République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Dr. Yahia FARES de Médéa



Faculté des Sciences et de la Technologie

Département de Génie de la Matière

Mémoire de fin d'étude

Présenté pour l'obtention du Diplôme

De master en Génie Mécanique

Option : Mécatronique

Thème:

Etude, Conception et Réalisation d'un prototype de profileuse en vue de la commande par API (Step-7 200) de siemens

Proposé et dirigé par :

Dr. BOUHADDA Mounir

Réalisés et présentés par :

FARES Idriss

BAL Mounir

Promotion 2011/2012

Remerciements

Nous n’aborderons pas notre travail sans remercier toutes les personnes qui ont contribué directement ou indirectement à l’élaboration de notre travail

Nous citerons d’abord nos enseignants pour le savoir qu’ils nous ont dispensé tout au long de notre cycle de formation

Aux responsables et aux personnels de notre département qui ont fait montre de disponibilité durant notre présence et qui ont été toujours à l’écoute de nos problèmes

Nous remercions particulièrement notre promoteur en l’occurrence M. Mounir BOUHEDDA qui à su nous guider dans notre tache et qui n’a à aucun moment hésiter a nous apporter son précieux concours et à nous prodiguer conseils et encouragements

A nos parents père et mère auprès de qui, nous avons trouvé tout le soutien moral et matériel notamment dans les moments les plus difficiles

Nos remerciements vont enfin à monsieur le président et les membres du jury qui auront à se pencher sur l’étude de notre travail

Dédicace

Au nom de dieu le tout puissant et le très miséricordieux par la grâce duquel j'ai pu réaliser ce travail que je dédie à :

Mes chers parents que cet humble travail représente l'accomplissement de tous vos sacrifices et le reflet de
Mon immense reconnaissance et amour.

Mon frère Fouad, Amine et Hamza

Toute la famille B A L .

Toutes mes amies :Manar, Nabil, Hossam, Hanni, Mounir...

Toute la promotion Mécatronique 2011-2012 de L'université Yahia Faress Médéa.

Et enfin ma gratitude à mon encadreur avec qui j'ai l'honneur de travailler.

A tous ceux qui ont contribué de prés ou de loin pour réaliser cette étude.

Mounir.

Sommaire

Introduction général ……

Chapitre I : Etude théorique sur les machines profileuses

I.1) Les profileuses…….. 1

I.2) Définition et Principe de profilage……1

I.3) Les capacités du profilage……3

I.3.1) De l'épaisseur ……4

I.3.2) de la largeur……4

I.3.3) de la longueur……4

I.3.4) Des nuances d'acier……5

I.4) Différents types de profileuse……..5

I.4.1) Profileuse HP (Hydraulique Presse)……5

I.4.2) Profileuse EP (Electrique Presse)……5

I.5) Equipement d'appareillage…….6

I.6) préparation de la machine……6

I.7) Matériaux utilisé pour le profilage……..7

I.8) conclusion ……..7

Chapitre II : automate programmable

II.1) les avantages de l'automatisation ……8.

II.2) Systèmes automatisés de production ……..8

II.3) Place des automates programmables ……10

II.4) structure interne d'un API ……..11

II.4.1) Matériel …….11

II.4.2) Modules d'entrées/sorties (E/S) ……12

II.4.3) Eléments de stockage et de liaison ……12

II.4.4) Processeur ……13

II.4.5) Auxiliaires ……14

II.5) Programmation et fonctionnement ……14

II.5.1) Déroulement du programme ……..15

II.5.2) Jeu d'instructions ……..16

II.5.2.1) Langage à contacts (LD Ladder Diagram)……..16

II.5.2.2) Grafcete ……17

II.6) Automate STEP 7/200…….. 16

II.6.1) Environnement matériel requis ……18

II.6.2) Options de communication ……19

II.6.2.1) Raccordement de l'alimentation à la CPU S7-200 ……20

II.6.2.2) Connexion de la CPU S7-200 ……21

II. 6.3) Composantes principales de l'automate programmable S7-200 ……21

II.6.3.1) CPU S7-200……22

II.6.3.2) Modules d'extension ……22

II.6.4) Progiciel de programmation STEP 7-Micro/WIN ……23

II.6.4.1) Mode de représentation du langage de programmation STEP 7 …23

II.6.4.2) Saisie d'un programme « CONT » ……...24

II.6.4.2.1) Création d'un projet ……24

II.6.4.2.2) Ouvrir un projet existant ……..25

II.6.4.2.3 Compilation en CONT ……25

II.6.4.2.4) Correction des erreurs à l'aide de la fenêtre ……25

II.6.4.2.5) Enregistrement du travail ……26

II.6.4.2.6) Etablissement et communication et chargement…….. 26

II.6.4.2.6.1) Sélection d'un type d'AP……..26

II.6.4.2.6.2) Lecture du type d'AP éloigné par STEP 7-Micro/WIN 32

II.6.4.2.6.3) Test du réseau de communication.……27

II.6.4.2.6.4) Chargement d'un programme dans la CPU ……27

Chapitre III : Conception et réalisation du prototype

III.1) Partie mécanique ……29

III.1.1) Sous système (1) dérouleur……..30

III.1.2) Sous-système (2) station de perforation ……31

III.1.3) Sous-système (3) défileuse ……34

III.1.4) Sous-système (4) station de découpage …….35

III.2) Partie électrique …….35

III.2.1) contacteur (solénoïde) …….36

III.2.1.1) Solénoïde ……..36

III.2.1.2) Définition …….36

III.2.1.3) Constituant solénoïde ……37

III.2.2) Codeur optique …….37

III.2.2.1) Généralité sur les codeurs optiques……..37

III.2.2.2) Définition d'un codeur optique ……38

III.2.2.3) Différent types de codeurs ……38

III.2.2.3.1) Codeur incrémental (ou générateur d'impulsions)

III.2.2.3.1.1) RESOLUTION (Nb de points par tour……39

III.2.2.3.2) Codeur « absolu »……39

III.2.2.3.2.1) Exemple d'un codeur absolu 3 bits (avec le code binaire

III.2.2.3.2.2) Choix du codeur……42

III.2.2.4) Codeur optique utilisé sur le prototype ……..43

III.2.3) moteur à courant continue……44

III.2.3.1) Principe de fonctionnement ……44

III.2.3.2) schéma équivalent du moteur c c……..44

III.3) les relais ……46.
III.4) Les transistors ……46
III.4.1) Principe de fonctionnement ……47
III.4.2) analyse fonctionnelle du prototype de profileuse ……47
III.5) Parte automatism……48
III.5.1) Les opération utilisée dans notre programme ……49
III.5.1.1) Opération de temporisation T37 …….49
III.5.1.2) Operation set (S) ET reset (R)……50
III.5.1.3) OPERATION M0.2 ……50
III.5.1.4) Opération CTU …….51
III.5.1.5) Opération CEI(=)……..51
III.5.1.6) Operation CEI-(AND,OR) ……..48
III.5.1.7) Tableau de mnémonique ……52
III.5.1.8) Description du programme …….54
Conclusion générale……..
Référence

Liste des figures :

Figure I.1: Fleur de profilage……2

Figure I.2: commande d'un moteur par automate……..3

Figure I.3 : défirent formes réaliser par une profileuse……..4

Figure I.4 : défirent sous-système existant sur une profileuse……6

Figure II.1 : fonctionnement d'un SAP……9

Figure II.2 : structure interne d'un API……..11

Figure II.3 : Déroulement du programme. **……16**

Figure II.4 : exemple d'un langage Ldder……..17

Figure II.5: symboles et fonction des instructions du Grafcet……18

Figure II.6: Automate programmable S7-200…….19

Figure II.7: alimentation d' S7-200…….20

Figure II.8: connexion avec un câble RS-232/PPI multi-maitre……..21

Figure II.9: Unité centrale avec module d'extension…….23

Figure II.10: Système cible…….24

Figure II.11: Sélection d'un type d'AP. ……..25

Figure II.12: emplacement de l'icône de communication……26

Figure II.13: boite de dialogue « communication »……..27

Figure II.14: chargement d'un programme dans la CPU……28

Figure III.1 : Prototype de la profileuse et ces sous-systèmes……29

Figure III.2 : dérouleur…….31

Figure III.3 : collier de fixation……31

Figure III.4 : support du détecteur…….32

Figure III.5 : sous-système n°2…….33

Figure III.6 : sous-système n°3……34

Figure III.7 : sous système n°4…….35

Figure III.8 : contacteur……36

Figure III.9 : composants d'un contacteur…….37

Figure III.10 : codeur optique……39

Figure III.11 : zone opaque et transparent du disque de codeur incrémental……..39

Figure III.12 : signaux carrés délivrer par le codeur…….40

Figure III.13 : disque d'un codeur absolu……41

Figure III.14 : exemple d'un codeur absolu……41

Figure III.15 : exemple de calcul…….43

Figure III.16 : Codeur optique utilisé sur le prototype…….44

Figure III.17 : les composants d'un moteur à c.c……..46

Figure III.18 : schéma équivalent d'un moteur à CC……46

Figure III.19 : alimentation d'un moteur à C.C……47

Figure III.20 : explication de fonctionnement des deux sens de rotation……...47

Figure III.21 : schéma explique les pertes de puissance…….48

Figure III.22 : la réversibilité des moteurs……50

Figure III.23 : schéma explicatif du freinage d'un moteur à C.C…….51

Figure III.24 : schéma d'un relais…….52

Figure III.25 : photo d'un transistor……..52

Liste des tableaux :

Tableau II.1 : symboles et fonction des instructions du langage Ladder.

Tableau III.1 : Exemple d’un codeur absolu 3 bits (avec le code binaire).

Les abréviations :

H P : Hydrolique Presse.

E P : Electrique Presse.

A P I : Automate Programmable Industriele.

PLC : Programmable Logique Controller.

SAP : Système Automatiser de Production.

TOR : Tout Ou Rien.

ROM : Read Only Memory.

EPROM : Efassable Programme Read Only Memory.

RAM : Random Acces Memory.

E/S : Entrée/ Sortie.

PC : programme counter.

UC : Unité Centrale.

LD : Ladder Diagramme.

FBD : Function Block Diagramme.

SFC : SEcquential Function Chart.

IL : Instruction List.

ST : Struction Text.

NF : Normalement Fermée.

NO : Normalement Ouvert.

CP : Processus de Communication.

CC : Courant Continue.

MSB Most Sinificant Bit.

LSB : Least .Significant Bit.

الملـــخـــــــص

إن الهدف الرئيسي لهذه الدراسة هو إنشاء نموذج profileuse بهدف إعطاء رؤية حول مستقبل هذه التكنولوجيا التي تعتمد على البرمجة والآليات لتسيرها مثل STP7-200 وهو عبارة عن آلية تحكم وتسير الآلة بفضل برنامج مدروس لتحصل على النتيجة المطلوبة وتحكم بها حسب متطلبات السوق مثلا الثقب مستمر أو الثقب متناوب.

Résumé

Le but de ce travail est de traiter l'aspect conception et réalisation d'un prototype d'une nouvelle machine dénommée profileuse en vue de la commende

Le principe de fonctionnement de cette machine repose sur le profilage, qui est un procédé de pliage par étapes successives au moyen d'une profileuse.

La profileuse adopte, à la fois pour son fonctionnement, la technologie de la conception mécanique et celle de l'automatisme qui est basé sur la programmation numérique.

Les résultats obtenus démontrent que le procédé du prototype, sujet de notre étude, pourrait s'appliquer à une perforation continue et perforation alternative

Ce projet a également pour objectif d'ouvrir des voies innovantes de développement de procédés applicables à une profileuse.

ABSTRACT

The ain of this work is the design and realisation of roll forming prototype

The principle of operation of this machine is based on the profiling, which is a folding prcess in successive stages by means of a profiling machine.

This profiling machine adopts, for its operation, both the technology of mechanical design and automation, this latter is based on digital programming stp7-200.

The obtained results shom that the method of prototype, subject of our study ,could be applied to a continuous perforation and an alternative perforation.

This project also aims to open lines of innovative development processes applicable to a profiling machine.

Chapitre I

Etude théorique sur les machines profileuses

I) Etude théorique sur les machines profileuses :

Introduction :

Le but de notre projet est de proposer une commande automatique de la profileuse. Pour cela une étude détaillée s'avère importante afin de pouvoir comprendre le fonctionnement détaillée de chaque sous-système, ces détaille nous permettrons de concevoir un prototype de profileuse.

I.1) Les profileuses :

Ces machines ligne(s) de profilage servent à profiler les tôles (feuillards), feuillard acier, feuillard aluminium, feuillard inox, donc à réaliser les divers types de profils, a plier le feuillard qui passe dans les diverses têtes de la machine à profiler (ligne de profilage). Pour obtenir la forme du profil souhaité.

La machine se compose d'un dérouleur manuelle ou bien automatisé, d'une table d'entrée pour le feuillard de la profileuse, des têtes de profilage (axe supérieur, axe inférieur, support des galets), du dévrilleur, boite de sortie, table de récupération du profil, de l'armoire électrique, commande motorisation, l'avance de la ligne de profilage (vitesse de 20 à 120 mètres par minute) est suivant les modèles des lignes de profilage et de leur utilisation.

I.2) Définition et Principe de profilage :

Le profilage est un procédé de pliage par étapes successives au moyen d'une profileuse. A chaque étape de pliage correspond une station de profilage équipée d'un jeu de galets dans les quels la tôle pénètre pour y être mise en forme. Les produits profilés sont réalisés à partir de tôles planes en feuille ou en bobine, de largeur constante. Le profilage permet la réalisation de pièces dont les sections sont très variées (forme en U, C, Z, L, W, glissières de siège, pare-chocs automobiles, supports de faux plafonds, luminaires,...). Le profilage est un procédé utilisé par tous les secteurs de l'industrie (automobile, électroménager, bâtiment, ...). Ces profilés peuvent être divisés en trois grandes familles:

- les profilés larges: il s'agit de produits de grande largeur (600 à 1500 mm environ) comportant plusieurs ondes de forme équivalente ou

similaire. On rencontre notamment dans cette catégorie les profilés de couverture.

- les tubes: il s'agit de produits à section fermé réalisés par profilage avec une soudure en continue.
- les profilés étroits: il s'agit de produits de section ouverte ou fermée sans soudure ne comportant pas de répétition d'une forme d'onde. Cette catégorie comprend, entre autres, les profilés de structure, les chemins de câbles, les glissières de sécurité.

Le profilage peut être considéré comme un pliage en continu. Au cours du processus, la tôle est pliée plastiquement et progressivement par chacune des têtes, jusqu'à obtenir les angles et la forme souhaités, (figure I.1).

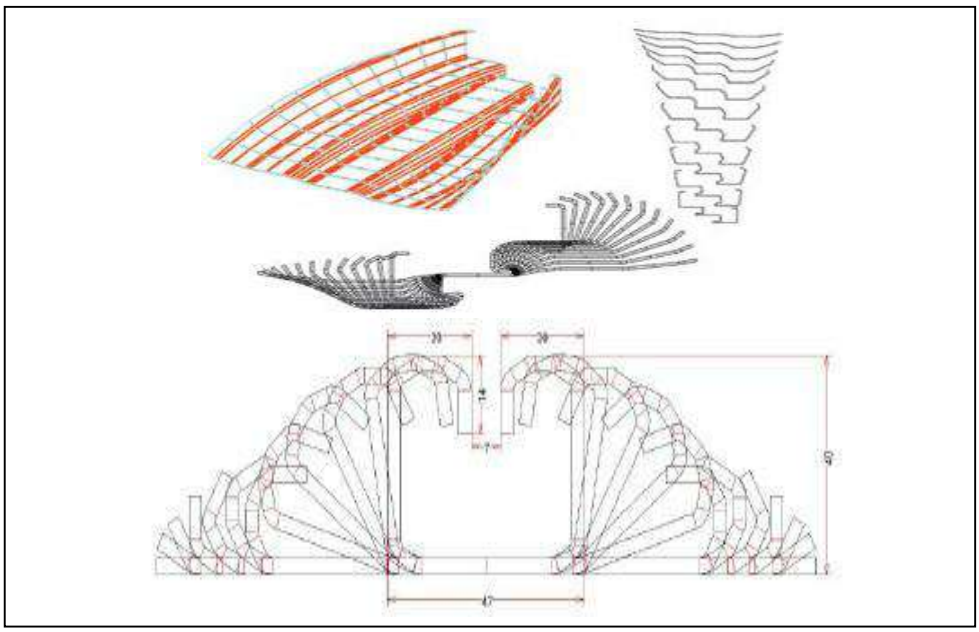


Figure I.1: Fleur de profilage

La bande de métal est entraînée entre des pièces appelées (galets) inférieur et supérieur, réalisés en acier trempé ou prétraité. La tôle va passer entre plusieurs têtes de profilage qui va former progressivement la tôle, jusqu'à la forme finale. Cette machine, possède diverses têtes de profilage (galets supérieur-galets inférieur) de 6 à 30 têtes environ suivant sa capacité. Elle est équipée de plusieurs appareillages, dérouleur, défileuse, outil de coupe, outil de poinçonnage, d'une armoire à commande numérique...

Cette armoire est équipée par des équipements électrique nécessaire au fonctionnement de la machine, elle peut même être équipée par un automate, qui va donner les ordres nécessaire aux moteurs qui compose la profileuse selon un programme bien définie comme montré à la (figure I.2).

