

MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE SCIENTIFIQUE  
UNIVERSITÉ MOULOUD MAMMÈRI, TIZI-OUZOU



FACULTE DE GENIE ELECTRIQUE ET DE L'INFORMATIQUE  
DEPARTEMENT D'ELECTRONIQUE

# Mémoire de fin d'études

Présenté en vue de l'obtention du Diplôme d'Ingénieur d'Etat  
en Electronique

Option : Contrôle

*Thème:*

**Automatisation de la bobineuse par un API S7-200  
et un panel tactile TP 177A**

**Proposé par:**

Mr : A.HARBIT (Electro-Industries)

**Encadré par :**

Mme : Z.AMIROU

**Etudié et réalisé par:**

Mr: AZZOUG Massinissa

Mr: BOUZEMBRAK Madjid

Année universitaire 2010/2011.

## Sommaire

Introduction générale.....	1
<b>Chapitre I : Description de la bobineuse</b>	
I. Présentation de l'organisme d'accueil .....	3
I.1. Présentation du complexe d'Electro-Industries .....	3
I.1.1 Unité de transformateurs .....	3
I.1.2 Unité moteurs, alternateurs et groupes électrogènes (UMAGE) .....	3
II. Présentation des différents blocs constituant la machine .....	4
II.1. Pupitre de commande .....	4
II.2. Panneau de commande .....	5
II.3. Unité de pose .....	6
II.4. Mandrin de bobinage .....	6
III. Parties constituant la machine .....	7
III.1. Partie commande .....	7
III.1.1. Pupitre de commande .....	7
III.1.2. Commande de la machine .....	7
III.2. Partie opérative .....	8
III.2.1. Pré actionneurs .....	8
a. Distributeurs pneumatiques .....	8
b. Les électrovannes .....	9
III.2.2. Les actionneurs .....	9
III.2.3 Les capteurs .....	11
IV. Description de l'armoire électrique .....	12
IV.1. Protection .....	13
IV.1.1. Sectionneurs .....	13
IV.1.2. Les fusibles .....	13
IV.1.3. Disjoncteurs .....	13
IV.1.4. Relais à minimum de tension .....	13
IV.2. Les contacteurs .....	13
V. Le transformateur galvanique (380V/220V) .....	14
VI. Le variateur de vitesse .....	14
VII. Encodeur rotatif .....	15
VII. Fonctionnement de la Bobineuse .....	15

Problématique .....	18
Objectifs .....	18
Conclusion .....	19

## **Chapitre I : Modélisation par GRAFCET**

I. Introduction .....	20
II. Définition .....	20
III. Concepts de base .....	20
III.1. Les étapes .....	21
III.2. Les actions associées à une étape .....	22
III.3. Les transitions et les réceptivités associées .....	22
III.3.1. Transitions .....	22
III.3.2. La réceptivité .....	22
III.3.3. Les liaisons orientées .....	23
IV. Les règles d'évolution du GRAFCET .....	23
V. Niveaux d'un GRAFCET .....	24
VI. GRAFCET de la bobineuse .....	24
VI.1. GRAFCET niveau 1 .....	24
VI.1. GRAFCET niveau 2 .....	29
Conclusion .....	35

## **Chapitre III : Description de l'automate programmable**

I. Introduction .....	36
II. Place de l'API dans le système automatisé .....	36
II.1. Structure d'un système automatisé .....	37
II.2. Domaines d'emploi des automates .....	38
II.3. Nature des informations traitées par l'automate .....	38
III. Architecture des automates .....	39
IV. Les objectifs de l'automatisation .....	41
V. Fonctions d'un automate programmable .....	41
VI. Présentation de l'automate programmable S7-200 .....	42
VII. Modules d'extensions .....	43
VIII. Choix de la CPU 224 .....	45

Conclusion .....	46
------------------	----

## **Chapitre IV : Automatisation**

I. Introduction .....	47
II. Fonctionnement du programme de commande .....	47
III. Accès aux données de l'API S7-200 .....	48
IV. Accès aux données dans les zones mémoire .....	49
IV.1. Mémoire image des entrées (I) .....	49
IV.2. Mémoire image des sorties (Q) .....	49
IV.3. Mémoire des variables (V) .....	49
IV.4. Mémentos (M) .....	49
IV. 5. Mémentos spéciaux (SM) .....	49
IV.6. Compteur (C) .....	50
IV.7. Accumulateurs (AC) .....	50
IV.8. Compteurs rapides (HC) .....	50
V. Fonctionnement des compteurs rapides .....	50
VI. Programmation des compteurs rapides .....	51
VI.1. Définition du mode de comptage .....	51
VI.2. Définition de l'octet de commande .....	53
VI.3. Définition de valeur en cours et de la valeur prédéfinie .....	54
VI.4. L'octet d'état .....	54
VI.5. Association du programme d'interruption à événements .....	54
VI.6. Validation du programme d'interruption .....	55
VI.7. Activation du compteur rapide .....	55
VI.8. Programme d'interruption (EG_COMPT) .....	55
VII. Organisation du programme de commande .....	55
VII.1. Programme principale : PPAL (OB1) .....	55
VII.2. Sous-programmes : SBR_i .....	56
VII.3. Programmes d'interruption : INT_i .....	56
VIII. Programmation de l'automate .....	56
VIII.1. Présentation du logiciel STEP7-MICRO/WIN .....	57
VIII.2. Création et enregistrement d'un programme sous STEP7 MICRO /WIN .....	58
VII.3. Choix de la CPU .....	59

VIII.4. Transfert du programme dans la CPU de l'automate .....	59
VIII.5. Mettre la CPU à l'état marche .....	60
Conclusion .....	61

## **Chapitre V : Configuration du panel tactile avec WinCC Flexible**

I. Introduction .....	62
II. WinCC flexible.....	62
II .1 Elements de WinCC flexible.....	62
II.1.1.WinCC flexible Engineering System.....	62
II.1.2. WinCC flexible Runtime.....	62
III. Critères de choix du panel tactile TP 177A .....	62
IV. Etapes de configuration .....	63
IV.1. Créer un projet .....	63
IV.1.1. Démarrer WinCC flexible .....	63
IV.1.2. Créer un projet vide .....	63
IV.1.3 .Sélectionner le pupitre .....	64
IV.2.Configuration de la liaison .....	65
IV.3. Création de vues .....	66
IV.4. Création de variables .....	67
IV.5. Ajouter des changements de vues .....	67
IV.6. Test et simulation du projet .....	69
Conclusion .....	72
Conclusion générale .....	73

L'histoire de l'automatisation est une science qui remonte à très loin. Dès la fin du XVIII<sup>e</sup> siècle, les processus de production ont été découpés en plusieurs étapes successives, ce qui a permis une croissance de la production. L'identification des tâches élémentaires qui composent tout cycle de production permet de construire des machines reproduisant les tâches correspondantes initialement réalisées par l'opérateur humain, ouvrant la voie à la constitution de chaînes de montage et d'assemblage.

Ces méthodes ont débutées dans l'industrie automobile, notamment dans les usines Ford aux Etats Unis avant d'être généralisées dans toute l'industrie. Ce qui a permis à Ford d'augmenter sa productivité.

L'introduction de l'informatique dans les processus de fabrication a considérablement accéléré le développement de l'automatisation. En effet, avec l'arrivée des ordinateurs et des machines à commande numérique, les mouvements sont enregistrés sur une unité de stockage, et peuvent accomplir plusieurs opérations d'usinage différentes. Si, en 1950, les API étaient encore peu courants, moins d'un demi-siècle plus tard, ce sont des millions d'API qui peuvent accomplir des tâches complexes : de contrôle, de pilotage de machines, de traitement de données, de circulation de l'information et de simulation. Ils sont utilisés à tous les niveaux de production (conception, production, contrôle). L'opérateur reste cependant maître des décisions, mais dispose ainsi à tout moment d'informations précises sur la production et les stocks.

Pour automatiser une machine, l'ingénieur doit étudier simultanément sa structure et son fonctionnement. L'aspect structurel d'une machine décrit l'organisation et la combinaison des différents éléments qui la compose. L'aspect fonctionnel présente les opérations qu'elle doit réaliser et la manière dont s'effectuent les échanges entre les différents éléments et leur environnement.

Les automates programmables prennent une place importante dans les systèmes de commande automatique des machines. Ils remplacent avantageusement les systèmes en logique câblée (à relais ou électroniques) dans la plupart des applications industrielles. Les fonctions d'automatisme sont programmées, ce qui permet d'adapter facilement l'application chargée dans la mémoire de l'automate aux conditions de fonctionnement de la machine.

Depuis l'ouverture de l'entreprise Electro-Industries d'AZAZGA en 1983, elle s'est spécialisée dans la fabrication et la commercialisation des moteurs électriques, alternateurs, transformateurs de distribution et groupes électrogènes. A son ouverture, cette entreprise s'est équipée de machines à commande électromagnétiques et de logique câblée. Avec l'évolution

de l'industrie, l'entreprise a évolué en s'équipant progressivement de machines automatisées et en automatisant les machines existantes pour remédier aux pannes fréquentes et améliorer sa productivité.

Dans ce contexte, les responsables de l'entreprise nous ont confié l'automatisation d'une machine bobineuse. Le cahier des charges contient la suppression de la logique câblée et des cartes électroniques et leur remplacement par un automate programmable, ainsi que le remplacement du pupitre de commande actuel par un écran tactile.

Pour réaliser cette tâche, nous avons étudié l'état actuel de la machine puis donné sa description à travers le premier chapitre.

Le fonctionnement automatique de la machine est modélisé par un GRAFCET que nous développons dans le deuxième chapitre. Ce GRAFCET est traduit en langage machine exécutable par l'automate S7-200 dont l'étude fait objet du troisième chapitre.

Les chapitres quatre et cinq sont respectivement consacrés à la programmation de l'automate et à la configuration du Panel tactile.

Le travail sera terminé par une conclusion générale.



## **I. Présentation de l'organe d'accueil**

Electro-Industries est spécialisé dans la fabrication et la commercialisation des moteurs électriques, alternateurs, transformateurs de distribution et groupes électrogènes.

### **I.1. Présentation du complexe d'Electro-Industries**

L'entreprise est divisée en trois unités :

- Unité de fabrication des transformateurs.
- Unité de fabrication de moteurs électriques, alternateurs et groupes électrogènes.
- Unité prestation technique.

#### **I.1.1 Unité de transformateurs**

Elle produit des transformateurs de puissance variant entre 50 et 2000 KVA avec une capacité de 5000 transformateurs par an.

Cette unité dispose de l'ensemble des équipements nécessaires à la fabrication et aux essais des produits finis.

##### **Transformateurs de distribution**

- Puissance : 50 à 2000 KVA
- Tension primaire : 5,5 -10-30 KV
- Tension secondaire : 400 V

#### **I.1.2. Unité moteurs, alternateurs et groupes électrogènes (UMAGE)**

Elle vient en second place après l'unité de transformateurs en matière d'activité de l'Electro-industries.

Sa capacité de production est de :

- 45000 moteurs de 0,25 à 15 KW.
- 4500 moteurs de 1,5 à 40 KW.
- 2000 alternateurs de 16 à 180 KW.
- Montage de 400 groupes électrogènes (22, 35, 52 KV).

Les produits de cette unité sont : Alternateurs, groupes électrogènes et les moteurs.

##### **Alternateurs synchrones**

- Puissance : 17,5 à 200 KVA
- Tension : 380 V
- Fréquence : 50HZ

## **II. Présentation des différents blocs constituant la machine**

La bobineuse est une machine industrielle qui réalise les bobinages des machines tournantes (machines à courant alternatif, machines à courant continu) et des transformateurs. Le bobinage est l'enroulement d'un fil de cuivre autour d'un circuit magnétique.

Elle se compose de différentes unités, on a :

- Le pupitre de commande.
- Panneau de commande.
- Unité de pose.
- Le mandrin.

### **II.1. Pupitre de commande**

Il contient les boutons de commandes suivants :

- Sélecteur de mode : choisir le mode de fonctionnement de la machine Automatique ou Ajustage.
- Bouton Start : démarre le cycle automatique.
- Bouton d'arrêt d'urgence.
- Bouton verrou en avant.
- Bouton retour du verrou.
- Bouton avance de l'unité de pose.
- Bouton retour de l'unité de pose.
- Bouton vérin en avant.
- Bouton retour du vérin.
- Bouton mâchoire de bobinage ventiler (Serrer).
- Bouton mâchoire de bobinage tendre (Desserrer).
- Bouton buse de bobinage en haut.
- Bouton buse de bobinage en bas.
- Bouton porte de protection ouverte.
- Bouton porte de protection fermer.
- Bouton pince d'entraînement en haut.
- Bouton pince d'entraînement en bas.
- Bouton empivoter la pince.
- Bouton dé pivoter la pince.

- Bouton fermer la pince.
- Bouton ouvrir la pince.
- Bouton changement ou coupure d'un fil.
- Potentiomètre pour régler la vitesse d'avance de l'unité de pose.

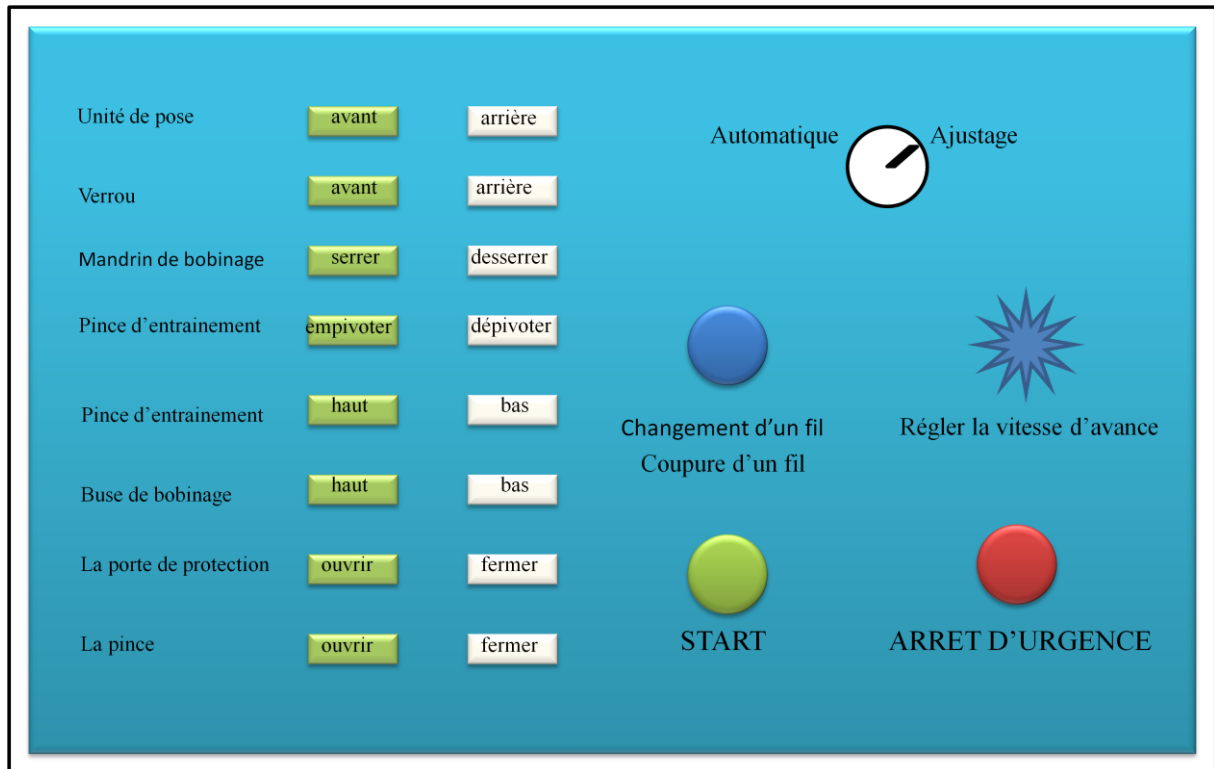
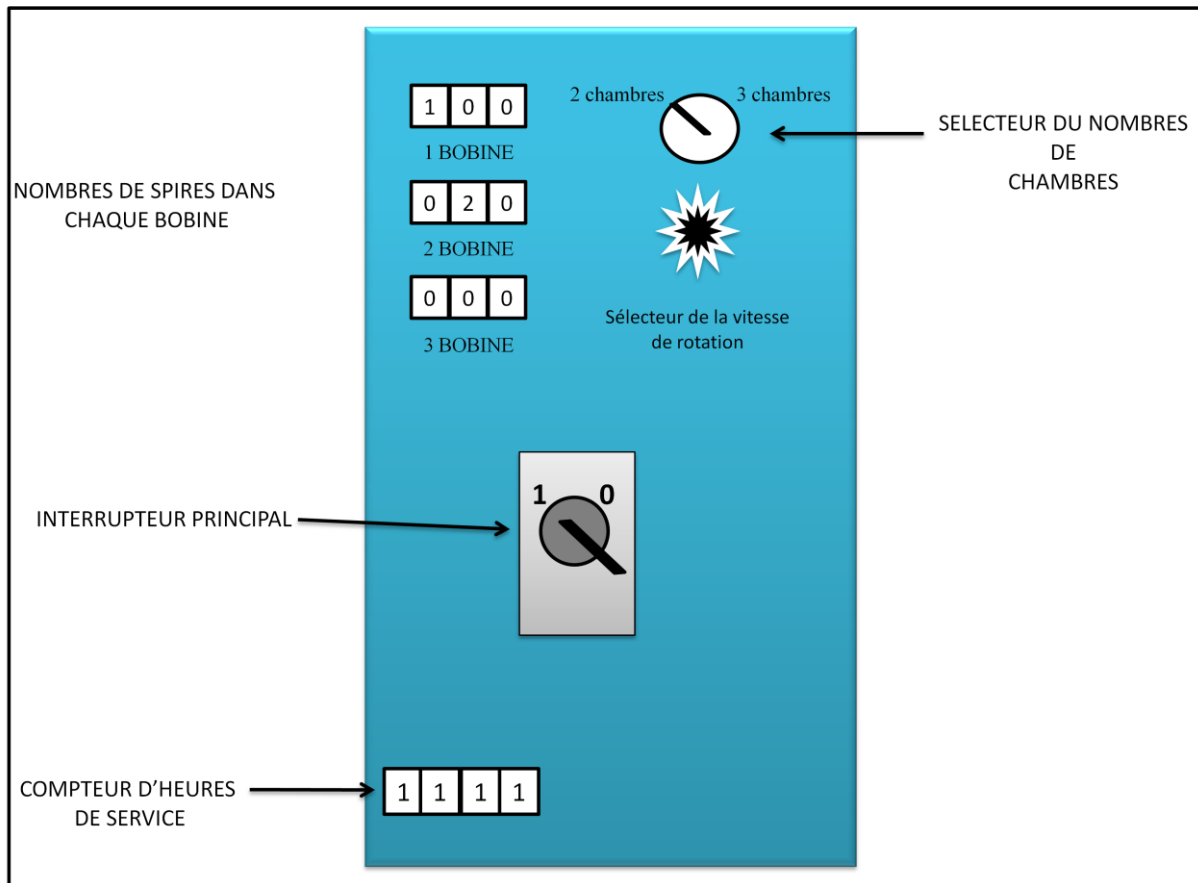


Figure 1. Pupitre de commande.

