2,

S4:Dr./Ga.Tête3,

S5:Dr./Ga.Tête4,

S6 :Inc. /Exc. Distributeur avance,

S7 : Me. /De. Tête soudeuse,

S8 : sélecteur marche continu/intermittente/à cycle.

P2: arrêt d'urgence,

P4 : marche commande,

P5 : arrêt,

-presse/cyclo distributeur,

P6 : marche,

P7: arrêt,

-distributeur avance,

P8 : marche,

P9 : arrêt

-moteur extraction,

P10 : marche,

P11: marche cycle,

P12 : manuel al. Pressel,

P13 : coup manuel,

P14: remettre sécurités distributeurs.

II.3.3 Partie commande

La partie “commande” élabore des ordres destinés à la partie opérative en fonction

- du programme qu'elle contient,
- des informations reçues par les capteurs,

- des consignes données par l'utilisateur ou par l'opérateur.

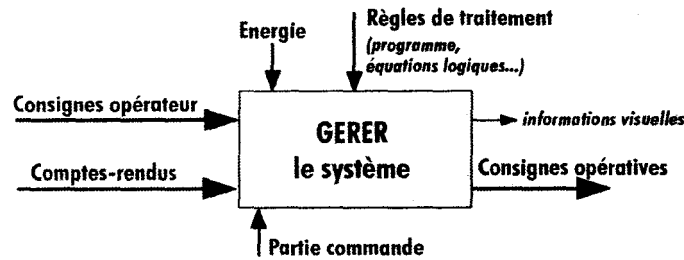


Figure II.33: fonction de la partie commande

II.3.3.1 Programmeur à cames

a)description et principe de fonctionnement

La fonction d'un programmeur à cames est de délivrer, suivant la position angulaire d'un arbre ou d'un vilebrequin, des ordres au circuit de commande d'une machine. Ces ordres ont généralement pour but d'assurer des fonctions d'automatismes (comptage, synchronisation de mouvement, etc.).

Les programmeurs électromécaniques sont généralement constitués des éléments suivants :

- 1).des fins de course qui délivrent un signal électrique au circuit de commande de la machine. Ils sont soit mécaniques, soit inductifs;
- 2).des disques de cames qui permettent le réglage des points de commutation des interrupteurs;
- 3).des poussoirs également appelés « suiveurs » qui suivent le profil des disques de cames et viennent actionner les interrupteurs;
- 4).un arbre qui supporte les différents disques de cames et qui permet l'accouplement du programmeur à l'élément de transmission à contrôler;
- 5).un carter qui permet la protection et la fixation de l'ensemble.

1- fins de course

-2- disques de cames

-3- poussoirs ou suiveurs

-4- arbre

-5-carter

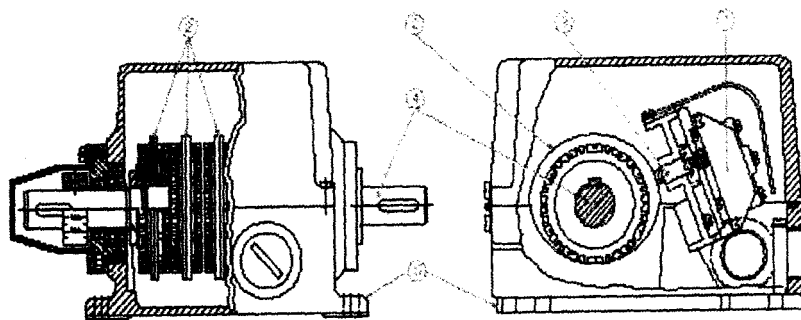


Figure II.34: programmeur à came électromécanique

c) programmeur à cames de la GE48

Pour que le programmeur à cames puisse s'adapter au cycle de fonctionnement de la machine, les positions angulaires des disques des 25 cames de notre programmeur sont données dans le tableau suivant:

Borne cames	N	(A)	(B)	commentaire
2-3	01	25	27	Poinçon
1-3	02	10	23	Ames internes
2-3	03	20	50	Ames externes
1-3	04	06	27	Blocage interne
1-3	05	25	43	Blocage externe
1-3	06	10	13	Bloc tête
1-3	07	53	00	Pliage interne
1-3	08	33	44	Pliage externe
1-3	09	36	25-29	Relevage soudure
1-3	10	05	25	Bloc pièce
	11	35	00	
	12	35	05	
	13	05	45	Entréefil
	14	30	58	
	15	28	57	

	16	25	55	
	17	25	57	
	18	24	55	
	19	24	53	
	20	24	43	
2-3	21	50	51	
1-3	22	50	35	Soudure
1-3	23	24	55	Compte pièce
2-3	24	42	40	Moteur 360
	25	35	20	process

III. Le cycle machine

La machine pour obtenir périmètre des gilles de table effectue un cycle constitué de sept phases, qui sont:

- Etampage bosses.
- Avancement et dressage du ruban.
- Coupage du ruban.
- Placement du ruban dans la zone de cintrage.
- Cintrage.
- Soudure.
- Extraction de la pièce soudée.

La machine est programmée de façon telle que plusieurs pièces puissent se trouver en exécution simultanément, ce qui permet une productivité élevée.

III.1 Etampage bosses

Le premier usinage mécanique se produit dans cette phase du cycle grâce au groupe presses.

Les bosses sont réalisées simultanément par quatre presses ayant été préalablement espacés grâce à une règle millimétrique horizontale.



Figure II.35: la forme du ruban après l'étampage

III.2 Avancement et dressage du ruban

Le groupe d'avancement sert à alimenter la machine de la quantité nécessaire de ruban. Ce dernier ayant été redressé par le groupe de dressage.

Un encodeur est placé à la sortie du groupe d'avancement, servant à mesurer la longueur du ruban et la compare à une valeur de consigne fixée sur le pupitre de contrôle. Lorsque les deux valeurs sont identiques, la servovalve revient à la position de repos et le moteur oléo-hydraulique, du groupe d'avancement, s'arrête.

Le dérouleur motorisé A.M.I 1500 travaille uniquement durant cette première phase du cycle.

III.3 Coupage du ruban

Le ruban est coupé par le groupe coupe ruban.

Pour raison de sécurité un capteur de proximité a été prédisposé pour détecter l'avancée du ruban. Cette dernière autorise le coupage du ruban lorsque les quatre presses d'étampage sont en position de repos.

III.4 Placement du ruban dans la zone de cintrage

Il ne s'agit pas d'une véritable phase d'usinage, mais d'un transport afin de pouvoir rendre accessible les stations du travail précédentes, ce qui permet l'usinage simultanée d'une autre pièce.

Le transport du ruban coupé se fait comme suit:

Premièrement, une barrière décente pour permettre au ruban de se placer dans une place bien déterminée, puis deux pinces tiennent celui-ci et la barrière se lève. Enfin le ruban est déplacé vers la zone de cintrage au moyen de deux vérins pneumatiques.

III.5 Cintrage

Durant cette phase le ruban prend sa forme géométrique définitive au moyen des quatre têtes de cintrage.

La réception du ruban, dans la zone de cintrage, se déroule de la manière suivante:

Une fois le ruban transporté par le groupe approprié il sera positionné sur le blocage (le groupe de cintrage) ensuite les âmes sortent pour saisir le ruban comme un étau, le bloque et l'outillage de cintrage effectuent le cintrage externe. Les blocages extérieurs lâchent le ruban et les âmes extérieures rentrent pour libérer ce dernier pour le cintrage interne qui s'effectue de la même façon que le cintrage externe.



Figure II.36: la forme de la pièce après le cintrage externe.



Figure II.37: la forme de la pièce après le cintrage interne.

III.6 Soudure

C'est la dernière opération mécanique du cycle de production, elle se produit grâce au groupe de soudure.

Une fois la pièce est cintrée, le groupe de soudure se glisse pour saisir la pièce au moyen de deux pinces, puis remonte pour libérer le châssis de la zone de cintrage afin de rendre cette station opérative. Un vérin pneumatique est actionné pour joindre les deux bouts de la pièce afin de commencer la soudure qui s'effectue en trois étapes, qui sont:

- Le temps de pressurisation: c'est le temps de maintenir les deux bouts de la pièce sous la pression avant d'alimenter les électrodes pour effectuer la soudure. il dure environ une seconde;
- Temps de mis en électricité: c'est le temps de mettre les électrodes sous tension pour effectuer le soudage qui dure environ deux secondes;
- Une fois la pièce soudée et avant de la lâcher on attend environ une seconde, c'est le temps de détention.

III.7 Extraction

Dès que la dernière opération du cycle est achevée, la pièce finie est libérée des pinces qui la retiennent pour l'évacuer grâce au tapis en mouvement. Fin de cycle

Discussion

Dans ce chapitre, nous avons présenté les différents composants de la machine ainsi que son cycle de fonctionnement.

Le développement de la technologie et la compétition économique impose à l'industrie de produire en qualité et en quantité pour répondre à la demande.

Pour ces raisons, la réussite et la performance d'une installation automatique pilotée par l'automate industriel (API) repose essentiellement sur une bonne compréhension de l'installation et de la qualité des actionneurs et capteurs qu'elle comporte.

Chapitre III
Modélisation par le
GRAFCE

Préambule

Un API est programmé à l'aide de langages spécialisés, fournis par son constructeur et utilisables au travers d'une interface (un logiciel sur PC, un pupitre...). Ces langages peuvent être classés en cinq grandes familles. Cependant, deux langages de la même famille fournis par deux constructeurs différents ne sont pas forcément compatibles, ce qui est de nature à nuire à la portabilité des applications et à limiter la réutilisation du code. C'est pour cette raison que la commission électrotechnique internationale a entrepris un grand effort de normalisation visant à uniformiser les langages utilisés dans le domaine de la programmation des API, ce qui a donné naissance à la norme IEC 61131-3 [TEC93]. Ce standard définit cinq langages correspondant aux familles de langages les plus utilisées pour la programmation des API [Lew98].

I. Les différents langages**I.1 Instruction List (IL)**

Un langage textuel de type assembleur.

PROGRM And

VAR_INPUT

I1 :BOOL;

I2 : BOOL;

END_VAR

VR_OUTPUT

O :BOOL;

END_AR

LD I1

AND I2

STO

ENDPROGRAM

Un API exécutant ce programme IL joue le rôle d'une porte ET.

I.2 StructuredText (ST)

Un langage textuel structuré similaire au Pascal.

```

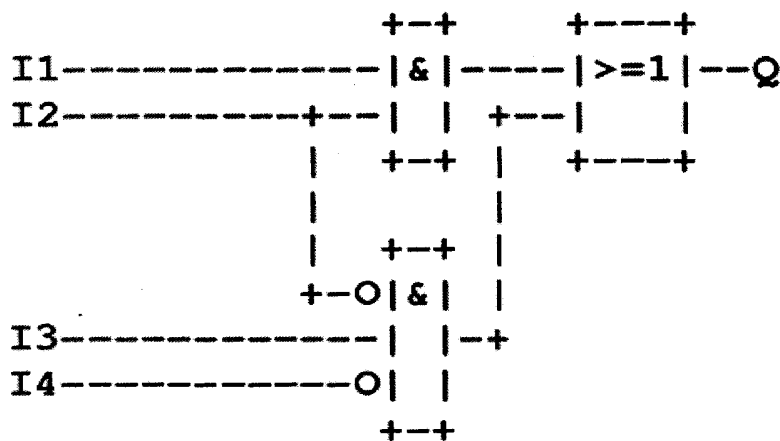
IF RUN THEN XOUT :=XOUT + K * (XIN — XOUT) ;
ELSE XOUT:=XIN ;
K ; =TIME_TO_REAL(CYCLE) / TIME_TO_REL(CYCLE + TAU)
END_IF ;

```

Fragment de code ST donné comme exemple dans [IEC93]

I.3 Function Block Diagram (FBD)

Un langage graphique permettant d'exprimer le comportement des fonctions, des blocs fonctionnels ou des programmes comme un ensemble de boîtes noires interconnectées (à la manière des portes logiques en électronique).




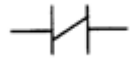
Ce programme FBD réalise l'opération suivante:

$Q := (I1 \text{ and } I2) \text{ or } (\text{not } (I2) \text{ and } I3 \text{ and } \text{flot } (I4))$

I.4 LadderDiagram (LD)

Un langage graphique, très utilisé en milieu industriel, car il s'inspire des circuits de commande basés sur la logique électrique, les équations combinatoires étant câblées à l'aide de contacts et de relais. Un programme est décrit par un diagramme sous forme d'échelle. Chaque échelon de l'échelle contient un ensemble de symboles graphiques qui peuvent être des contacts ou des bobines. Un contact permet la lecture d'une variable booléenne tandis qu'une bobine permet d'affecter une valeur à une variable booléenne.

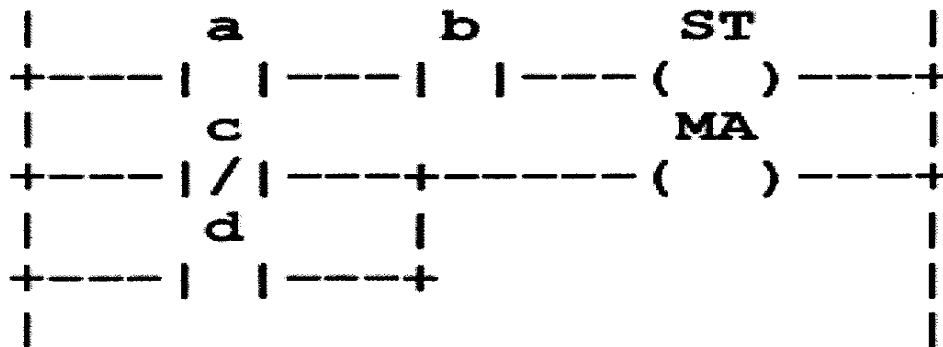
Les composants graphiques élémentaires d'un diagramme LD sont:

 Variable d'entrée ou contact à fermeture
 Variable d'entrée complémentaire ou contact à ouverture

—()— Variable de sortie

—(s)— Sortie mise à un, mémorisée (S=set)

—(R)— Sortie mise à zéro, mémorisée (R=reset)



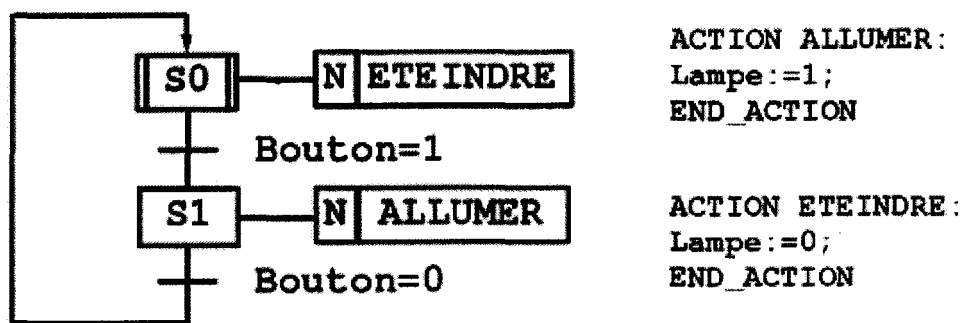
Ce programme Ladder réalise les opérations suivantes:

ST: =aandb

MA: =not(c) or d

I.5 Sequential Function Charts (SFC)

Un langage graphique permettant de structurer tout comportement séquentiel pouvant être décrit dans l'un des quatre autres langages de la norme.