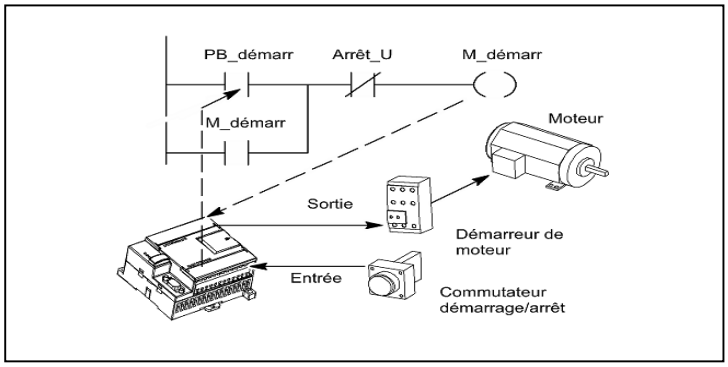


Figure I.2: commande d'un moteur par automate

- Les principaux avantages de ce type de procédés sont la faible perte de matière, la réalisation de formes complexes et les bonnes propriétés mécaniques, ce procédé est rapide et productif.
- Les inconvénients sont liées à la prise en compte du retour élastique lors de l'étape de mise en forme mais également la prédiction de l'apparition de défauts tels que les plis, la striction, la rupture, le coût d'outillages et investissements en capital élevés, n'est économique que pour des niveaux de production très importants.

I.3) Les capacités du profilage :

Le profilage est un processus très souple. Ceci se retrouve dans ses capacités de production, dans les formes réalisabables (figure I.3), même si elles doivent rester de section constante dans le sens de la longueur.

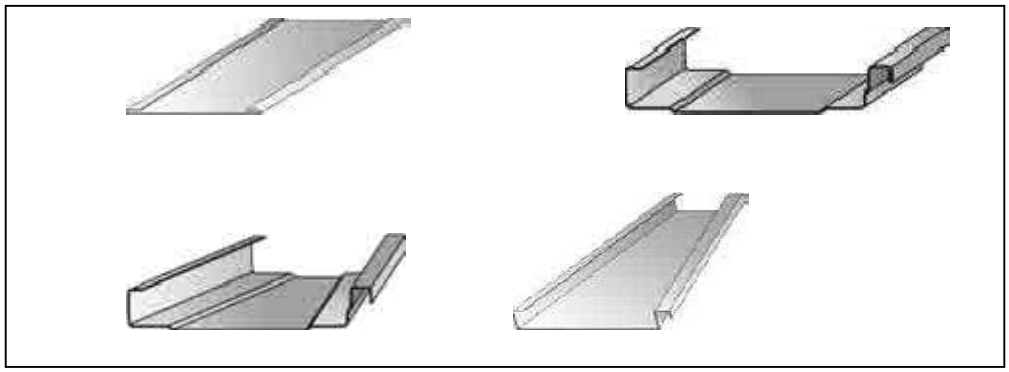


Figure I.3 : défirent formes réaliser par une profileuse

I.3.1) De l'épaisseur : elle peut varier, elle peut atteindre 12 mm.

I.3.2) de la largeur : il n'y a pas de valeur minimale du fait des capacités de refendasge des bobines. Pour la valeur maximale, la borne est en réalité due au format réalisable des bobines.

I.3.3) de la longueur : il n'existe pas de maximum pour la longueur des profilés si ce n'est celui qui existe pour la manutention, le transport, ect… des produits.

I.3.4) Des nuances d'acier : les nuances très ductiles avec une faible limite d'élasticité (aux environs de 150 MPa) sont utilisables ainsi que celles très dures avec limites d'élasticité de 1000 MPa et plus. Avec une technologie adaptée, il est tout à fait possible de réaliser des profilés de qualité dans toutes les nuances existantes.

I.4) Différents types de profileuses :

Il existe plusieurs types de profileuse selon leur fonction, marque, type, nombre de galets qu'elle peut porter, épaisseur maximal de la tôle à traiter et on peut citer quelques profileuses qu'on trouve dans les ateliers de travail.

I.4.1) Profileuse HP (Hydraulique Presse) : elle est utilisée pour la fabrication d'un rabat creux dans les rainures, de pliures pour l'eau, des avant-toits etc... A partir d'une tôle coupée.

I.4.2) Profileuse EP (Electrique Presse) : Machine universelle, elle est parfaite pour les profilés courts, coniques ou extrémité en pointe. Par le profilage unilatéral, les profilés sont réalisés rapidement en deux étapes de travail et sans régler la largeur. D'un poids léger et en utilisant des poignées de transport, elle est facile à déplacer et sont utilisation est facile.

I.5) Equipement d'appareillage :

Une profileuse est composée de plusieurs sous-système :

Dérouleurs : Ils servent à emmener la tôle (feuillard) dans la première tête de la profileuse pour la réalisation des profils dans la ligne de profilage. Il est utilisé pour des charges légères ou bien pour des charges lourdes, avec motorisation. Suivant le poids de la bobine de feuillard. Ces rouleaux de feuillards peuvent être en acier, aluminium, inox, titane, cuivre, laiton, ect...

Outils de perforations et de découpage : Les outils de coupes sont utilisés en complément sur la profileuse, ils servent à faire des découpes diverses avec des formes diverses. Soit à l'entré du feuillard ou sortie de la profileuse soit à l'arrêt ou bien en marche. Ils servent aussi à couper les bandes à l'entrée de la profileuse ou bien à la sortie, à la longueur désirée, lors du passage de la bande de feuillard.

La figure (I.4) montre les différent sous-systèmes qui compose la profileuse en effet elle peut être devisé en quatre sous-système. Cette décomposition nous permet de voir les choses en claire de point de vue commande, qui est l'un des buts essentiels de notre travail.

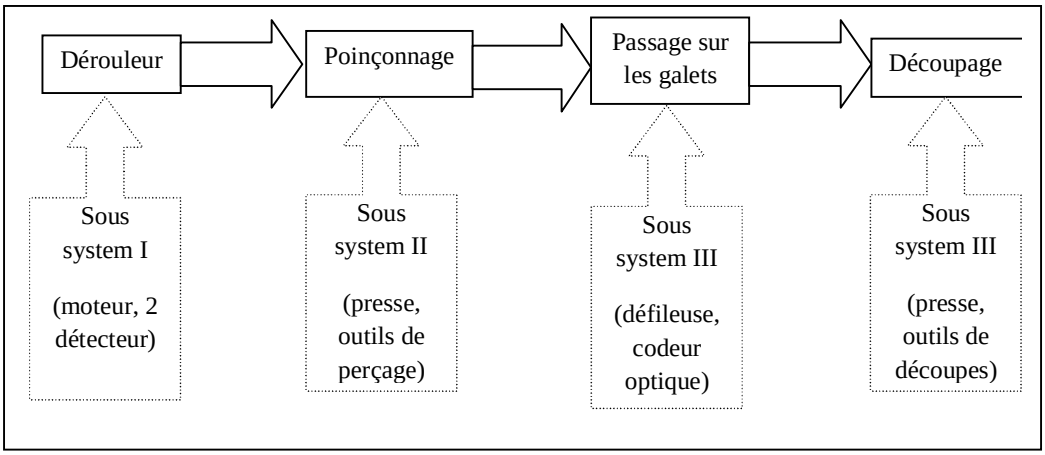


Figure I.4 : déffirent sous-système existant sur une profileuse.

I.6) préparation de la machine :

La mise en marche de la machine:

- Touts d'abord il faut que la bobine soit montée sur le dérouleur, on dessert le mandrin jusqu'à ce qu'il puisse recevoir largement le diamètre intérieur de la bobine, puis on introduit la bobine en la laissant suspendue puis serrer, les mors, en tournant la manivelle dans le sens des aiguilles d'une montre.
- Introduire le bout du feuillard entre les axes de mise en hauteur puis dans le guide bande jusqu'à attendre la première station de profilage, puis avec la commande manuelle introduire le feuillard entre les différents galets. Guider ensuite le profilé dans la boite à galets.
- Il faut s'assurer que le jeu entre les galets soit égal à l'épaisseur des feuillards, et ce en introduisant une cale d'épaisseur égale à celle du feuillard, entre les galets. Si le jeu entre les galets est trop petit, on fait tourner les noix au dessus des arcades dans le sens horaire.
- On introduit la feuille dans l'étau du banc de coupe.
- Apres vient l'étape de réglage du profile, il ya trois réglage à faire :
  - Au niveau de la table d'entrée.
  - Au niveau de la boite à galets.
  - Au niveau de la butée de longueur.

I.7) Matériaux utilisé pour le profilage : elles peuvent être constituées: Acier - Acier galvanisé - Acier prés laqué - Acier inox - Cuivre - Laiton - Aluminium – Titanium.

I.8) conclusion :

Dans ce chapitre nous avons essayé de donner une description générale des profileuses, montrer ses composant, ses sous-système, et son mode de fonctionnement, cette description est faite pour pourvoir appliquer la commande.

Chapitre II

Automate programmable

Introduction : L'automate programmable industriel API (ou Programmable Logique Controller PLC) est aujourd'hui le constituant le plus répandu des automatismes. On le trouve non seulement dans tous les secteurs de l'industrie, mais aussi dans les services (gestion de parkings, d'accès à des bâtiments) et dans l'agriculture (composition et délivrance de rations alimentaires dans les élevages). Il répond aux besoins d'adaptation et de flexibilité de nombres d'activités économiques actuelles.

Pour cella on va mètre en évidence :

- ses caractéristiques propres, matérielles et logicielles.
- sa capacité à s'intégrer dans un ensemble plus large, et donc à répondre aux besoins d'un système automatisé de production SAP.

II.1) les avantages de l'automatisation :

L'évolution des techniques s'est traduite pour l'automatisation, car cette solution à de nombreux avantages comme :

- Améliorer la qualité du produit.
- Adaptation à des contextes particuliers, tel que les environnements hostiles pour l'homme (milieu toxique, dangereux, nucléaire...), adaptation à des tâches physiques ou intellectuelles pénibles pour l'homme (manipulation de lourdes charges, tâches répétitives, parallélisées...).
- Augmenter la sécurité.
- Accroître la productivité du système.
- une approche de plus en plus globale des problèmes.
- une intégration dès la conception de l'installation.

On est ainsi passé du stade de la machine automatisée à celui du système automatisé de production

II.2) Systèmes automatisés de production :

Un système automatisé de production SAP (notion assez large qui inclut des systèmes de contrôle, de conditionnement, d'analyse), reçoit un flux de matière ou de produits et 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. Tout cela, entraine 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. La solution reposant sur un processeur central unique s'étant vite révélée peu économique du point de vue câbles, complexe quant à la maintenance voire dangereuse en cas d'incident, l'utilisation de processeurs spécialisés et interconnectés s'est aujourd'hui largement imposée.

L'architecture décentralisée qui en résulte facilite la conception et l'installation en permettant de fractionner les études, la mise en place, les tests, permet d'amélioré aussi la maintenance (modification aisée des programmes, de parties du système automatisé) et se traduit par plus de flexibilité et de disponibilité. Elle entraine toutefois, des multiples sous-ensembles fonctionnels, un fort accroissement des besoins de communication et de gestion. La (figure II.1) illustre le fonctionnement d'un SAP, tel que l'on rencontre notamment, mais pas exclusivement, dans l'industrie manufacturière (chaines d'usinage, de montage, de conditionnement).

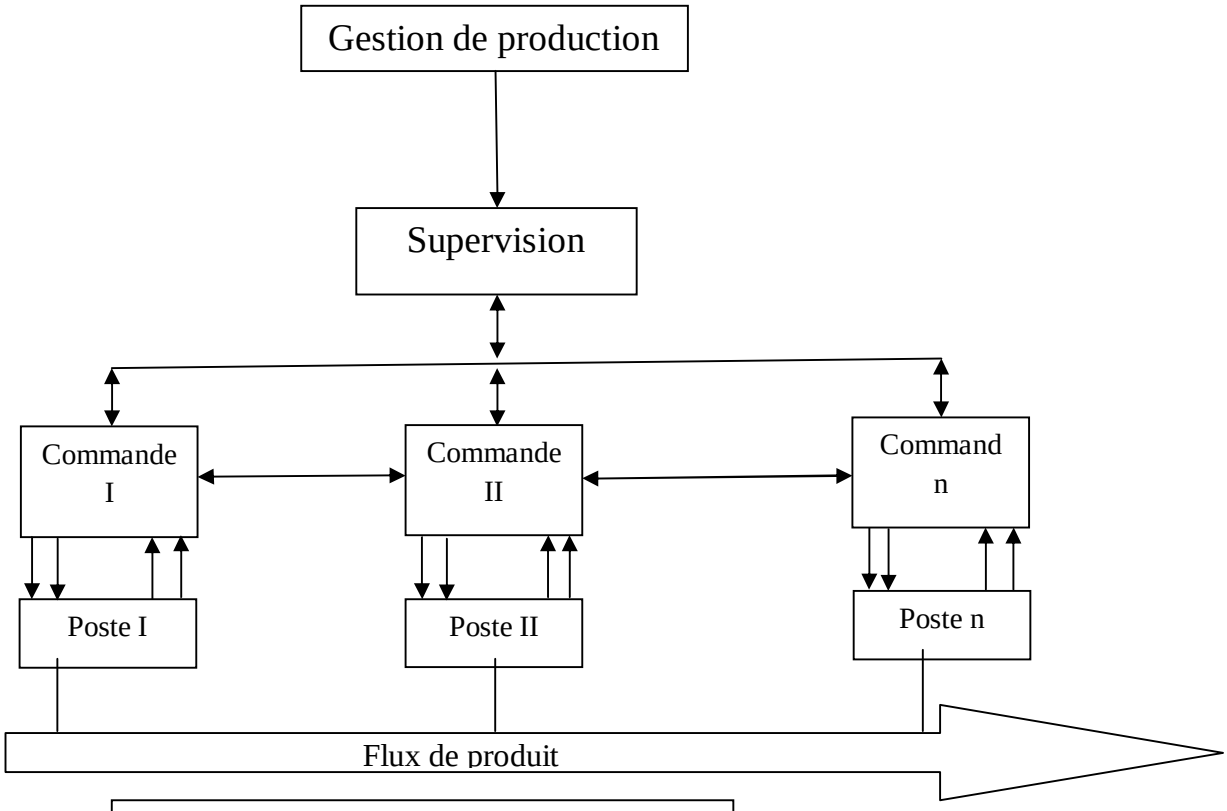


Figure II.1 : fonctionnement d'un SAP

II.3) Place des automates programmables :

Dans ces systèmes de traitement de l'information, les API occupent une place de choix. Les équipements notés commande sont souvent des automates.

Remplaçant initialement des ensembles en technologie câblée (relais électromagnétiques ou statiques, composants pneumatiques), ils constituent de plus en plus un maillon fiable et efficace entre le calculateur, qui a plutôt un rôle de gestion et l'appareillage de terrain (capteurs et actionneurs). Cet appareillage pouvant lui-même aujourd'hui contenir un processeur, il nous faut préciser la définition de l'API, tout en sachant que dans ce domaine comme dans beaucoup d'autres touchant à l'informatique, les frontières sont floues.

Nous considérerons comme automate programmable un système :

- construit autour d'un processeur numérique, spécifique ou non.
- pouvant être relié à de nombreux signaux physiques.
- fonctionnant grâce à une protection adaptée dans des conditions industrielles.
- doté d'un logiciel de programmation permettant un traitement simple des variables booléennes (Tout ou Rien, **TOR** en abrégé).
- doté de possibilités d'échanges avec d'autres processeurs.

Cela constitue un noyau susceptible a de nombreuses extensions et compatible avec la définition assez lourde. A partir de ces définitions, nous distinguerons dans les fonctions que l'automate doit remplir :

- un rôle de commande, ou il est un composant d'automatisme, élaborant des actions, suivant une algorithmique appropriée, à partir des informations que lui fournissent des détecteurs (Tout ou Rien) ou des capteurs (analogiques ou numériques).
- un rôle de communication dans le cadre de la production :

1. Avec des opérateurs humains : c'est le dialogue d'exploitation.

2. Avec d'autres processeurs, hiérarchiquement supérieurs (calculateur de gestion de production), égaux (autres automates intervenant dans la même chaine) ou inférieurs (instrumentation intelligente).

II.4) structure interne d'un API :

Ce rôle constitue sa base historique et sa spécificité par rapport à d'autres matériels. Pour étudier cet équipement connecté à des systèmes réels en milieu industriel, il nous faut passer en revue l'aspect matériel, l'aspect logiciel et devant les conditions d'emploi, nous posons la question de la sureté de fonctionnement.

II.4.1) Matériel :

La structure matérielle interne d'un API obéit au schéma donné sur la (figureII.2).