## II.2. Panneau de commande

Sur la face arrière le panneau contient les circuits d'alimentation, de commande et des cartes électroniques. Sur la face avant, sont fixés : un commutateur principal pour la mise en marche, trois compteurs pour donner le nombre de spires, un potentiomètre pour régler la vitesse de rotation et un sélecteur de bobine.



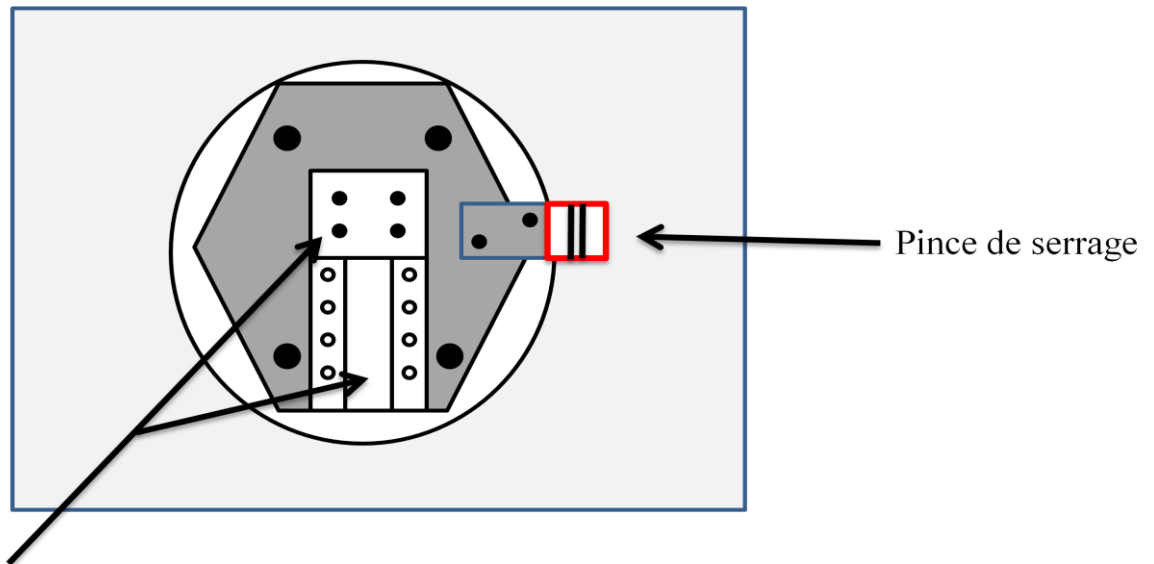
**Figure 2.** Panneau de commande.

### II.3. Unité de pose

L'unité de pose est une unité importante dans la machine, elle comporte la buse principale, la buse de bobinage et elle a sept positions pour se déplacer (de A1 jusqu'à A7). Le fil passe de la buse de bobinage vers la buse principale, cette dernière comporte la pince d'entraînement qui sert à couper le fil et à le ramener vers le mandrin de bobinage (pince de serrage).

### II.4. Mandrin de bobinage

Le bobinage va s'effectuer sur les mâchoires qui vont être fixées sur le mandrin de (figure 3) selon le diamètre de la bobine qu'on veut réaliser. Le fil en cuivre sera coincé dans la pince de serrage afin que le bobinage se fasse sur les mâchoires lorsque le mandrin est en rotation.



Fixation des deux mâchoires de bobinage

**Figure 3.** Schéma fonctionnel du mandrin de bobinage.

### III. Parties constituant la machine

La machine est essentiellement composée de deux parties, une partie pour commander la machine et une autre partie pour exécuter les commandes.

#### III.1. Partie commande

##### III.1.1. Pupitre de commande

Le pupitre de commande nous offre les moyens d'intervenir sur la machine en cas de problème en actionnant le bouton d'arrêt d'urgence, et d'effectuer les réglages nécessaires pour le bon fonctionnement de la machine avant son démarrage. L'opérateur a besoin de savoir si les fonctions de la machine s'effectuent correctement.

##### III.1.2. Commande de la machine

La bobineuse fonctionne en deux modes, manuel et automatique :

- **Le mode manuel :** le mode manuel est choisi en tournant la clé du sélecteur en mode « Ajustage ». L'opérateur fait les réglages nécessaires ou les conditions initiales avant le démarrage de la machine. Ce mode est aussi choisi en cas de rupture du fil.

- **Le mode automatique :** le mode automatique est choisi en tournant la clé du sélecteur en mode « Automatique ». Le cycle de production s'effectue automatiquement et ne s'arrête que s'il y'a une rupture du fil ou en actionnant le bouton STOP cycle automatique ou un arrêt d'urgence.

### III.2. Partie opérative

Elle comporte le processus à commander. Elle contient :

- Les pré-actionneurs.
- Les actionneurs.
- Les capteurs.

#### III.2.1. Pré actionneurs

Un pré-actionneur est un constituant gérant l'énergie de commande afin de la distribuer vers les actionneurs. Dans ce qui suit, seront décrits les distributeurs pneumatiques et les électrovannes « Tout Ou Rien ».

##### a. Distributeurs pneumatiques

Leur rôle essentiel est de distribuer de l'air dans les canalisations qui vont arriver dans les chambres des vérins. Les distributeurs sont des pré-actionneurs qui permettent de commander les vérins pneumatiques.

La bobineuse comprend des vérins a double effet, comporte donc deux orifices qu'il faut alterner, les états de pression et d'échappement. Elle contient également les distributeurs 3/2, 4/2 et 5/2.

**Distributeur 3/2 :** c'est un corps muni de trois orifices pour le circuit de commande (un vers le vérin, un vers l'échappement, un vers la source de pression) et deux orifices pour la commande. Sa représentation schématique est la suivante :

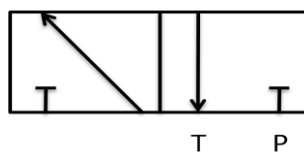


Figure 4. Distributeur 3/2.

**Distributeur 4/2 :** c'est un corps muni de quatre orifices pour le circuit de commande (deux vers le vérin, un vers l'échappement, un vers la source de pression). Et deux orifices pour la commande. Sa représentation schématique est la suivante :

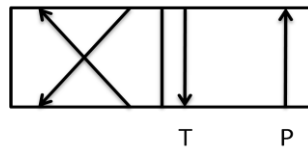


Figure 5. Distributeur 4/2.

**Distributeur 5/2 :** c'est un corps muni de Cinq orifices pour le circuit de puissance (deux vers le vérin, un vers échappement 1, un vers échappement 2, un vers la source de pression) et deux positions. Sa représentation schématique est la suivante :

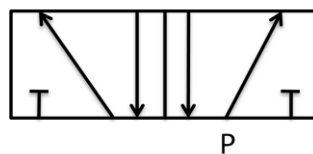


Figure 6. Distributeur 5/2.

## b. Les électrovannes [9]

Le traitement des informations se fera par automate. Tous les distributeurs sont équipés d'une ou de deux électrovannes, leur rôle principale est de transformer le signal électrique en un signal pneumatique.

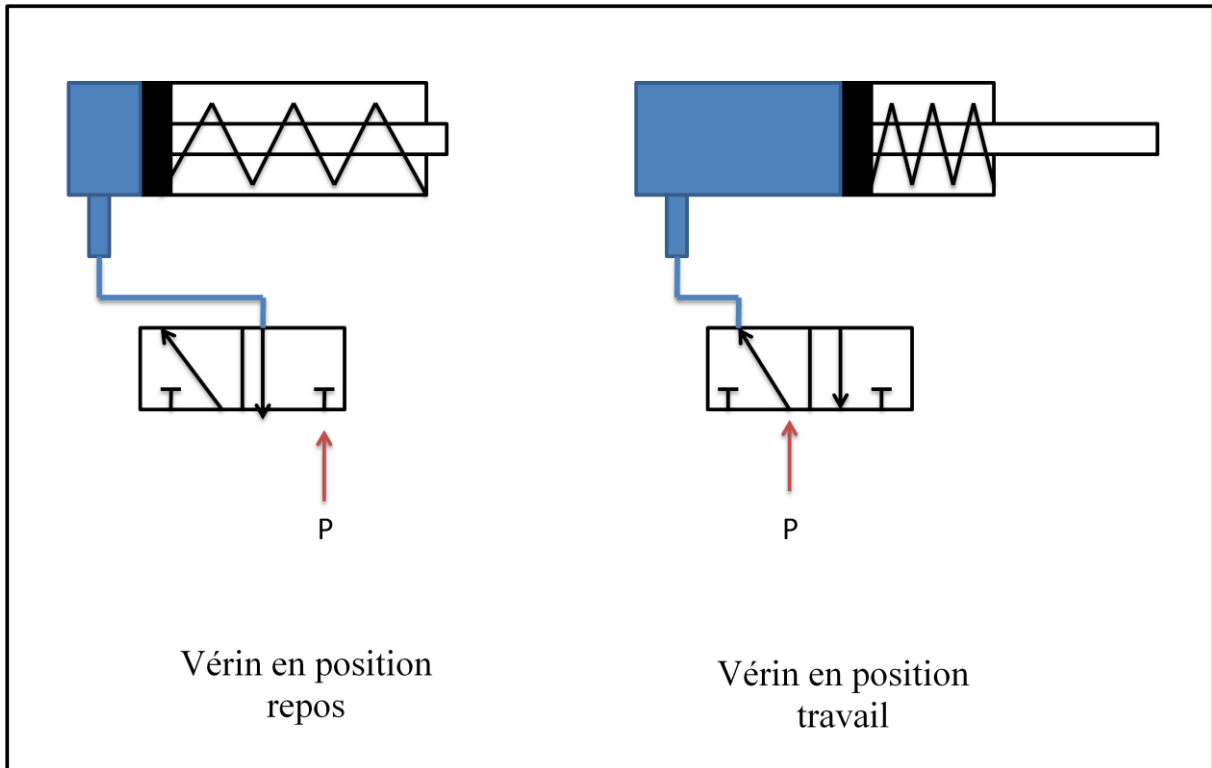
L'électrovanne ou l'électrovalve est un dispositif commandé électriquement. Il permet d'autoriser ou d'interrompre par action mécanique la circulation d'un fluide ou d'un gaz dans le circuit. L'électrovanne TOR à deux positions seulement, soit elle s'ouvre entièrement soit elle se ferme entièrement.

### III.2.2. Les actionneurs

Ils convertissent l'énergie pneumatique en énergie mécanique de translation. Parmi les actionneurs, on citera les vérins pneumatiques à simple effet et les vérins pneumatiques à double effet.

- **Vérin pneumatique simple effet**

L'ensemble tige-piston se déplace dans un seul sens sous l'action du fluide sous pression et le retour est effectué par un ressort de rappel. Un vérin simple effet ne comporte qu'un seul orifice. Il est cependant très peu utilisé dans les Systèmes Automatisés. On lui préfère le vérin double effet.



**Figure 7.** Vérin simple effet.

- **Vérin pneumatique double effet**

L'ensemble tige-piston peut se déplacer dans les deux sens sous l'action du fluide sous pression (air comprimé). Un vérin double effet comporte deux orifices sur lesquels il faut alterner les états de pressions et d'échappement, le distributeur utilisé doit comporter deux orifices de sorties.

✚ Ce vérin est utilisé dans notre machine pour quelques fonctions qui sont :

- Mettre le verrou en avant ou en arrière.
- Serrer ou desserrer les mâchoires de bobinage.

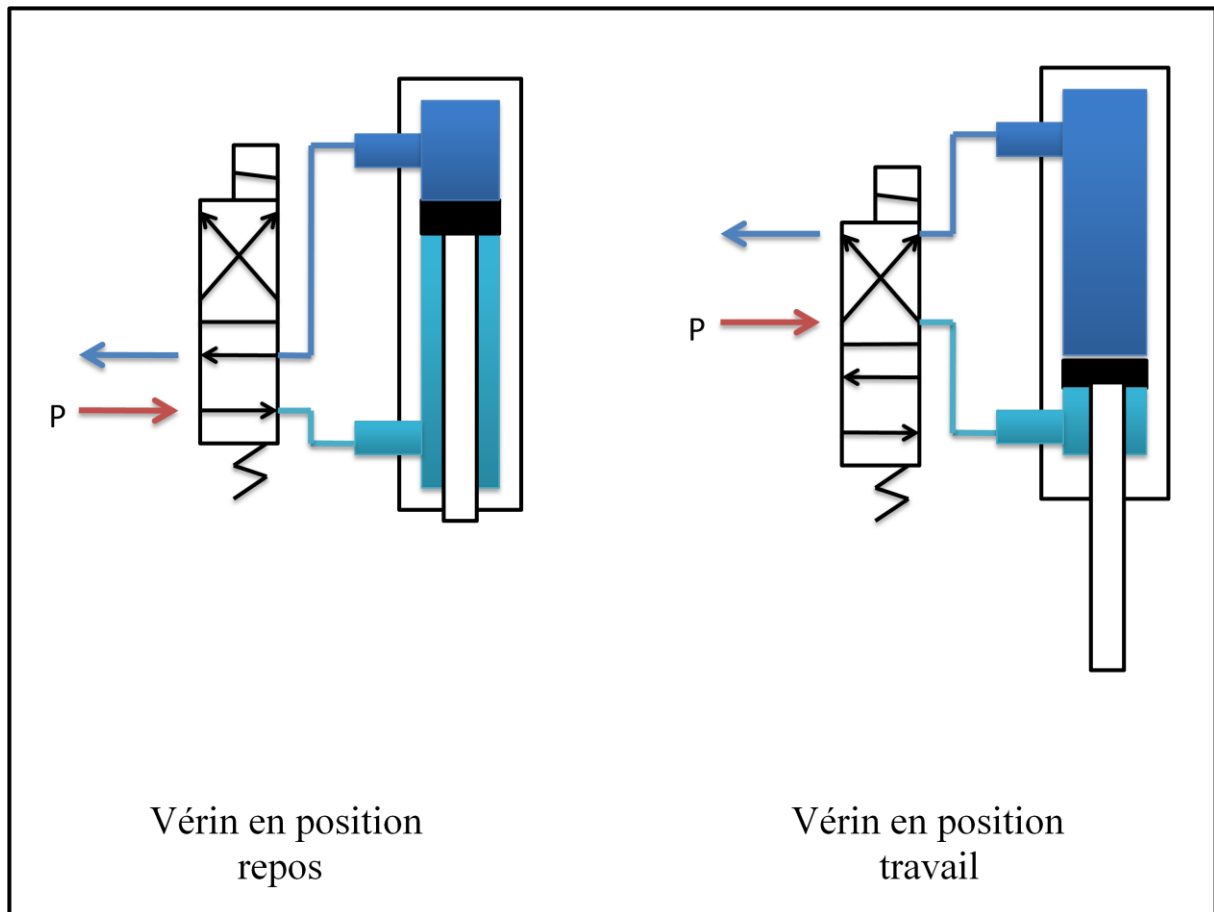


Figure 8. Vérin double effet.