Initialement, l'étape S0 est activée. Quand la réceptivité Bouton=1 devient vraie alors l'étape S0 est immédiatement désactivée et l'étape S1 immédiatement activée : la transition entre ces deux étapes est franchie. Si l'étape S1 est activée et la réceptivité Bouton=0 est vraie, alors

l'étape Si est désactivée et l'étape S0 activée. Le qualificateur d'action N indique que l'action associée à l'étape est exécutée durant l'activation de l'étape. Les actions ETEINDRE ET ALLUMER associées respectivement aux étapes SO et Si sont décrites en ST (de même que les réceptivités).

II. Définition d'un grafcet

Le GRAFCET (Graphe Fonctionnel de Commande Etape-Transition) est un outil graphique de description du comportement déterministe de la Partie Commande.

Le GRAFCET décrit les interactions informationnelles à caractère déterministe à travers la frontière d'isolement entre la Partie Commande et la Partie Opérative d'un système isolé.

Il établit une correspondance à caractère séquentiel et combinatoire entre:

- les ENTREES, c'est-à-dire les transferts d'informations de la Partie Opérative vers la Partie Commande,
- et les SORTIES, transferts d'informations de la Partie Commande vers la Partie Opérative.

III. Niveau d'un grafcet

III.1 Grafcet du niveau 1 ou spécification fonctionnelles

On décrit le comportement de la partie commande vis-à-vis de la partie opérative et du monde extérieur. On définit seulement les différentes fonctions, informations et commandes impliquées dans l'automatisation de la partie opérative du système automatisé, sans préjuger en aucune façon des technologies qui seront employées.

III.2 Grafcet du niveau 2 ou spécifications technologiques et opérationnelles.

Les spécifications technologiques viennent compléter les spécifications fonctionnelles en précisant la façon dont vont se faire les échanges entre l'automatisme et le processus. A ce niveau on indique comment les actions sont réalisées en pratique, compte tenu du matériel défini pour les capteurs et les actionneurs, des contraintes de sécurité, d'exploitation,... etc

IV. Eléments de base d'un grafcet

Le fonctionnement d'un automatisme peut être représenté graphiquement par un ensemble:

- d'étape** auxquelles sont associées des actions
- de **transitions** auxquelles sont associées des **réceptivités**

-de liaisons(ou arcs) orientées

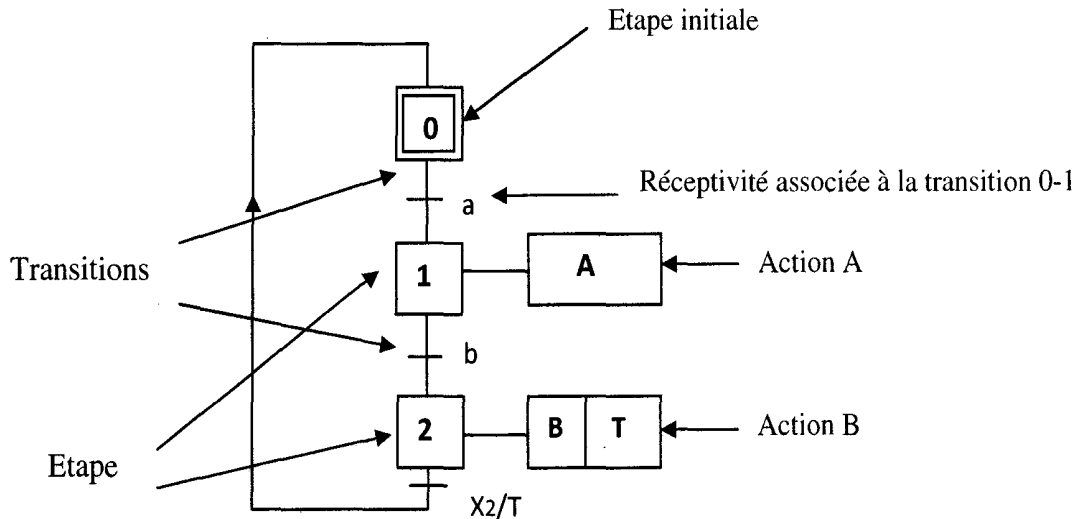


Figure III.1:Symbolisation d'un grafcet

- **Etape** : correspond à une situation dans laquelle le comportement de tout ou partie du système par rapport à ses entrées et sorties est invariant
- **Action**: a chaque étape on peut associer une ou plusieurs actions.il se également qu'aucune action ne soit associée à une étape.une action peut être associée à plusieurs étapes.
- **Transitions** : est une condition de passage d'une étape à une autre. Elle n'est que logique (dans son sens Vrai ou Faux), sans notion de durée. La condition est définie par une RECEPTIVITE.
- **Réceptivité** : est généralement une expression booléenne (c.à.d. avec des ET et des OU) de l'état des CAPTEURS.
- **Liaisons orientées** : est un arc orienté (ne peut être parcouru que dans un sens). A une extrémité d'une liaison il y a une (et une seule) étape, à l'autre une transition. On la représente par un trait plein rectiligne, vertical ou horizontal. Une verticale est parcourue de haut en bas, sinon il faut le préciser par une flèche. Une horizontale est parcourue de gauche à droite, sinon le préciser par une flèche
- La représentation graphique de tous ces éléments est décrite dans le tableau suivant:



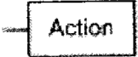
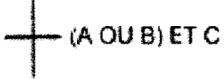
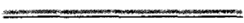

	ÉTAPES INITIALES NUMÉROTÉES
	ÉTAPES NUMÉROTÉES
	ACTIONS ASSOCIÉES AUX ÉTAPES
	TRANSITIONS ET RÉCEPTIVITÉS ASSOCIÉES
	LIAISONS DE CONVERGENCE / DIVERGENCE
	LIAISONS ORIENTÉES

Tableau III.1 : la représentation graphique des éléments de base d'un grafcet

V. Syntaxe et règles d'évolution

V.1 Syntaxe

- L'alternance étape-transition et transition-étape doit toujours être quelle que soit la séquence parcourue.
- Deux étapes ou deux transitions ne doivent jamais être reliées par une liaison orientée. la liaison orientée relie obligatoirement une étape à une transition ou une transition à une étape.

V.2 Règles d'évolution

Le grafcet fonctionne en suivant cinq règles d'évolution

➤ Règle 1 : situation initial

- Étapes actives (possibilité de plus d'une étape initiale) au début du fonctionnement. Correspond habituellement au comportement au repos du procédé.
- Si l'automatisme n'est pas cyclique (état initial dépendant des entrées dès la mise en marche du procédé), alors il peut être utile de « forcer » des étapes initiales.

- En logique, l'activation de ces étapes se fait suivant un créneau unitaire et unique provoqué par le démarrage de l'appareil.

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

- Une transition est dite validée lorsque toutes les étapes immédiatement précédentes reliées à cette transition sont actives

- Le franchissement d'une transition se produit si les deux conditions suivantes sont vraies:

- la transition est validée

- la réceptivité associée à cette transition est vraie

- Une transition franchissable est obligatoirement et automatiquement franchie.

➤ **Règle 3 : évolution des étapes actives**

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

➤ **Règle 4 : évolutions simultanées**

- Plusieurs transitions simultanément franchissables sont simultanément franchies (automatique).

➤ **Règle 5: activation et désactivation simultanées**

Si une étape est simultanément activée et désactivée au cours d'un même balayage, alors cette étape reste active.

VI. Structure de base

VI.1 Divergence et convergence en ET

- 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". Des étapes de synchronisations sont souvent indispensables avant la convergence "en et" car la durée des différentes branches est très rarement synchrone.

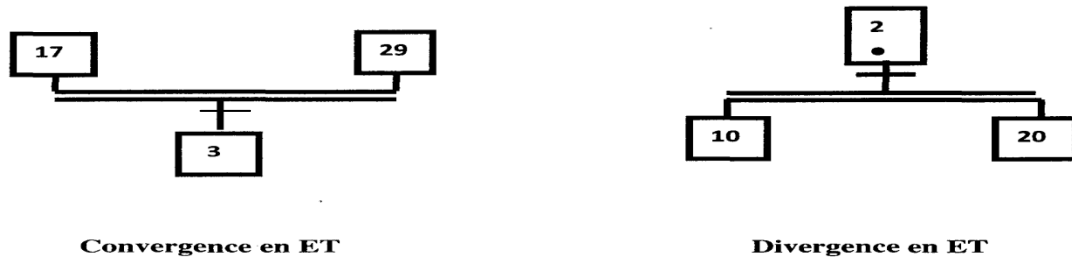


Figure III.2: convergence et divergence en ET

- 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”.

VI.2 Divergence et convergence en OU

- 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” (respectivement de divergence “en ou”).
- Les transitions lors d’une divergence “en ou” doivent avoir un caractère exclusif. Cela peut apparaître dans la réceptivité ou sur la partie opérative elle-même

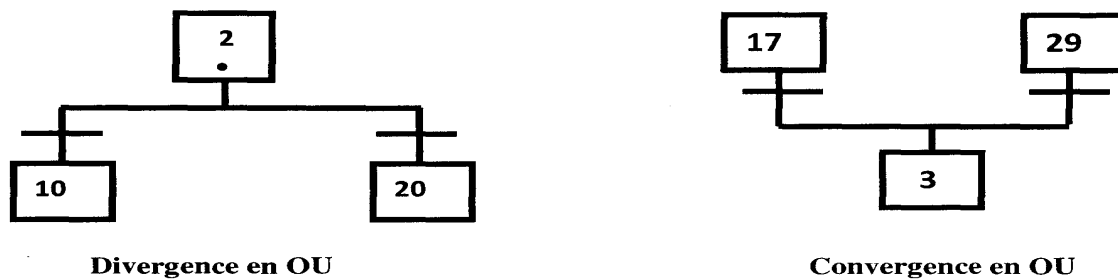


Figure III.3 : divergence et convergence en OU

VI.3 Saut en avant (saut d’étape)

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.

Le saut en avant permet de sauter une ou plusieurs étapes lorsque les actions à réaliser deviennent inutiles.

Dans la figure suivante, si la réceptivité e est vraie et l'étape initiale et active, alors les actions associées aux-étapes 11 et 12 ne se produisent pas. La prochaine action qui va se produire est celle associée à l'étape 13, c'est un saut d'étape.

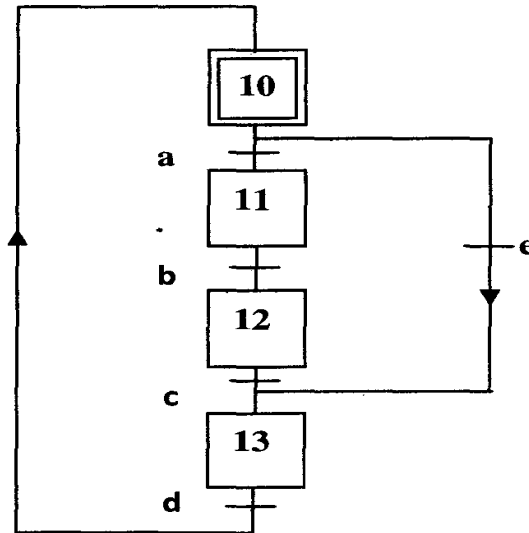


Figure III.4: Saut d'étape dans un grafcet.

VI.4 Saut en arrière (reprise de séquence)

Le saut en arrière permet de reprendre une séquence lorsque les actions à réaliser sont répétitives.

La reprise de séquence représentée dans la figure suivante permet de recommencer la séquence qui comporte les étapes 21, 22, 23 une ou plusieurs fois tant que la condition fixée i n'est pas obtenue.

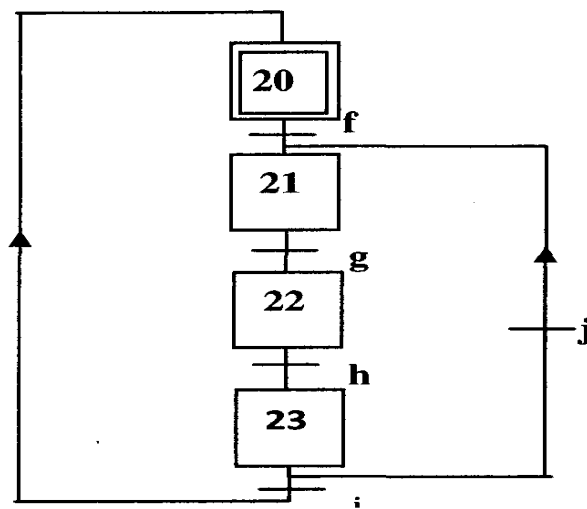


Figure III.5 : Reprise de séquence dans un grafcet.

VII. Mise en équation d'un grafcet

Soit la partie du grafcet représentée par la figure ci-après pour décrire l'activité de l'étape i , on utilise la notation i est active et : $X_i=1$ si l'étape i est active et $X_i=0$ si l'étape i est inactive.

La réceptivité t_i , étant une variable binaire, a pour valeur: $t_i=1$ si la réceptivité associée à la transition (2) est vraie et $t_i=0$ si la réceptivité associée à la transition (2) est fausse.

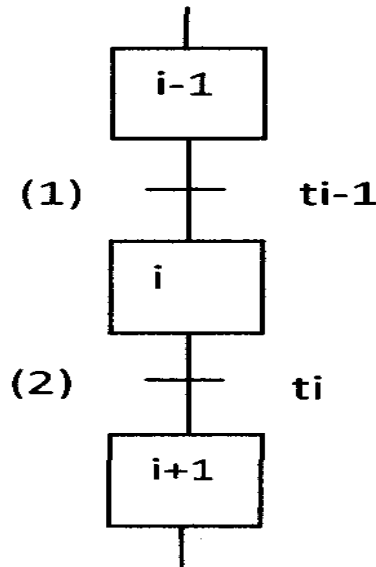


Figure III.6: Model simple du grafcet.

L'équation logique d'une transition contient l'état de l'étape validant cette transition (ici c'est l'étape i identifiée X_i) et la réceptivité de cette transition (t_i).

L'équation logique de la transition y_i est: **$Y_i = X_i \cdot t_i$**

L'équation logique d'une étape vérifie l'état de la transition précédant cette étape et celui de la transition qui suit.

L'équation logique de l'étape X_i est: **$X_i = Y_{i-1} + X_i \cdot Y_i$**

VIII. Le passage du Grafcet au langage Ladder

Une fois les conditions d'activation et de désactivation de chaque étape et de chaque action sont obtenues, on passe facilement au diagramme LADDER. L'exemple suivant illustre ce passage.