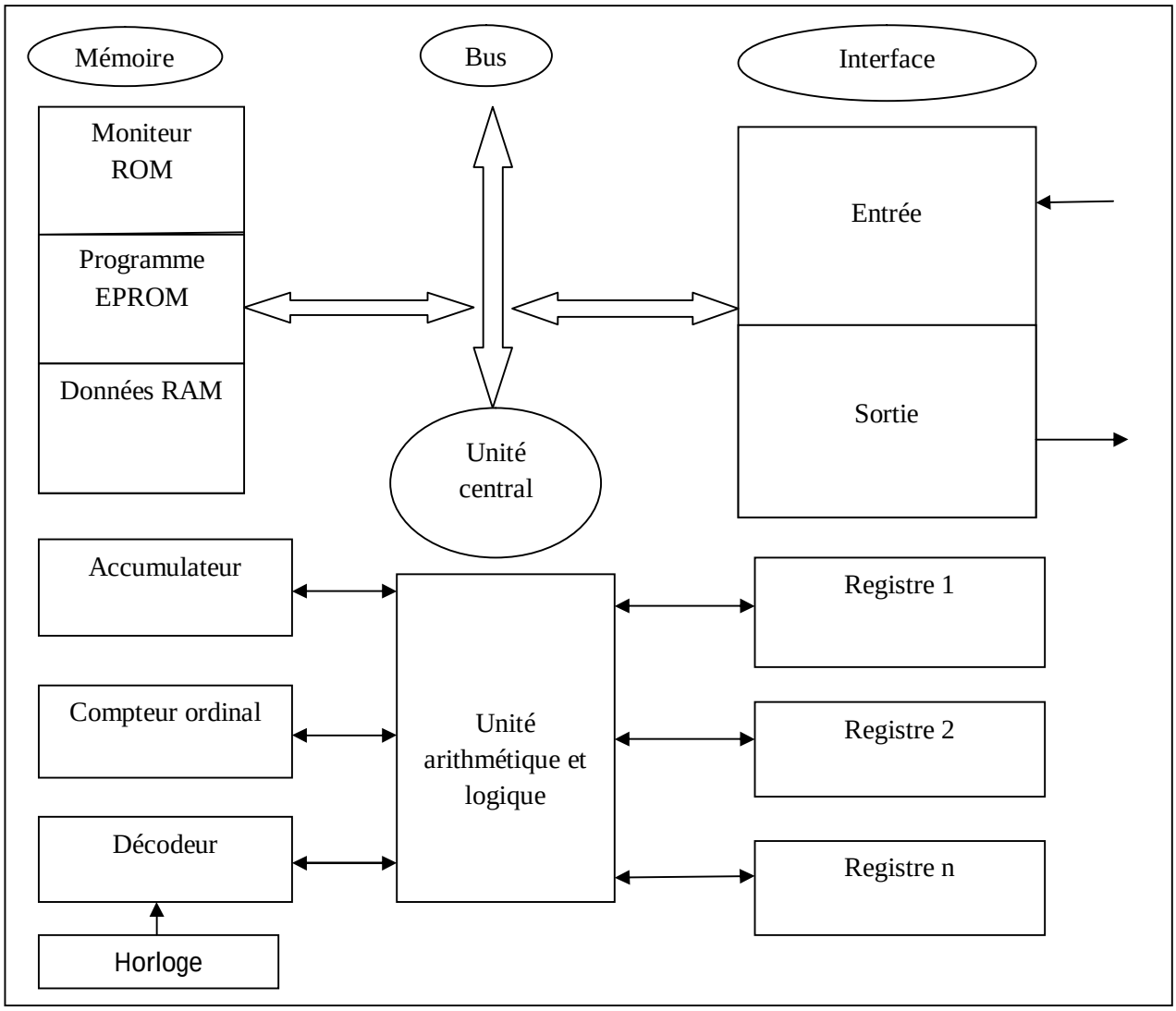


Figure II.2 : structure interne d'un API

II.4.2) Modules d'entrées/sorties (E/S) :

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

Pour ce faire, ils doivent :

- regrouper des variables de même nature, pour diminuer complexité et cout.
- assurer le dialogue avec l'UC.
- traduire les signaux industriels en informations.

Beaucoup d'automates assurent cet interfaçage par des modules amovibles comportant un nombre fixé de voies d'un certain type du point de vue nature informationnelle, mais aussi électrique, et signal.

Le nombre total de modules est évidemment limité, pour des problèmes physiques (taille du châssis et/ou de l'alimentation électrique) ou de gestion informatique. La possibilité de configurer des voies d'accès en entrée ou en sortie est rarement utilisée, pour des raisons de sécurité.

D'autres automates, tels le TSX17 de Schneider Electric, ont une structure figée monobloc, avec des modules intégrés dans le châssis de base. Le nombre de voies de ces modules reflète alors le rapport moyen entre entrées et sorties dans les problèmes industriels, qui se traduit par un déséquilibre au profit des entrées.

II.4.3) Eléments de stockage et de liaison :

Le stockage des données et des programmes s'effectue dans des mémoires. La mémoire vive (RAM) est volatile mais secourue par batterie. La mémoire morte (ROM), dont l'utilisateur ne peut que lire le contenu, éventuellement programmable (PROM) à l'aide d'outils spéciaux, contient le système d'exploitation, tandis que les programmes au point et utilisables peuvent se stocker dans des mémoires reprogrammables (EPROM), là encore avec un matériel spécifique.

La mémoire de données contient elle-même plusieurs zones :

- une zone de bits, dont certains peuvent être secourus en cas de défaillance de l'alimentation électrique.
- une zone de mots, permettant de soumettre des données à un traitement plus large que le traitement booléen (traitement numérique ou alphanumérique).

II.4.4) Processeur :

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

Les premiers API étaient équipées de processeurs spécifiques, à cycle de **scrutation** unique : on exécutait en permanence un programme gérant essentiellement des variables binaires. On est passé ensuite à des processeurs plus performants, issus du monde de l'informatique. Cette évolution a permis de baisser les couts, d'accroitre les possibilités opérationnelles. Les processeurs généralistes étant toutefois ponctuellement moins efficaces que les spécialisés.

L'unité centrale UC est une carte électronique bâtie autour de la (ou des) puce(s) processeur(s), qui assure au moins les fonctions suivantes :

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

Il existe trois technologies de réalisation :

1. la technologie câblée, la plus rapide mais aussi la plus couteuse, réservée à des usages particuliers.
2. la technologie à microprocesseur, la plus économique dès lors que l'on utilise un microprocesseur standard produit en grande série.
3. la technologie mixte, certaines opérations étant réalisées en câblé pour en accroitre la rapidité.

Pour assurer la liaison entre l'UC et les cartes d'entrées/sorties, un réseau, certaines consoles ou unités de dialogue, il faut une carte électronique spécialisée d'interfaçage : le coupleur. Il existe au moins un coupleur de base, éventuellement des coupleurs vers d'autres châssis contenant des entrées/sorties supplémentaires.

II.4.5) Auxiliaires :

Il s'agit principalement :

- de l'alimentation électrique qui a pour rôle de fournir les tensions continues que nécessitent les composants (5V, 12V,….) avec de bonnes performances, notamment face aux microcoupures du réseau. Sa source d'énergie est normalement le réseau électrique parfois du 24V continu. Il ne faut pas oublier que les châssis d'extension et les entrées/sorties déportées doivent aussi disposer d'une alimentation. Il est parfois nécessaire, pour lutter contre les perturbations électriques, d'introduire un transformateur d'isolement. C'est notamment le cas pour les raccordements à un réseau électrique à neutre flottant. Un onduleur évite les risques de coupure si celles-ci risquent de dépasser les tolérances admises.
- D'un ventilateur, indispensable dans les châssis comportant de nombreux modules ou dans le cas ou la température ambiante est susceptible de devenir assez élevée (plus de 40°C).
- du support mécanique il peut s'agir d'un **rack** (structure métallique accueillant des cartes avec généralement un raccordement arrière), l'automate se présentant alors sous forme d'un ensemble de cartes, d'une armoire, d'une grille et des fixations correspondantes.
- D'indicateurs d'état concernant la présence de tension, l'exécution du programme (mode RUN), la charge de la batterie, le bon fonctionnement du coupleur…

II.5) Programmation et fonctionnement :

C'est un des atouts majeurs des API puisqu'elle permet une multitude de traitements des informations reçues sans toucher à la configuration matérielle. Certaines modifications peuvent même s'effectuer alors que l'automate est en marche. Il faut toutefois comprendre le fonctionnement du processeur, qui impose certaines

contraintes et choisir le langage le plus approprié dans le cadre du problème à résoudre.

L'automate programmable reçoit les informations relatives à l'état du système et puis commande les pré-actionneurs suivant le programme inscrit dans sa mémoire.

Généralement les automates programmables industriels ont un fonctionnement cyclique. Le microprocesseur réalise toutes les fonctions logiques ET, OU, les fonctions de temporisation, de comptage, de calcul... Il est connecté aux autres éléments (mémoire et interface E/S) par des liaisons parallèles appelées « **BUS** » qui véhiculent les informations sous forme binaire. Lorsque le fonctionnement est dit synchrone par rapport aux entrées et aux sorties, le cycle de traitement commence par la prise en compte des entrées qui sont figées en mémoire pour tout le cycle.

Le processeur exécute alors le programme instruction par instruction en rangeant à chaque fois les résultats en mémoire. En fin de cycle les sorties sont affectées d'un état binaire, par mise en communication avec les mémoires correspondantes. Dans ce cas, le temps de réponse à une variation d'état d'une entrée peut être compris entre un ou deux temps de cycle (durée moyenne d'un temps de cycle est de 5 à 15 ms)

II.5.1) Déroulement du programme :

Il doit assurer en permanence un cycle opératoire qui comporte trois types de taches (figureII.3) :

- L'acquisition de la valeur des entrées (lecture).
- le traitement des données.
- L'affectation de la valeur des sorties (écriture).

Le premier et le troisième point s'effectuent par mise en communication successive de chacune des voies d'entrée ou de sortie (scrutation).

L'exécution des instructions se fait dans l'ordre de leur rangement dans la mémoire programme, l'incrémentation d'un compteur ordinal contenant initialement l'adresse du début de mémoire permet le passage à l'instruction suivante, sauf si un saut avant, permet de se positionner plus loin, à une adresse plus élevée, en n'exécutant pas un ensemble d'instructions.

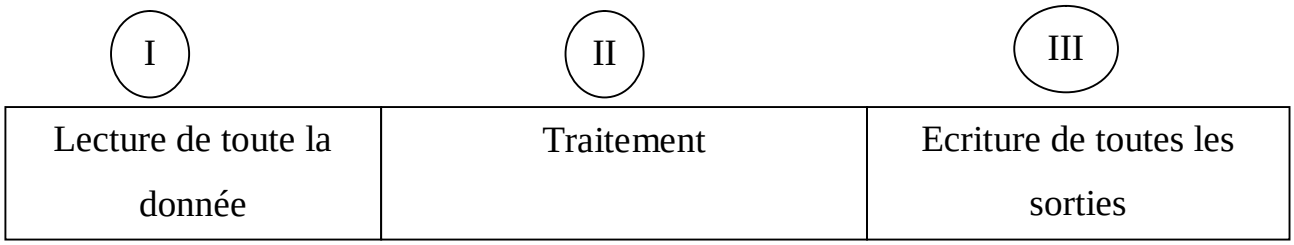


Figure II.3 : Déroulement du programme

II.5.2) Jeu d'instructions :

Le processeur peut exécuter un certain nombre d'opérations logiques, l'ensemble des instructions booléennes des instructions complémentaires de gestion de programme (saut, mémorisation, adressage ...) constitue un jeu d'instructions.

Chaque automate possède ses propres jeux d'instructions. Mais, les constructeurs proposent tous une interface logicielle de programmation. Cette norme définit cinq langages de programmation utilisables, qui sont :

Les langages graphiques :

- LD : Ladder Diagramme (Diagrammes échelles)
- FBD : Function Block Diagram (Logigrammes)
- SFC : Sequential Function Chart (Grafcet)

Les langages textuels :

- IL : Instruction List (Liste d'instructions).
- ST: Structured Text (Text structure).

II.5.2.1) Langage à contacts (LD Ladder Diagram):

Le langage à relais (Ladder Diagramme) est basé sur un symbolisme très proche de celui utilisé pour les schémas de câblage classiques. Les symboles les plus utilisés sont donnés au (Tableau II.1).

Il utilise une forme graphique (figureII.4), d'où son appellation de schéma à contacts (il est encore appelé diagramme à relais).

L'application est représentée par un ou plusieurs réseaux. Un réseau est formé d'éléments graphiques, et éventuellement de blocs fonctionnels, connectés entre eux,

partant d'une barre d'alimentation à gauche, et se terminant par une barre à droite (**facultative**). Il correspond au cheminement d'un flux :

- D'énergie (cas d'une application ou le programme pourrait être remplacé par un ensemble de relais électromécaniques, avec passage éventuel de courant électrique), il circule alors conventionnellement de gauche à droite.
- De signaux (cas des blocs fonctionnels, ce qui permet d'introduire des éléments tels que temporisateurs, compteurs, opérateurs arithmétiques).

Dans le cas de traitement **booléen**, les éléments fondamentaux sont des contacts normalement fermés (NF) ou normalement ouverts (NO), et des bobines, avec possibilité de mémorisation de l'information reçue.

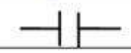
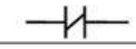
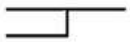
Fonction	**Symbole**	
	Européen	Américain
Contact ouvert au ropos	---o o---	
Contact fermé au ropos	---o o---	
Début de branchement		
Fin de branchement		
Affectation	---( )---	---( )

Tableau II.1 : symboles et fonction des instructions du langage ladder



Figure II.4 : exemple d'un langage Ldder.

Cela traduit le fait que la sortie repérée (B) doit être mise à 1 si, et seulement si, l'entrée (NO) est à 1 (le contact NO est alors fermé) et l'entrée (NF) à 0.

II.5.2.2) Grafcete :

Le GRAFCET est très fonctionnel, il comporte un nombre restreint de symboles conventionnels, (figure II.5) et permet la correction d'une partie de séquence facilitant ainsi les modifications. De plus ce système étant conventionnel à l'avantage d'être facilement interprété par n'importe qu'elle personne le connaissant.

On notera, en ce qui concerne l'aspect graphique, que le réseau ci-dessus a été tracé en se servant exclusivement des symboles offerts par les touches d'un clavier normal de PC. Cela est volontaire et confirme la facilité de programmation à partir d'outils simples qui est un des atouts des API.

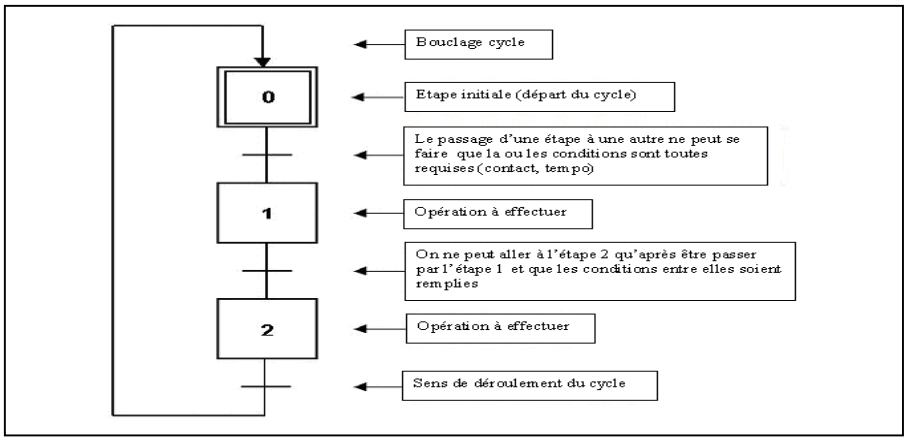


Figure II.5: symboles et fonction des instructions du Grafcet

II.6) Automate STEP 7/200 :

L'automate S7/200 combine un micro processeur, une alimentation intégrée, des circuits d'entrée et des circuits de sortie dans un boîtier compact (figure II.6), afin de créer un puissant micro-automate. Une fois que nous avons chargé notre programme, le S7/200 contient la logique nécessaire au contrôle et à la commande d'unités d'entrée et de sortie dans notre application.

Siemens fournit des modèles de CPU S7/200 différents, disposant de divers éléments et fonctions afin de nous aider à créer des solutions efficaces pour nos applications variées.

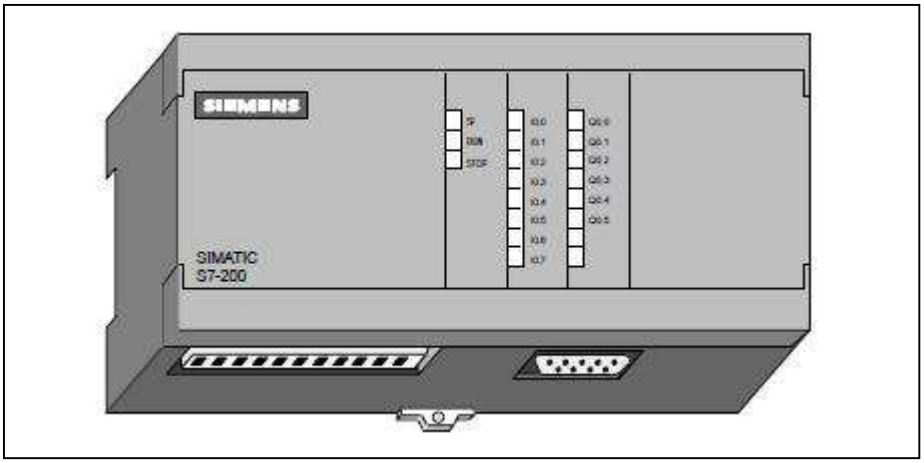


Figure II.6: Automate programmable S7-200

II.6.1) Environnement matériel requis :

STEP7-Micro/WIN s'exécute sur un ordinateur personnel (PC) ou sur une console de programmation (PG) Siemens, telle que la PG 760. Notre PC ou la PG doit satisfaire aux exigences minimales suivantes:

Système d'exploitation:

- Windows 2000, Windows XP, Vista.
- Au moins 350 Mo d'espace libre sur le disque dur.
- Souris (recommandée).