### III.2.3. Les capteurs

Un capteur est un organe de prélèvement de l'information qui délivre sous forme d'une grandeur physique une image de la grandeur mesurée, afin de commander les actionneurs et pré-actionneurs. On peut classer les capteurs selon la nature de la sortie délivrée en trois groupes : analogique, logique (TOR) ou numérique.

Sur la bobineuse, les capteurs existants sont de type TOR.

#### Capteur TOR (Tout Ou Rien)

Un capteur TOR délivre généralement une information qui est directement exploitable. C'est une information logique (0 ou 1) sous forme d'une tension avec deux niveaux qui doivent être compatibles avec l'unité de traitement.

➤ **Capteur de proximité**

• **Détecteur de proximité inductif**

Un détecteur inductif détecte exclusivement les objets métalliques qui se trouvent à proximité de la tête de détection. Tous les détecteurs de proximité inductifs sont équipés d'une diode électroluminescente de signalisation de l'état de sortie.



**Figure 9.** Capteur de proximité inductif [13].

 **Principe de fonctionnement**

Un détecteur inductif se compose essentiellement d'un oscillateur. Le champ électromagnétique est créé à l'avant de la face sensible. Toute pièce métallique pénétrant dans ce champ devient le siège de courants de Foucault, qui provoquent l'arrêt des oscillations. C'est l'arrêt des oscillations qui est détecté.

#### **IV. Description de l'armoire électrique**

L'armoire électrique est une enceinte qui comporte tous les accessoires de protection et de commande des mécanismes électriques (moteurs...).

## IV.1. Protection

Parmi les organes de protection on trouve :

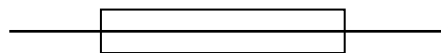
- Sectionneurs.
- Les fusibles.
- Disjoncteurs.

### IV.1.1. Sectionneurs [11]

Le sectionneur est destiné à fermer ou à ouvrir un circuit électrique. Il permet d'isoler électriquement une installation ou un circuit électrique. Il est équipé généralement de fusibles (protection contre les courts-circuits) et d'un dispositif de verrouillage pour une éventuelle consignation comme les opérations de maintenance, de dépannage ou de modification sur les circuits électriques.

### IV.1.2. Les fusibles [10]

La protection par fusible consiste à insérer dans le circuit à protéger, un élément faible (le moins résistant à la chaleur). En cas de surtension, le fusible est détruit afin d'ouvrir le circuit et ainsi de protéger le reste des composants.



**Figure 10.** Symbolisation d'un Fusible.

### IV.1.3. Disjoncteurs [10]

Un disjoncteur est un organe électromécanique, voire électronique, de protection, dont la fonction est de couper le courant électrique en cas d'incident sur un circuit électrique. Il peut interrompre un courant de surcharge ou un courant de court-circuit dans une installation.

## IV.2. Les contacteurs [11]

Les contacteurs électromagnétiques sont des appareils de commande pouvant établir ou interrompre le passage de l'énergie électrique. Ils assurent la fonction de commutation.

Dans la technologie des systèmes automatisés les contacteurs sont appelés pré-actionneurs puisqu'ils se trouvent avant les actionneurs.

## Fonctionnement

Lorsque la bobine se trouvant dans le circuit de commande et parcourue par un courant électrique (bobine excitée), le circuit magnétique tire les contacts de puissance et de commande, établissant ainsi la liaison électrique avec le moteur.

Lorsque les bobines ne sont plus parcourues par un courant (bobine dés excitée), les contacteurs s'ouvrent sous l'action du ressort de rappel.

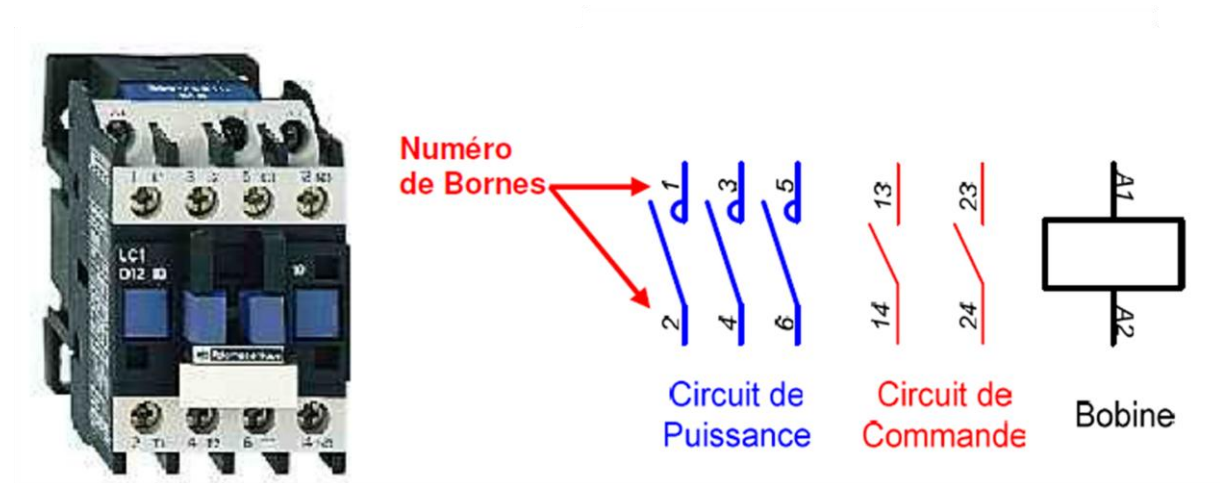


Figure 11. Contacteurs.

## V. Le transformateur galvanique (380V/220V) [9]

Le transformateur d'isolement est uniquement destiné à créer un isolement électrique entre plusieurs circuits pour des raisons de sécurité ou de résolution de problèmes techniques.

- ✚ Dans notre machine le transformateur galvanique a été utilisé pour isoler électriquement le circuit de commande du circuit de puissance.

## VI. Le variateur de vitesse

Le variateur de vitesse de la machine est une carte électronique. Elle est utilisée pour régler la vitesse du moteur à l'aide d'un potentiomètre.

## VII. Encodeur rotatif

Le contrôle du déplacement et de la position d'un mobile est un problème couramment rencontré sur un grand nombre de systèmes automatisés. Pour résoudre ces problèmes il existe différents types de capteurs :

- Les capteurs simples (résistif, capacitif, inductif).
- Les codeurs rotatifs (absolue, incrémental).

✚ Dans notre projet nous allons nous intéresser au codeur incrémental qui est un élément très important.

L'encodeur rotatif ou codeur incrémental est un générateur d'impulsions, il est destiné au contrôle du déplacement d'un mobile par comptage et décomptage des impulsions qu'il délivre. Le codeur incrémental est un capteur de position angulaire.

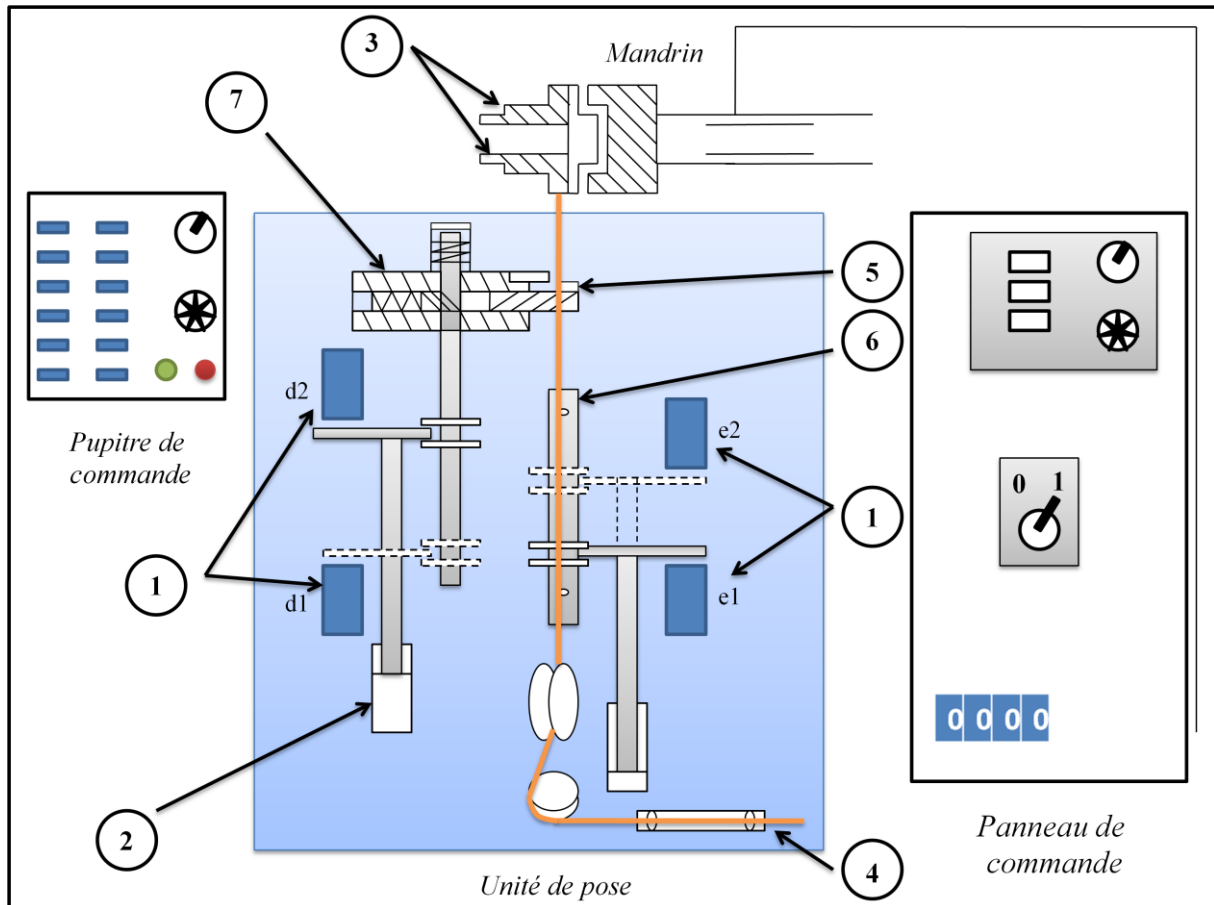
L'encodeur de notre machine est associé à l'arbre de rotation d'un moteur à courant continu ; quand ce dernier tourne, l'encodeur délivre des impulsions qui vont être traitées par des cartes électroniques liées à l'encodeur.



**Figure 12.** Codeur incrémental.

## VIII. Fonctionnement de la Bobineuse

Avant que la machine ne soit mise en marche, faire passer le fil de cuivre à bobiner par le casse-fil et les guides flexibles au tendeur, le bout du fil est guidé vers le haut en passant par la poulie de renvoi et à travers la buse principale.



1. d1,d2,e1,e2 : Capteurs inductifs.
2. Vérins.
3. Mâchoires de bobinage.
4. Fil en cuivre.
5. Pince.
6. Buse de bobinage.
7. Buse principale.

**Figure 13.** Le schéma fonctionnel de la bobineuse.

La pince coupe le fil et ce dernier reste coincé dans la pince. On fait monter la buse principale à la hauteur voulue. La buse principale se situe sur l'unité de pose et assure ainsi un parfait guidage du fil durant le déroulement de bobinage.

Le système de pinçage est disposé parallèlement au guide du fil. Son rôle consiste à coincer le bout du fil de la bobine achevée, de la couper et d'amener le nouveau bout du fil au mandrin.

En actionnant le bouton de desserrage du mandrin « UNITE PIVOTANTE VENTILER » la griffe latérale (pince de serrage) s'ouvre, on y introduit le bout du fil. Le bouton « TENDRE » peut maintenant être actionné. La griffe de serrage se ferme, le bout du fil est fixé au mandrin.

Au pupitre de commande on commute de la position « Ajustage » vers la position « Automatique », la machine est maintenant prête à fonctionner et peut être mise en marche.

Après avoir actionné le bouton « START » le bobinage démarre, il s'arrête selon le nombre de chambre sélectionnées deux ou trois. Le cycle étant terminé (bobine terminée), la buse inférieure se déplace vers le bas, la buse principale du système d'alimentation du fil est dépivotée et se déplace par la suite à la hauteur du couteau ou il sera cisailé. L'opérateur supervisant la machine actionne le bouton « MACHOIRES DE BOBINAGE VENTILER » provoquant ainsi le desserrage des mâchoires et le retrait de la bobine au moyen d'un adaptateur puis son transfert à la machine d'insertion.

La vitesse de bobinage ainsi que le nombre de spires par bobine sont à sélectionner par les éléments de commande. Les compteurs correspondants sont incorporés dans l'armoire électrique.

Après le retrait de la bobine des gabarits, le fil reste coincé dans la pince. Un nouveau cycle automatique peut alors être entamé.

## Problématique

Après une étude approfondie sur le fonctionnement et la commande de la bobineuse, on a pu détecter plusieurs insuffisances:

- Difficulté de maintenance en cas de pannes.
  - Pannes assez fréquentes de cartes électroniques et leur non disponibilité sur le marché.
  - Cartes électroniques trop volumineuses.
  - La logique câblée (bascules, compteurs, portes logiques...) est une technique figée, la modification de circuits est quasiment impossible.
  - Le nombre de chambre de bobinage est fixe (deux ou trois).
- ✚ Afin de remédier à ces insuffisances et aux problèmes rencontrés auparavant, nous avons opté pour l'automatisation de la bobineuse avec un automate programmable industriel (API) et la commande par un écran tactile 177A pour mieux suivre le fonctionnement de la machine en nous fixant les objectifs suivants :

## Objectifs

- ❖ Sur l'armoire de commande, toute la logique câblée ainsi que les cartes électroniques reliées à l'encodeur seront supprimé.
- ❖ L'alimentation des différents capteurs et des électrovannes se fera directement à partir de l'automate, ce dernier va commander aussi les bobines des contacteurs de manière à ouvrir ou fermer les contacts de puissance.
- ❖ La machine peut faire une ou deux ou trois chambres de bobines.
- ❖ On peut savoir à n'importe quel moment le nombre de spires effectué grâce au compteur.
- ❖ Création de plusieurs menus.

**Conclusion**

Dans ce chapitre nous avons passé en revue la description de la bobineuse ainsi que son fonctionnement. A partir de la, nous avons posé une problématique ou les principales insuffisances de la machine ont été notées. Ces insuffisances nous ont amenés à nous fixer quelques objectifs dont les détails seront développés dans les prochains blocs.



## **I. Introduction**

Le GRAFCET (graphe de commande étapes-transitions) est un outil graphique de représentation du cahier des charges d'un automatisme séquentiel. Il est à la fois simple à utiliser et rigoureux sur le plan formel.

Il décrit les ordres émis par la partie commande vers la partie opérative en mettant en évidence les actions engendrées et les événements qui les déclenchent. Cette représentation est étroitement liée à la notion d'évolution du processus.

## **II. Définition**

Le **GRAFCET (GRAF de Commande Etapes Transitions)**, est un langage graphique qui sert à décrire, étudier, et réaliser le fonctionnement des automatismes séquentiels, en tout ou rien, et cela en suivant un cahier des charges. Il est simple à utiliser et rigoureux sur le plan formel. Le GRAFCET est devenu plus qu'un outil de description, mais un langage de programmation graphique.

Le cahier des charges est le descriptif littéral fourni par l'utilisateur au concepteur de l'automatisme pour lui indiquer les différents modes de marches et les sécurités que devra posséder l'automatisme. Il sert à décrire le comportement de la partie opérative par rapport à la partie commande.

## **III. Concepts de base**

Le GRAFCET se compose d'un ensemble :

1. D'étapes auxquelles sont associées des actions (activités).
2. De transitions auxquelles sont associées des réceptivités.
3. Des liaisons orientées reliant les étapes aux transitions et les transitions aux étapes.