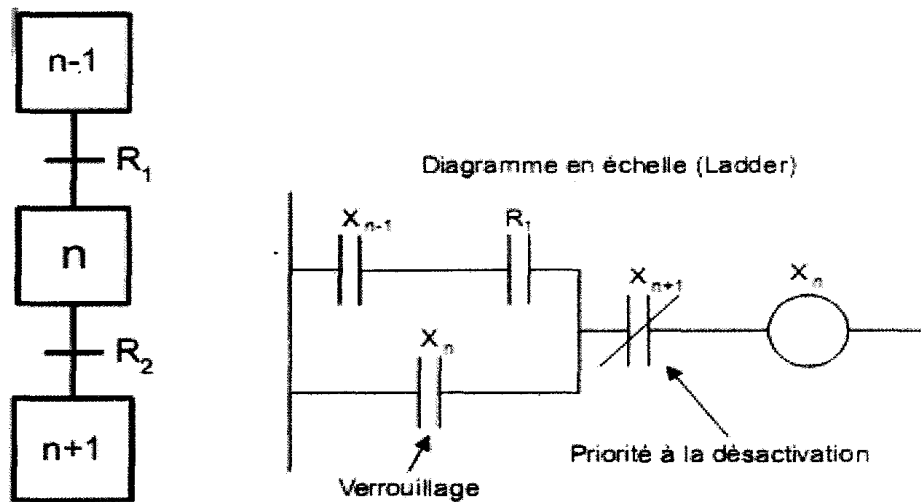


Figure 111.7: le passage du grafcet en Ladder.

IX. Les modes de marche

Dans notre machine on constate deux modes de marche, automatique et manuelle. Le mode de marche manuel, comme l'indique son nom est le mode où l'opérateur fait marcher la machine manuellement. Dans notre cas, quand le mode de marche manuel est sélectionné, il n'y a que la coupe ruban et le groupe d'avancement qui marchent manuellement.

Le mode de marche automatique est divisé en deux modes essentiels, marche continue et marche intermittente.

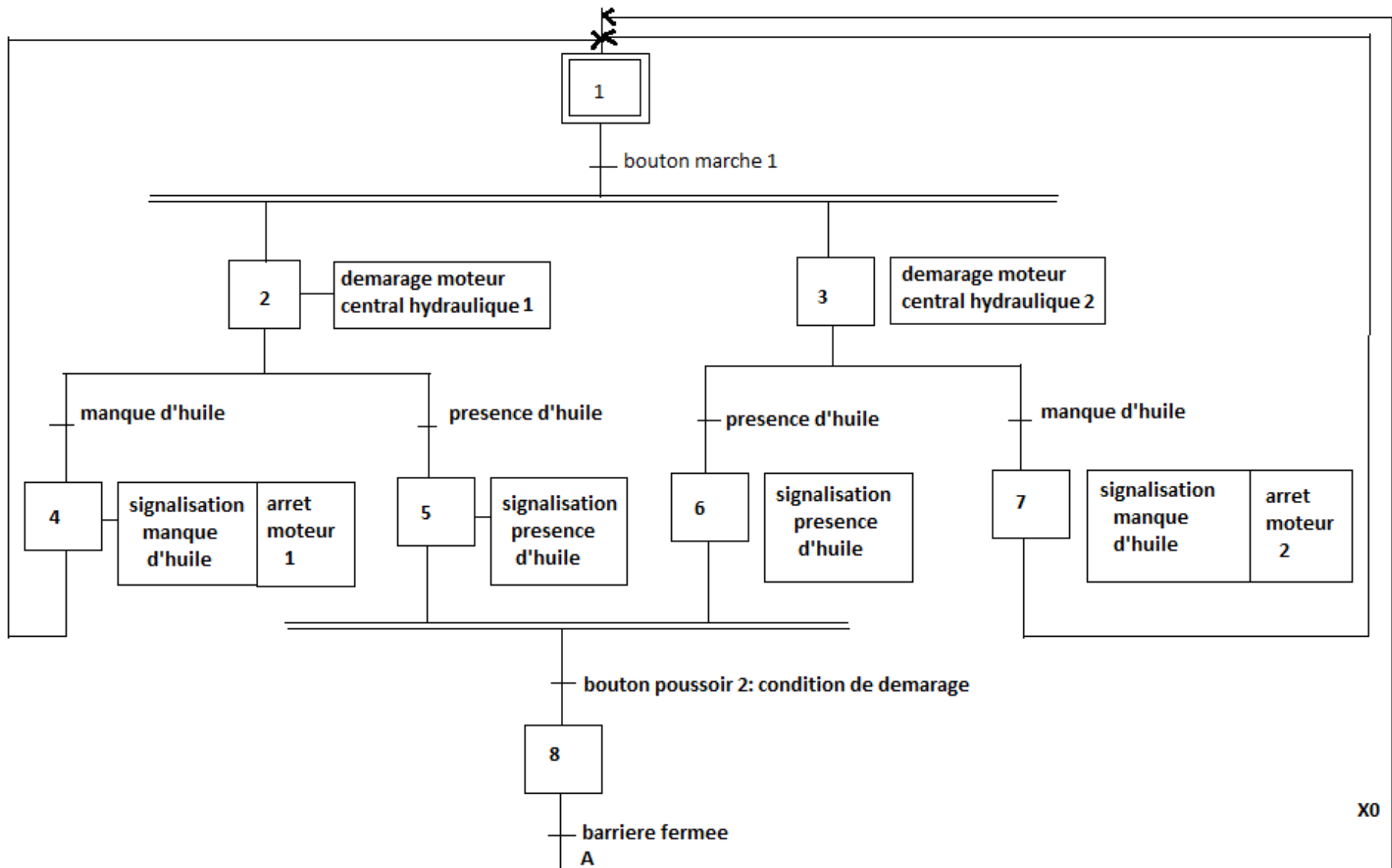
La marche continue est l'accomplissement du cycle de fonctionnement et le répéter sans cesse.

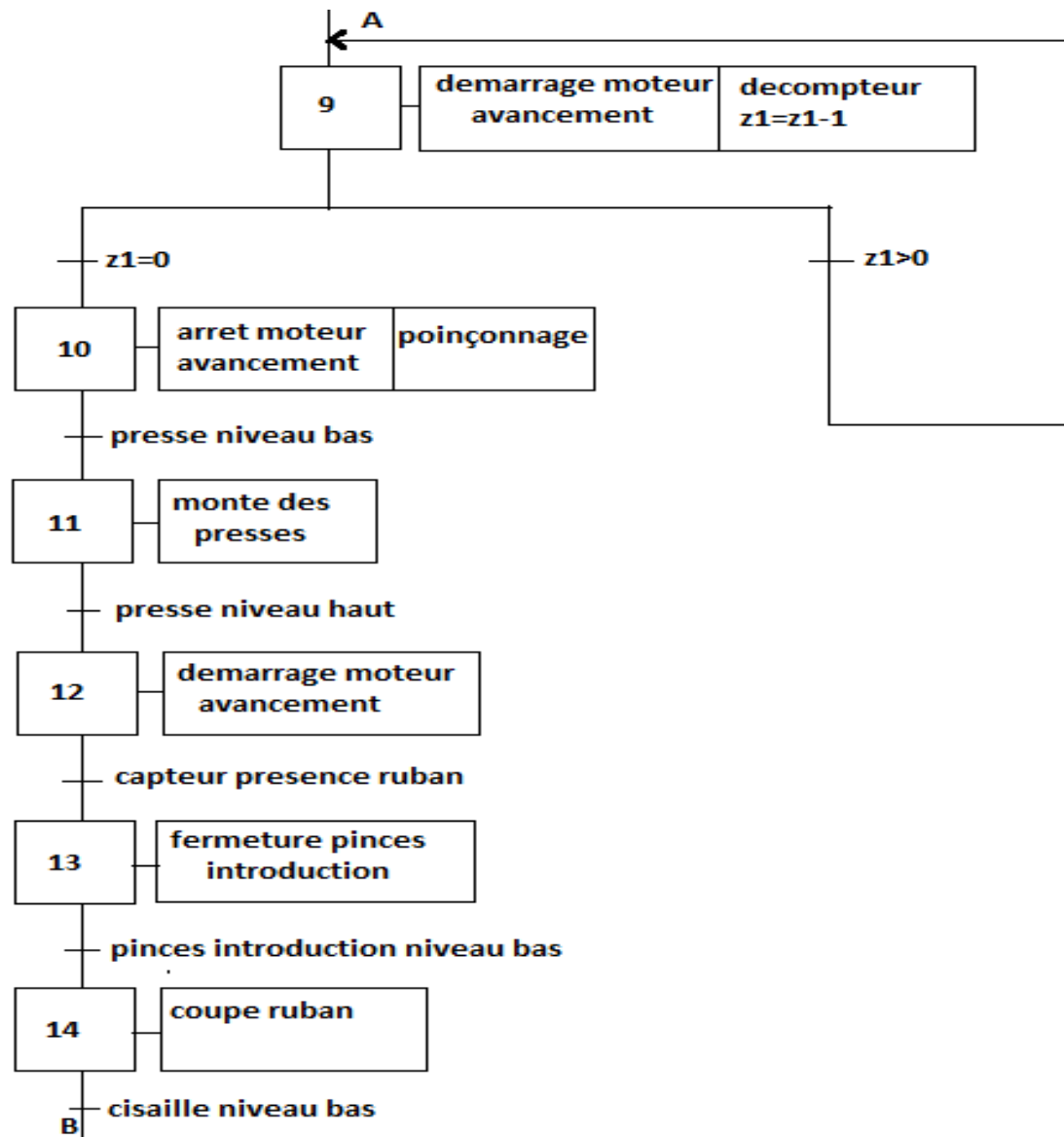
La marche intermittente est l'accomplissement du cycle de fonctionnement étape par étape en appuyant chaque fois sur le bouton départ cycle.

X. Application du grafcet pour modéliser le système

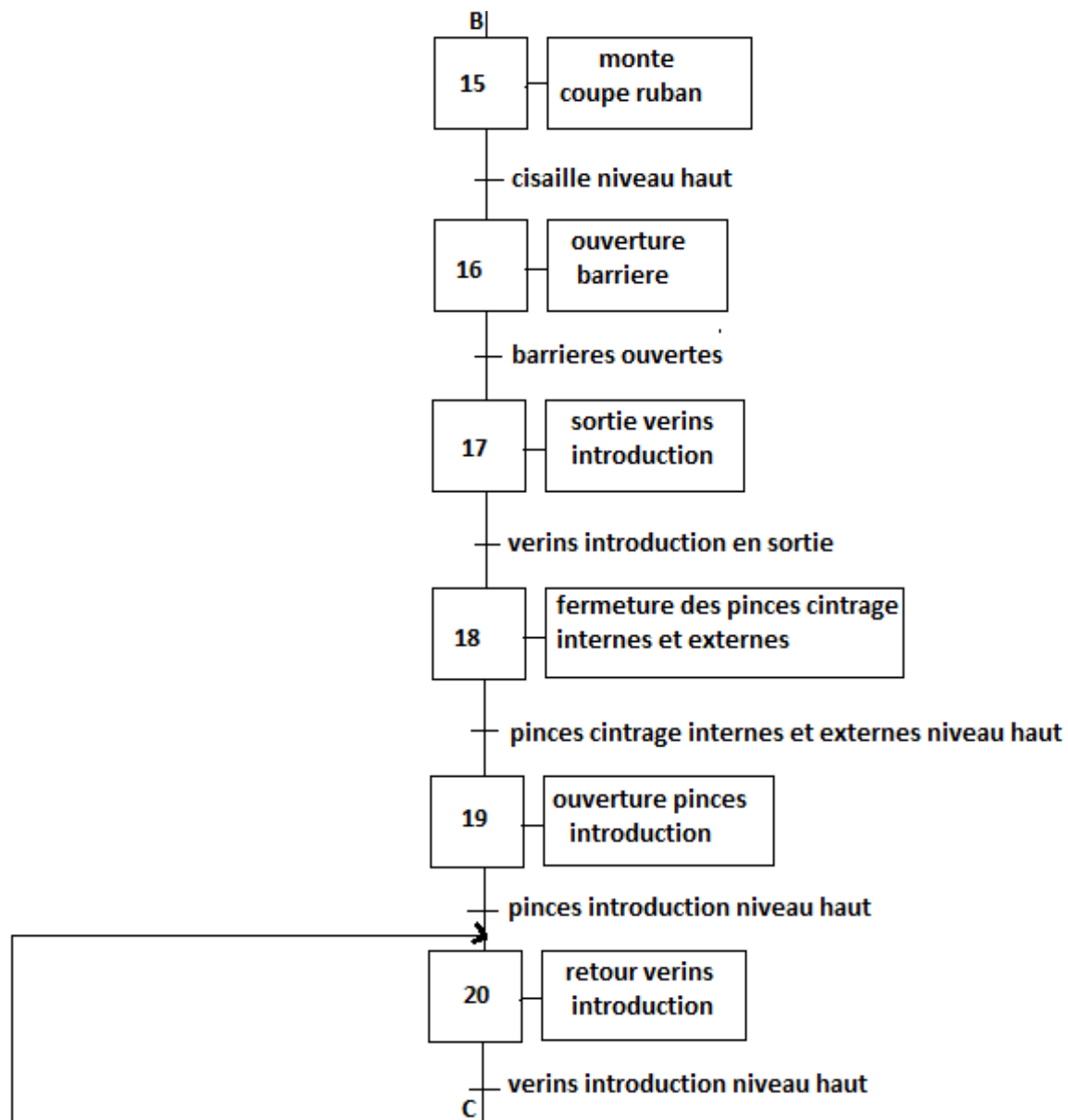
X.1 grafcet niveau 1 :