II.6.2) Options de communication :

Siemens propose deux options de programmation pour la connexion de notre ordinateur à notre S7-200 : une connexion directe à l'aide d'un câble PPI multi-maître ou une carte CP (processeur de communication) avec un câble MPI. Le câble de programmation PPI multi-maître constitue la solution la plus commune et économique pour connecter notre ordinateur au S7-200. Ce câble relie l'interface de communication du S7-200 au port de communication série de notre ordinateur. Nous pouvons également nous en servir pour raccorder d'autres unités de communication au S7-200.

II.6.2.1) Raccordement de l'alimentation à la CPU S7-200 :

La première étape consiste à raccorder le S7-200 à une source de courant. La (figure II.7), montre les connexions de câblage pour un modèle en courant continu et un modèle en courant alternatif de la CPU S7-200.

Tout d'abord il faut s'assurez, avant d'installer ou de démonter tout appareil électrique, que cet appareil a été mis hors tension. On prend toujours toutes les mesures de sécurité nécessaires et que l'alimentation de l'automate S7-200 est coupée avant de tenter d'installer ou de démonter le S7-200.

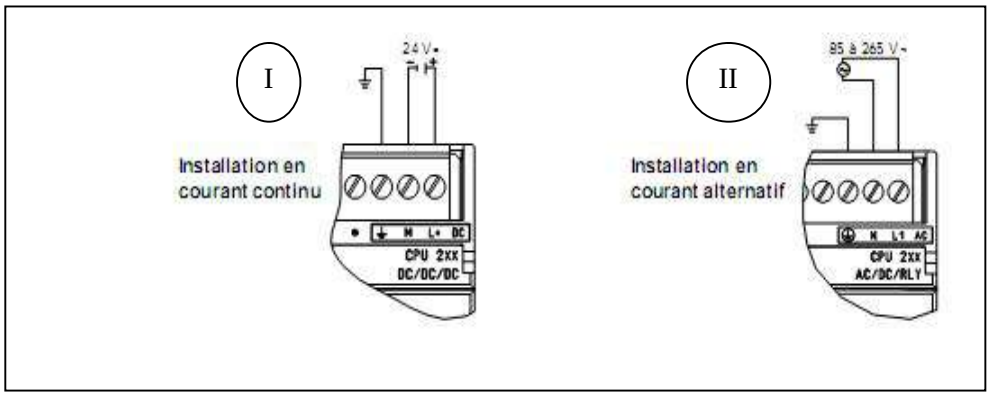


Figure II.7: alimentation d' S7-200

Remarque :

Tenter d'installer ou de câbler le S7-200 ou des équipements reliés alors qu'ils sont sous tension peut provoquer un choc électrique ou un dysfonctionnement du matériel. La non désactivation de l'alimentation complète du S7-200 et des appareils qui y sont reliés pendant les procédures d'installation ou de désinstallation peut entraîner des dommages matériels importants.

On prend toujours toutes les mesures de sécurité nécessaires et s'assurez que l'alimentation de l'automate S7 200 est coupée avant de tenter d'installer ou de démonter le S7-200 ou des matériels reliés.

II.6.2.2) Connexion de la CPU S7-200 :

La connexion de notre S7-200 est facile. Il nous suffit de raccorder l'alimentation à notre CPU S7-200, puis de connecter le câble de communication entre la console de programmation et la CPU S7-200.

La (figure II.8) montre un câble RS-232/PPI multi-maître reliant le S7-200 à la console de programmation. On procède comme suit pour connecter le câble :

- Raccordez le connecteur RS-232 (identifié par PC) du câble RS 232/PPI multi-maître à l'interface de communication de la console de programmation.
- Raccordez le connecteur RS-485 (identifié par PPI) du câble RS 232/PPI multi-maître à l'interface de communication 0 ou1du S7-200.
- S'assurer que les commutateurs multiples du câble RS-232/PPI multi maître sont réglés comme illustré à la (figure II.8).

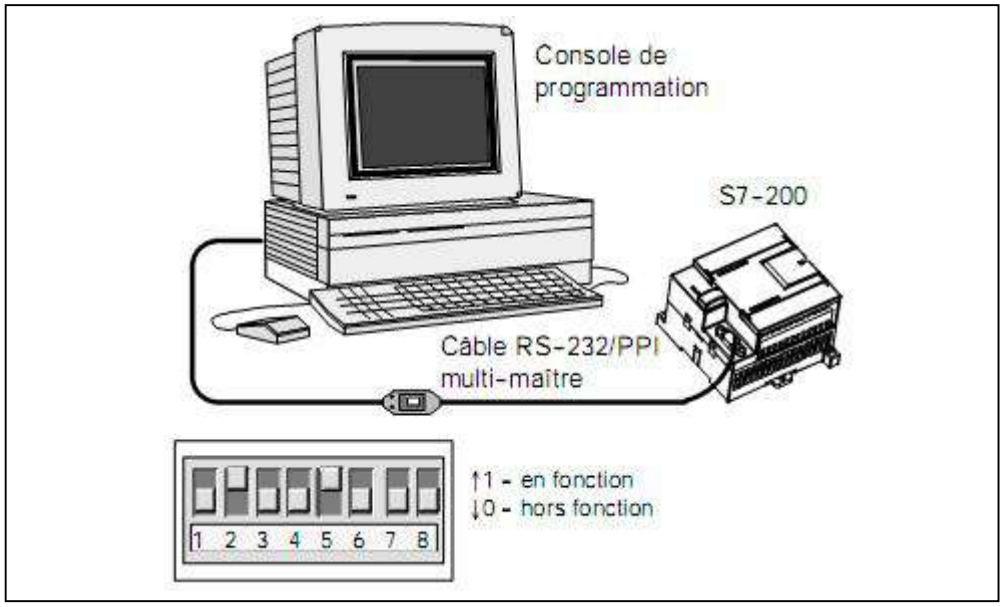


Figure II.8: connexion avec un câble RS-232/PPI multi-maitre.

II. 6.3) Composantes principales de l'automate programmable S7-200 :

Un automate programmable S7-200 consiste en une CPU S7-200 seule ou complétée de divers modules d'extension facultatifs.

II.6.3.1) CPU S7-200 :

Le module CPU S7-200 est un appareil autonome compact comprenant une unité centrale (CPU ou UC), une alimentation et des entrées/sorties discrètes.

- La CPU exécute le programme et sauvegarde les données pour la commande du processus ou de la tâche d'automatisation.
- L'alimentation fournit de l'énergie électrique à l'appareil de base et à tout module d'extension connecté.
- Les entrées et les sorties sont les points de commande du système : les entrées surveillent les signaux des appareils sur site (tels que capteurs et commutateurs) et les sorties commandent pompes, moteurs et autres appareils dans votre processus.
- L'interface de communication vous permet de connecter la CPU à une console de programmation ou à d'autres appareils. Certaines CPU S7-200 disposent de deux interfaces de communication.
- Des témoins (LED) d'état donnent des informations visuelles sur l'état de fonctionnement de la CPU (Marche - RUN - ou Arrêt - STOP), l'état en vigueur des entrées/sorties locales et la détection éventuelle d'une défaillance système.

II.6.3.2) Modules d'extension :

La CPU S7-200 comporte des entrées/sorties locales. Les modules d'extension permettent d'ajouter des entrées/sorties à l'appareil de base. Comme illustré à la (figure II.9), un connecteur de bus fourni avec le module d'extension permet de connecter ce dernier à l'appareil de base.

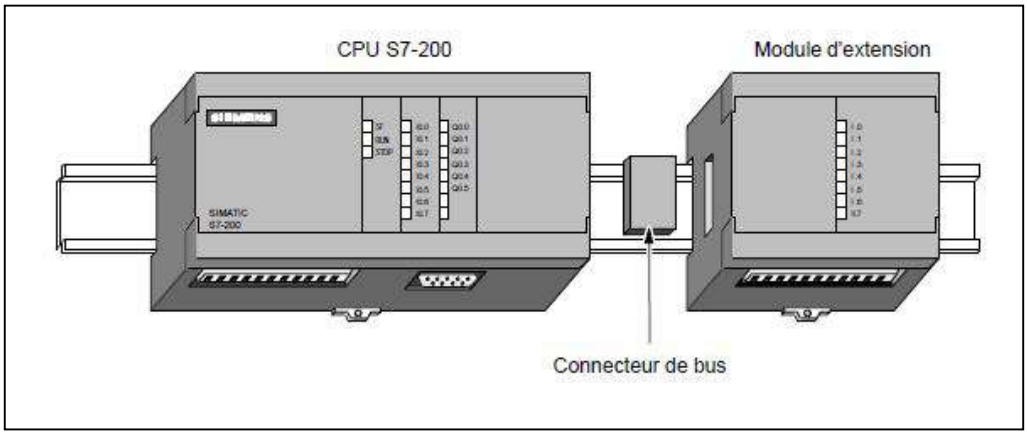


Figure II.9: Unité centrale avec module d'extension

II.6.4) Progiciel de programmation STEP 7-Micro/WIN :

Le progiciel de programmation STEP 7-Micro/WIN fournit un environnement convivial pour concevoir, éditer et surveiller la logique nécessaire à la commande de notre application. STEP 7-Micro/WIN comprend des éditeurs de programme, ce qui s'avère très pratique et efficace pour la mise au point du programme de commande de notre application.

II.6.4.1) Mode de représentation du langage de programmation STEP 7 :

Le langage de programmation STEP 7 dispose de plusieurs modes de représentation (**LOG, LIST, CONT)**, selon les goûts et l'état des connaissances, le programme peut être conçu sous forme de liste d'instruction puis converti en un autre mode de représentation. La représentation que nous allons utiliser c'est la **CONT**.

II.6.4.2) Saisie d'un programme « CONT » :

II.6.4.2.1) Création d'un projet :

Double-cliquez sur l'icône STEP 7-Micro/WIN 32  ou on sélectionne Simatic > STEP 7-Micro/WIN 32 dans le menu Démarrer, afin de lancer l'application. Un nouveau projet STEP 7-Micro/WIN 32 s'ouvre.

II.6.4.2.2) Ouvrir un projet existant :

Dans STEP 7-Micro/WIN 32, il faudra activer le menu Fichier et sélectionnez l'une des commandes suivantes

- ✓ Un projet existant et de l'ouvrir.
- ✓ Nom de fichier : Si nous avons travaillés sur un projet dernièrement, il apparaîtra dans le menu Fichier, où nous pourrons le sélectionner directement sans passer par la boîte de dialogue "Ouvrir".

Nous pouvons également faire appel à l'explorateur Windows pour sélectionner le répertoire approprié et ouvrir notre projet sans avoir à lancer STEP 7-Micro/WIN 32 lors d'une étape précédente. Dans STEP 7-Micro/WIN 32, version 3.0 ou plus, un projet est contenu dans un fichier unique d'extension.

II.6.4.2.3 Compilation en CONT :

La compilation s'effectuer à l'aide de boutons dans la barre d'outils ou de Commandes du menu Système cible (figure II.10).

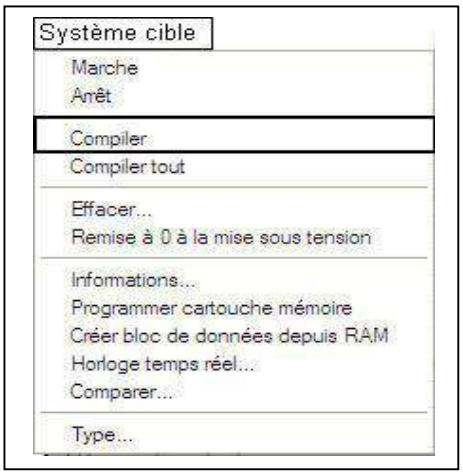


Figure II.10: Système cible.

Compiler : permet de compiler un seul élément du projet. La fenêtre (éditeur de programme, bloc de données système ou bloc de données) sélectionnée lorsque nous Exécuterons "Compiler" sera compilée, les deux autres fenêtres ne le seront pas.

Compiler tout : compile l'éditeur de programme, le bloc de données système et le bloc de données, et ce quelle que soit la fenêtre sélectionnée lorsque nous exécuterons la commande "Compiler tout".

II.6.4.2.4) Correction des erreurs à l'aide de la fenêtre :

Lors de la compilation, la fenêtre « Erreurs et avertissements » énumère toutes les erreurs qui apparaissent. Ces erreurs sont identifiées par leur emplacement (réseau, ligne ou colonne), ainsi que par leur type. Double-cliquez sur une ligne d'erreur pour afficher le réseau contenant cette erreur dans l'éditeur de programme.

II.6.4.2.5) Enregistrement du travail :

Pour enregistrer le travail, on clique sur le bouton "Enregistrer"  dans la barre d'outils, cette action permet de sauvegarder rapidement toutes les modifications apportées à notre travail. Lors du premier enregistrement d'un projet, le système nous demande de confirmer ou de modifier les choix de nom, et de répertoire par défaut, pour le projet en cours, ou en exécutons la commande Enregistrer ou Enregistrer sous du menu Fichier, nous permet de modifier le nom, ou le répertoire pour notre projet en cours.

Par défaut, STEP 7-Micro/WIN donne le nom "Projet1.mwp", au projet lors de sa création. Nous pouvons le modifier, si le nom nous convient, il deviendra "Projet2.mwp " par défaut pour le nouveau projet. Il s'enregistre aussi par défaut, dans le dossier "Project" situé dans le répertoire "STEP 7-Micro WIN". Ne nous somme pas obligé d'utiliser ce dossier pris par défaut.

II.6.4.2.6) Etablissement et communication et chargement d'un programme :

II.6.4.2.6.1) Sélection d'un type d'AP :

On peut sélectionner un AP en choisissant un modèle dans la liste de la boîte de dialogue "Type d'AP" (figure II.11). Si vous disposez, la CPU 224 par exemple. S'affiche :



Figure II.11: Sélection d'un type d'AP.

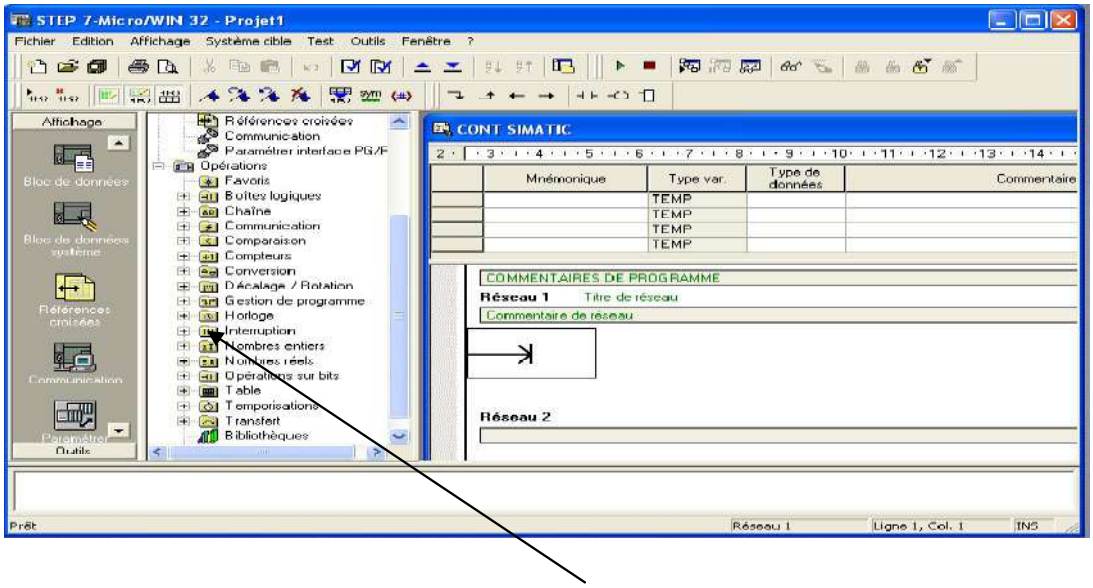
Lorsque nous effectuons cette sélection, les restrictions de plages et de fonctions pour la dernière version de microprogramme de la CPU 224sont appliquées. Mais, nous pouvons avoir une version de microprogramme plus ancienne de la CPU 224, qui ne prend pas en charge les fonctions disponibles dans la dernière version de microprogramme de la CPU 224 Aussi, pour être sûr que le modèle de notre AP et notre version de microprogramme sont tous deux pris en considération lors de la vérification de plages, nous pouvons demander que STEP 7-Micro/WIN 32 lise ces informations directement dans L'API.

II.6.4.2.6.2) Lecture du type d'AP éloigné par STEP 7-Micro/WIN 32 :

Pour que STEP 7-Micro/WIN 32 lise le modèle d'AP et la version du microprogramme, on clique sur le bouton dans la boîte de dialogue "Type d'AP".

II.6.4.2.6.3) Test du réseau de communication :

- Dans STEP 7-Micro/WIN 32, cliquez sur l'icône Communication  dans la barre d'exploration (figure II.12) ou sélectionnez Affichage > Composante > Communication dans la barre des menus.



Icône de communication

Figure II.12: emplacement de l'icône de communication

- Dans la partie droite de la boîte de dialogue "Communication", on clique sur le texte "Double clique pour rafraîchir".