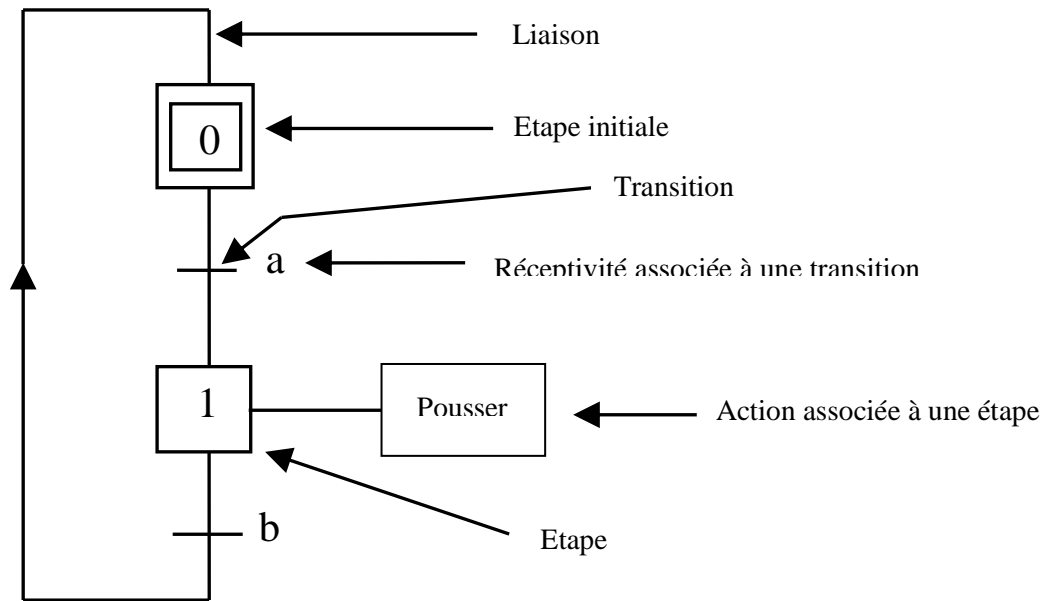


Figure 1. Symbolisation du GRAFCET.

### III.1. Les étapes

Une étape caractérise une situation dans laquelle le comportement d’une partie ou la totalité de l’automatisme est invariant (stable) : pendant une étape, les organes de commande et les capteurs ne changent pas d’états.

- ✚ Une étape simple est représentée par un rectangle numéroté tandis que l’étape initiale est représentée par un double rectangle. L’étape initiale précise les étapes actives au début du fonctionnement.

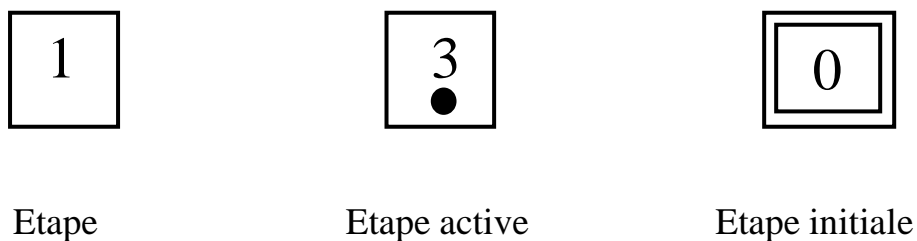


Figure 2. Différentes étapes du GRAFCET.

### III.2. Les actions associées à une étape

A chaque étape sont associées, en général, une ou plusieurs actions élémentaires ou complexes. Ces actions réalisent des travaux à chaque fois que l'étape à laquelle elles sont associées, est active. Elles sont inscrites de façon littérale ou symbolique dans un rectangle latéral relié à l'étape par un trait.

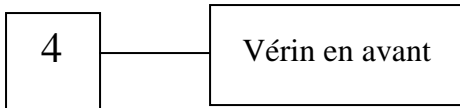


Figure 3. Etape à une seule action.

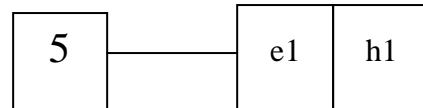


Figure 4. Etape à plusieurs actions.

### III.3. Les transitions et les réceptivités associées

#### III.3.1. Transitions

Une transition indique la possibilité d'évolution d'une étape à une autre. Cette évolution est accomplie par le franchissement de la transition.

Le franchissement de la transition consiste à désactiver l'étape précédente (reliée à la transition) et d'activer l'étape suivante.

#### II.3.2. La réceptivité

La réceptivité est la condition logique pour l'évolution du GRAFCET. Si la réceptivité est vrai (=1) le cycle peut évoluer. Les réceptivités proviennent du pupitre de commande, des fins de courses ou d'information provenant de la partie opérative.

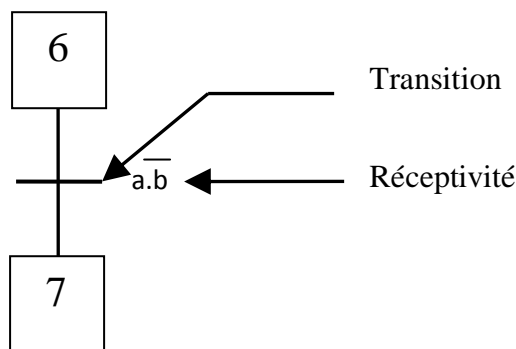


Figure 5. Réceptivité associée à l'étape 6.

### III.3.3. Les liaisons orientées

Une liaison orientée relie soit une ou plusieurs étapes à une transition, soit une transition à une ou plusieurs étapes. Les liaisons sont horizontales ou verticales et servent à indiquer les voies d'évolution du GRAFCET. Dans le cas général, les liaisons qui se font du haut vers le bas ne comportent pas de flèches, mais dans les autres cas, il faut utiliser les flèches. De plus, pour éviter toute ambiguïté, il est préférable d'éviter les croisements continus des lignes de liaison.

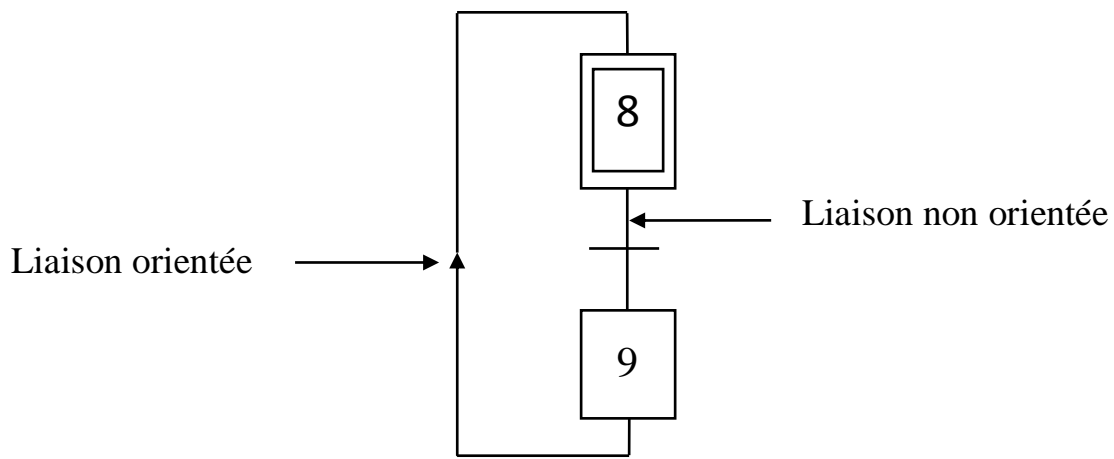


Figure 6. Liaisons orientées et non orientées.

## IV. Les règles d'évolution du GRAFCET

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

### Règle 1 : Situation initiale

L'ensemble des étapes actives au début de fonctionnement est l'ensemble des étapes initiales. C'est la situation initiale, elle correspond aux étapes actives à la mise en énergie du système de commande de la machine. Elles sont souvent des étapes d'attente pour ne pas effectuer une action dangereuse.

**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 lorsque la transition est validée et que la réceptivité associée à cette transition est vraie.

**Règle 3 : Evolution des étapes actives**

Le franchissement d'une transition entraîne simultanément l'activation de toutes les étapes de sortie de la transition et la désactivation de toutes les étapes d'entrée.

**Règle 4 : Evolution simultané**

Plusieurs transitions simultanément franchissables sont simultanément franchies.

**Règle 5 : Activation et désactivation simultané d'une même étape**

Si, au cours du fonctionnement, une étape active est simultanément activée et désactivée, alors elle reste active.

**V. Niveaux d'un GRAFCET**

Le GRAFCET est représenté selon deux niveaux :

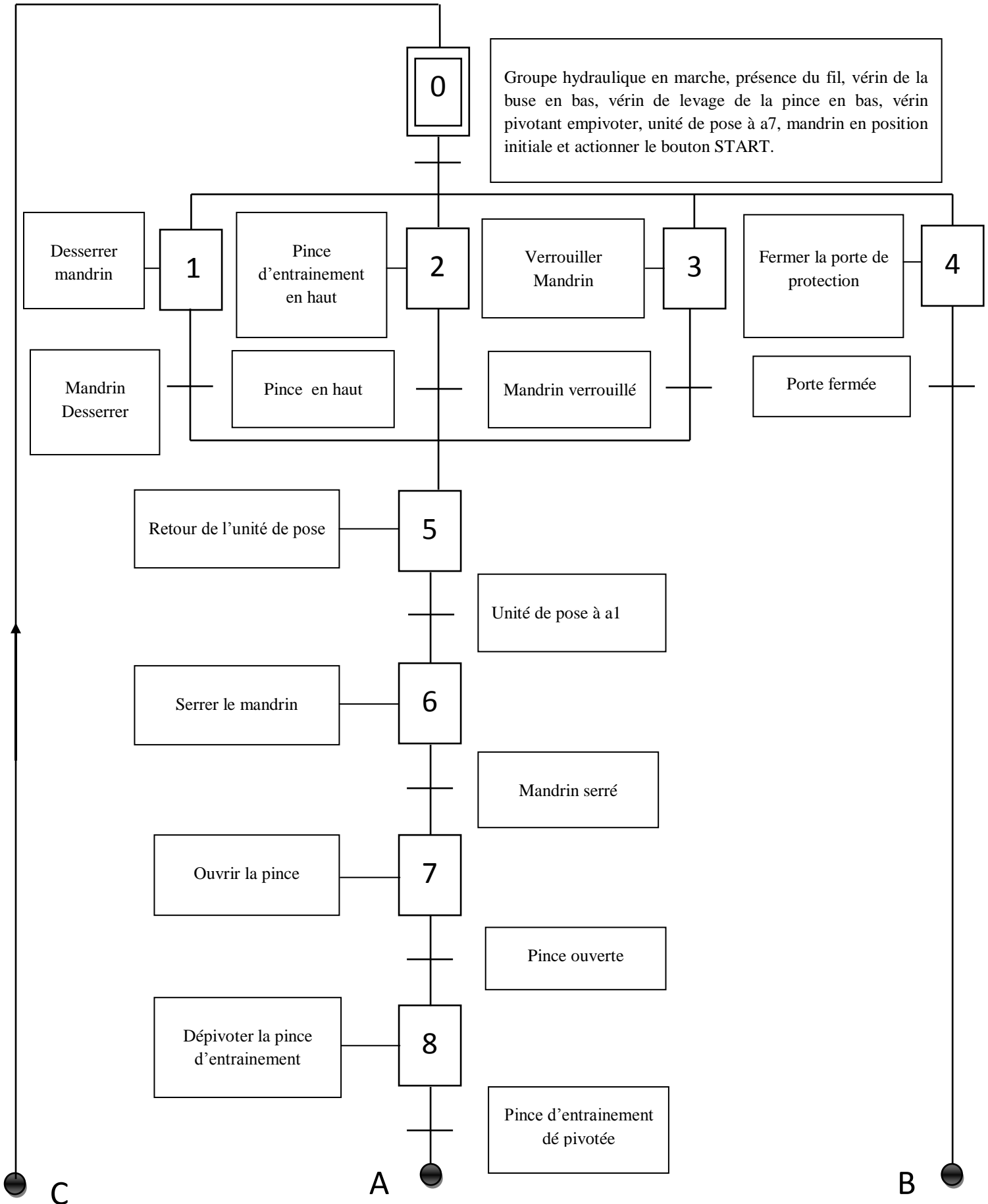
**GRAFCET de niveau 1 : Spécifications fonctionnelles**

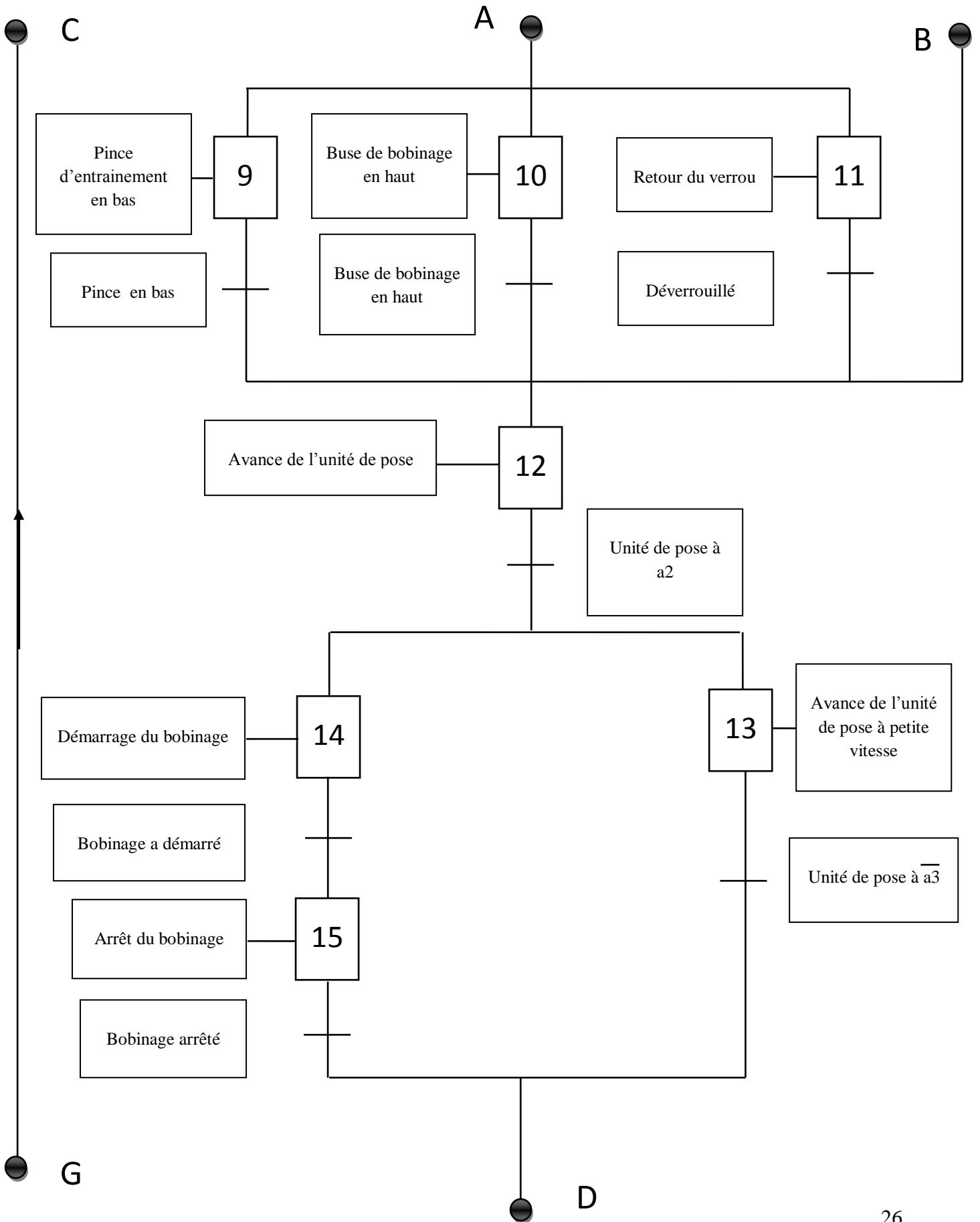
- Représentation de la séquence de fonctionnement de l'automatisme sans se soucier de la technologie des actionneurs et des capteurs.
- Description littérale des actions et de la séquence de l'automatisme.