Grafcet niveau 1

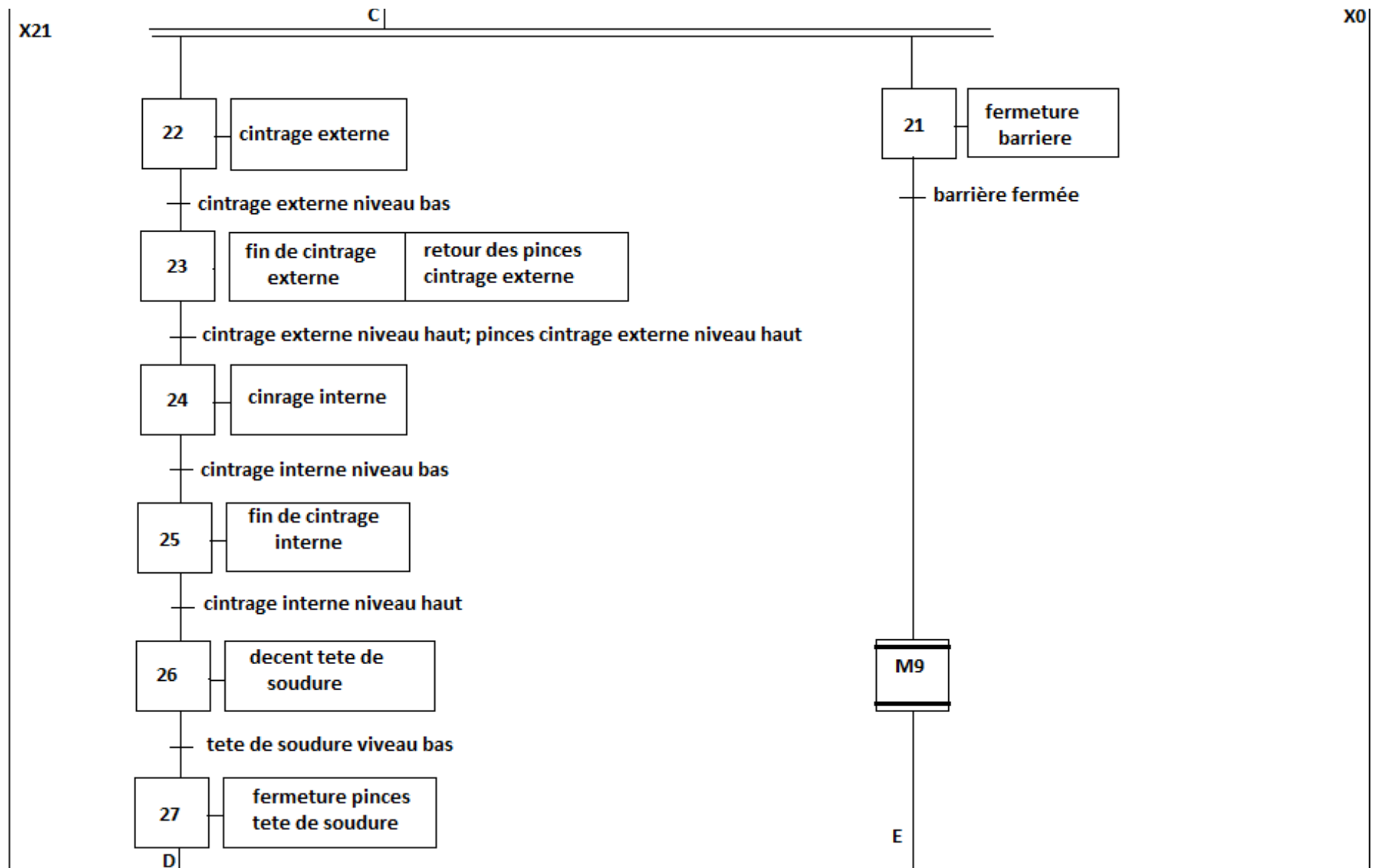


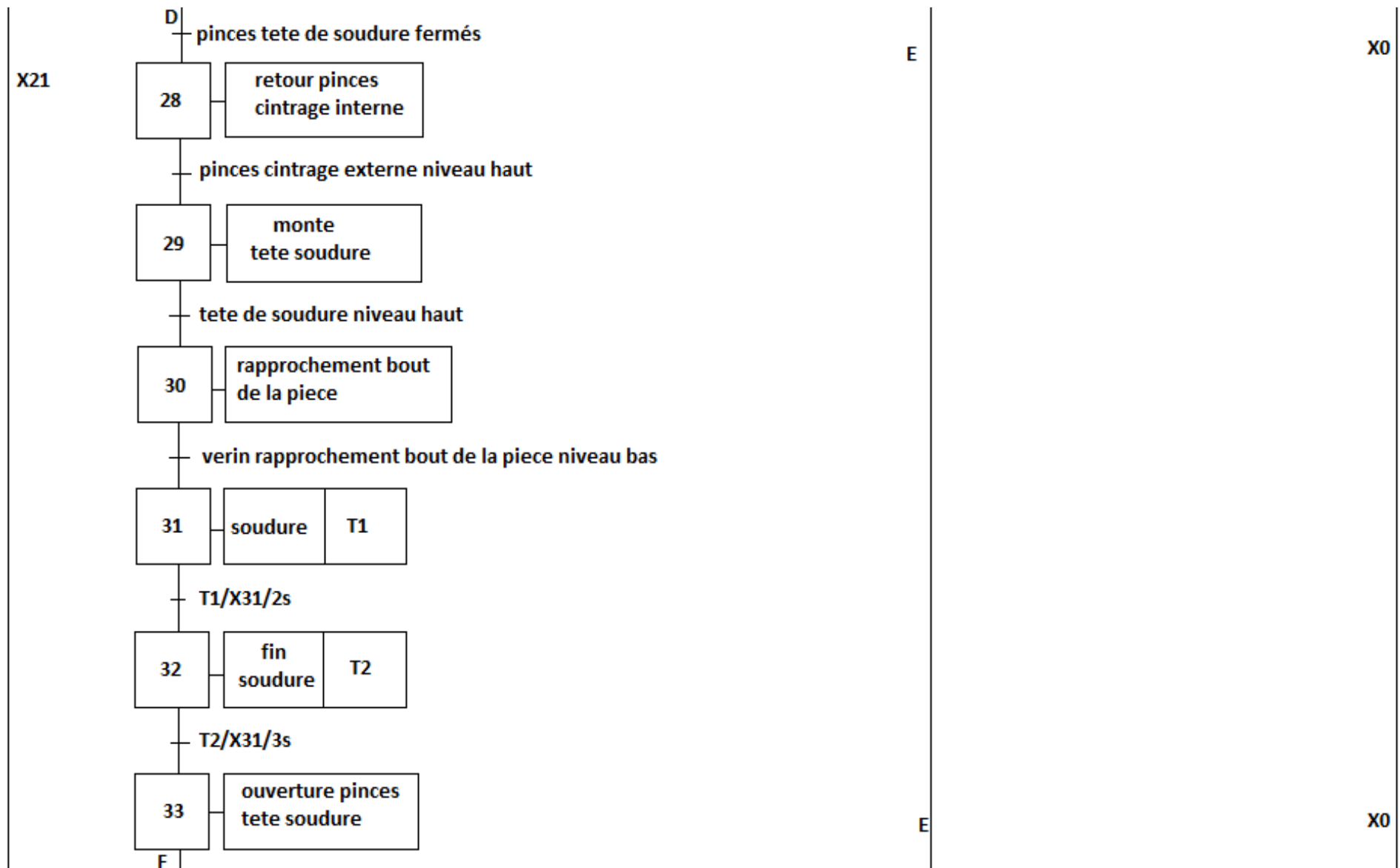


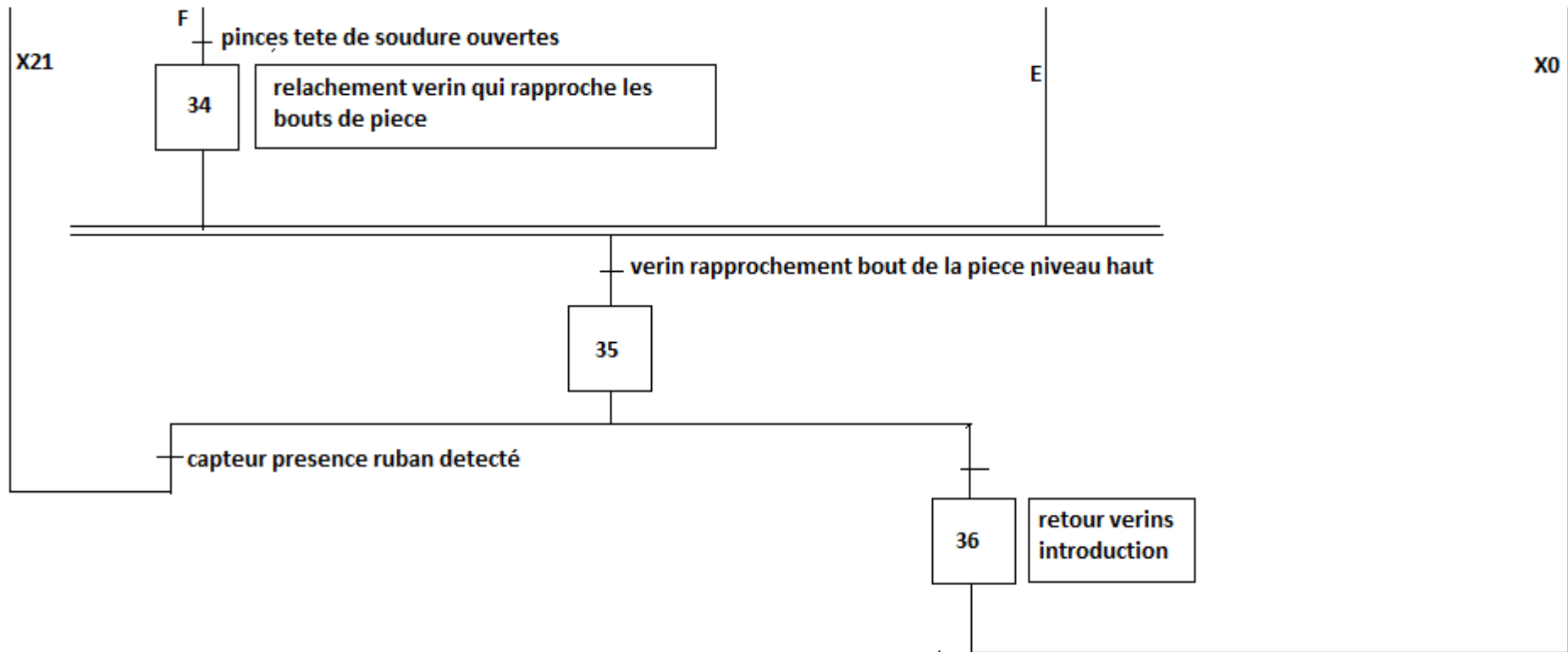
X0



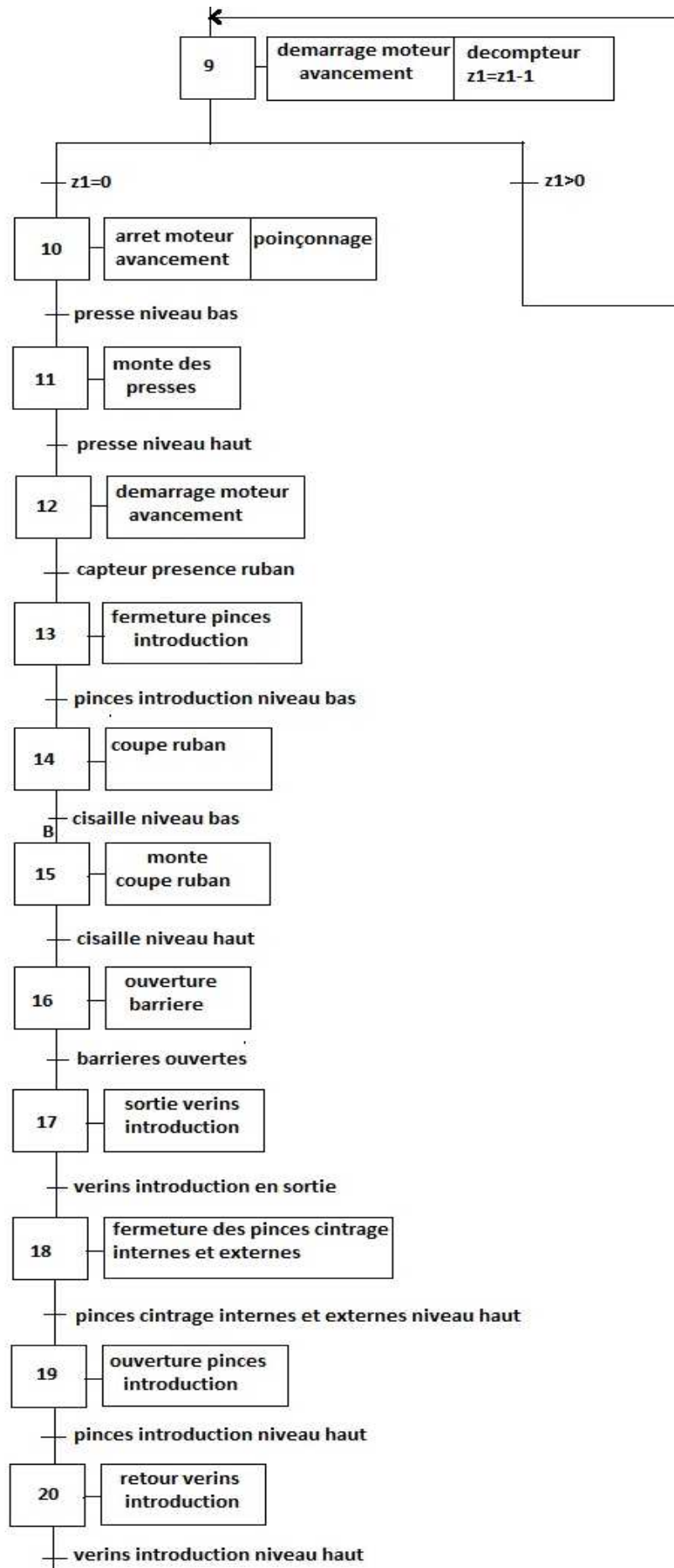
X0







La tâche (M9)



Discussion:

Le langage GRAFCET permet de combler les lacunes des diagrammes à relais et la combinaison des deux permet une programmation efficace des automatismes séquentiels.

Le modèle grafcet que j'ai développé est caractérisé par sa fiabilité, sa simplicité à comprendre et à mettre en œuvre.

Chapitre IV
Description et choix de
l'automate

Préambule :

Après la modélisation de la soudeuse par le GRAFCET, l'étape suivante consiste à concevoir le programme qui sera implanté dans l'automate S7-300, et avant d'entamer la programmation j'ai jugé utile de présenter l'automate utilisé et citer les critères sur lequel mon choix est basé.

I. Les critères de choix de l'automate programmable S7-300 :

Les critères de choix de l'automate programmable S7-300 sont désignés par le cahier de charge établi au préalable et qui sont :

- Les compétences/expériences de l'équipe d'automaticiens en mise en œuvre et en programmation de la gamme d'automate.
- La qualité du service après-vente.
- Les capacités de traitement du processeur (vitesse, données, opérations, temps réel...).
- Le type des entrées/sorties nécessaires (numériques, analogiques, booléennes).
- Le nombre d'entrées/sorties nécessaires.
- La fiabilité.
- La situation pédagogique.

II. Présentation de l'automate programmable S7-300 :

L'automate programmable S7-300 est un mini automate modulaire pour des applications d'entrées et de milieu de gamme fabriqué par firme SIEMENS, il est conçu pour des solutions systèmes innovantes dédiées au secteur manufacturier et constitue une plate forme d'automatisation universelle et optimale pour les applications dans architectures centralisées et décentralisées.

La gamme des modules comprend :

- Unités centrales (CPU) de capacités différentes, certaines avec des entrées/sorties intégrées (ex: CPU314C) ou avec interface PROFIBUS intégrée (ex: CPU315-2DP).
- Modules d'alimentation PS avec 2A, 5A ou 10A.
- Modules d'extension IM pour configuration de plusieurs lignes du SIMATIC S7-300.
- Modules de signaux SM pour entrées et sorties numériques et analogiques.
- Modules de fonction FM pour fonctions spéciales (ex: pilotage d'un moteur pas à pas).
- Processeurs de communication CP pour la connexion réseau.
- Des modules d'alimentations pour le raccordement du S7-300 sur le secteur **120/230V**.

II.1. Caractéristiques l'automate S7-300

L'automate programmable S7-300 offre les caractéristiques suivantes :

- Le mini-automate modulaire pour applications de puissance petite à moyenne.
- Une gamme étoffée de modules permettant l'adaptation optimale à la tâche d'automatisation.

- Souplesse d'utilisation grâce à la simplicité de réalisation d'architectures décentralisées et aux multiples possibilités de mise en réseau.
- Facilité de mise en œuvre procurée par le confort de manipulation et une constitution simple et sans ventilateur.
- Evolutivité permettant de suivre la croissance des installations.
- Haut niveau de performance procuré par les nombreuses fonctions intégrées.