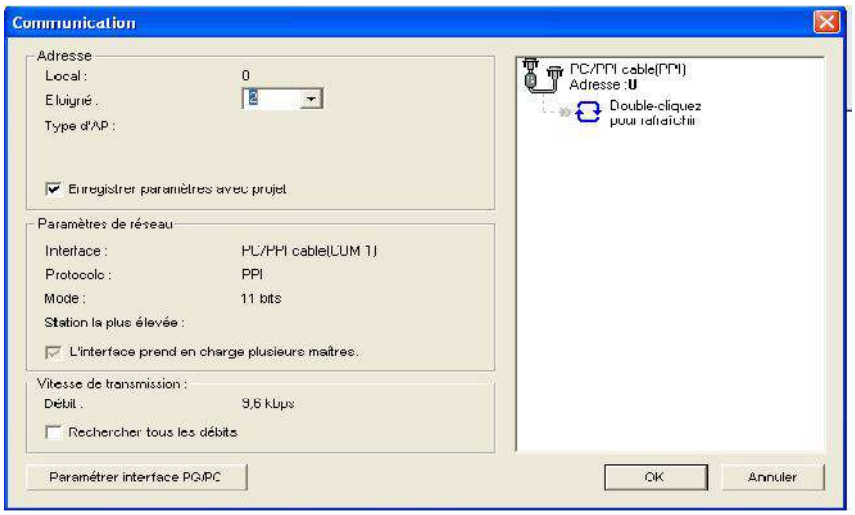


Figure II.13: boite de dialogue « communication »

La figure II.13 s'affiche, si on a réussi à établir la communication entre notre ordinateur personnel et les unités dans notre réseau, (avec leur type de modèle et leur adresse de station).

STEP 7-Micro/WIN 32 communique avec un seul AP à la fois. Une boîte rouge apparaît autour de l'AP qui est en train de communiquer avec STEP 7-Micro/WIN 32. On peut double cliquer sur un autre AP pour communiquer avec lui.

II.6.4.2.6.4) Chargement d'un programme dans la CPU :

Si on a établir la communication entre l'ordinateur personnel sur lequel s'exécute STEP 7-Micro/WIN 32 et un AP, on peut charger le programme dans cet AP. Procèdent comme suit (figure II.14):

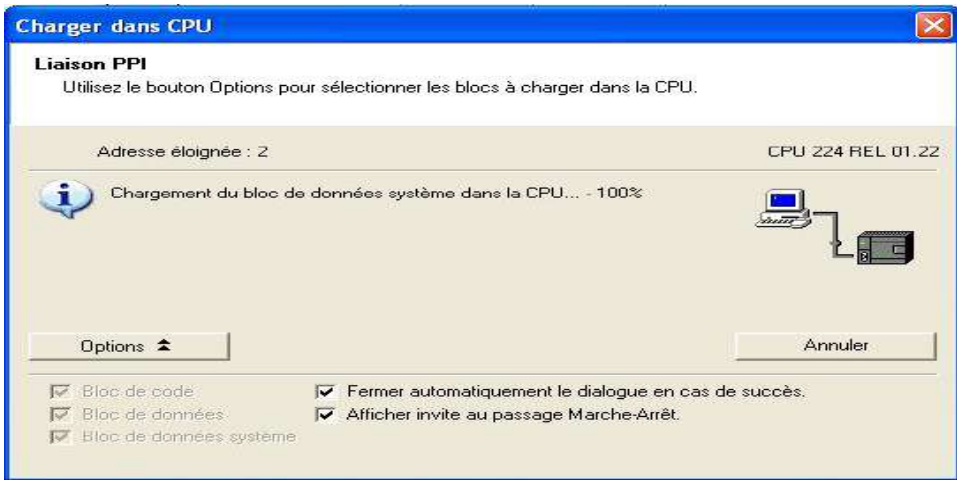


Figure II.14: chargement d'un programme dans la CPU

Conclusion :

Le S7-200 présente plusieurs caractéristiques spéciales nous permettant de personnaliser ses fonction sa fin qu'elles correspondent mieux à notre application il :

- Permet au programme de lire ou d'écrire directement dans les E/S.
- Permet d'allouer du temps de processus pour l'édition à l'état marche et la visualisation d'exécution.
- Permet de définir l'état des sorties **TOR** pour l'état **Arrêt** (STOP).
- Permet de configurer la valeur des sorties analogiques.
- Permet de définir la mémoire à conserver en cas de coupure de courant.
- Permet d'effectuer un filtrage des entrées **TOR**.
- Permet d'effectuer un filtrage des entrées analogiques.
- Permet de capturer des impulsions de courte durée.
- Fournit une **LED** commandée par l'utilisateur.
- Le S7-200 constitue un historique des événements majeurs concernant la CPU.
- Permet d'augmenter la mémoire de programme utilisateur disponible.
- Fournit une protection par mot de passe.
- Dispose d'entrées/sorties rapides.

III) Conception et réalisation du prototype :

Introduction :

Apres avoir donné une idée sur les profileuses ainsi sur les API dans les deux chapitres précédents. Nous nous intéressons dans ce chapitre à la réalisation du prototype, qui peut être divisé en trois parties : électrique, mécanique et commande par API.

Chaque partie fera l'objet d'une description et explication bien détaillée.

Le prototype de la profileuse, à été réalisé dans deux ateliers de travail, l'atelier de fabrication mécanique, et le laboratoire d'automatique. Le cadre de ce système est de réaliser un prototype fonctionnel pour faire la commande automatique de la profileuse, à l'aide d'un automate programmable S7-200 de SIEMENS.

III.1) Partie mécanique :

La fabrication et le montage complet de notre prototype (profileuse) est présenté par la (figure III.1), l'ensemble peut être décomposé en quatre sous-système, qui sont : le dévidoir (1), mécanisme de perforation (2), le système qui fait passer la bonde de papier (3), le mécanisme de découpage (4)

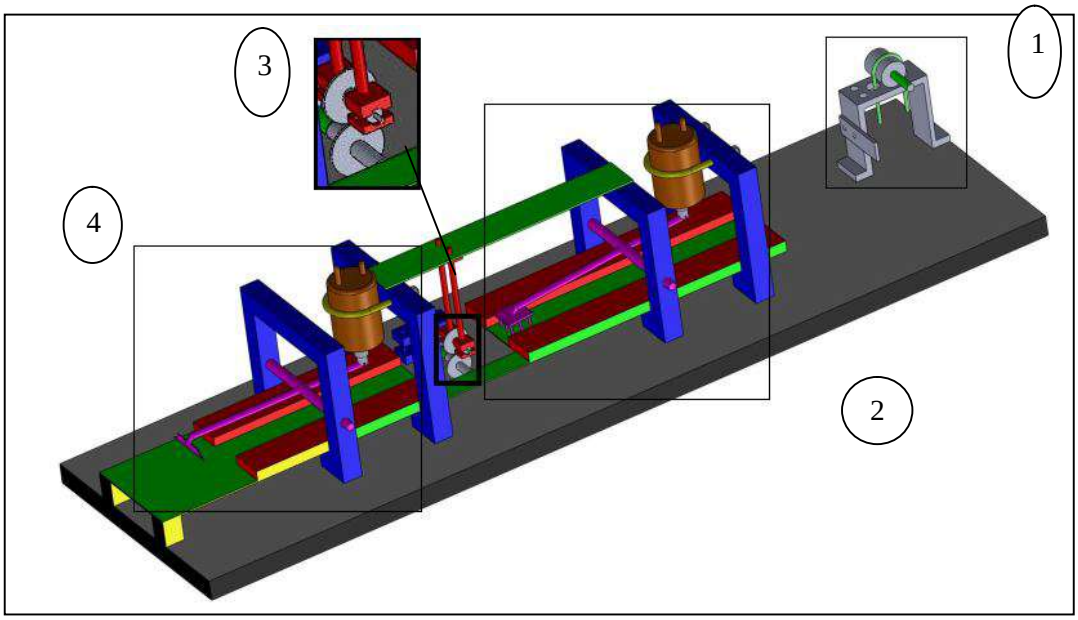


Figure III.1 : Prototype de la profileuse et ces sous-systèmes.

Lorsqu'on charge le programme dans l'automate et, on pressant sur le bouton de démarrage, le prototype fonctionnera selon des séquences bien définies : déroulement du papier sur la table, la perforation puis le découpage après une certaine longueur.

III.1.1) Sous système (1) dérouleur :

Il se compose d'un moteur à courant continu (moteur récupéré d'une imprimante), et de deux détecteurs. Ils sont fixés sur le dévidoir de façon à détecter la position du papier, si le papier est trop tendue le premier détecteur qui se trouve en haut fera démarrer le moteur, si le papier commence à se tirer le deuxième détecteur sera actionné par le poids du papier, et fera arrêter le moteur du dévidoir.

Touts d'abord on a construit un support qui porte le moteur (7), et les détecteurs de façon à ce qu'il soit sur le même niveau que la table. Alors nous avons pris une barre de fer plat (6), et à l'aide de la plieuse, nous avons plié la barre jusqu'à obtenir la forme de la (figure III.2).

7 : moteur. 6 : fer plat. 5 : collier. 8 : axe de rotation. 9 : surface n° (1). 10 : détecteurs.

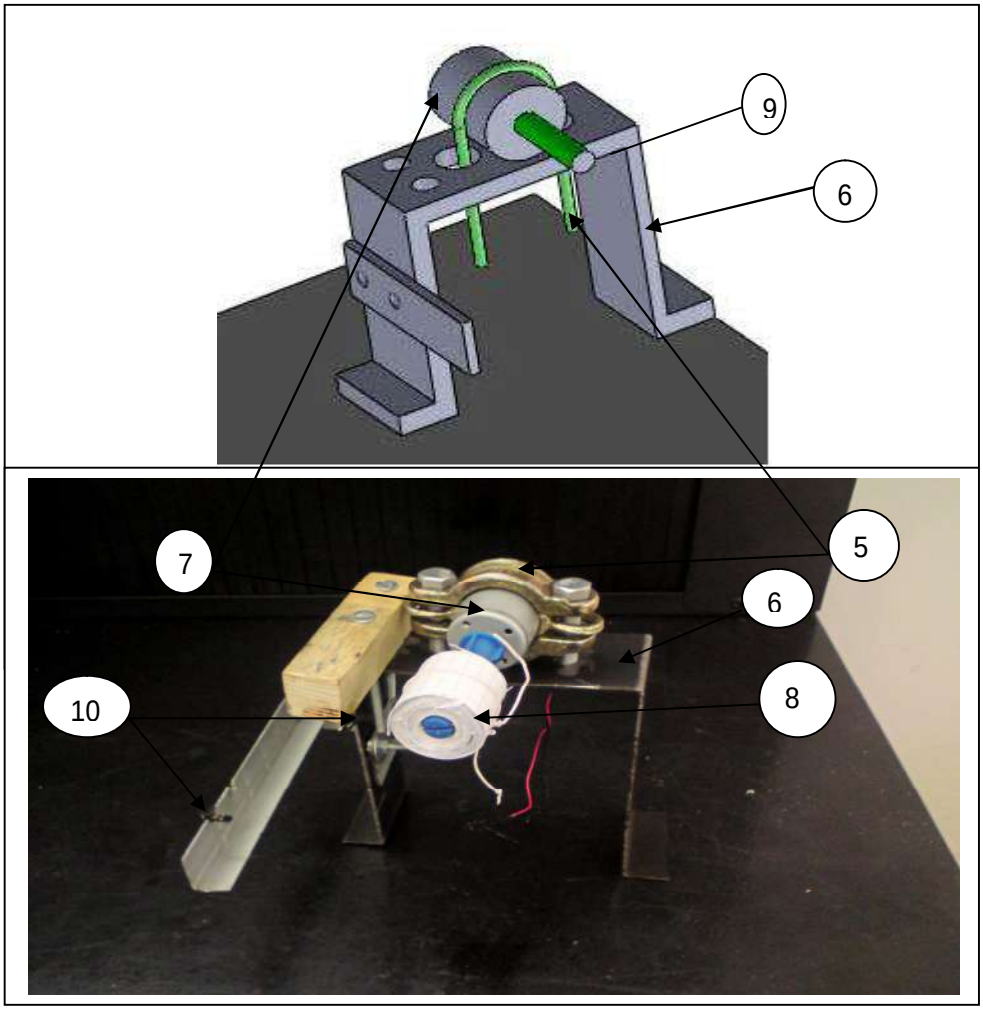


Figure III.2. : Dérouleur

Pour que le moteur soit fixe nous avons percé la surface (n°9) de deux trous d'un diamètre de 10 mm, et à l'aide d'un collier (figure III.3), et aussi nous avons pu sérer le moteur à courant continue.



Figure III.3 : collier de fixation

Pour les détecteurs :

- Le premier qui fait marcher le moteur : il est fixé à une planche en bois de 90 mm avec des vis écrou, et on a collé sur son extrémité le détecteur normalement ouvert (NO).
- Le deuxième qui fait arrêter le moteur : il est fixé sur une barre métallique (figure III.4).

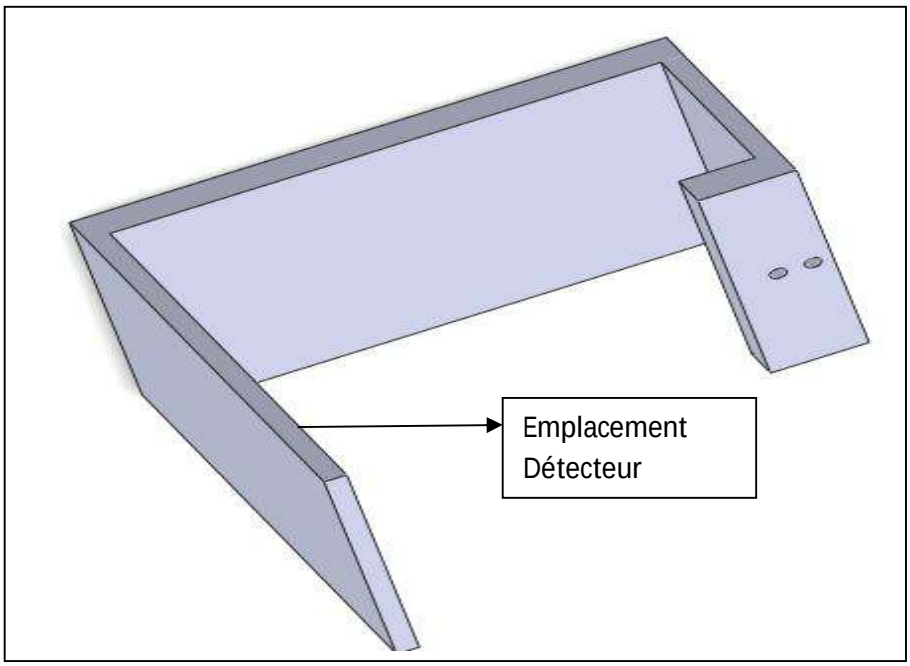
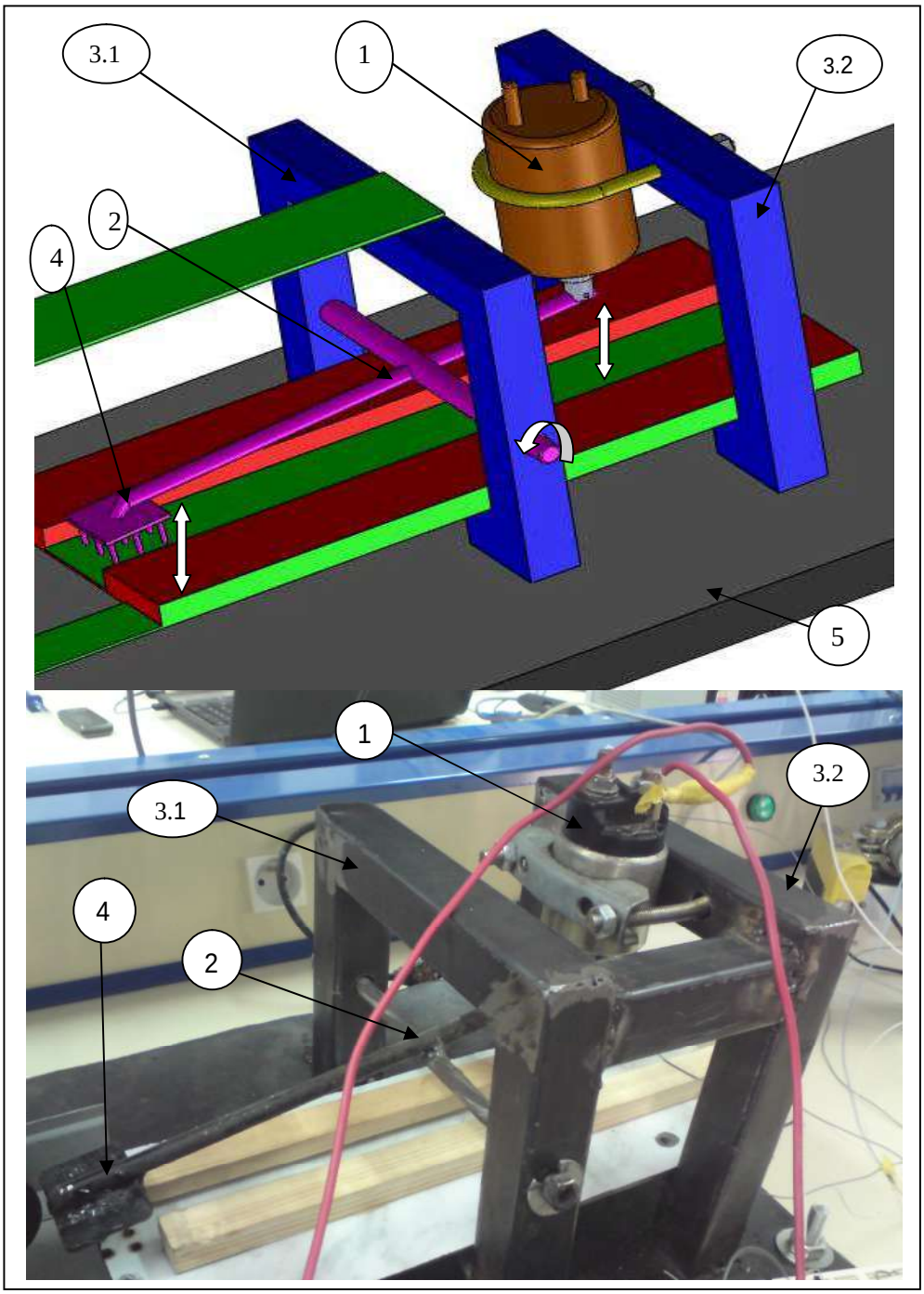


Figure III.4 : support du détecteur.