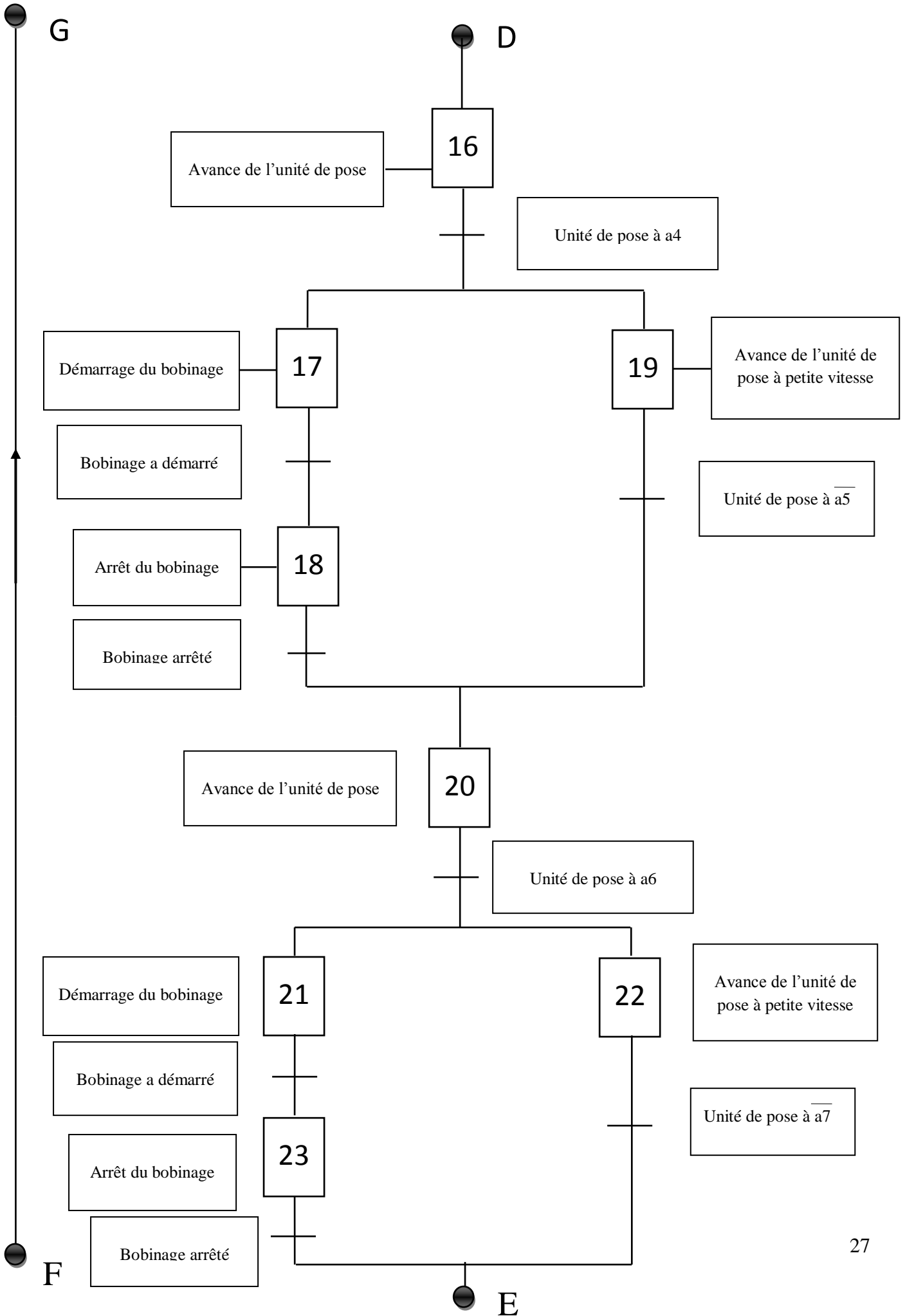
**GRAFCET de niveau 2 : Spécifications technologiques**

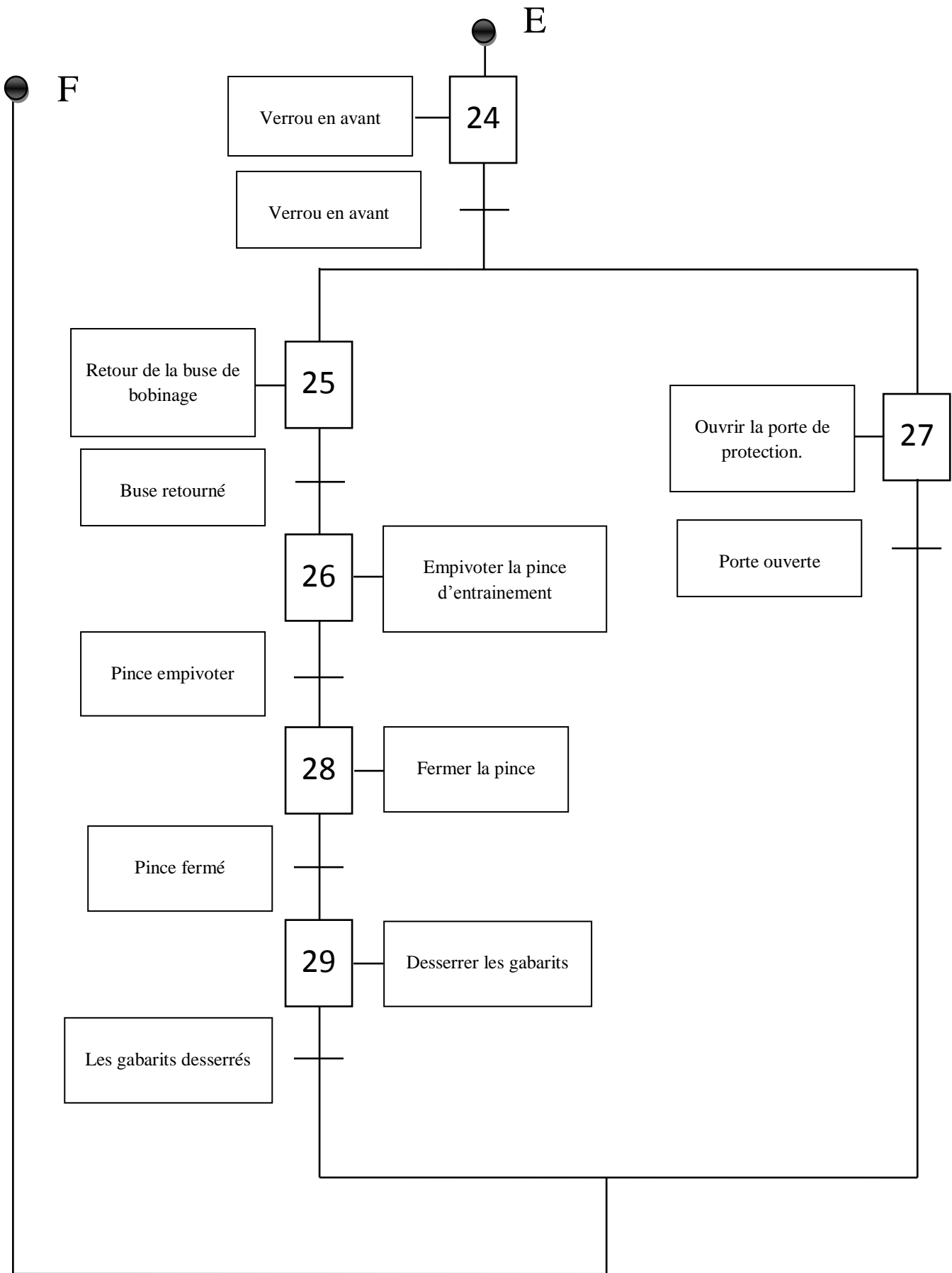
- Prise en compte de la technologie des actionneurs et des capteurs.
- Description symbolique des actions et de la séquence de l'automatisme.