II.2. Constitution de l'automate programmable S7-300 :

S7-300 : modules

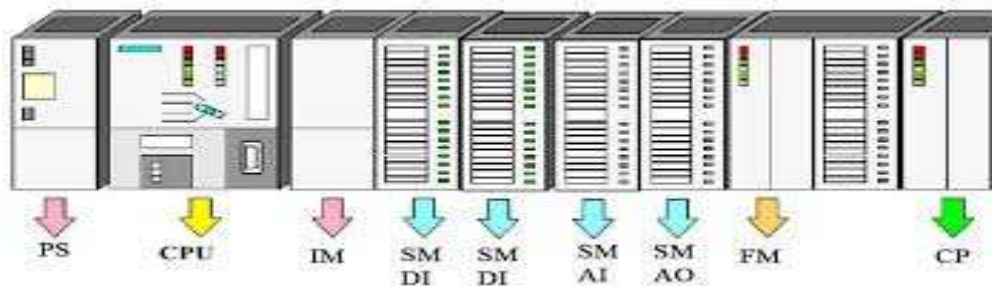


Figure II.2 : constitution de l'automate S7-300

II.2.1. Module d'alimentation (PS) :

Pour alimenter la CPU de l'automate S7-300 un réseau de **24V** suffit.

PS 307 :

Caractéristiques	Référence	Entrée	Sortie	Rendement
CS				
2A	6ES7 307-1BA00-0AA0	120/230V AC	24V DC	83% environ
5A	6ES7 307-1EA80-0AA0	120/230V AC	24V DC	87% environ
10A	6ES7 307-1KA01-0AA0	120/230V AC	24V DC	87% environ

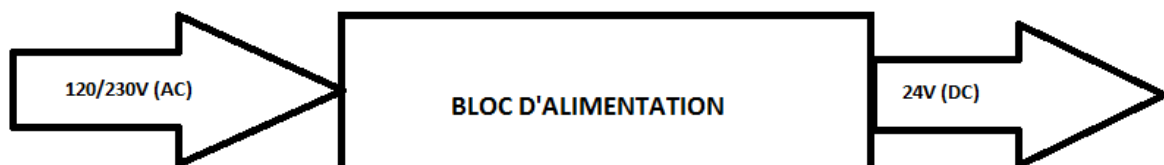


Figure II.2.1 : Modules d'alimentation

II.2.2. L'unité centrale (CPU) : Central Processing Unit) :

c'est le cerveau de l'automate, elle lit les états des entrées puis elle exécute le programme utilisateur en mémoire pour qu'elle puisse commander les sorties (donner des ordres à partir du programme utilisateur).

La gamme S7-300 offre une grande variété de CPU et chacune possède des caractéristiques différentes des autres selon le type de système automatisé choisi. Elle est constituée des blocs suivants :

A. Interface (MPI : Interface Multipoints) :

C'est l'interface de communication de l'unité centrale vers PG/PC (Console de programmation ou ordinateur avec logiciel STEP7) ou vers d'autres stations au sein d'un sous réseau MPI.

B. Signalisation d'état :

Pour contrôler le fonctionnement de la CPU, cette dernière comporte plusieurs LED de signalisation suivante :

- **SF** (rouge) :
Ce signal apparaît s'il y a :
 - ✓ Un défaut du matériel.
 - ✓ Une erreur de programmation.
 - ✓ Une erreur de paramétrage.
 - ✓ Une erreur de calcul.
- **BAFT** (rouge) :
Ce signal apparaît s'il y a un défaut dans la pile :
 - ✓ Défectueuse.
 - ✓ Absente.
 - ✓ Déchargé
- **DC 5V** (verte) :
Alimentation 5Vcc pour la CPU et le bus S7-300, si ce signal apparaît cela veut dire que 5V sont présents mais dès qu'il clignote alors il y a une surcharge de courant.
- **FRCE** (jaune) :
Ce signal apparaît s'il y a un forçage permanent.
- **RUN** (verte) :
C'est un état de fonctionnement, il clignote en cas de démarrage de la CPU.
- **STOP** (jaune) :
Ce signal apparaît si la CPU ne traite aucun programme utilisateur, s'il clignote donc la CPU demande un effacement général.

C. Commutateur de mode :

Le commutateur de mode et les éléments d'affichage de toutes les CPU sont identiques, ils ont les mêmes rôles et fonctions. On définit ici les quatre positions principales de la CPU d'un S7-300 :

- **RUN-P (mode de fonctionnement RUN de programme) :**

La CPU traite le programme utilisateur et la clé ne peut pas être retirée, il est possible de lire des programme de la CPU avec une PG et de transférer les programme dans la CPU.

- **RUN :**

La CPU traite le programme utilisateur, dans cette position la clé peut être retirée pour éviter qu'une personne non habilitée change de mode de fonctionnement.

- **STOP :**

La CPU ne traite aucun programme utilisateur, la clé peut être retirée pour éviter le changement de mode inattendue mais on peut lire et écrire dans la CPU.

- **MRES :**

Effacement générale, c'est une position instable du commutateur de mode de fonctionnement ; en vue de l'effacementgénéral de la CPU le contenu mémoire de chargement rémanente intégrée reste inchangé après l'effacement.

D. Carte mémoire :

Son rôle est de conserver le contenu du programme en cas de coupure de courant, même en absence de pile.

E. La pile :

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

F. Borne pour l'alimentation et la terre fonctionnelle :

Ce module est commun à la majorité des CPU des S7-300, on trouve les différentes bornes d'alimentation de la CPU tel que :

- ✓ Cavalier amovible pour le montage sans liaison à la terre.
- ✓ La terre.

II.2.3. Coupleur (IM) :

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

Les coupleurs IM 306/IM 361 ou IM365 permettent de réaliser des configurations à plusieurs châssis.

II.2.4. Modules de communication (CP) :

Ces modules permettent d'établir des liaisons Homme-Machine qui sont effectués à l'aide des interfaces de communication.

II.2.5. Modules des signaux (SM) :

Ils comportent plusieurs types tels que : STOR, ETOR, SANA, EANA ou E/SANA et E/STOR, ils ont comme fonction d'adaptation des niveaux de signaux entre le processus et le S7-300.

II.2.6. 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 différents modules tels que les modules d'entrées/sorties et d'alimentation.

II.2.7. Modules de simulation (S7-300) :

Il possède les caractéristiques suivantes :

- Simulation de :
 - ✓ 16 entrées.
 - ✓ 16 sorties.
 - ✓ 8 entrées et 8 sorties.
- Fonction pouvant être sélectionnée à l'aide d'un tournevis.
- LED simulant l'état des entrées ou sorties.

II.3. Console de programmation PG ou PC avec logiciel de simulation :

Elle a pour fonction la configuration, paramétrage, programmation et teste de S7-300.

III. Mode de fonctionnement de l'automate S7-300 :

L'automate programmable S7-300 exécute le programme cyclique qui commence par l'acquisition des entrées issues par des capteurs sur l'état du processus et fini par l'envoi des sorties sous forme d'ordres aux actionneurs.

III.1. Réception des informations sur l'état 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 jours la mémoire image des entrées au début de chaque cycle de programme, en transférant le nouveau 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.

III.2. Exécution du programme utilisateur :

Après l'analyse des informations d'entrées et exécution du système d'exploitation, la CPU passe à l'exécution du programme utilisateur qui contient la liste des instructions pour faire fonctionner le procédé. Il est composé essentiellement en :

- ✓ Bloc de donnée.
- ✓ Bloc de code.
- ✓ Bloc d'organisation.

III.3. La commande de processus :

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

IV. Programmation de l'automate S7-300 :

Le **STEP7** est le logiciel de base pour la configuration et la programmation de systèmes d'automatisés, il fait partie de l'industrie logicielle SIMATIC et il existe en plusieurs versions, il offre les fonctions suivantes pour l'automatisation d'une installation :

- La création et la gestion de projets.
- La configuration et le paramétrage du matériel et de la communication.
- Le chargement de programme dans des systèmes cible.
- La gestion des mnémoniques.
- La création de programmes, par exemple pour les systèmes cible S7.
- Le test de l'installation d'automatisation.
- Le diagnostic lors de perturbations de l'installation.

La programmation en STEP7 présente trois modes de représentations possibles qui peuvent être combinés dans une même application :

1. Le schéma logique (LOG).
2. Le schéma contact (CONT).
3. Liste d'instruction (LIST).

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

IV.1. Création d'un projet STEP7 :

Pour créer un projet STEP7 on a deux possibilités :

1. Commencer par la configuration matérielle.
2. Commencer par la création de programme.

Une fois Windows démarré, on trouve dans l'interface Windows une icône pour



SIMATIC Manager qui permet d'accéder au logiciel STEP7 SIMATIC Manager. On démarre rapidement STEP7 en effectuant un double clic sur l'icône « SIMATIC Manager ». La fenêtre de SIMATIC Manager s'ouvre alors. De là, on peut accéder à toutes les fonctions qu'on a installées aussi bien du logiciel de base que des logiciels optionnels.

L'autre méthode consiste à lancer SIMATIC Manager via le bouton « Démarrer » dans la barre des tâches du système d'exploitation (sous « Simatic »).

SIMATIC Manager constitue l'interface d'accès à la configuration et à la programmation.

On peut :

- Créer des projets.
- Configurer et paramétrer le matériel.
- Configurer le fonctionnement en réseau du matériel.
- Programmer des blocs.
- Tester et mettre en œuvre nos programmes.

Avec SIMATIC Manager, on peut travailler :

- Hors ligne, c'est-à-dire avec un automate soit raccordé
- En ligne, c'est-à-dire avec un automate raccordé.

La figure suivante illustre la constitution d'un projet :

1. La fenêtre suivante permet la création d'un projet.

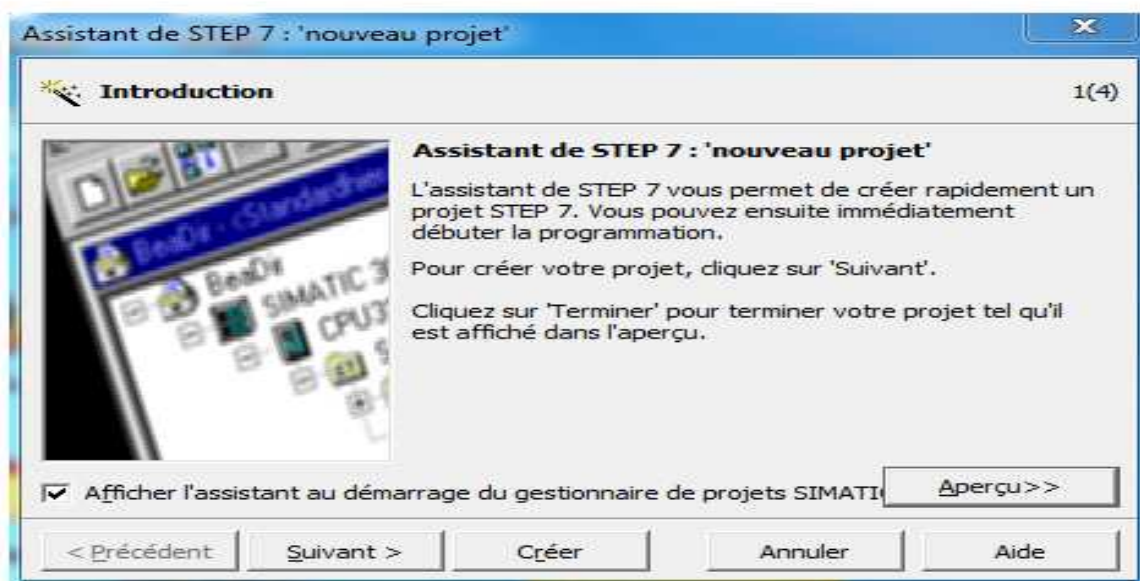


Figure IV.1.a : Fenêtre de création d'un projet.

2. On clique sur suivant, la fenêtre suivante nous permet de choisir la CPU.

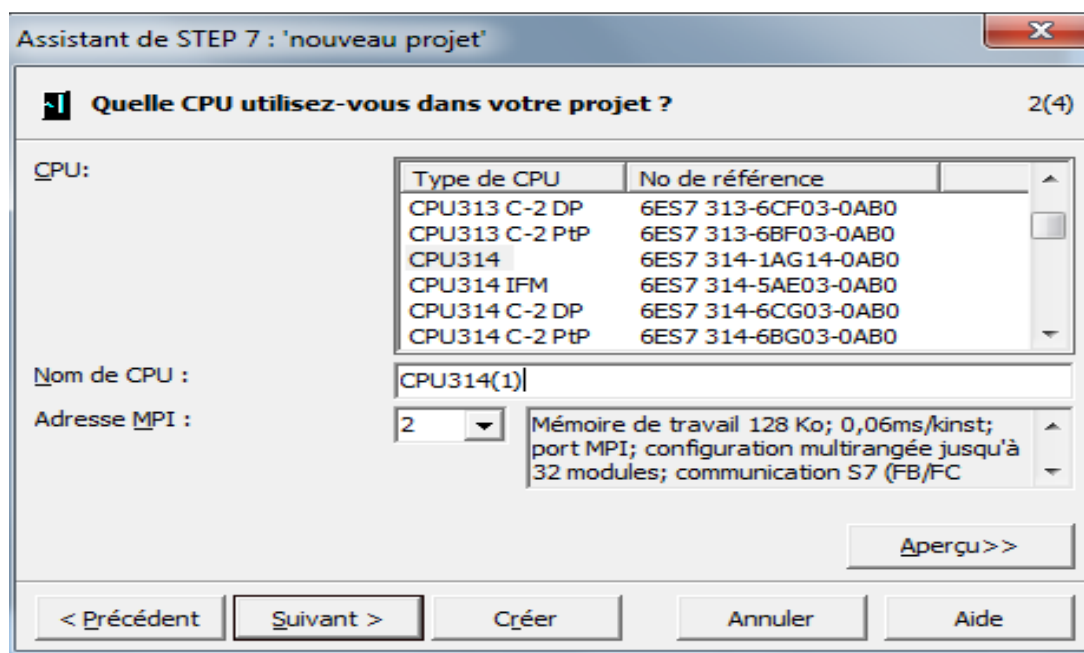


Figure IV.1.b : CPU choisie

- Après validation de la CPU, une fenêtre qui apparaît permet de choisir les blocs et le langage de programmation à insérer.

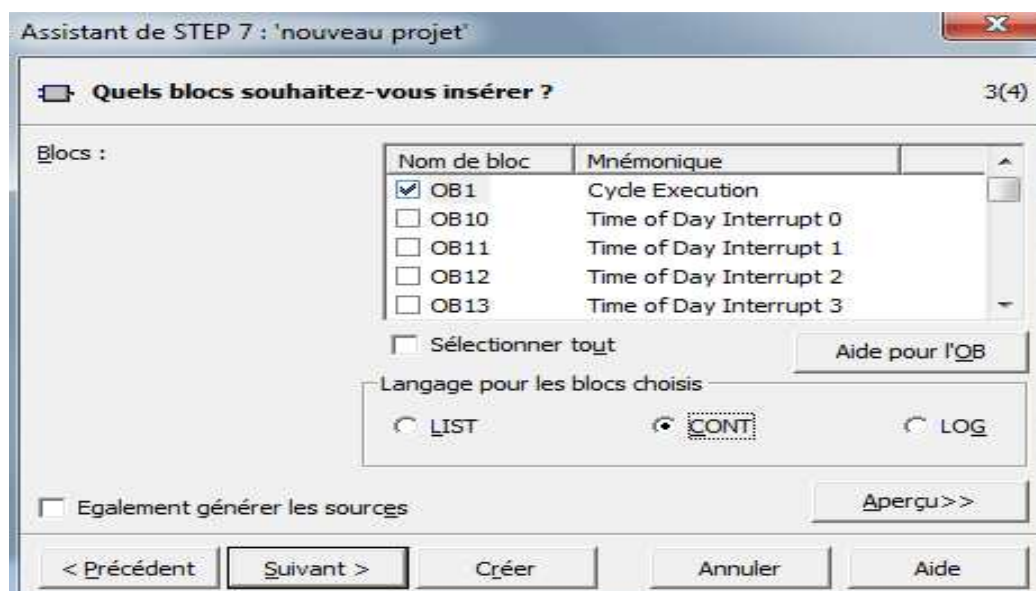


Figure IV.1.c : Sélection des blocs et le langage de programmation (CONT)

- En cliquant sur suivant, une dernière fenêtre pour la création du projet apparaît pour le nommer.