Ces deux détecteur s sont mis en place de façon à détecter la position haute et basse de la bande de papier, à chaque fois qu'elle devient tendue elle actionne le détecteur de haut, qui fait marcher à son tour le derouleur, le deuxième détecteur sera actionné avec le poids du papier (figure III.5).

III.1.2) Sous-système (2) station de perforation :

Cette station se compose de plusieurs pièces (figure III.5), un contacteur (1), fourchette de perçage (2), des supports en forme de U (3.1) ; (3.2), une table (4) avec des guide papier qui sert à remettre le papier sur sa trajectoire exacte.



1 : contacteur.

2 : fourchette.

3.1 : support.

3.2 : support.

4 : table.

5 : base.

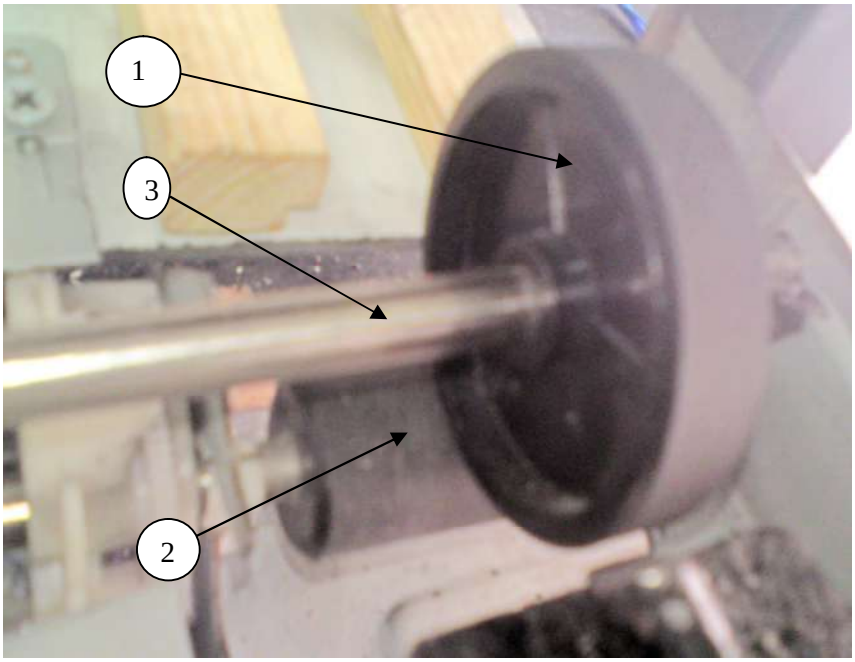
Figure III.5 : sous-système n°2

Les supports en forme U sont soudés avec un angle de 90° avec la base (5), un contacteur est fixé verticalement sur le support (3.1), une fourchette est fixée sur l'axe de contacteur de façon à inverser son

mouvement, (figure III.5), si on alimente le contacteur la tête de la fourchette se dirige vers le bas (car le contacteur rentre), si l'alimentation est perdue elle revient vers sa position initiale (haut), ce qui permet de trouer le papier.

III.1.3) Sous-système (3) défileuse :

C'est un système qui assure le déroulement de papier sur la table (figure III.6), un défileur de papier de récupération d'une imprimante est utilisé, ce système permettra le déroulement du papier, de porte sur son axe le codeur optique qui envoi des signaux, afin de calculer la distance parcouru par le ruban de papier.



1 : roue superieur.

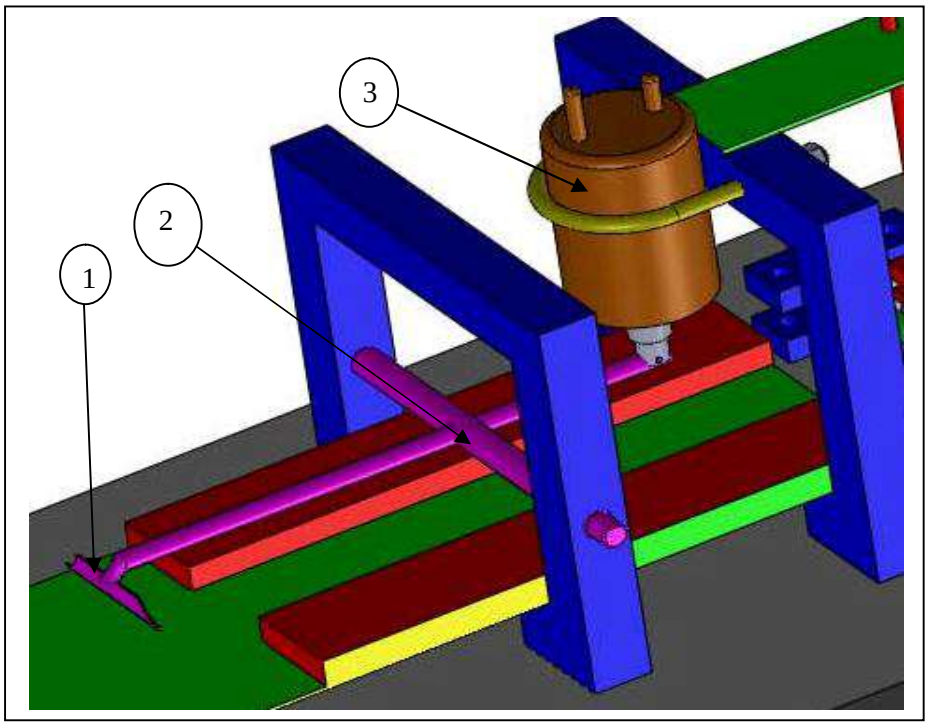
2 : roue inferieur.

3 : axe de rotation superieur

Figure III.6 : sous-système n°3.

III.1.4) Sous-système (4) station de découpage :

C'est le même système que le précédent la différence est seulement dans l'outil de travail. Cet outil est une lame de cutter (1) placé sur la fourchette (2) de manière à couper le papier lorsque l'automate donne une impulsion électronique au contacteur (3), comme montrer sur la (figure III.7). Cette coupe est faite si une certaine longueur est atteinte.



3 : contacteur.

2 : fourchette.

1 : outil de coupe.

Figure III.7 : sous système n°4.

III.2) Partie électrique :

Pour cette partie nous allons citer les différents actionneurs électriques qui composent notre prototype. Ces actionneurs sont relier à un automate S7-200 à travers des relais.

III.2.1) contacteur (solénoïde) :

III.2.1.1) Solénoïde :

III.2.1.2) Définition :

Bobine cylindrique dont la longueur est très supérieure aux dimensions transversales.

Un solénoïde se définit comme une bobine de fil dont la forme courante est un long cylindre, qui quand elle est traversée par un courant se comporte comme un aimant droit qui attire un noyau mobile dans la bobine quand un courant circule. Une définition plus simple est qu'un solénoïde est une bobine et un noyau ferreux mobile utilisés pour convertir de l'énergie électrique en énergie mécanique.

C’est une solution qui est utilisé par les constructeurs automobiles spécialement en coordination avec le démarreur (figure III.8), car ce relais effectue deux fonctions essentielles pour le démarrage d’un moteur à combustion interne. La première fonction qui est de commander le circuit de puissance du

démarreur grâce à un courant de commande faible. La deuxième engager le pignon de démarreur sur la roue dentée du volant moteur grâce à un électro-aimant.

Le mécanisme a été appelé un solénoïde parce que c'est un actionneur linéaire avec des fonctions tant mécaniques qu'électriques. Étant donné la grande force exigée de la part de la bobine du solénoïde, celle-ci doit tirer typiquement 8 à 10 ampères pendant le démarrage du moteur.



Figure III.8 : contacteur

III.2.1.3) Constituant solénoïde :

Il se compose principalement d'un axe, un ressort de rappelle, une bobine longue, un noyau (figure III.9).

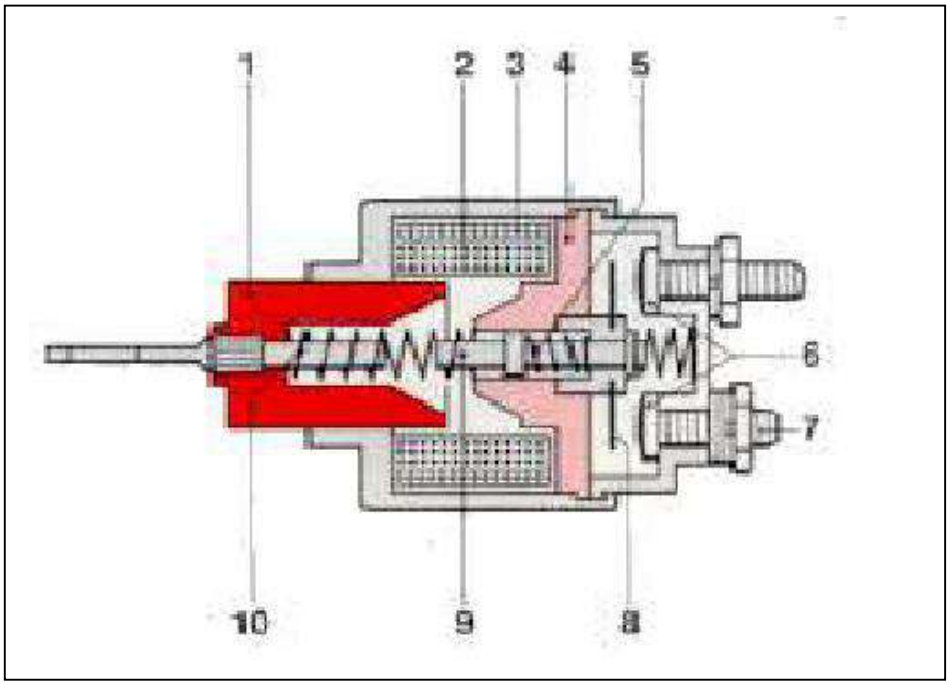


Figure III.9 : composant d'un contacteur.

1 : noyau plongeur.

2 : enroulement d'appel.

3 : enroulement d'appel et de maintien.

4 : noyau magnétique.

5 : ressorts de contact.

6 : contacts.

7 : connexions de puissance.

8 : pont de contact.

9 : axe.

10 : ressorts de rappel.

III.2.2) Codeur optique :

III.2.2.1) Généralité sur les codeurs optiques :

La croissance de la puissance des systèmes de traitement ainsi que les impératifs de productivité appellent dans tous les domaines de production industrielle un besoin d'information continu sur le déplacement, la position ou la vitesse des outils.

Les systèmes de détection conventionnels (interrupteurs et détecteurs de positions), qui ne peuvent fournir que des informations « TOUT OU RIEN » à des endroits prédéterminés, ne répondent que partiellement aux besoins de précision et de flexibilité.

Dans le cas d'un codeur optique, le positionnement du mobile est entièrement maîtrisé par les systèmes de traitement et non plus réalisé physiquement par le positionnement d'un interrupteur de position sur la machine. Le codeur optique est un dispositif électromécanique dont la sortie électrique représente sous forme numérique une fonction mathématique de la position angulaire de l'axe d'entrée. Il existe deux types de codeurs rotatifs :

- le codeur incrémental (ou générateurs d'impulsions).

- le codeur absolu.

III.2.2.2) Définition d'un codeur optique :

Un codeur optique est un dispositif électromécanique dont la sortie électrique représenté sous forme numérique la position angulaire de l'axe d'entrée.

Un codeur optique rotatif (figure III.10) est un capteur angulaire de position. Lié mécaniquement à un arbre qui l'entraîne, son axe fait tourner un disque qui comporte une succession de zones opaques et transparentes. La lumière émise par des diodes électroluminescentes arrive sur des photodiodes à chaque fois qu'elle traverse les zones transparentes du disque. Les photodiodes génèrent alors un signal électrique qui est amplifié et converti en signal carré avant d'être transmis vers l'unité de traitement.

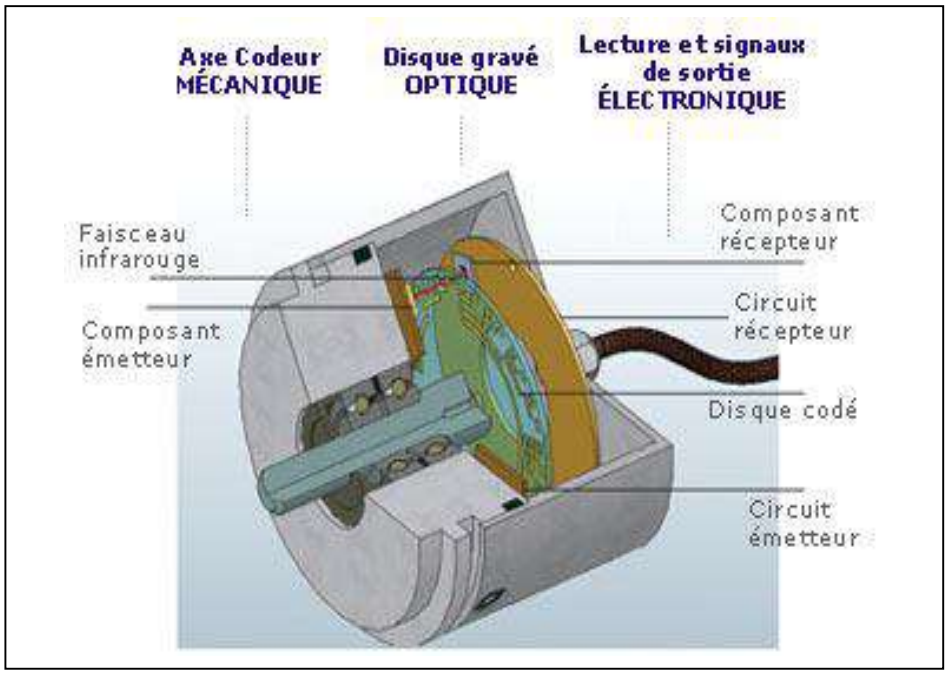


Figure III.10 : codeur optique.

III.2.2.3) Différent types de codeurs :

Il existe deux types de codeurs optiques : « **incrémental »** et **« absolu ».**

III.2.2.3.1) Codeur incrémental (ou générateur d'impulsions) :

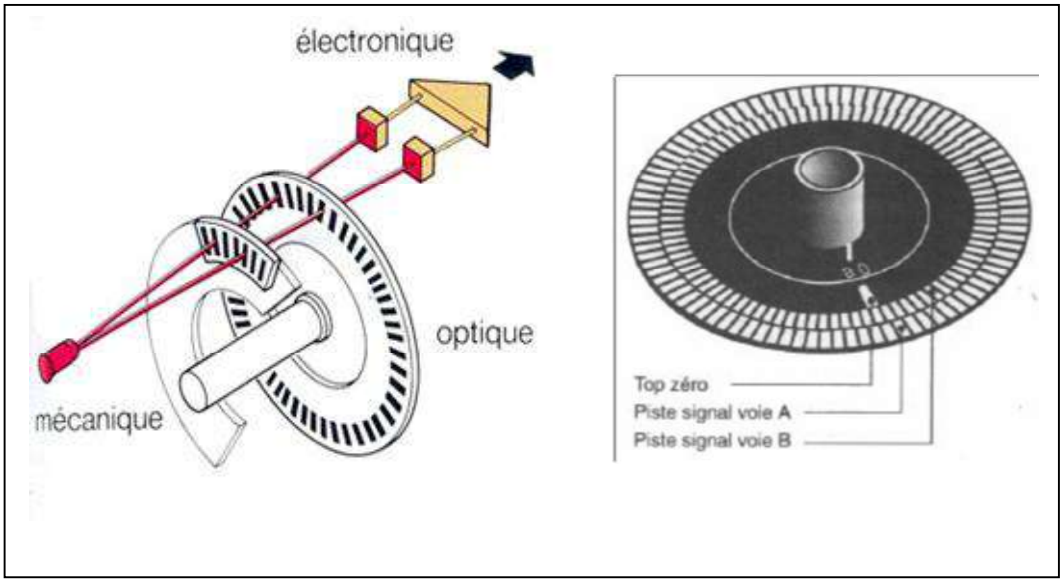


Figure III.11 : zone opaque et transparent du disque de codeur incrémental.

Le disque comporte au maximum **3 pistes** (figure III.11)**.** Une ou deux pistes extérieures divisées en (n) intervalles d'angles égaux alternativement opaques et transparents.

Pour un tour complet du codeur, le faisceau lumineux est interrompu (n) fois et délivre (n) signaux carrés (A et B) en quadrature (figure III.12).