**VI. GRAFCET de la bobineuse****VI.1. GRAFCET niveau 1**

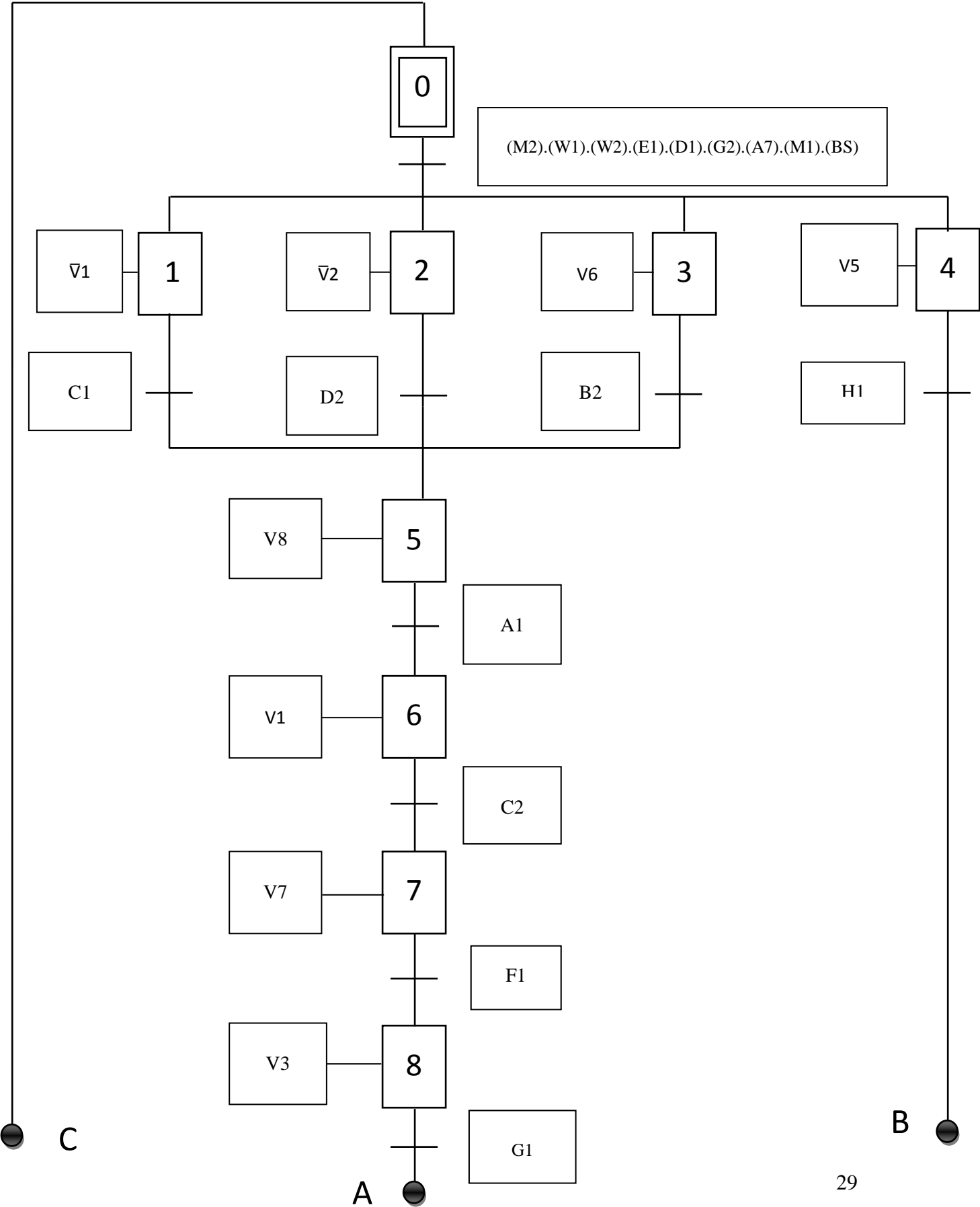


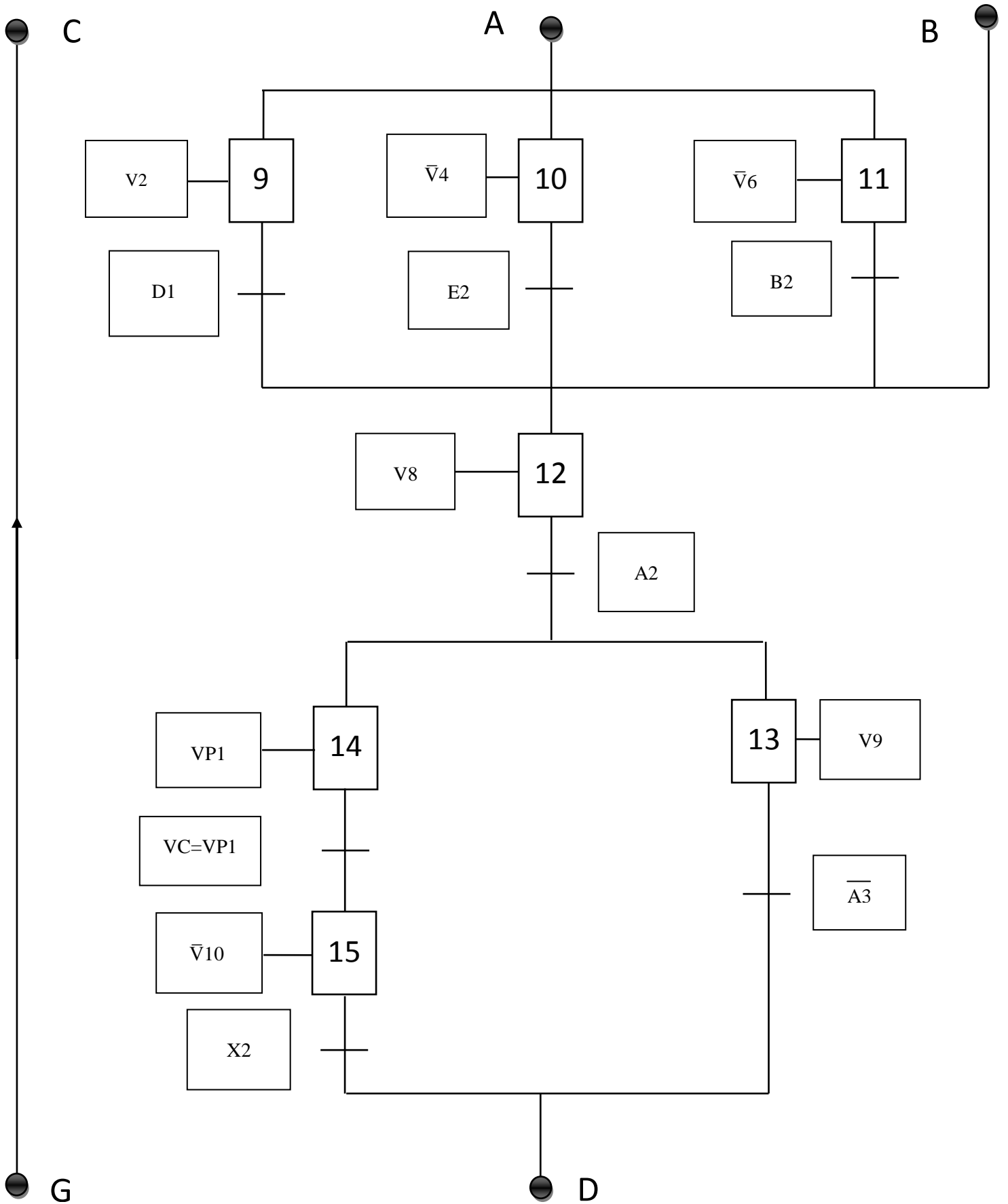


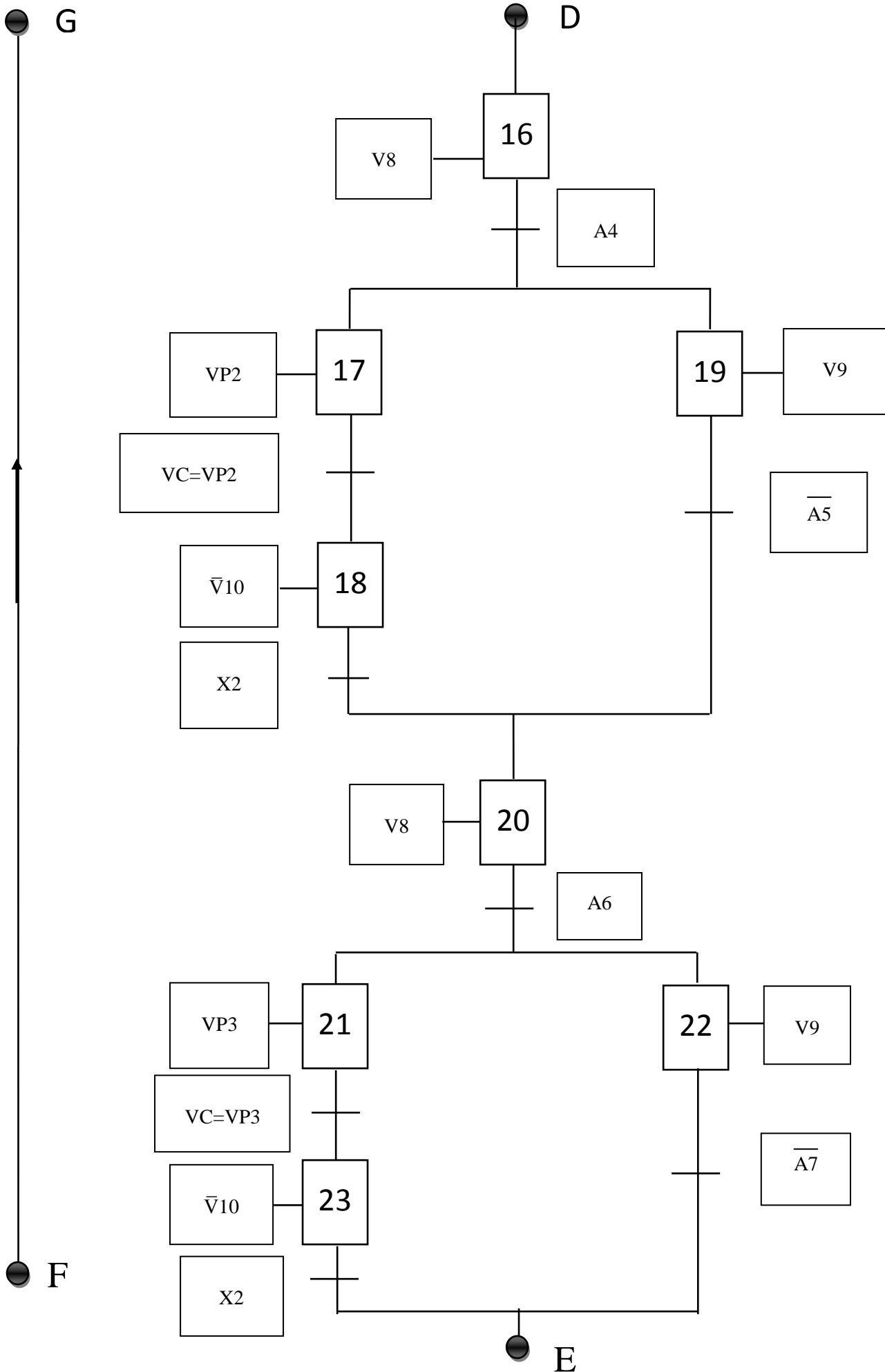


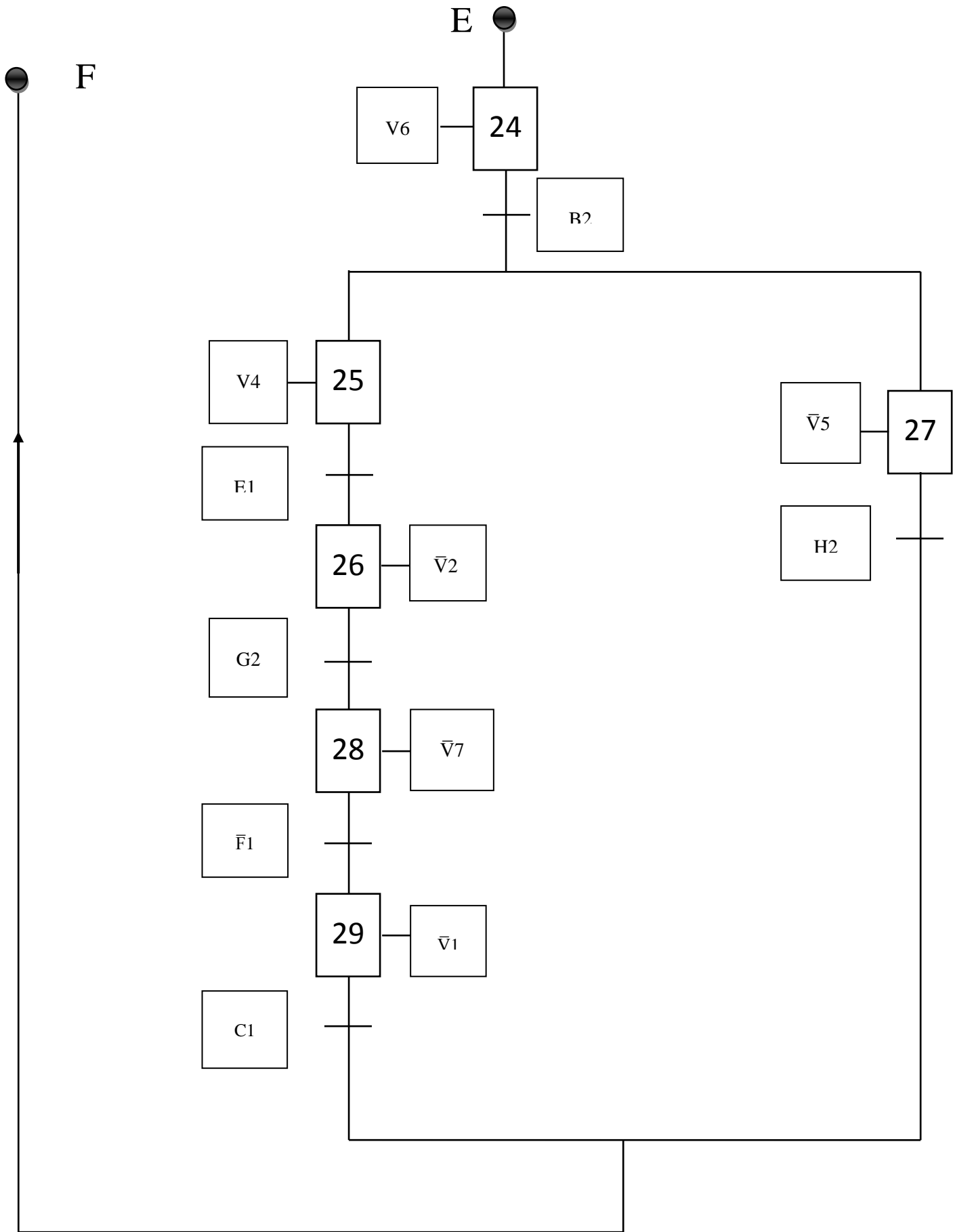


VI.2. GRAFCET niveau 2









➤ **Liste des capteurs, actionneurs et pré actionneurs**

• **Les capteurs :**

**BS** : bouton poussoir Star.

**A1** : unité de pose en position a1.

**A2** : unité de pose en position a2.

**A3** : unité de pose en position a3.

**A4** : unité de pose en position a4.

**A5** : unité de pose en position a5.

**A6** : unité de pose en position a6.

**A7** : unité de pose en position a7.

**B1** : mandrin de bobinage déverrouillé.

**B2** : mandrin de bobinage verrouillé.

**C1** : pressostat vérin d'écartement desserré.

**C2** : pressostat vérin d'écartement serré.

**D1** : vérin de levage de la pince en bas.

**D2** : vérin de levage de la pince en haut.

**E1** : vérin de la buse en bas.

**E2** : vérin de la buse en haut.

**F1** : présence de l'air dans le pressostat de la pince d'entraînement (ouverte).

**$\bar{F}1$**  : non présence de l'air dans le pressostat de la pince d'entraînement (fermée).

**G1** : vérin pivotant dépivoté.

**G2** : vérin pivotant enpivoté.

**H1** : la porte de protection fermée.

**H2** : la porte de protection ouverte.

**M1** : mandrin de bobinage en position initiale.

**M2** : groupe hydraulique en marche.

**W1** : présence du fil de bobinage.

**W2** : présence du fil de bobinage.

- **Les actionneurs et pré actionneurs**

**V1** : Serrer le vérin d'écartement (mandrin).

**V̄1** : Desserrer le vérin d'écartement (mandrin).

**V2** : Mettre le vérin de levage de la pince en bas.

**V̄2** : Mettre le vérin de levage de la pince en haut.

**V3** : Dépivoter le vérin pivotant.

**V̄2** : Empivoter le vérin pivotant.

**V4** : Mettre le vérin de la buse en bas.

**V̄4** : Mettre le vérin de la buse en haut.

**V5** : Fermer la porte de protection.

**V̄5** : Ouvrir la porte de protection.

**V6** : Verrouiller le mandrin de bobinage

**V̄6** : Déverrouiller le mandrin de bobinage.

**V7** : Ouvrir la pince.

**V̄7** : Fermer la pince.

**V8** : unité de pose à grande vitesse.

**V9** : unité de pose à petite vitesse.

**V10** : Démarrage du bobinage.

**V̄10** : Arrêt du bobinage.

**VC** : Valeur en cours du compteur de spires.

**VP1, VP2, VP3** : Valeur prédéfinie du compteur de spires 1, 2 et 3 respectivement.

**Conclusion**

Dans ce chapitre, nous avons présenté l'outil GRAFCET, qui nous a permis de traduire le cahier des charges et de mieux modéliser le fonctionnement de la bobineuse et de décrire les différentes étapes.

Dans les prochains chapitres, ce GRAFCET sera traduit en un programme qui pourra être transféré vers la mémoire de l'automate programmable.

# Description de l'automate programmable



**I. Introduction**

L'Automate Programmable Industriel (API) est aujourd'hui le constituant de base de la plupart des automatismes. Apparu vers les années soixante-dix, à la demande des constructeurs d'automobiles qui souhaitaient augmenter leur productivité avec un moindre coût, il s'est répandu par la suite dans tous les secteurs de l'industrie grâce à ses multiples avantages. Tous les API, qu'ils soient de la firme OMRON, ALLEN BRADLEY, ABB, SIEMENS, SCHNEIDER... sont essentiellement semblables dans leur structure de base et leur but général. Si différence il y a, c'est au niveau de leur puissance de travail, dimensions physiques, programmation et de leur prix, mais ce sont là des critères qui évoluent très rapidement avec le temps et selon le besoin de la manufacture.

Ce chapitre permet d'une part, de présenter l'architecture et les composants de base des API et d'autre part, de se familiariser avec l'automate SIEMENS, depuis la structure d'ensemble jusqu'aux règles de fonctionnement et de configuration de l'unité centrale.

L'Automate Programmable Industriel (API) est un appareil électronique programmable, adapté à l'environnement industriel, qui réalise des fonctions d'automatisme pour assurer la commande de pré-actionneurs et d'actionneurs à partir d'informations logique, analogique ou numérique.

**II. Place de l'API dans le système automatisé**

Un système automatisé est un ensemble d'éléments en interaction, et organisés dans un but, précis : agir sur une matière d'œuvre afin de lui donner une valeur ajoutée.

## II.1. Structure d'un système automatisé

Tout système automatisé peut se décomposer selon le schéma suivant :

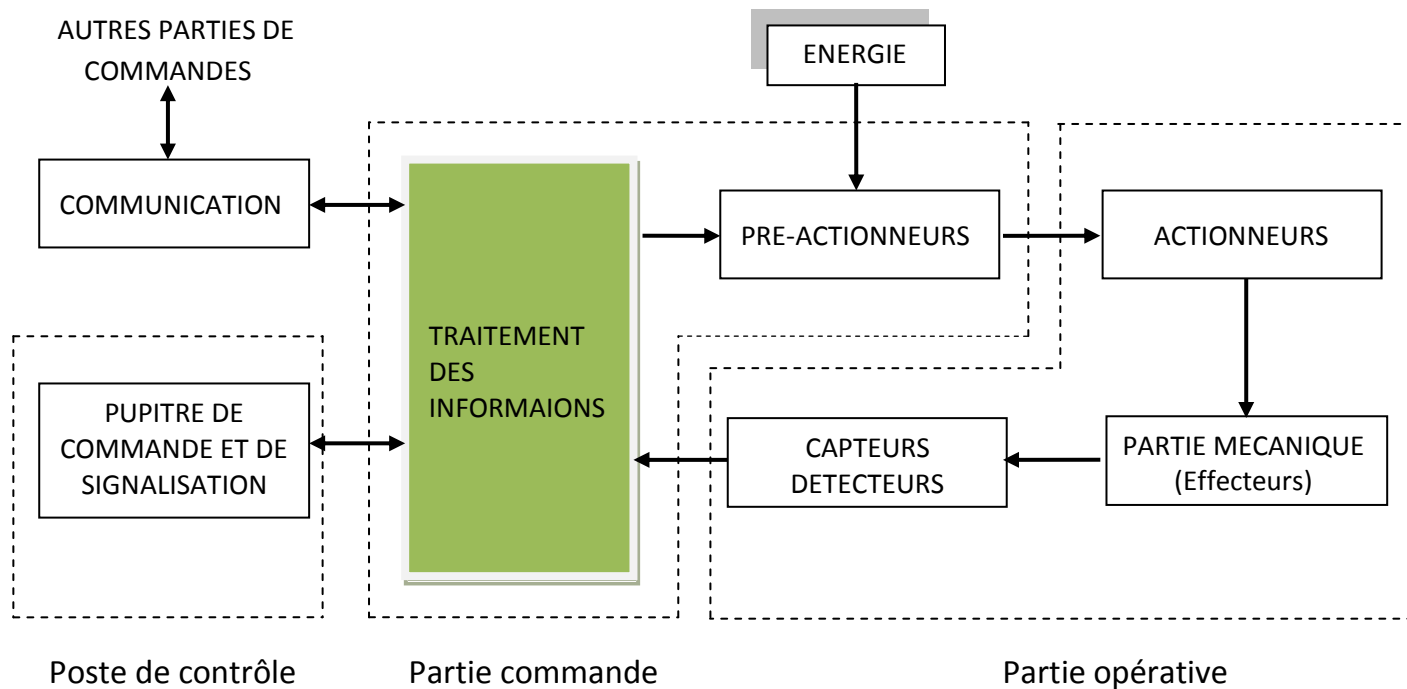


Figure 1. Structure interne d'un système automatisé.

- **Partie opérative :**

Elle agit sur la matière d'œuvre afin de lui donner sa valeur ajoutée.

Les actionneurs (moteurs, vérins) agissent sur la partie mécanique du système qui agit à son tour sur la matière d'œuvre.

Les capteurs / détecteurs permettent d'acquérir les divers états du système.

- **Partie commande :**

La partie commande donne les ordres de fonctionnement à la partie opérative.

Les pré-actionneurs permettent de commander les actionneurs ; ils assurent le transfert d'énergie entre la source de puissance (réseau électrique, pneumatique ...) et les actionneurs.

Exemple : contacteur, distributeur ...

Ces pré-actionneurs sont commandés à leur tour par le bloc traitement des informations.

Celui-ci reçoit les consignes du pupitre de commande (opérateur) et les informations de la partie opérative transmises par les capteurs / détecteurs.

- **Poste de contrôle :**

Composé de pupitre de commande et de signalisation, il permet à l'opérateur de commander le système (marche, arrêt, départ cycle ...). Il permet également de visualiser les différents états du système à l'aide de voyants, de terminal de dialogue ou d'interface homme-machine (IHM).

## **II.2. Domaines d'emploi des automates**

On utilise les API dans tous les secteurs industriels pour la commande des machines (convoyage, emballage ...) ou des chaînes de production (automobile, agroalimentaire ...) ou il peut également assurer des fonctions de régulation de processus (métallurgie, chimie ...).

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

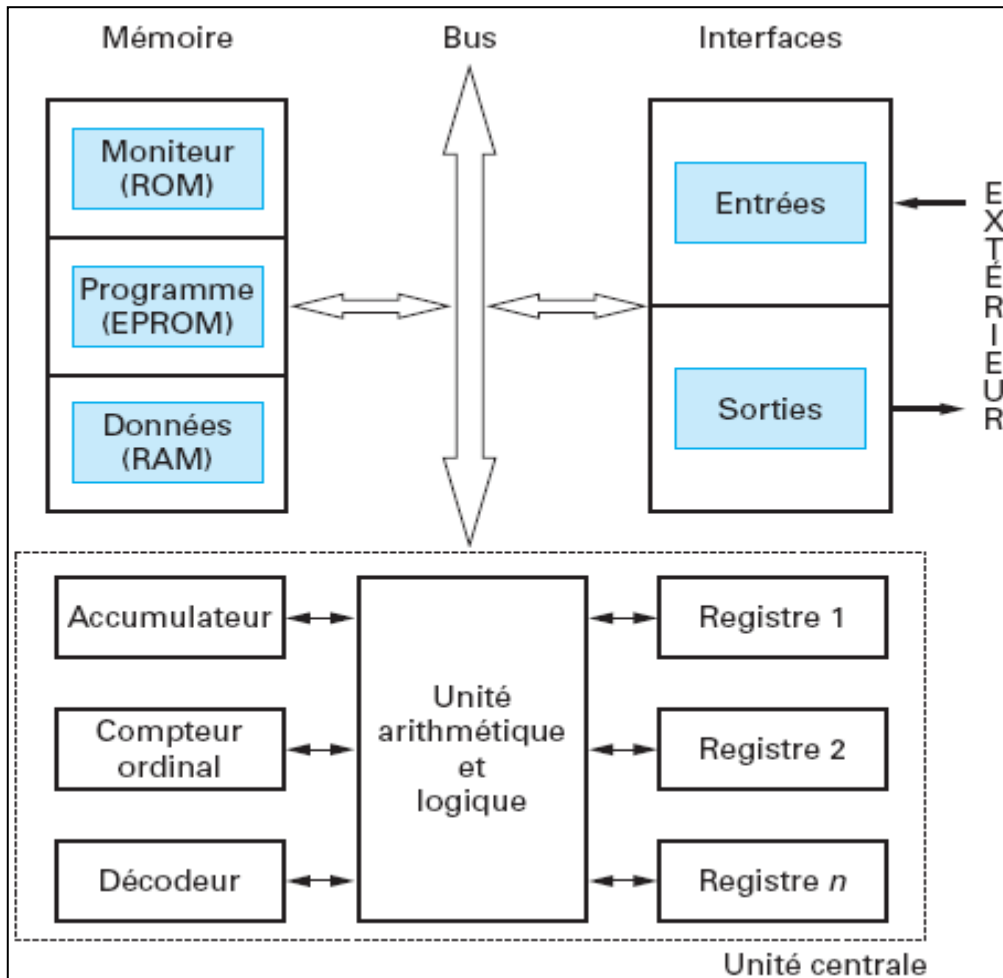
## **II.3. Nature des informations traitées par l'automate**

Les informations peuvent être de type :

- **Tout ou rien (TOR) :** l'information ne peut prendre que deux états (vrai/faux, 0 ou 1 ...).  
C'est le type d'information délivrée par un détecteur, un bouton poussoir ...
- **Analogique :** l'information est continue et peut prendre une valeur comprise dans une plage bien déterminée. C'est le type d'information délivrée par un capteur (pression, température ...).
- **Numérique :** l'information est contenue dans des mots codés sous forme binaire ou bien hexadécimale. C'est le type d'information délivrée par un ordinateur ou un module intelligent.