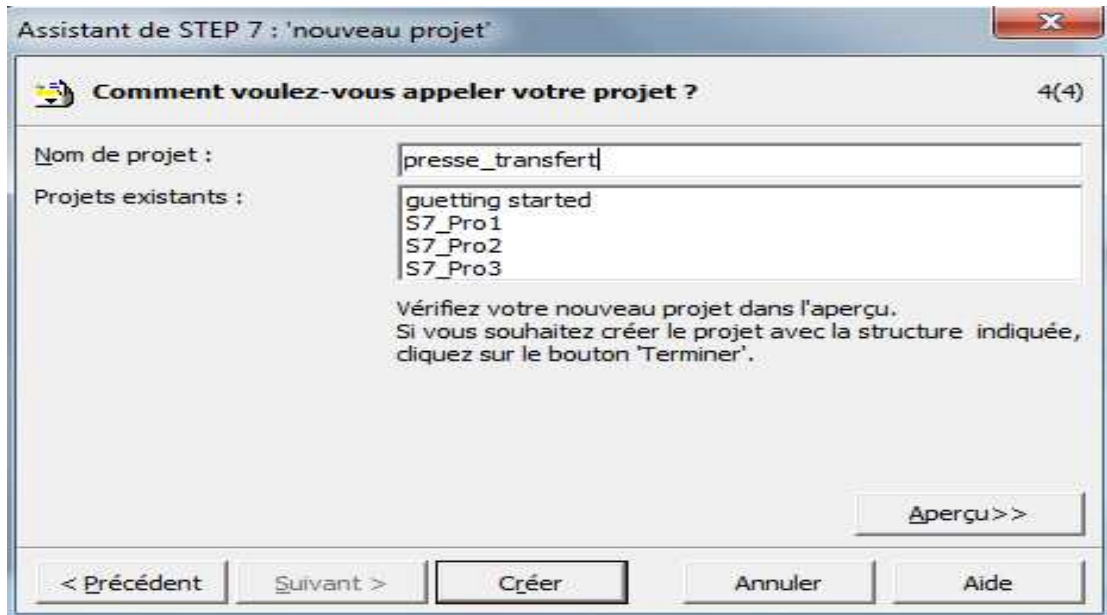


Figure. IV.1.d : Nomination du programme.

5. Une fois le projet créé, il est nécessaire de configurer le matériel à utiliser comme le montre la (voir la figure. IV.1.f).

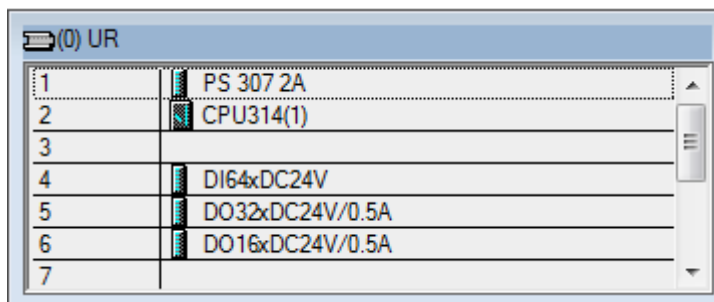


Figure. IV.1.e : Configuration matériels.

6. Ensuite en passe au programme utilisateur que j'ai écrit pour commander la machine, ce dernier est composé d'objets définis dans l'environnement de STEP7.

V. 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 du 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) :** à cette étape 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 derniers vers les modules de sorties.

- **Phase (5) :** à la fin du cycle, le système d'exploitation exécute les travaux en attente, tel que le changement et l'effacement des blocs ou la réception et l'émission des données globales.
- La CPU revient alors au début du cycle et démarre à nouveau la surveillance du temps de cycle.

VI. Le rôle des systèmes d'exploitation 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.

VI.1. Rôle du système d'exploitation :

Il organise toutes les fonctions et procédures qui ne sont pas liées à une tâche d'automatisation spécifique, ces tâches sont :

- Le déroulement du démarrage et du redémarrage.
- Actualisation de la mémoire image des entrées et émission de la mémoire image des sorties.
- Appel de programme utilisateur.
- Enregistrement des alarmes et l'appel des OB d'alarme.
- 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.

VI.2. Rôle du programme utilisateur :

Il se charge dans la CPU et il contient toutes les fonctions nécessaires au traitement des tâches d'automatisation, il doit prendre en charge :

- La détermination des conditions pour le démarrage et le redémarrage de la CPU.
- La réaction aux alarmes.
- Le traitement des données du processus (exp : combiner des signaux binaires ; lire et exploiter des valeurs analogiques, fixer les signaux binaires pour les sorties et écrire des valeurs analogiques).
- Traitement des perturbations dans le déroulement normal du programme.

VI.3. Les blocs dans le programme utilisateur :

Le STEP 7 offre la possibilité de structurer notre programme utilisateur, c'est-à-dire le subdiviser en différentes parties autonome qui donne les avantages suivants :

- Ecritures des programmes importants clairs.
- Standardiser certaines parties du programme.
- Simplification de l'organisation du programme.

- Modification facile du programme.
- Simplifier le teste de programme, car on peut l'exécuter section par section.
- Faciliter la mise en service.

VI.4. Les différents blocs du programme utilisateur :

On peut programmer notre automate très simplement, en créant un programme utilisateur qu'on charge ensuite dans la CPU de notre S7-300. Ce programme utilisateur à créer comprend différents blocs.

Le logiciel de base STEP 7 dans ces différents langages de programmation possède un nombre important de ces blocs d'utilisateur, destiner à structurer le programme utilisateur et permet aux blocs, on cite les plus importants :

VI.4.1. Blocs d'organisations :

Un bloc d'organisation (OB) constitue l'interface entre le système d'exploitation de la CPU et notre programme utilisateur. L'OB spécifie l'ordre selon lequel son traités les blocs du programme utilisateur. Le dispositif de commande est informé dans cet OB par des commandes d'appel de blocs, de quels blocs de programme il doit traiter.

VI.4.2. Bloc fonctionnel (FB) :

Le FB est à disposition via un espace mémoire correspondant. Si un FB est appelé, il lui est attribué un bloc de données (DB). On peut accéder aux données de cette instance DB par des appels depuis le FB. Un FB peut être attribué à différents DB.

D'autre FB et d'autres FC peuvent être appelés dans un bloc de fonction par des commandes d'appel de blocs.

VI.4.3. Fonction (FC) :

Une fonction (FC) est un bloc de code qui est sans « mémoire », mais qui peut transmettre des paramètres. L'utilisation de ce bloc est particulièrement intéressant pour programmer des fonctions qui reviennent fréquemment. Une FC ne possède pas un espace mémoire attribué. Les données locales d'une fonction sont perdues après le traitement de la fonction. D'autres FB et FC peuvent être appelés dans une fonction par des commandes d'appel de blocs.

VI.4.4. Bloc de données (DB) :

Les DB sont employés afin de mettre à disposition de l'espace mémoire pour les variables de données. Il y a deux catégories de blocs de données. Les DB globaux où toute les OB, FB et FC peuvent lire des données enregistrées et écrire eux-mêmes des données dans DB. Les instances DB sont attribuées à un FB défini.

Discussion :

J'ai présenté l'automate programmable industriel et en particulier l'automate S7-300 a été choisi comme solution adéquate et extensible, facile à adapter aux diverses conditions non seulement industrielles mais aussi différents secteurs.

Vu le degré de complexité de notre processeur, l'utilisation de la programmation structurée est indispensable.

Une fois tout les FC programmés, j'ai inséré ces derniers dans le bloc d'organisation OB1 pour la phase de simulation, ce qui sera l'objectif du prochain chapitre.

Chapitre V
Simulation avec le
logiciel S7-PLCSIM

I. Logiciel de simulation S7-PLCSIM:

S7-PLCSIM est un logiciel optionnel de STEP7. Son utilisation suppose que la version de base de STEP7 est déjà installée.

Le domaine d'application du logiciel S7-PLCSIM est le teste du programmeSTEP7 pour les automates S7-300 et S7-400 qu'on ne peut pas tester immédiatement sur le matériel. Ceci peut avoir différentes raisons:

Petits blocs de programme qui ne peuvent pas encore être testés dans une séquence unique sur la machine.


L'application est critique, car elle peut occasionner des dommages matériels ou blessures corporelles en cas d'erreurs de programmation. La simulation permet de supprimer ces erreurs dès la phase de test et dans le cas ou nous ne disposons pas d'un automate.

II. Mise en route du logiciel S7-PLCSIM :

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

On ouvre le <SIMATIC Manager> par un double clic sur son icône.



Cliquer sur  ou sélectionner la commande Outils-simulation de modèles, cela lance l'application S7-PLCSIM et ouvre une fenêtre CPU (Figure II.1)

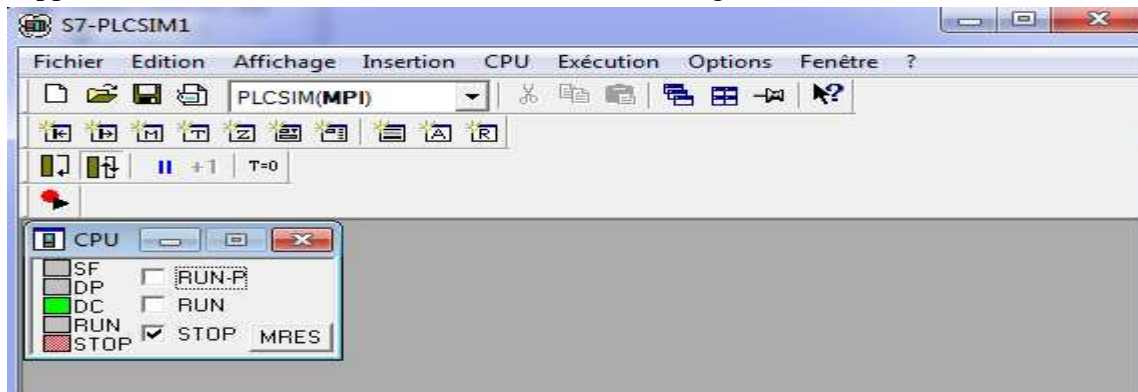





Figure II.1 : fenêtre du S7-PLCSIM.

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

- ✓ Pour ajouter une entrée on clique sur 
- ✓ Pour ajouter une sortie on clique sur 
- ✓ Pour ajouter un memento on clique sur 

Mettre la CPU de simulation en marche en cliquant sur l'une des cases à cocher RUN ou RUN-P (figure II.2).

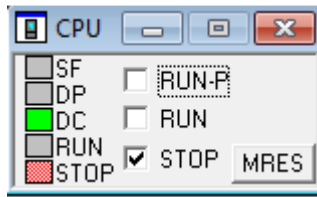


Figure II.2 : Mise en marche de la CPU.

Une fois toutes les fenêtres d'entrées et sorties sont prêtes, nous insérons la table des mnémoniques pour lire l'état des sorties (Figure II.3)

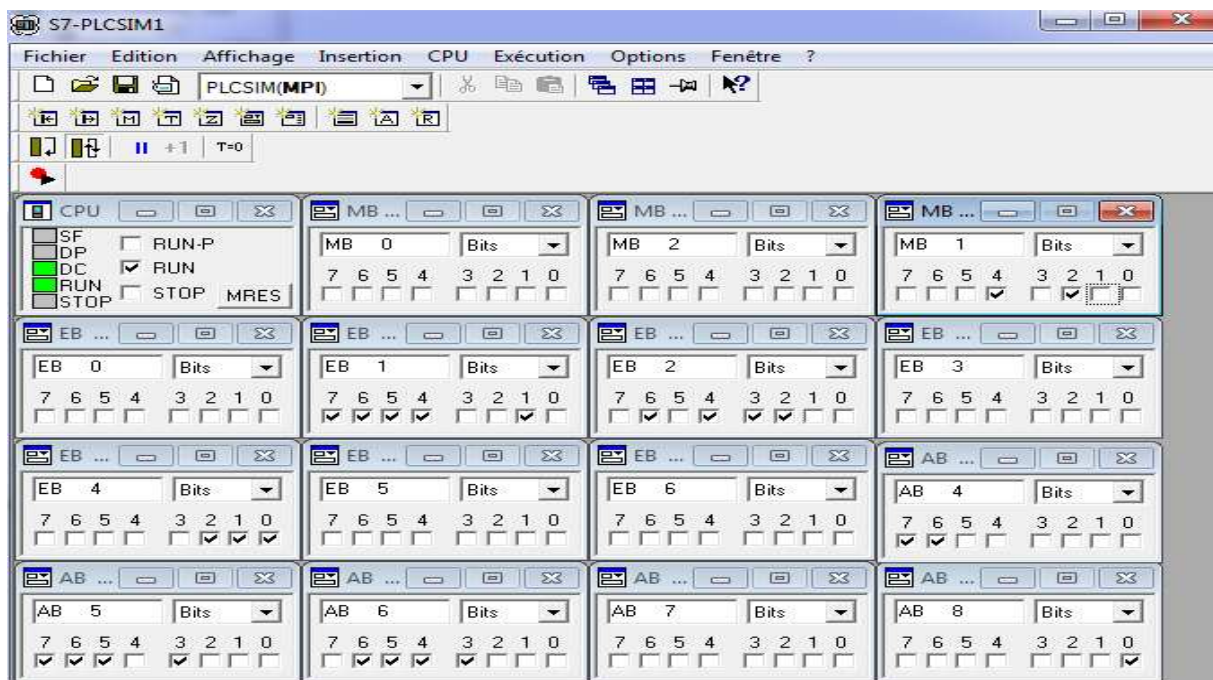



Figure II.3 : simulateur S7-PLCSIM.

III. 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'icone  ou sélectionnant la commande **Test-Visualiser**.

Discussion :

Le programme utilisateur que j'ai développé pour la machine a été validé grâce à l'utilisation du S7-PLCSIM. Ce logiciel dispose d'une interface permettant de surveiller et de modifier le programme développé afin de le rendre opérationnel pour une éventuelle implantation réelle sur un automate programmable industriel.

Conclusion générale

Conclusion générale

Le travail que j'ai effectué dans le cadre de ce projet de fin d'étude, avec l'appui du stage pratique à l'entreprise nationale des industries de l'électroménager ENIEM, m'a permis de découvrir la réalité de l'activité d'un complexe industriel, de mettre ainsi la mise en pratique de nos connaissances théoriques et de me familiariser avec les automatismes industriels dans le but de concevoir une solution programmable pour la machine soudeuse grille de table (GE48).