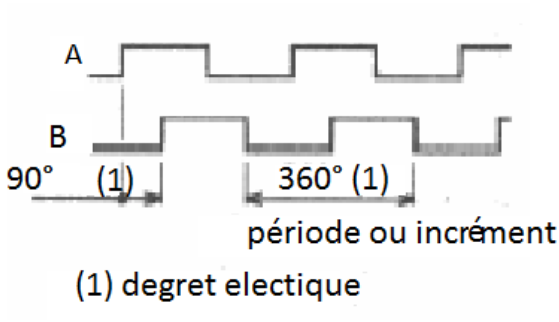


Figure III.12 : signaux carrés délivrer par le codeur.

Le **comptage-décomptage** des impulsions par l'unité de traitement permet de définir la position du mobile.

III.2.2.3.1.1) RESOLUTION (Nb de points par tour) :

On peut trouver trois cas qui peuvent se présenter :

- Le système de traitement n'utilise que les fronts montants de la voie A (exploitation simple). La résolution est égale au nombre de points (n).
- Le système de traitement utilise les fronts descendants et montants de la voie A (exploitation double). La résolution est multipliée par 2 (2 x n).
- Le système de traitement utilise les voies A et B (exploitation quadruple). La résolution est multipliée par 4 (4 x n).

III.2.2.3.2) Codeur « absolu » :

Il ressemble au codeur incrémental sauf au niveau des disques, car le disque des codeurs absolus comporte un nombre « n » de pistes concentriques divisées en segments égaux alternativement opaques et transparents (figure III.13).

A chaque piste est associé un couple émetteur / récepteur optique. Chaque piste a donc son propre système de lecture.

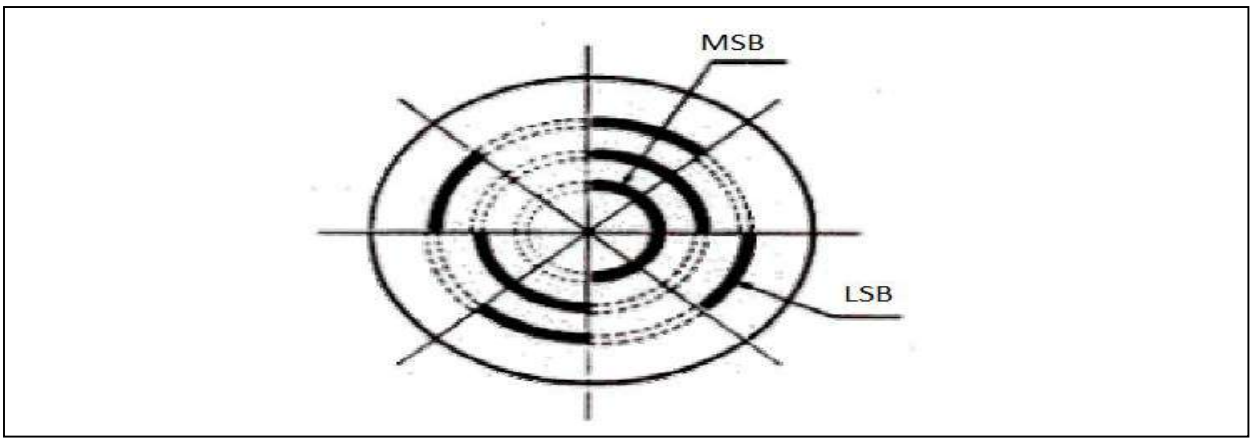


Figure III.13 : disque d'un codeur absolu.

III.2.2.3.2.1) Exemple d'un codeur absolu 3 bits (avec le code binaire) :

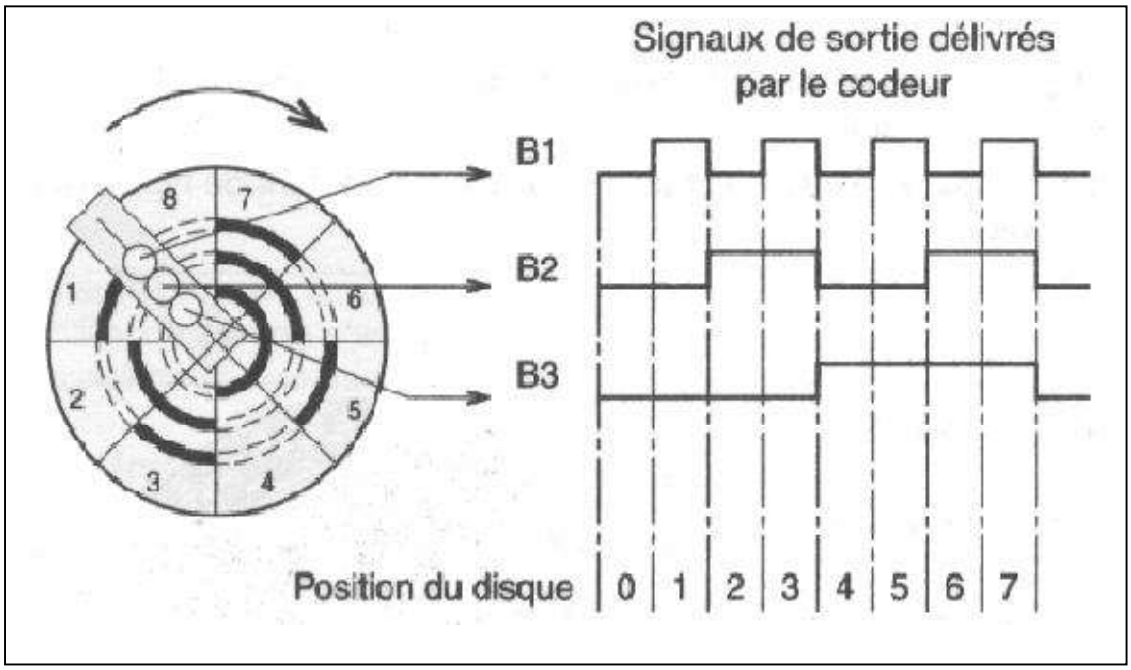


Figure III.14 : exemple d'un codeur absolu

Pour chaque position angulaire de l'axe (figure III.14), le disque fournit un code binaire de 3 bits correspondant à 1/8ème de tour. **3 bits ↔ 2^n= 8 positions.** Le nombre de sorties parallèles est le même que le nombre de bits ou de pistes sur le disque. Elles sont ici désignées par B1, B2, B3 car c'est un disque en binaire pur (tableau III.2).

Position du disque	Code binaire		
	B1	B2	B3
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

Tableau III.1 : Exemple d'un codeur absolu 3 bits (avec le code binaire)

III.2.2.3.2.2) Choix du codeur :

Sans tenir compte des jeux et imprécisions mécaniques, le nombre de points est calculé à l'aide de la formule suivante :

$$\text{Nombre de point} = 1/Ps*R*P \quad \text{.....(6)}$$

Avec :

Ps = précision souhaitée en mm,

R = rapport de réduction entre l'engrenage du mouvement entraînant le codeur et le dernier engrenage entraînant le mobile.

et P = rapport de conversion du mouvement de rotation en mouvement de translation.

Exemple :

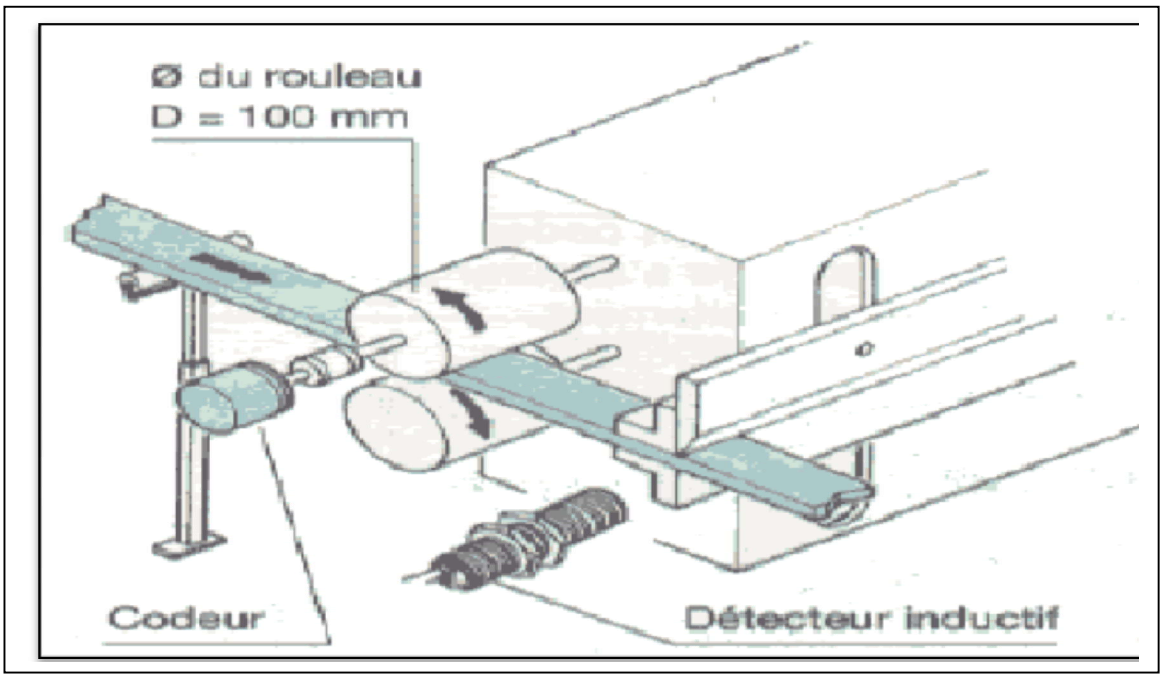


Figure III.15 : exemple de calcul

$$P = \pi * D = 3.14 * 100 = 314 \text{ (mm)}$$

Cela veut dire que si le rouleau fait un tour complet sur lui-même le tapi avancera de 314 (mm).

III.2.2.4) Codeur optique utilisé sur le prototype :

Au début on a pensé à utiliser le codeur optique qui se trouve dans la souris d'un ordinateur PS/2, mais on à tout de suite trouver des difficultés à retrouver, l'alimentation du circuit, et les sorties du signal.

C'est pour cela que nous avons opté pour une solution plus adéquate, qui consisté à ramener un détecteur de position (utilisé dans les imprimantes) (2), et on a fabriqué nous-mêmes un codeur avec une seule piste de 4 trous (1), ce codeur est placé sur l'axe de rotation de la défileuse (figure III.16).

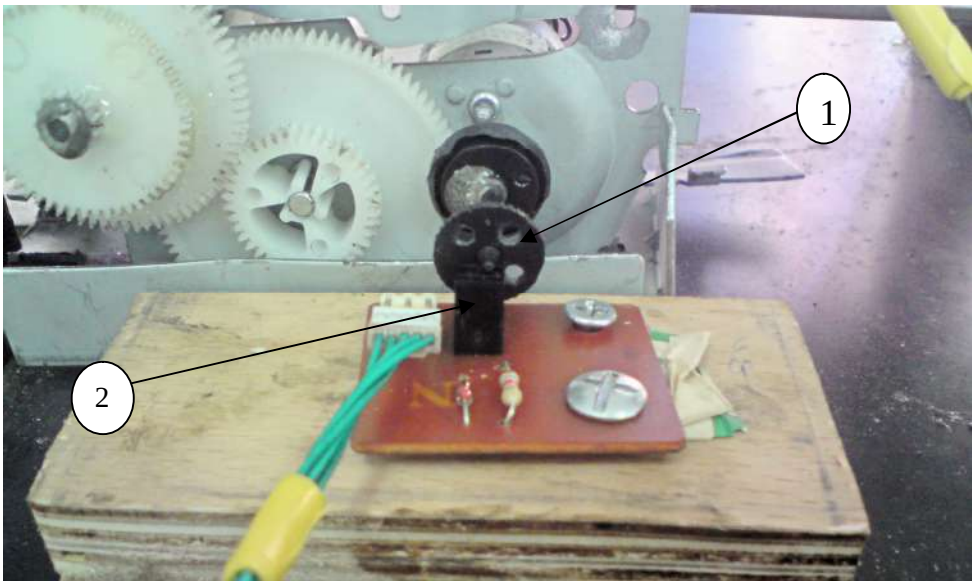


Figure III.16 : Codeur optique utilisé sur le prototype.

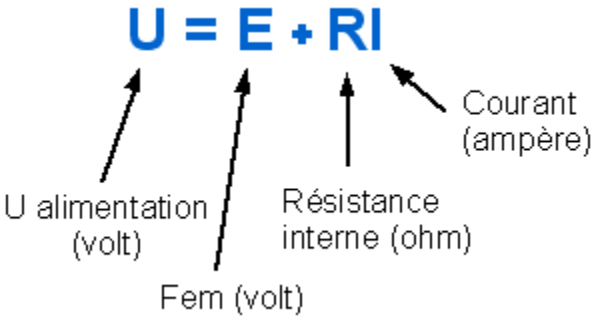
III.2.3) moteur à courant continue

III.2.3.1) Principe de fonctionnement :

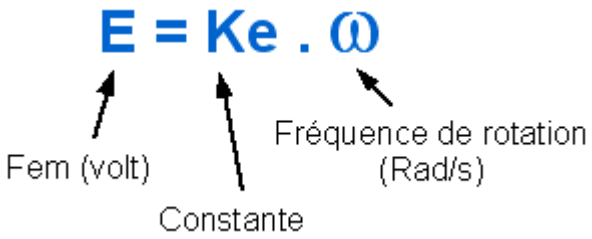
Si un conducteur en forme de spire parcouru par un courant I, est placé dans un champ magnétique, il est soumis à des forces de Laplace. Ces forces créent un couple de rotation qui fait tourner la spire sur son axe. Quand la spire a fait un demi-tour, il faut inverser la polarité pour inverser le sens des forces et continuer le mouvement. Ce sera le rôle du collecteur.

Le rotor, (partie tournante), est constitué d'un noyau métallique avec un bobinage de cuivre, le stator comporte des aimants permanents qui engendrent un champ magnétique dont le flux traverse le rotor. L'espace étroit entre le rotor et le stator est nommé entrefer.

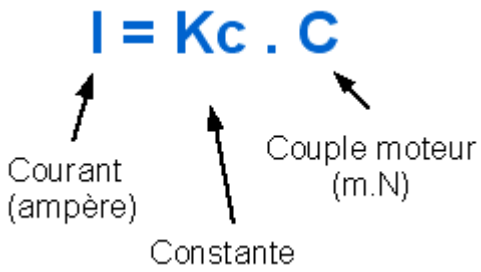
Les équations caractéristiques du moteur sont les suivantes
équation découle directement du schéma équivalent ci-dessus



La tension fem est proportionnelle à la fréquence de rotation



Le courant consommé par le moteur est directement lié au couple résistant sur l'arbre



Ke et Kc sont des constantes qui caractérisent le moteur.

On peut en déduire que :

Pour faire varier la fréquence de rotation, il faut faire varier E et donc la tension d'alimentation U.
Pour inverser le sens de rotation, il faut inverser E et donc la tension d'alimentation à ses bornes.
Le courant varie avec le couple, on peut aussi limiter le courant pour limiter le couple.

III.3) les relais :

Un relais est un organe électrotechnique permettant la commutation de liaisons électrique. Il est charger de transmettre un ordre de la partie commande à la partie puissance d'un appareil électrique et permet, entre autres, un isolement galvanique entre les deux parties.

Le relais est une solution à la commande en puissance.il assure en outre une isolation galvanique en mettant en œuvre un mouvement mécanique. On alimentant la bobine, le contact mobile est déplacé fermant ainsi le contacte électrique. En l'absence de courant dans la bobine le ressort de rappel maintient le contact ouvert (**Figure III.17**).



Figure III.17 : schéma d'un relais.

III.4) Les transistors :



Figure III.18 : photo d'un transistor.

Le **transistor** est un composant électronique actif utilisé :

- Comme interrupteur dans les circuits logiques,
- Comme amplificateur de signal,
- Pour stabiliser une tension, moduler un signal ainsi que de nombreuses autres utilisations.

Un transistor est un dispositif semi-conducteur à trois électrodes actives, qui permet de contrôler un courant (ou une tension) sur une des électrodes de sortie (*le collecteur* pour le transistor bipolaire et *le drain* sur un transistor à effet de champ) grâce à une électrode d'entrée (*la base* sur un transistor bipolaire et *la grille* pour un transistor à effet de champ).

III.4.1) Principe de fonctionnement :

Les transistors bipolaires fonctionnent comme suite :

- Le transistor bipolaire est un amplificateur de courant, on injecte un courant dans l'espace base/émetteur afin de créer un courant multiplié par le gain du transistor entre l'émetteur et le collecteur.

- Les transistors bipolaires NPN (négatif-positif-négatif) qui laissent circuler un courant de la base (+) vers l'émetteur (–), sont plus rapides et ont une meilleure tenue en tension que les transistors PNP base (–) émetteur (+).

Pour le prototype on à utiliser 3 transistors, car il a trois composants à alimenté : le moteur de la défileuse, le moteur du dévidoir, et en fin l'automate S7-200.

III.4.2) analyse fonctionnelle du prototype de profileuse :

Suite à la réalisation du prototype de la profileuse. L'analyse de ces détails visait à obtenir une meilleure performance.

1. Problème : il était difficile de trouver un système qui assure la perforation et le découpage du papier. Au début nous avions opté pour un moteur raccorder à un visse infinie avec un écrou, qui monte et descend, suivant la rotation du moteur. Cette solution ne nous à pas tellement aider, car nous n'avions pas trouvé sur le marché de tels mécanismes, et puis un moteur avec deux rotations exige un circuit électronique supplémentaire pour changer le sens de rotation.