### III. Architecture des automates

La structure matérielle interne d'un API obéit au schéma donné sur la figure.



**Figure 2.** Architecture d'un API.

- ✚ **Unité centrale :** L'unité centrale (UC) est une carte électronique à base de microprocesseur, elle réalise toutes les fonctions logiques, arithmétiques et de traitement numérique (transfert, comptage, temporisation ...).

Le microprocesseur est généralement constitué de trois parties :

- **Unité de contrôle :** Sa fonction est de rechercher les instructions en mémoire, les décoder et générer les signaux de commande nécessaires à l'exécution des instructions.

- **Unité arithmétique et logique** : Elle exécute les opérations arithmétiques telles que l'addition, la soustraction, l'incrément, la décrémentation... et des opérations logiques telles que : AND, OR et XOR.
- **Les registres** : L'un des plus importants registres est le compteur programme, appelé aussi compteur ordinal. Il contient en permanence l'adresse de l'instruction en cours d'exécution ou la suivante, pour accélérer sa recherche. Son évolution est automatique : incrément de +1 à la fin de l'instruction, sauf en cas d'instruction de saut où l'adresse de la nouvelle instruction lui est imposée. Un autre registre est utilisé pour emmagasiner les codes des instructions une fois décodées, il s'agit du registre d'instruction. L'accumulateur est utilisé dans toutes les opérations arithmétiques et logiques. Comme son nom l'indique, il accumule les résultats des opérations arithmétiques et logiques.

#### ✚ Modules d'E/S :

Ils assurent le rôle d'interface de la Partie Commande (PC) dans le schéma classique de la (Figure3), qui distingue une Partie Opérative (PO), où les actionneurs agissent physiquement sur le processus, et une partie commande (PC) qui récupère l'état du processus à travers des capteurs (sous forme d'un signal électrique) via une interface d'entrées, coordonnant en conséquence les actions (via l'interface de sorties) pour atteindre les objectifs prescrits (matérialisés par des consignes).

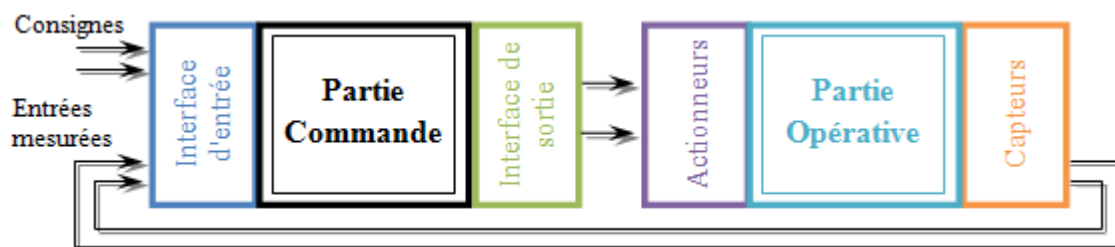


Figure 3. Partie opérative et partie commande.

- ✚ **Bus** : Il permet la communication de l'ensemble des blocs de l'automate et des éventuelles extensions.

✚ **Mémoire** : Elle permet de stocker le système d'exploitation (ROM ou PROM), le programme (EEPROM) et les données système lors du fonctionnement (RAM). Cette dernière est généralement secourue par pile ou batterie.

✚ **Bloc d'alimentation** :

Il permet de fournir à l'automate l'énergie nécessaire pour son fonctionnement, à partir d'une alimentation de 220 Volts alternatifs. Le bloc d'alimentation fournit à l'automate la tension dont il a besoin : 5 ou 12 ou 24 Volts.

NB : les entrées TOR sont souvent alimentées en 24V avec commun sur le 0V ou sur le 24V.

#### **IV. Les objectifs de l'automatisation**

Parmi les objectifs de l'automatisation :

- Production en qualite.
- Fournir les quantites ncessaires au bon moment.
- Accroitre la productivité et la flexibilité de l'outil.
- Augmenter la sécurité de l'opérateur.
- Réduire au maximum l'intervention de l'opérateur.

#### **V. Fonctions d'un automate programmable**

- **Acquérir les informations** : Ce sont les capteurs qui permettent de connaître toutes les informations nécessaires au bon fonctionnement du système.
- **Dialoguer avec l'opérateur** : C'est les ordres et les comptes-rendus qui permettent à l'opérateur de savoir à chaque instant l'état du système et son évolution.
- **Communiquer** : C'est tout ce qui permet au système de communiquer avec d'autre système pour une gestion automatisé de la production par exemple.
- **Traiter les données** : C'est le cœur du système, cette fonction est celle ou l'on effectue tous les calculs nécessaires au bon fonctionnement du système.

**➤ Critères de Choix d'un automate programmable industriel**

Les critères de choix essentiels d'un automate programmable industriel sont :

- Le nombre et la nature des entrées/sorties nécessaire.
- Les capacités de traitement du processeur (vitesse, données, opérations, temps réel..).
- La qualité du service après-vente.
- La fiabilité et robustesse.
- La nature du traitement (temporisation, comptage...).
- Puissance de l'unité centrale.
- L'alimentation (doit couvrir les besoins énergétique le l'unité centrale et de toutes les extensions).
- Considérations économiques.

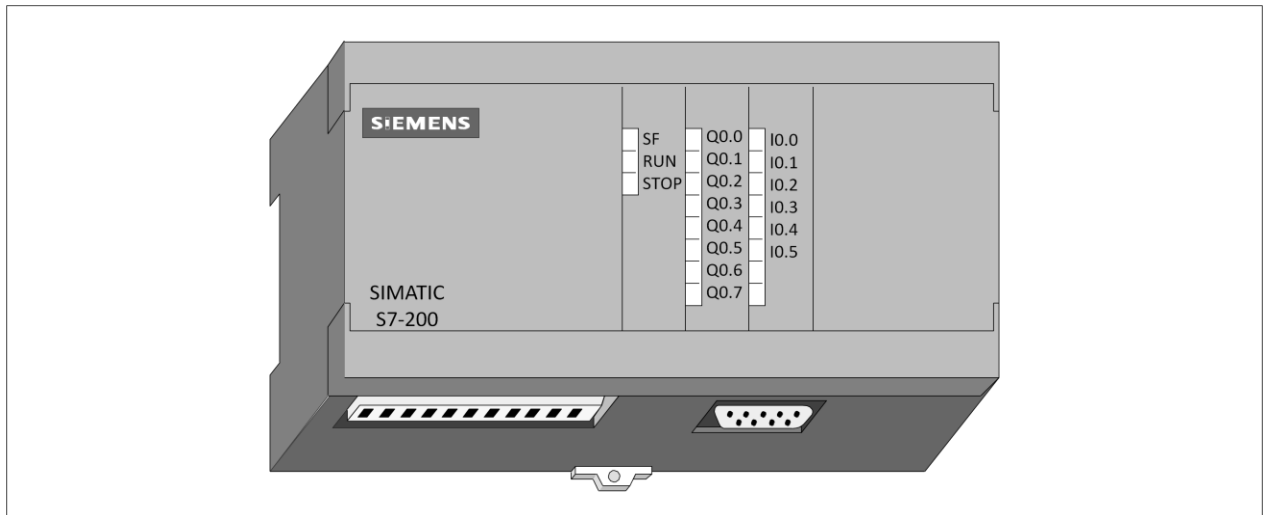
**VI. Présentation de l'automate programmable S7-200**

Pour automatiser notre machine, nous avons choisi l'API S7-200 (CPU 224) de la firme SIEMENS qui convient le mieux vue qu'il met à notre disposition des compteurs rapides non disponibles dans les autres automates programmables SIEMENS tel que le S7-300 et le S7-400. Par ailleurs, cet API est disponible et son nombre d'entrées et sorties est suffisant.

Le choix de la firme SIEMENS est fait par rapport à l'évolution de l'Entreprise Electro-Industries (ENEL), celle ci a toujours été fidèle à la firme SIEMENS depuis la première génération d'automate jusqu'au S7, cette dernière est très connue par sa technologie évolutif et offre la fiabilité et la robustesse de ses produits.

La famille S7-200 est constituée de micro-automates programmables qui peuvent commander une large gamme d'appareils afin de répondre à nos besoins en matière d'automatisation. Le S7-200 surveille les entrées et commande les sorties selon les instructions du programme utilisateur. Le programme peut contenir des opérations booléennes, des opérations de comptage, des opérations de temporisation, des opérations arithmétiques complexes et des opérations de communication avec d'autres unités intelligentes.

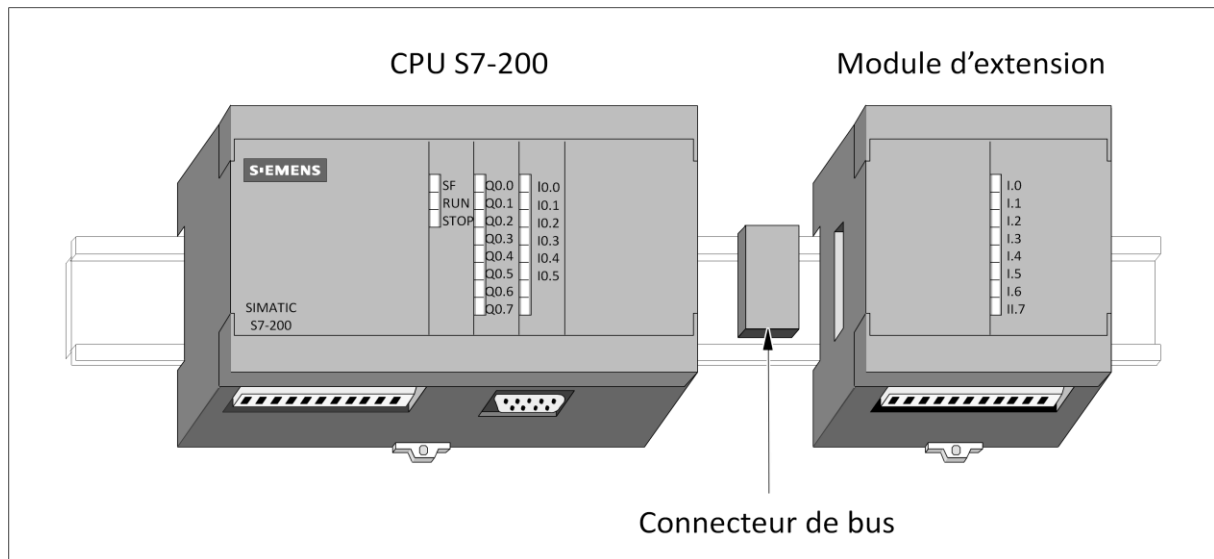
Sa forme compacte, sa configuration souple et son important jeu d'opérations en font une solution idéale pour la commande d'applications très variées.



**Figure 4.** Automate programmable S7-200.

## VII. Modules d'extensions

Le raccordement d'autres modules d'extension sur le côté droit de l'automate de base permet d'ajouter des fonctions ou séquences d'entrées /sorties supplémentaire. Un connecteur de bus est fournit avec le module d'extension pour permettre la connexion de ce dernier a l'automate de base. La figure si dessous nous montre un exemple de raccordement de module d'extension.



**Figure 5.** Unité centrale avec module d'extension.

1. Module d'entrées Tout Ou Rien (TOR): lorsqu'on a toutes les entrées d'un automate de base utilisées, on peut rajouter un module d'extension TOR qui ne possède que des entrées pour crée d'autre entrées supplémentaires.
2. Module de sortie Tout Ou Rien (TOR) : lorsqu'on a toutes les sorties d'un automate de base utilisées, on peut rajouter un modules d'extension TOR qui ne possède que des sorties pour crée d'autre sortie supplémentaires.

**VIII. Choix de la CPU 224**

Le choix de la CPU 224 a été fait par rapport à sa haute performance compacte, elle a 14 entrées et 10 sorties intégrées, et possède une capacité extensible de 7 modules maximum.

**+ Caractéristiques techniques de la CPU 224 :**

- 14 entrées à 24 Vcc.
- 10 sorties à 24 Vcc.
- Mémoire de programmation 8192 octets.
- Mémoire de données 5120 octets.
- Deux potentiomètres analogiques intégré 8bits.
- Sauvegarde des données utilisateur (7 jours par le biais d'un condensateur).
- Un Interface de communication RS 485.
- Deux voies de comptages pour le compteur rapide Incrémentation/Décrémentation, de 6 à 30 KHz pour les compteurs rapides a une phase et de 4 à 20 KHz pour les compteurs rapides à deux phases.
- Taille de la mémoire image des E/S TOR, 128 entrées et 128 sorties.
- Vitesse de traitement d'instructions sur bits 0,22  $\mu$ s.

**Conclusion :**

Dans ce chapitre, nous avons présentes l'automate programmable industriel (API) que nous avons utilisés dans notre projet (S7-200), le fonctionnement de l'API, les principaux composants d'un API et ses différentes caractéristiques.

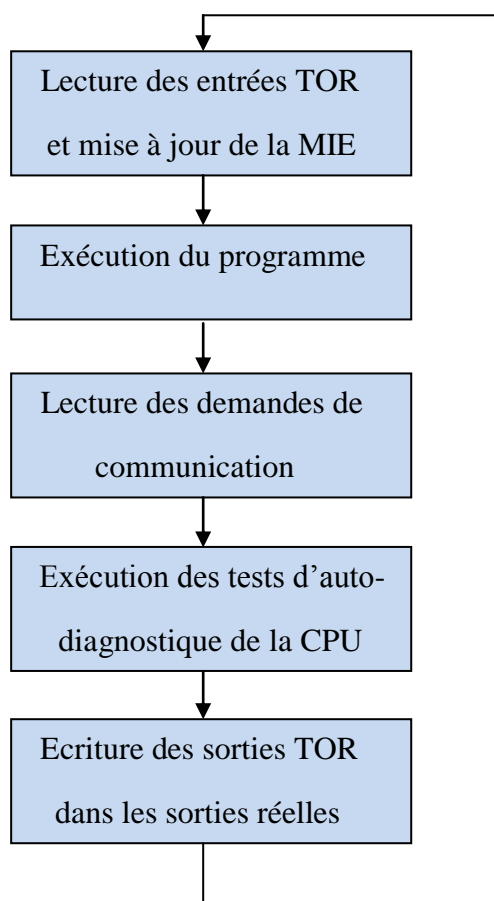
Le chapitre suivant a été consacré à la programmation qui a été faite à l'aide du logiciel de programmation : STEP-7 Micro Win v4.0 qui est un intermédiaire entre l'opérateur et l'API, ce dernier ne peut pas remplir son rôle sans programme.

## I. Introduction

Pour automatiser la machine, nous avons choisi l'automate programmable S7-200. Le logiciel de programmation STEP 7-Micro/WIN fournit un environnement convivial pour concevoir, éditer et surveiller la logique nécessaire à la commande d'une application. Le projet permet d'associer le programme à toutes les informations nécessaires pour pouvoir communiquer avec l'API et charger le programme dans cet API.

## II. Fonctionnement du programme de commande

Le S7-200 exécute en cycles continus la logique de commande du programme, en lisant et en écrivant des données. Après avoir chargé le programme dans l'API, et faire passer l'API à l'état de fonctionnement "Marche" (RUN), la CPU S7-200 exécute le programme dans l'ordre suivant :



- Lecture des entrées TOR : Le S7-200 copie l'état des entrées physiques dans la mémoire image des entrées (MIE).
- Exécution du programme : Le S7-200 exécute les instructions du programme et sauvegarde les valeurs dans la zone mémoire de sorties ou mémoire image des sorties (MIS).
- Traitement de toute demande de communication : Le S7-200 exécute toute tâche nécessaire pour la communication.
- Exécution des tests d'auto-diagnostic de la CPU : Le S7-200 s'assure que le microprogramme, la mémoire de programme et les modules d'extension présents fonctionnent correctement.
- Ecriture des sorties TOR : Les valeurs enregistrées dans la mémoire image des sorties (MIS) sont écrites dans les sorties physiques.

**Figure 1.** Etapes du fonctionnement du programme de commande.

### III. Accès aux données de l'API S7-200

Pour accéder à un bit dans une zone de mémoire, on doit préciser son adresse composée d'un identificateur de zone de mémoire, de l'adresse d'octet et du numéro de bit. La figure suivante montre comment accéder à un bit dans une mémoire par la méthode appelée adressage « octet.bit ».