Après l'étude de la machine, j'ai proposé une solution de commande automatisée à base d'API S7-300 et ce grâce à l'outil puissant de modélisation qui est le GRAFCET.

J'ai constaté aussi durant notre projet, que les facteurs de sécurité et fiabilité doivent être pris en considération par l'ingénieur, c'est-à-dire qu'il faut s'assurer de la fiabilité des composants, pour garantir un bon fonctionnement du procédé. Comme il doit penser à la sécurité du personnel et du matériel par intégration de capteurs et de conditions d'arrêt d'urgence. Ces mesures sont très importantes car elles permettent d'arrêter les actions pouvant avoir des conséquences dangereuses en cas d'une défaillance humaine ou matérielle.

J'espère que ce projet puisse être réalisé sur le plan pratique et qu'il puisse être un guide pour tout projet d'automatisation utilisant les API S7-300 et le langage de programmation STEP7.

Bibliographie

Bibliographie :

✓ **Les ouvrages :**

Titre	Auteur	Edition
Application industriel de GRAFCET	Sylvain Thelliez, Jean Marc Toulotte	Eyrolles, 1982
Automate programmable industriel	Patrick. Jacquard et Serge Sandre	Dunod paris 1995
Automatismes industriels	J-M. Bleux J-L. Fanchon	Dunod paris 1995
Les capteurs	Georges Asch et Collaborateur	Dunod
Graphcet	Noel Elbouteille, Paul Brad	2 ^{eme} édition 1995

✓ **Les thèses :**

Titre	Cote	Année
Automatisation d'une soudeuse a grille a l'entreprise ENIEM	45 ING ELN	2010
Conception d'une commande automatisée d'une soudeuse de grilles a l'entreprise ENIEM	20 ING AUTO	2008
Etude et automatisation par automates programmables S7-300 d'une presse transfert a l'entreprise ENIEM	35 ING ELN	2009

✓ **Les sites internet :**

- www.eniem.dz.com
- <http://www.tecoltant.fr/grafcet/>.
- http://philippe.berger2.Free.fr/automatique/cour/eltspneu/les_elements_pneumatique.htm.

✓ **documentation d'ENIEM :**

- Doc. Technique de la machine grille de table GE48

Propriétés de la table des mnémoniques

Nom : Mnémoniques

Auteur :

Commentaire :

Date de création : 29/09/2015 23:36:46

Dernière modification : 30/09/2015 19:43:55

Dernier filtre sélectionné : Tous les mnémoniques

Nombre de mnémoniques : 123/123

Dernier tri : Opérande ordre croissant

Etat	Mnémonique	Opérande	Type de données	Commentaire
	MD	A 0.0	BOOL	moteur derouleur
	MA	A 0.1	BOOL	moteur d'avancement
	MTE	A 0.2	BOOL	moteur tapis extraction
	EV1P	A 0.3	BOOL	electrovanne verin 1 presse
	EV2P	A 0.4	BOOL	electrovanne verin 2 presse
	EV3P	A 0.5	BOOL	electrovanne verin 3 presse
	EV4P	A 0.6	BOOL	electrovanne verin 4 presse
	EVC	A 0.7	BOOL	electrovanne verin cisaille
	EV1PI	A 1.0	BOOL	electrovanne verin 1 pince introduction
	EV2PI	A 1.1	BOOL	electrovanne verin 2 pince introduction
	EV1I	A 1.2	BOOL	electrovanne verin 1 introduction
	EV2I	A 1.3	BOOL	electrovanne verin 2 introduction
	EV1B	A 1.4	BOOL	electrovanne verin 1 barriere
	EV2B	A 1.5	BOOL	electrovanne verin 2 barriere
	EV1LE	A 1.6	BOOL	electrovanne verin 1 lame externe
	EV2LE	A 1.7	BOOL	electrovanne verin 2 lame externe
	EV1LI	A 2.0	BOOL	electrovanne verin 1 lame interne
	EV2LI	A 2.1	BOOL	electrovanne verin 2 lame interne
	EV1CE	A 2.2	BOOL	electrovanne verin 1 cintrage externe
	EV2CE	A 2.3	BOOL	electrovanne verin 2 cintrage externe
	EV1CI	A 2.4	BOOL	electrovanne verin 1 cintrage interne
	EV2CI	A 2.5	BOOL	electrovanne verin 2 cintrage interne
	EV1TS	A 2.6	BOOL	electrovanne verin 1 tete de soudure
	EV2TS	A 2.7	BOOL	electrovanne verin 2 tete de soudure
	EV1PS	A 3.0	BOOL	electrovanne verin 1 pince de soudure
	EV2PS	A 3.1	BOOL	electrovanne verin 2 pince de soudure
	EVRB	A 3.2	BOOL	electrovanne verin rapprochement bout de piece
	MDHC1	A 3.3	BOOL	manque d'huile centrale hydraulique 1
	MDHC2	A 3.4	BOOL	manque d'huile centrale hydraulique 2
	LMA	A 3.5	BOOL	leds moteur avancement
	LMD	A 3.6	BOOL	led moteur derouleur
	LMTE	A 3.7	BOOL	led tapis extraction
	SAR	A 4.0	BOOL	signalisation d'arret d'urgence
	MCH1	A 4.1	BOOL	moteur centrale hydraulique 1
	MCH2	A 4.2	BOOL	moteur centrale hydraulique 2
	ESO	A 4.3	BOOL	electrovanne soudure
	NH1OK	A 4.4	BOOL	niveau d'huile centrale 1 ok
	NH2OK	A 4.5	BOOL	niveau d'huile centrale 2 ok
	SRV	A 4.6	BOOL	servovalve
	bp1	E 0.0	BOOL	bouton poussoir 1
	bp2	E 0.1	BOOL	bouton poussoir 2
	init	E 0.2	BOOL	bouton initialisation
	ar	E 0.3	BOOL	bouton d'arret d'urgence
	NH1	E 0.4	BOOL	niveau d'huile centrale hydraulique 1
	NH2	E 0.5	BOOL	niveau d'huile centrale hydraulique 2
	CEN	E 0.6	BOOL	capteur encodeur

Etat	Mnémonique	Opérande	Type de données	Commentaire
	V1PH	E 0.7	BOOL	verin 1 presse haut
	V2PH	E 1.0	BOOL	verin 2 presse haut
	V3PH	E 1.1	BOOL	verin 3 presse haut
	V4PH	E 1.2	BOOL	verin 4 presse haut
	V1PB	E 1.3	BOOL	verin 1 presse bas
	V2PB	E 1.4	BOOL	verin 2 presse bas
	V3PB	E 1.5	BOOL	verin 3 presse bas
	V4PB	E 1.6	BOOL	verin 4 presse bas
	CH	E 1.7	BOOL	cisaille haut
	CB	E 2.0	BOOL	cisaille bas
	CPR	E 2.1	BOOL	capteur presence ruban
	V1PIH	E 2.2	BOOL	verin 1 pince introduction haut
	V2PIH	E 2.3	BOOL	verin 2 pince introduction haut
	V1PIB	E 2.4	BOOL	verin 1 pince introduction bas
	V2PIB	E 2.5	BOOL	verin 2 pince introduction bas
	V1IH	E 2.6	BOOL	verin 1 introduction haut
	V2IH	E 2.7	BOOL	verin 2 introduction haut
	V1IB	E 3.0	BOOL	verin 1 introduction bas
	V2IB	E 3.1	BOOL	verin 2 introduction bas
	V1BH	E 3.2	BOOL	verin 1 barriere haut
	V2BH	E 3.3	BOOL	verin 2 barriere haut
	V1BB	E 3.4	BOOL	verin 1 barriere bas
	V2BB	E 3.5	BOOL	verin 2 barriere bas
	V1LEH	E 3.6	BOOL	verin 1 lame externe haut
	V2LEH	E 3.7	BOOL	verin 2 lame externe haut
	V1LEB	E 4.0	BOOL	verin 1 lame externe bas
	V2LEB	E 4.1	BOOL	verin 2 lame externe bas
	V1LIH	E 4.2	BOOL	verin 1 lame interne haut
	V2LIH	E 4.3	BOOL	verin 2 lame interne haut
	V1LIB	E 4.4	BOOL	verin 1 lame interne bas
	V2LIB	E 4.5	BOOL	verin 2 lame interne bas
	V1CEH	E 4.6	BOOL	verin 1 cintrage externe haut
	V2CEH	E 4.7	BOOL	verin 2 cintrage externe haut
	V1CEB	E 5.0	BOOL	verin 1 cintrage externe bas
	V2CEB	E 5.1	BOOL	verin 2 cintrage externe bas
	V1CIH	E 5.2	BOOL	verin 1 cintrage interne haut
	V2CIH	E 5.3	BOOL	verin 2 cintrage interne haut
	V1CIB	E 5.4	BOOL	verin 1 cintrage interne bas
	V2CIB	E 5.5	BOOL	verin 2 cintrage interne bas
	V1TSH	E 5.6	BOOL	verin 1 tete de soudure haut
	V2TSH	E 5.7	BOOL	verin 2 tete de soudure haut
	V1TSB	E 6.0	BOOL	verin 1 tete de soudure bas
	V2TSB	E 6.1	BOOL	verin 2 tete de soudure bas
	V1PSH	E 6.2	BOOL	verin 1 pince de soudure haut
	V2PSH	E 6.3	BOOL	verin 2 pince de soudure haut
	V1PSB	E 6.4	BOOL	verin 1 pince de soudure bas
	V2PSB	E 6.5	BOOL	verin 2 pince de soudure bas
	VRBH	E 6.6	BOOL	verin rapprochement bout de piece haut
	VRBB	E 6.7	BOOL	verin rapprochement bout de piece bas
	VSOB	E 7.0	BOOL	verin soudure bas
	VSOH	E 7.1	BOOL	verin soudure haut
	G7_STD_3	FC 72	FC 72	
	VPH	M 0.0	BOOL	verins presse haut
	CD	M 0.1	BOOL	condition de demarrage
	VSH	M 0.2	BOOL	verin soudeuse haut

Etat	Mnémonique	Opérande	Type de données	Commentaire
	VCH	M 0.3	BOOL	verins cintrage haut
	VIH	M 0.4	BOOL	verins introduction haut
	VPB	M 0.5	BOOL	verins presse bas
	PIB	M 0.6	BOOL	verins pinces introduction bas
	VBH	M 0.7	BOOL	verins barriere haut
	VIB	M 1.0	BOOL	verins introduction bas
	VLB	M 1.1	BOOL	verins lame extere et interne bas
	PIH	M 1.2	BOOL	verins pinces introduction haut
	VBB	M 1.3	BOOL	verins barriere bas
	VPIH	M 1.4	BOOL	verins pinces introduction haut
	VPIB	M 1.5	BOOL	verins pinces introduction bas
	VCEB	M 1.6	BOOL	verins cintrage externe bas
	VCEH	M 1.7	BOOL	verins cintrage externe haut
	VLEH	M 2.0	BOOL	verins lame externe haut
	VCIB	M 2.1	BOOL	verins cintrage interne bas
	VCIH	M 2.2	BOOL	verins cintrage interne haut
	VTSB	M 2.3	BOOL	verins tete de soudure bas
	VTSH	M 2.4	BOOL	verins tete de soudure haut
	VPSB	M 2.5	BOOL	verins pinces de soudure bas
	VPSH	M 2.6	BOOL	verins pinces de soudure haut
	VLIH	M 2.7	BOOL	verins lame interne haut
	TIME_TCK	SFC 64	SFC 64	

OB1 - <offline>

" "

Nom :
Auteur :
Horodatage Code :
Interface :
Longueur (bloc/code /données locales) :

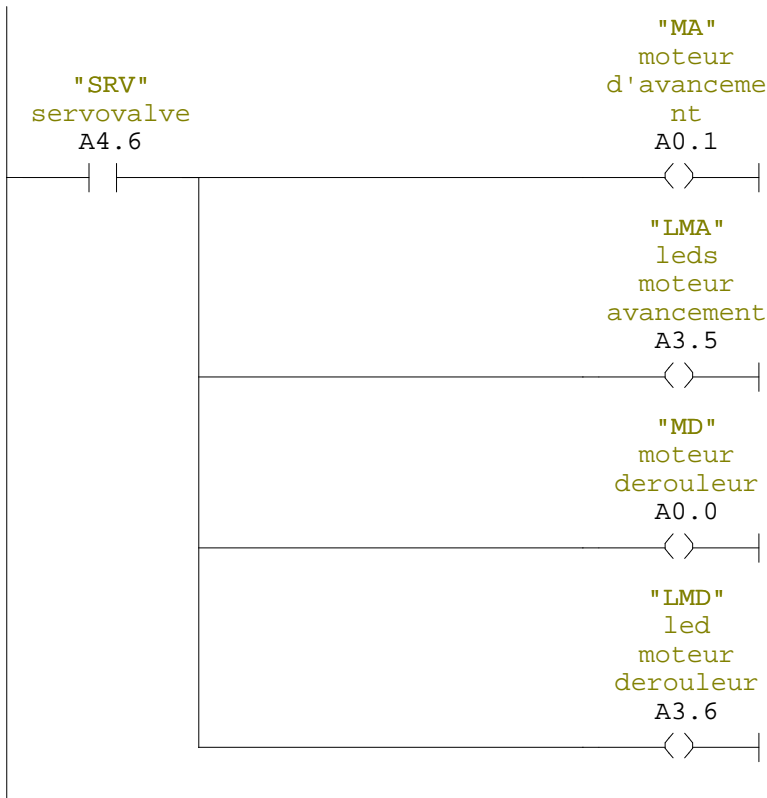
Famille :
Version : 0.1
Version de bloc : 2
30/09/2015 02:06:19
15/02/1996 16:51:12
00416 00242 00020

Nom	Type de données	Adresse	Commentaire
TEMP		0.0	
OB1_EV_CLASS	Byte	0.0	Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
OB1_SCAN_1	Byte	1.0	1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1)
OB1_PRIORITY	Byte	2.0	Priority of OB Execution
OB1_OB_NUMBR	Byte	3.0	1 (Organization block 1, OB1)
OB1_RESERVED_1	Byte	4.0	Reserved for system
OB1_RESERVED_2	Byte	5.0	Reserved for system
OB1_PREV_CYCLE	Int	6.0	Cycle time of previous OB1 scan (milliseconds)
OB1_MIN_CYCLE	Int	8.0	Minimum cycle time of OB1 (milliseconds)
OB1_MAX_CYCLE	Int	10.0	Maximum cycle time of OB1 (milliseconds)
OB1_DATE_TIME	Date_And_Time	12.0	Date and time OB1 started