- Solution : un contacteur d'automobile est un actionneur électrique puissant, une impulsion électrique génère un champ magnétique, ce champs déplace l'électro-aimant, l'absence de cette impulsion remet l'électro-aimant à sa place initial à l'aide du ressort intégrer sur le solénoïde.

2. Problème : quand on alimente le contacteur, l'électro-aimant rentre dans le contacteur, c'est le mouvement inverse qui nous intéresse.

- Solution : construire une fourchette qui inverseras le mouvement, cette solution est inspirée, par le système utilisé pour les moteurs automobiles.

3. Problème : le déroulement du papier n'était pas synchronisé avec la défileuse, un encombrement est très vite apparu.

- Solution : placer des détecteur sur le dévidoir de façon à détecter la position haute et basse du papier, si la bande de papier est tendu le premier détecteur signale ou moteur pour actionné, si le papier commence à s'enrouler trop vite

il actionnera avec sont poids le deuxième qui coupera l'alimentation du moteur.

4. Problème : la défileuse ne marcher pas, elle ne défile pas le papier sur la table, à cause de la roue supérieur, elle ne touche pas la roue inferieure.

. Solution : nous avons utilisé un ressort de façon à attirer l'axe de rotation de la roue supérieur contre l'axe de rotation de la roue inferieur.

5. Problème : le temps de réponse du contacteur électrique est très long.

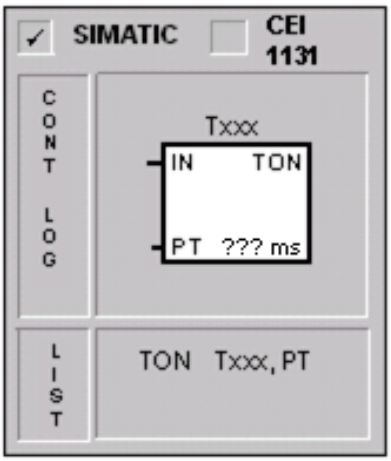
. Solution : jouer sur les instructions du programme Ladder. En utilisant les temporisateurs

III.5) Partie Automatisme:

III.5.1) Les opération utilisée dans notre programme :

III.5.1.1) Opération de temporisation T37 :

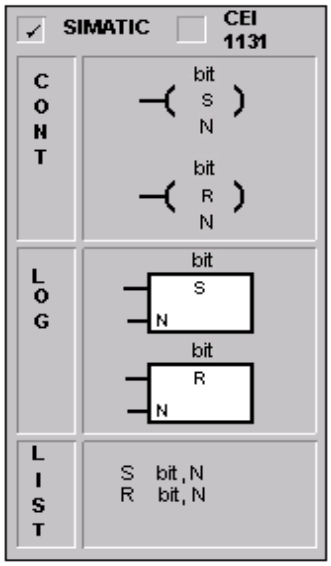
L'opération Démarrer temporisation sous forme de retard à la montée (TON) compte le temps qui s'écoule lorsque l'entrée de validation est activée. Lorsque la valeur en cours (Txxx) est supérieure ou égale au temps prédéfini (PT), le bit de temporisation est activé. Lorsque l'entrée de validation est désactivée, la valeur en cours de la temporisation TON est effacée. La temporisation TON poursuit le comptage une fois la valeur prédéfinie atteinte ; son exécution s'arrête à la valeur maximale (32767). **(Figure.III.19)**



(Figure.III.19) *Opération de temporisation T37*

III.5.1.2) Operation set (S) ET reset (R):

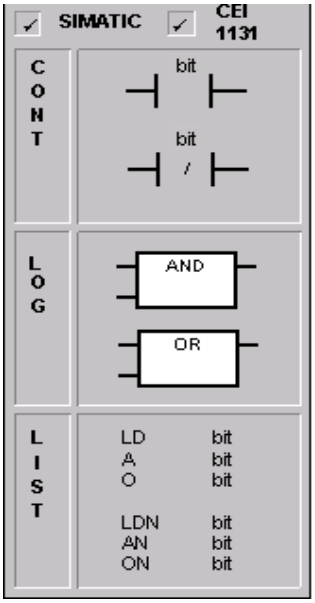
Les opérations Mettre à 1 (S) et Mettre à 0 (R) mettent à 1 (activent) ou mettent à 0 (désactivent) le nombre N de sorties indiquées, en commençant à l'adresse Bit indiquée. Vous pouvez mettre 1 à 255 sorties à 1 ou à 0.Si, pour l'opération Mettre à 0, le bit correspond à un bit de temporisation (T) ou de compteur (C), l'opération met le bit de temporisation ou de compteur à 0 et efface la valeur de comptage ou de temporisation en cours. **(Figure.III.20)**



(Figure.III.20) *Operation set (S) ET reset (R)*

III.5.1.3) OPERATION M0.2 :

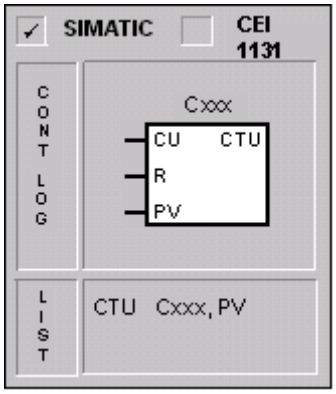
Ces opérations obtiennent la valeur référencée dans la mémoire ou dans la mémoire image si le type de données est I ou Q.



(Figure.III.21) *OPERATION M0.2*

III.5.1.4) Opération CTU :

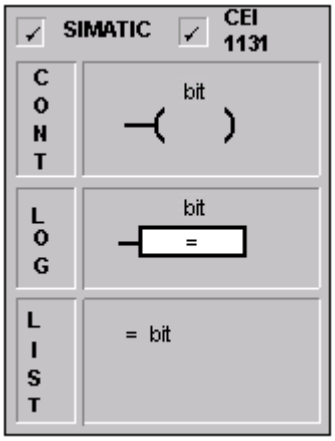
L'opération Compteur incrémental (CTU) incrémente en partant de la valeur en cours à chaque front montant de l'entrée d'incrémentation CU. Lorsque la valeur en cours « Cxxx » est supérieure ou égale à la valeur prédéfinie PV, le bit de compteur C est activé. Le compteur est remis à zéro lorsque l'entrée de remise à zéro R est activée ou que l'opération « Mettre à 0 » est exécutée. Le compteur incrémental arrête le comptage lorsqu'il atteint la valeur maximale 32 767.



(Figure.22) Opération CTU

III.5.1.5) Opération CEI(=):

L'opération Sortie (=) écrit la nouvelle valeur du bit de sortie dans la mémoire image.



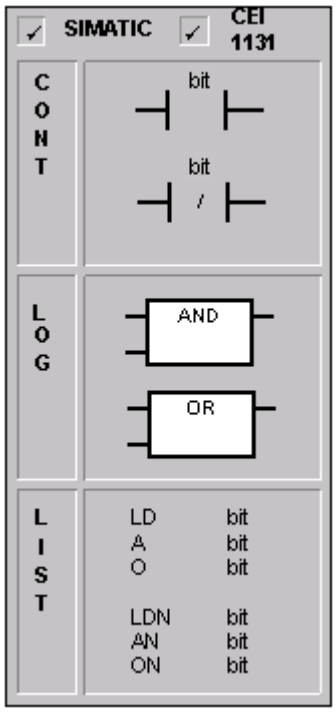
(Figuer.23.) Opération CEI(=):

III.5.1.6) Operation CEI-(AND,OR) :

Ces opérations obtiennent la valeur référencée dans la mémoire ou dans la mémoire image si le type de données est I ou Q.

I : entrée

Q : sorte



(Figuer.24.) Operation CEI-(AND,OR*)*

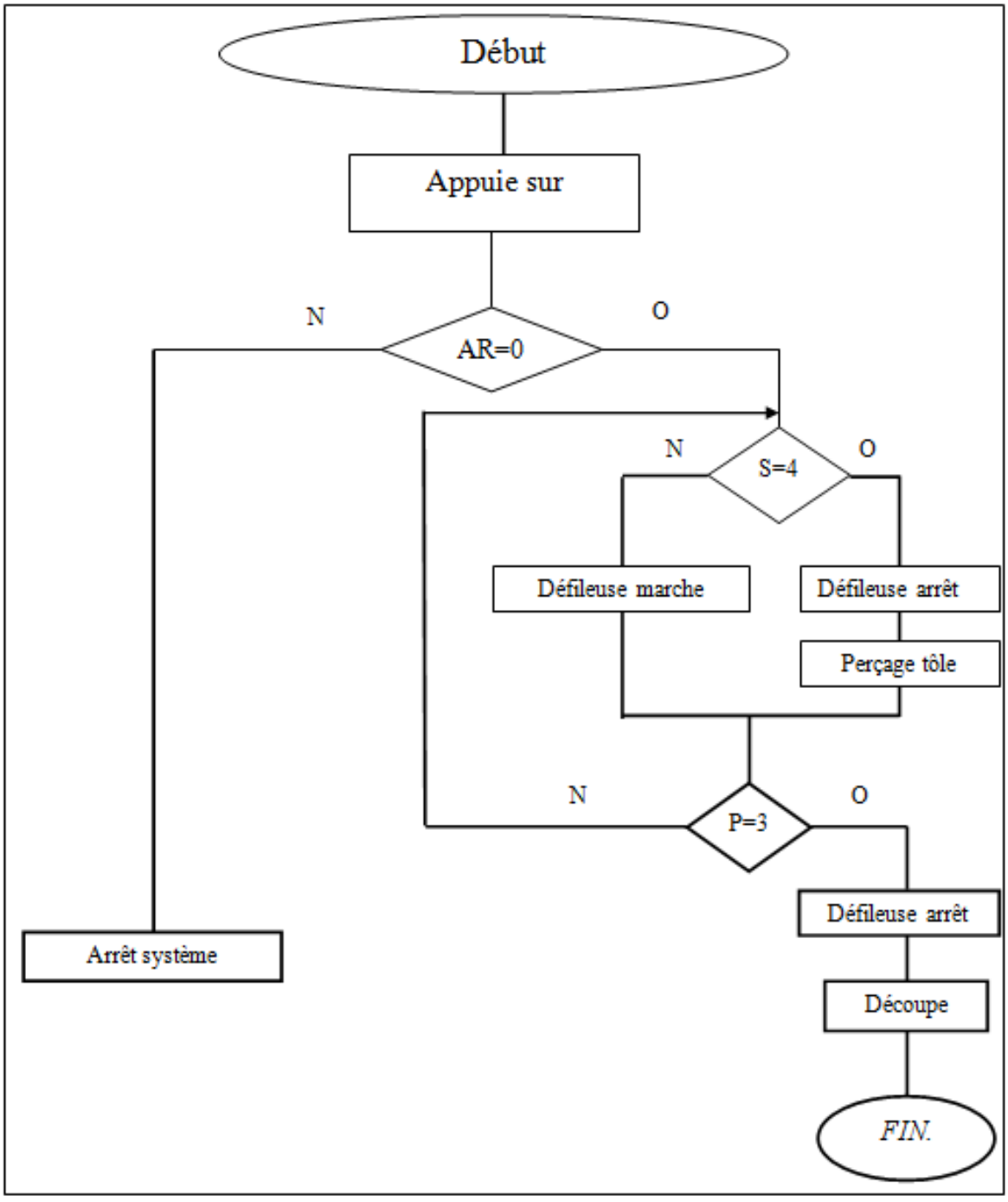
III.5.1.7) Tableau de mnémonique :

	mnémonique	adresse
1	Marche	I :0.0
2	Arrêt	I :0.1
3	Codeur	I :0.2
4	d-dérouleuse	I :0.3
5	A-dérouleuse	I :0.4
6	Dérouleur	Q :0.0
7	Défileuse	Q :0.1
8	Perforation	Q :0.2
9	découpe	Q :0.3

d-dérouleuse : démarrer dérouleuse

A-dérouleuse : arrêté dérouleuse

L'organigramme :



Algorithme de fonctionnement.

Avec :

D : bouton démarrage.

AR : bouton arrêt urgence.

S : nombre impulsion par le codeur optique.

P : nombre de perçage.

III.5.1.8) Description du programme :

Le programme à été fais pour géré la profileuse au début on appuie sur le bouton poussoir marche I :0.0 déflues de papier tourne est tire le papier après 1s, le compteur Q :0.1 commence à compte , la perforation Q :0.2 se fais puis il s'arrêt ; le temporisation T37 après 1s se sera le découpage Q :0.3

Conclusion :

Nous avons fabriqué et fait marcher ce prototype avec les moyens qu'ils ont été a notre disposition, et dans le plus part du cas de la récupération un programme ladder a été réalisé et adapté à nous composants pour automatiser le processus , nous avons remarqué les défaillances suivantes:

- La consommation considérable du courant électrique.
- Le système de découpage n'été pas convenable, car le papier ne ce découper pas.

Chapitre III
Conception
Et réalisation
Du prototype

Introduction générale

Le but de ce travail est de faire une réalisation d'un prototype de commande par automate d'une profileuse. Pour cela on doit bien comprendre le fonctionnement de la profileuse, et connaitre ses composants qui assurent son bon fonctionnement. Pour réaliser ce prototype nous devions choisir des petits mécanismes qui doivent accomplir et assurer approximativement les mêmes fonctions que celle de la profileuse réelle

La profileuse se compose de plusieurs composants (des moteurs, des axes de rotation, une armoire électrique, des buté, des vérins, des galets, des outils de coupes et de poinçonnage……), chaque composant assure une fonction bien précise.

Les importants composants de la profileuse sont, les deux vérins hydrauliques qui assurent le poinçonnage et le découpage de la tôle, deux moteurs, le premier pour le dévidoir, le deuxième est utilisé pour défiler la tôle sous la table à galet, un codeur qui calcule la position exacte de la tôle. Les principaux difficultés été de choisir des mécanismes qui assure le fonctionnement de ces composant, et le câblage nécessaire avec le S7-200 pour que la synchronisation soit compatible avec le passage de la feuille, le mouvement du poinçonnage et de la découpe.

La fabrication du prototype doit être réalisée dans un atelier qui contient le matériel et les matériaux nécessaire à ce travail, et un laboratoire d'automatique afin de réaliser la partie automatique.

Le plan de notre travail se décline donc comme suite. Le premier chapitre concerne l'étude générale sur les profileuses, cette étude va nous permettre de bien connaitre la profileuse, définir le profilage, avoir une vision sur son fonctionnement et les éléments qui la constituent.

L'automatisation se fait par un automate S7-200, dans le chapitre II. Cet automate à des caractéristiques, qui vont nous aider à bien commander notre prototype, les opérations arithmétiques et logiques, qui vont assurer le déroulement du programme de commande sont ainsi présenter.

Le développement et la réalisation de notre prototype, est divisé en deux parties. Le premier est la partie mécanique (les dessins des pièces nécessaires, et leurs dimensions). La deuxième est la partie électrique, les sous systèmes et leur composant est également reporté dans le chapitre III. La dernier chapitre est réservé pour l'automatisation du prototype, le câblage et le programme Ladder nécessaire, ainsi que l'algorithme qui explique le fonctionnement exacte de cet exemple.

Introduction générale

Conclusion générale

La réalisation de ce projet nous a permis d'explorer des divers domaines de la technologie d'une manière générale. Ceci étant donné que le travail demandé nécessite la fabrication de la profileuse d'une part et de la commander par un logiciel le travers une interface électronique qu'il faut concevoir.

L'utilisation des contacteurs nous a permis de faire les mouvements de perforation et de découpage et aussi nous avons utilisé un codeur optique qui mesure la longueur du papier.

Quant à la dérouleuse qui est indépendante, elle fait passer le papier. Ce projet nous l'avons réalisé grâce à notre logiciel.

Les résultats obtenus nous ont permis d'avoir un système de perforation à des distances voulues.

Les résultats de ce projet nous permettent d'envisager la conception d'autres systèmes basés sur les mêmes principes, telle que les machines de perforation pour les dessins sur les tôles.

Nous citons enfin que la réalisation de ce travail ne s'est pas effectuer sans problèmes tels que par exemple : absence de vérins qui coute très cher et aussi le codeur optique, l'alimentation du courant ou nous avons utilisé une batterie de voiture.

Finalement, on espère que ce projet soit un point de départ pour d'autres projets qui traitent la commande des systèmes automatiser.

Nomenclature :

N : Nombre de spires.

I : Intensité du courant [A].

L : Longueur de la bobine [m].

B : Champs magnétique tesla [T].

$\mu = 4 * \pi * 10^{-7}$: Perméabilité du vide [Tm/A].

n : Le nombre de tours de fil par unité de longueur [spires par mètre].

ϕ : Le flux magnétique [webers].

A : La section de solénoïde en [mètre carrées].

$d\phi_B / dt$: le taux de variation de flux magnétique.

dI / dt : le taux de variation du courant.

E : est la force électromotrice [V].

Ps = précision souhaitée en mm.

R = rapport de réduction entre l'engrenage du mouvement entraînant le codeur et le dernier engrenage entraînant le mobile.

P = rapport de conversion du mouvement de rotation en mouvement de translation.

Référence

[1] site web : WWW.comitaldu .com (05/05/2012)

[2] documentation se CERMO, Algérie 2012

[3] JEAN-JAQUES BERTRAND « profilage » BM 7574

[4]philipe lebrun ,automate programmable edition decembre 1999

[5] manuale d'utilisation du s7 200 , siemens 2008

[6] G. hatuios(roll forning haerd book, taylor francis,2006)