Bloc : OB1 "Main Program Sweep (Cycle)"
--

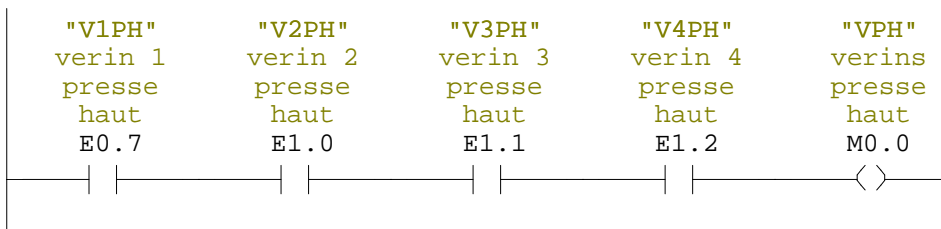
Réseau : 1 moteur d'avancement

MOTEUR AVANCEMENT



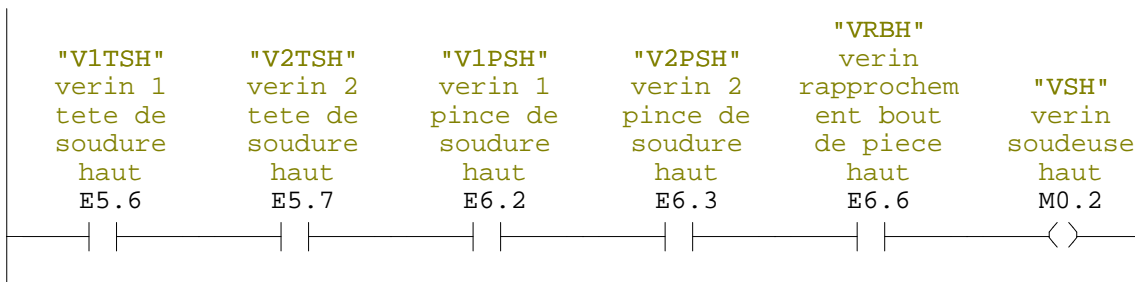
Réseau : 2 verins presse haut

VERINS PRESSE HAUT (VPH)



Réseau : 3 verin soudeuse haut

VERINS SOUDURE HAUT (VSH)



Réseau : 4 verins cintrage haut

VERINS CINTRAGE HAUT (VCH)

"V1LEH"	"V2LEH"	"V1LIH"	"V2LIH"	"V1CEH"	"V2CEH"	"V1CIH"	"V2CIH"	"VCH"
verin 1	verin 2	verin 1	verin 2	verin 1	verin 2	verin 1	verin 2	verins
lame	lame	lame	lame	cintrage	cintrage	cintrage	cintrage	cintrage
externe	externe	interne	interne	externe	externe	interne	interne	haut
haut	haut	haut	haut	haut	haut	haut	haut	haut
E3.6	E3.7	E4.2	E4.3	E4.6	E4.7	E5.2	E5.3	M0.3
								()

Réseau : 5 verins introduction haut

VERINS INTRODUCTION HAUT (VIH)

"V1IH"	"V2IH"	"V1PIH"	"V2PIH"	"V1BH"	"V2BH"	"VIH"
verin 1	verin 2	verin 1	verin 2	verin 1	verin 2	verins
introducti	introducti	introducti	introducti	barriere	barriere	introducti
on haut	on haut	on haut	on haut	haut	haut	on haut
E2.6	E2.7	E2.2	E2.3	E3.2	E3.3	M0.4
						()

Réseau : 6 condition de demarrage

CONDITION DE DEMARRAGE (CD)

"VPH"	"CH"	"VIH"	"VCH"	"VSH"	"CD"
verins	cisaille	verins	verins	verin	condition
presse	haut	introducti	cintrage	soudeuse	de
haut	haut	on haut	haut	haut	demarrage
M0.0	E1.7	M0.4	M0.3	M0.2	M0.1
					()

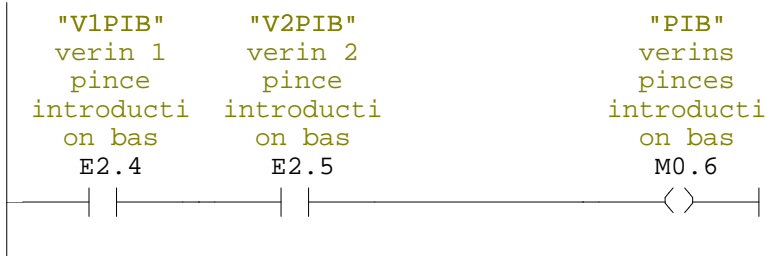
Réseau : 7 verins presse bas

VERINS PRESSE BAS (VPB)

"V1PB"	"V2PB"	"V3PB"	"V4PB"	"VPB"
verin 1	verin 2	verin 3	verin 4	verins
presse bas	presse bas	presse bas	presse bas	presse bas
E1.3	E1.4	E1.5	E1.6	M0.5
				()

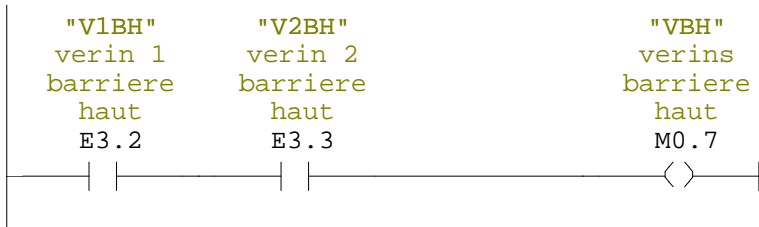
Réseau : 8 verins pinces introduction bas

PINCES INTRODUCTION BAS (PIB)



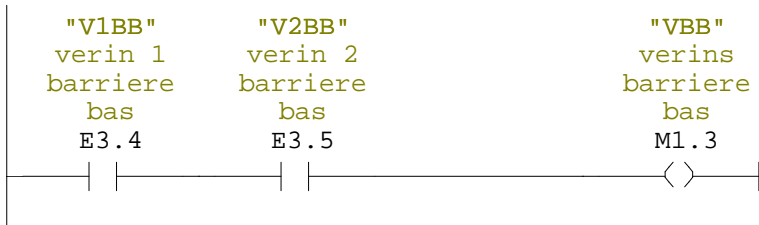
Réseau : 9 verins barriere haut

VERINS BARRIERE HAUT (VBH)



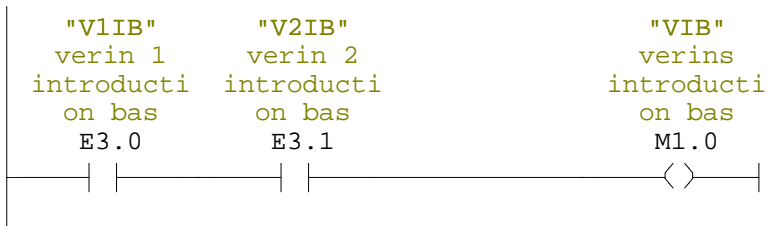
Réseau : 10 verins barriere bas

VERINS BARRIERE BAS (VBB)



Réseau : 11 verins introduction bas

VERINS INTRODUCTION BAS (VIB)



Réseau : 12	verins lame extere et interne bas
-------------	-----------------------------------

VERINS LAMES INTERNE ET EXTERNE BAS

				"VLB"
"V1LIB"	"V2LIB"	"V1LEB"	"V2LEB"	verins
verin 1	verin 2	verin 1	verin 2	lame
lame	lame	lame	lame	extere et
interne	interne	externe	externe	interne
bas	bas	bas	bas	bas
E4.4	E4.5	E4.0	E4.1	M1.1

Réseau : 13 verins pinces introduction haut

PINCES INTRODUCTION HAUT

"V1PIH"	"V2PIH"	"PIH"
verin 1	verin 2	verins
pince	pince	pinces
introducti	introducti	introducti
on haut	on haut	on haut
E2.2	E2.3	M1.2

Réseau : 14 verins introduction haut

VERINS INTRODUCTION HAUT

"V1IH"	"V2IH"	"VIH"
verin 1	verin 2	verins
introducti	introducti	introducti
on haut	on haut	on haut
E2.6	E2.7	M0.4

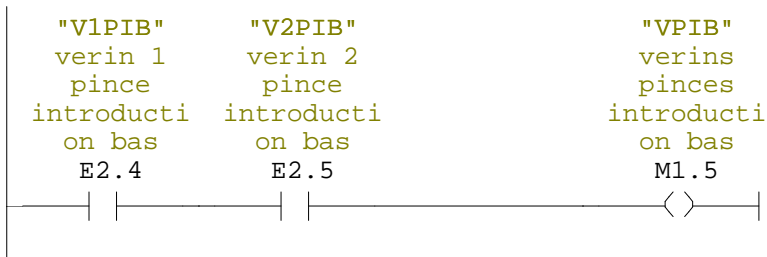
Réseau : 15 verins pinces introduction haut

VERINS PINCES INTRODUCTION HAUT

"V1PIH"	"V2PIH"	"VPIH"
verin 1	verin 2	verins
pince	pince	pinces
introducti	introducti	introducti
on haut	on haut	on haut
E2.2	E2.3	M1.4

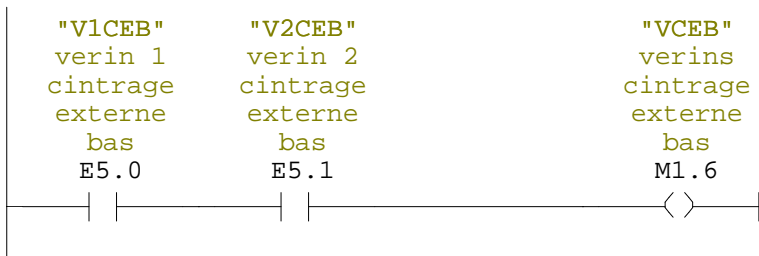
Réseau : 16 verins pinces introduction bas

VERINS PINCES INTRODUCTION BAS



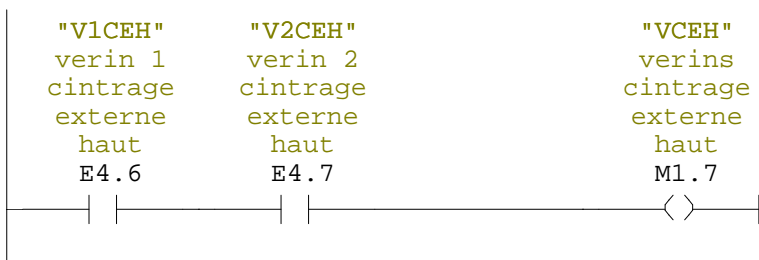
Réseau : 17 verins cintrage externe bas

VERINS CINTRAGE EXTERNE BAS



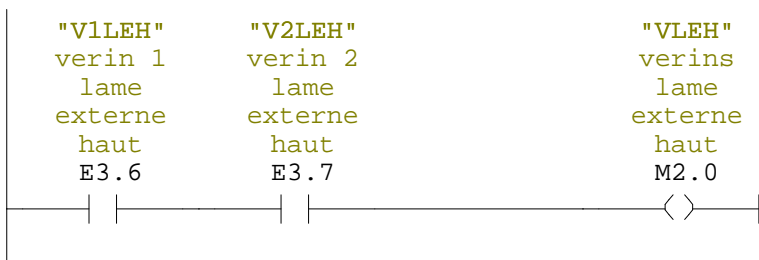
Réseau : 18 verins cintrage externe haut

VERINS CINTRAGE EXTERNE HAUT



Réseau : 19 verins lame externe haut

VERINS LAME EXTERNE HAUT (VLEH)



Réseau : 20 verins cintrage interne bas

VERINS CINTRAGE INTERNE BAS (VCIB)

"V1CIB"	"V2CIB"	"VCIB"
verin 1	verin 2	verins
cintrage	cintrage	cintrage
interne	interne	interne
bas	bas	bas
E5.4	E5.5	M2.1

Réseau : 21 verins cintrage interne haut

VERINS CINTRAGE INTERNE HAUT (VCIH)

"V1CIH"	"V2CIH"	"VCIH"
verin 1	verin 2	verins
cintrage	cintrage	cintrage
interne	interne	interne
haut	haut	haut
E5.2	E5.3	M2.2

Réseau : 22 verins tete de soudure bas

VERINS TETE DE SOUDURE BAS (VTSB)

"V1TSB"	"V2TSB"	"VTSB"
verin 1	verin 2	verins
tete de	tete de	tete de
soudure	soudure	soudure
bas	bas	bas
E6.0	E6.1	M2.3

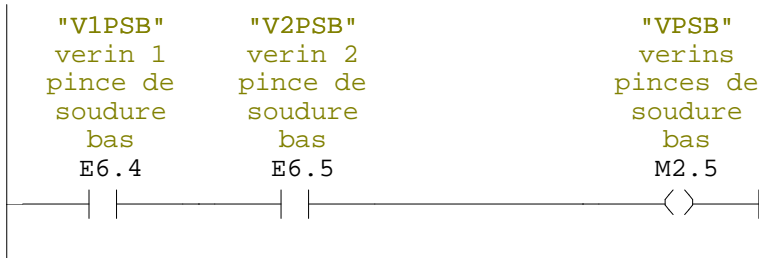
Réseau : 23 verins tete de soudure haut

VERINS TETE DE SOUDURE HAUT (VTSH)

"V1TSH"	"V2TSH"	"VTSH"
verin 1	verin 2	verins
tete de	tete de	tete de
soudure	soudure	soudure
haut	haut	haut
E5.6	E5.7	M2.4

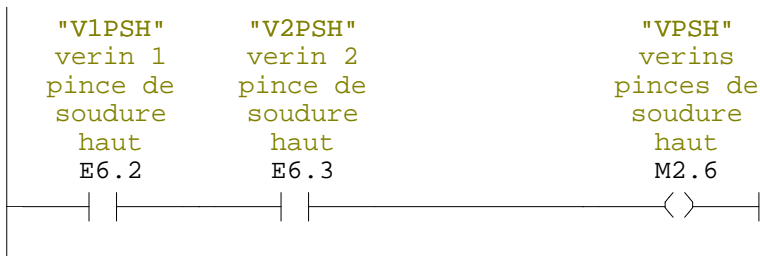
Réseau : 24 verins pinces de soudure bas

VERINS PINCES DE SOUDURE BAS (VPSB)



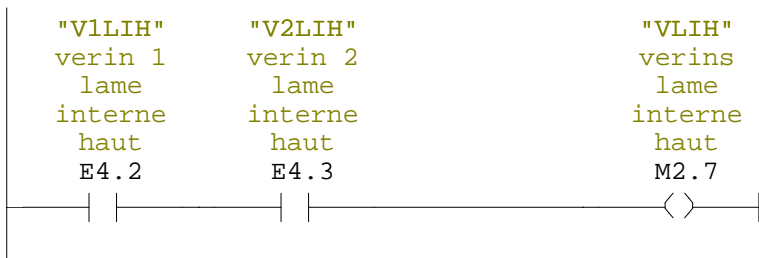
Réseau : 25 verins pinces de soudure haut

VERINS PINCES DE SOUDURE HAUT (VPSH)



Réseau : 26 verins lame interne haut

VERINS LAME INTERNE HAUT (VLIH)



Réseau : 27 compteur encodeur

COMPTEUR ENCODEUR



Réseau : 28 compteur encodeur

COMPTEUR ENCODEUR

"CEN"
capteur
encodeur
E0.6

Z1

(ZR)

Réseau : 29 temp de mise en electrisite

TEMP DE MISE EN ELECTRICITE

"VSOB"
verin
soudure
bas
E7.0

T1

(SE)

S5T#2S

Réseau : 30 temps de pressurisation

TEMPS DE PRESSURISATION

"VSOH"
verin
soudure
haut
E7.1

T2

(SE)

S5T#3S

Réseau : 31 signalisation d'arret d'urgence

signalisation d'arret d'urgence

"ar"
bouton
d'arret
d'urgence
E0.3

"SAR"
signalisat
ion
d'arret
d'urgence
A4.0

()

Réseau : 32 led tapis extraction
moteur tapis extraction