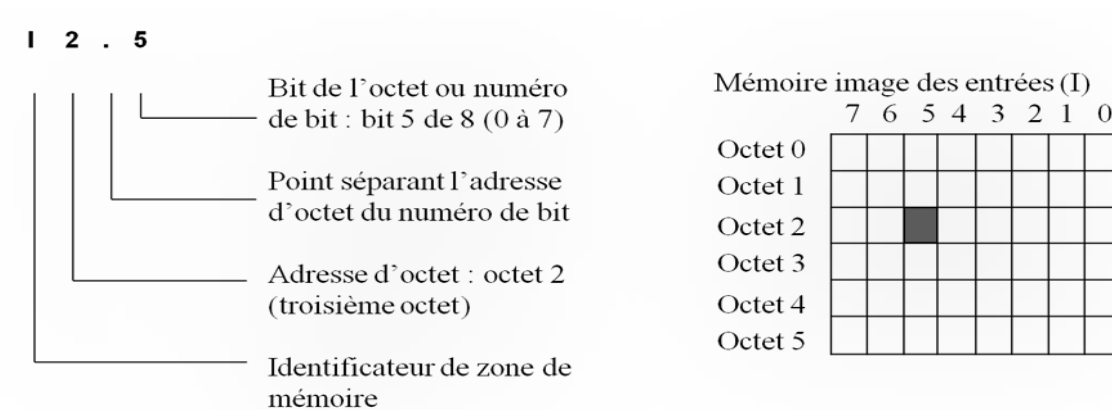
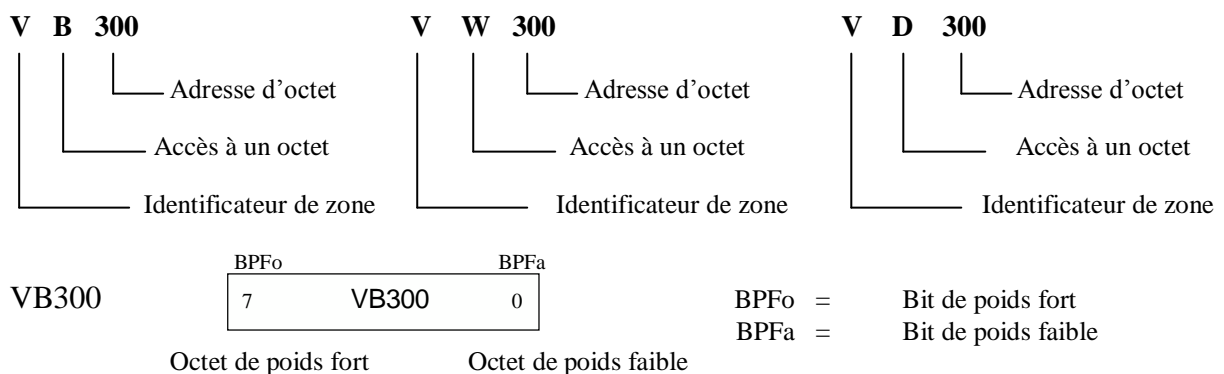
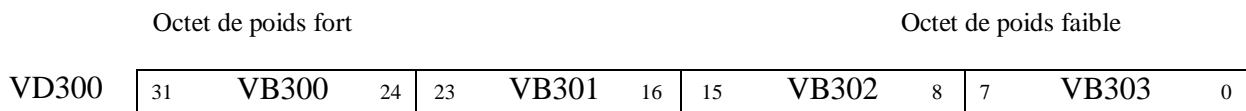


Figure 2. Adressage octet.bit [2]

Le format d'adresse d'octet permet d'accéder à des données dans la plupart des zones de mémoire (V, I, Q, M, S, L et SM) sous forme d'octets, de mots ou de doubles mots. L'adresse est composée d'un identificateur de zone, de la désignation de la taille des données et de l'adresse d'octet de départ de la valeur d'octet, de mot ou de double mot (Figure 3).

On accède aux données situées dans d'autres zones de mémoire (T, C, HC et AC) par un format d'adresse comprenant l'identificateur de zone et le numéro de l'élément en question.





**Figure 3.** Comparaison de l'accès par octet, mot ou double mot à la même adresse [2]

## IV. Accès aux données dans les zones mémoire [2]

### IV.1. Mémoire image des entrées (I)

Le S7-200 lit les entrées physiques au début de chaque cycle et écrit ces valeurs dans la mémoire image des entrées.

### IV.2. Mémoire image des sorties (Q)

A la fin du cycle, le S7-200 copie dans les sorties physiques les valeurs contenues dans la mémoire image des sorties.

### IV.3. Mémoire des variables (V)

La mémoire des variables peut servir à la sauvegarde des résultats intermédiaires d'opérations exécutées par la logique de commande.

### IV.4. Mémentos (M)

Les mémentos internes fournissent de l'espace mémoire pour l'état intermédiaire d'une opération ou d'autres informations de commande.

### IV.5. Mémentos spéciaux (SM)

Les mémentos spéciaux servent à échanger les informations entre la CPU et le programme. On peut faire appel à ces bits pour sélectionner et commander certaines fonctions spéciales de la CPU S7-200, on citera :

- SM0.0 : ce bit est toujours mis à 1 dès l'allumage de l'API.
- SM0.1 : ce bit est mis à 1 uniquement au premier cycle. Il sert aussi, parfois, à l'appel d'un sous-programme d'initialisation.
- SMD38 et SMD42 : ces doubles mots sont réservés aux compteurs rapides.

### **IV.6. Compteur (C)**

Le S7-200 contient trois modèles de compteurs qui comptent à chaque front montant des entrées de comptage : les compteurs incrémental (CTU), les compteurs décrémental (CTD) et les compteurs incrémental/décrémental (CTUD).

### **IV.7. Accumulateurs (AC)**

Les accumulateurs sont des éléments en lecture/écriture qui peuvent être utilisés comme mémoire. On peut utiliser les accumulateurs pour transmettre des paramètres à ou depuis des sous-programmes et pour sauvegarder des valeurs intermédiaires utilisées dans un calcul. Le S7-200 dispose de quatre accumulateurs de 32 bits : AC0, AC1, AC2 et AC3.

### **IV.8. Compteurs rapides (HC)**

Les compteurs rapides comptent des événements très rapides impossibles à gérer aux taux de cycle du S7-200. La fréquence de comptage maximale pour un compteur rapide dépend du modèle de la CPU S7-200. Ils ont une valeur de comptage (ou valeur en cours) entière signée de 32 bits. Pour y accéder, il faut indiquer l'adresse du compteur rapide, comprenant l'identificateur de zone HC et le numéro du compteur. L'accès à la valeur en cours des compteurs rapides est en lecture seulement et en format de double mot (32 bits).

## **V. Fonctionnement des compteurs rapides [2]**

Les compteurs rapides servent typiquement d'entraînement pour les dispositifs de comptage où un arbre en rotation à vitesse constante est muni d'un encodeur angulaire incrémental. L'encodeur angulaire fournit un nombre défini de valeurs de comptage par tour et une impulsion de remise à zéro par tour. Les horloges (A et B) et l'impulsion de remise à zéro du codeur angulaire constituent les entrées du compteur rapide.

La première parmi plusieurs valeurs prédéfinies est chargée dans le compteur rapide et les sorties désirées sont activées pour la durée où la valeur de comptage en cours est inférieure à la valeur prédéfinie. Le compteur est défini de telle façon qu'une interruption est générée si la valeur en cours est égale à la valeur prédéfinie ainsi que si une remise à zéro a lieu.

Une nouvelle valeur prédéfinie est chargée et l'état suivant des sorties est activé à chaque fois que l'événement d'interruption « Valeur en cours égale à valeur prédéfinie » se

produit. Lorsque l'événement d'interruption « Mise à zéro » survient, la première valeur prédéfinie ainsi que le premier état des sorties sont activés et le cycle se répète.

Comme les interruptions se produisent à une fréquence bien inférieure à la vitesse de comptage des compteurs rapides, il est possible de réaliser une commande précise des opérations rapides qui a un impact relativement mineur sur le cycle complet de l'automate programmable.

La méthode d'association d'interruptions à des programmes d'interruption permet d'effectuer chaque chargement d'une nouvelle valeur prédéfinie dans un programme d'interruption distinct.

Mais on peut aussi traiter tous les événements d'interruption dans un programme d'interruption unique.

## VI. Programmation des compteurs rapides

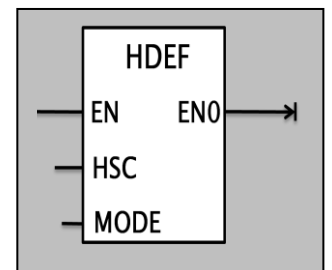
La configuration d'un compteur rapide se fait selon les étapes fondamentales suivantes :

### VI.1. Définition du mode de comptage

L'opération Définir mode pour compteur rapide (HDEF) affecte un mode de fonctionnement (MODE) au compteur rapide indiqué (HSCx).

La sélection du mode définit les fonctions d'horloge, de sens de démarrage et de remise à zéro du compteur rapide.

Le tableau 1 montre les entrées utilisées pour les fonctions d'horloge, de gestion du sens de comptage, de mise à zéro et de démarrage associées aux compteurs rapides. Une entrée ne peut servir pour deux fonctions différentes, mais toute entrée qui n'est pas utilisée pour le mode activé de son compteur rapide peut être utilisée à d'autres fins.



Mode	Description	Entrées			
	HSC0	I0.0	I0.1	I0.2	
	HSC1	I0.6	I0.7	I1.0	I1.1
	HSC2	I1.2	I1.3	I1.4	I1.5
	HSC3	I0.1			
	HSC4	I0.3	I0.4	I0.5	
	HSC5	I0.4			
0	Compteur monophasé avec gestion interne du sens de comptage	Horloge			
1		Horloge		Mise à 0	
2		Horloge		Mise à 0	Démarrage
3	Compteur monophasé avec gestion externe du sens de comptage	Horloge	Sens de comptage		
4		Horloge	Sens de comptage	Mise à 0	
5		Horloge	Sens de comptage	Mise à 0	Démarrage
6	Compteur biphasé avec deux entrées d'horloge	Horloge, incrémentation	Horloge, décrémentation		
7		Horloge, incrémentation	Horloge, décrémentation	Mise à 0	
8		Horloge, incrémentation	Horloge, décrémentation	Mise à 0	Démarrage
9	Compteur en quadrature de phase A/B	Horloge phase A	Horloge phase B		
10		Horloge phase A	Horloge phase B	Mise à 0	
11		Horloge phase A	Horloge phase B	Mise à 0	Démarrage
12	Seuls HSC0 et HSC3 acceptent le mode 12. HSC0 compte le nombre d'impulsions sortant de Q0.0. HSC3 compte le nombre d'impulsions sortant de Q0.1.				

**Tableau 1.** Entrées pour les compteurs rapides. [2]

- Dans notre programme, nous avons utilisé le compteur rapide HSC0 fonctionnant en mode 9 (quadrature de phase).
- Les compteurs rapides sont les éléments les plus importants dans notre programme, car ils nous permettent de supprimer toutes les cartes électroniques qui servent à la conversion (nombre de spires en impulsions et vis vers ça), et au comptage à partir de l'encodeur incrémental.

## VI.2. Définition de l'octet de commande

Les trois premiers bits de commande (0,1 et 2) permettent de configurer l'état actif pour les entrées de mise à zéro, de démarrage et de sélectionner la fréquence de comptage simple ou quadruple (pour compteurs en quadrature de phase), et ne servent qu'à l'exécution de l'opération HDEF. Ces trois bits doivent être mis à la valeur désirée avant d'exécuter l'opération HDEF, sinon, le compteur va prendre la configuration par défaut pour le mode sélectionné.

Le tableau 2 présente chacun des bits de commande pour HSC0.

HSC0	Description
SM37.0	Bit de commande du niveau d'activité pour la mise à zéro: 0 = niveau haut                      1 = niveau bas
- - -	Bit de commande du niveau d'activité pour le démarrage: 0 = niveau haut                      1 = niveau bas
SM37.2	Sélection de la vitesse de comptage pour compteurs en quadrature de phase : 0 = fréquence quadruple              1 = fréquence simple
SM37.3	Bit de commande du sens de comptage : 0 = décrémentation                      1 = incrémentation
SM37.4	Ecrire sens de comptage dans le compteur rapide : 0 = pas de mise à jour                      1 = mise à jour du sens
SM37.5	Ecrire nouvelle valeur prédéfinie dans le compteur rapide : 0 = pas de mise à jour                      1 = mise à jour de la valeur prédéfinie
SM37.6	Ecrire nouvelle valeur en cours dans le compteur rapide : 0 = pas de mise à jour                      1 = mise à jour de la valeur en cours
SM37.7	Valider le compteur rapide : 0 = inhiber le compteur rapide      1 = valider le compteur rapide

**Tableau 2.** Octet de commande pour HSC0. [2]

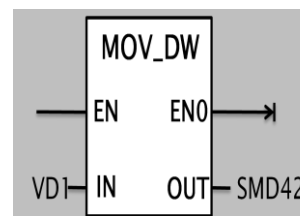
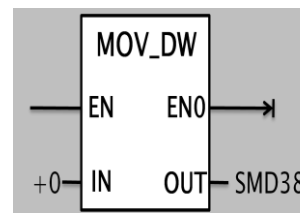
Dans notre programme, l'octet de commande qui définit le compteur rapide utilisé est le suivant :

SM37.7	SM37.6	SM37.5	SM37.4	SM37.3	SM37.2	SM37.1	SM37.0
1	1	1	1	1	0	0	0
F				8			

Donc l'octet de commande est : 16#F8 (hexadécimal).

### VI.3. Définition de valeur en cours et de la valeur prédéfinie

Chaque compteur rapide possède une valeur en cours (VC) et une valeur prédéfinie (VP) qui sont des nombres entiers de 32 bits signés. La valeur en cours est la valeur de comptage en cours du compteur alors que la valeur prédéfinie est la valeur de comparaison qu'on utilise optionnellement pour déclencher une interruption lorsque la valeur en cours atteint la valeur prédéfinie (VC=VP). Pour charger ces valeurs dans le compteur rapide, on doit définir l'octet de commande et les octets de mémentos spéciaux contenant ces valeurs.



Valeur à charger	HSC0
Nouvelle valeur en cours (VC)	SMD38
Nouvelle valeur prédéfinie (VP)	SMD42

Tableau 3. Mémentos spéciaux VC, VP. [2]

Compteur	HSC0
Valeur en cours	HC0

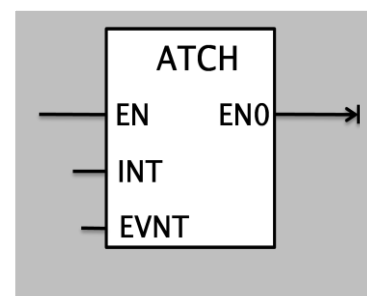
Tableau 4. Lecture de VC. [2]

### VI.4. L'octet d'état

L'octet d'état fournit des mémentos d'état indiquant le sens de comptage en cours et si la valeur en cours est supérieure ou égale à la valeur prédéfinie.

### VI.5. Association du programme d'interruption à événement s

L'opération ATCH sert à associer un événement (événement 12 pour HSC0) au programme d'interruption EG\_COMPT (auparavant programmé) lorsque la valeur en cours est égale à la valeur prédéfinie (VC=VP).

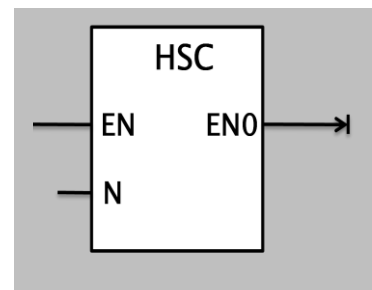


## VI.6. Validation du programme d'interruption

L'opération ENI sert à valider tous les événements d'interruption. Le passage à l'état de fonctionnement « Marche » (RUN) inhibe toutes les interruptions.

## VI.7. Activation du compteur rapide

L'opération « activer compteur rapide (HSC) » configure et commande le compteur rapide, selon l'état des mementos spéciaux pour compteurs rapides. Le paramètre N précise le numéro du compteur rapide.



## VI.8. Programme d'interruption (EG\_COMPT)

Les interruptions sont gérées par des événements. Avant d'appeler un programme d'interruption, il faut associer un événement d'interruption à la partie du programme à exécuter à l'apparition de cet événement (VC=VP). Pour cela il faut faire appelle à l'opération ATCH en précisant le numéro de l'événement d'interruption (événement 12) et le nom du programme d'interruption qui lui correspond (EG\_COMPT).

## VII. Organisation du programme de commande

Le programme de commande de la CPU S7-200 est composé de trois types d'unités d'organisation de programme (UOP) :

### VII.1. Programme principale : PPAL (OB1)

Nous y plaçons les opérations qui commandent notre application .Ces opérations sont exécutées séquentiellement, une fois par cycle de la CPU.

**VII.2. Sous-programmes : SBR\_i**

Un sous-programme est un ensemble facultatif d'opérations qui sont exécutées uniquement lorsque le programme principal, un programme d'interruption ou un autre sous-programme appellent le sous-programme.

**VII.3. Programmes d'interruption : INT\_i**

Ces éléments facultatifs du programme réagissent à des événements d'interruption spécifiques. Ils sont conçus pour traiter un événement d'interruption prédéfini. Ils ne sont pas appelés par le programme principal et seront exécutés seulement lorsque cet événement précis se produit.

**VIII. Programmation de l'automate**

Pour programmer l'automate, le GRAFCET a d'abord été traduit en langage à contact sur l'éditeur du MICRO/WIN qui à son tour le convertit en langage machine exécutable par la CPU.

VIII.1. Présentation du logiciel STEP7-MICRO/WIN

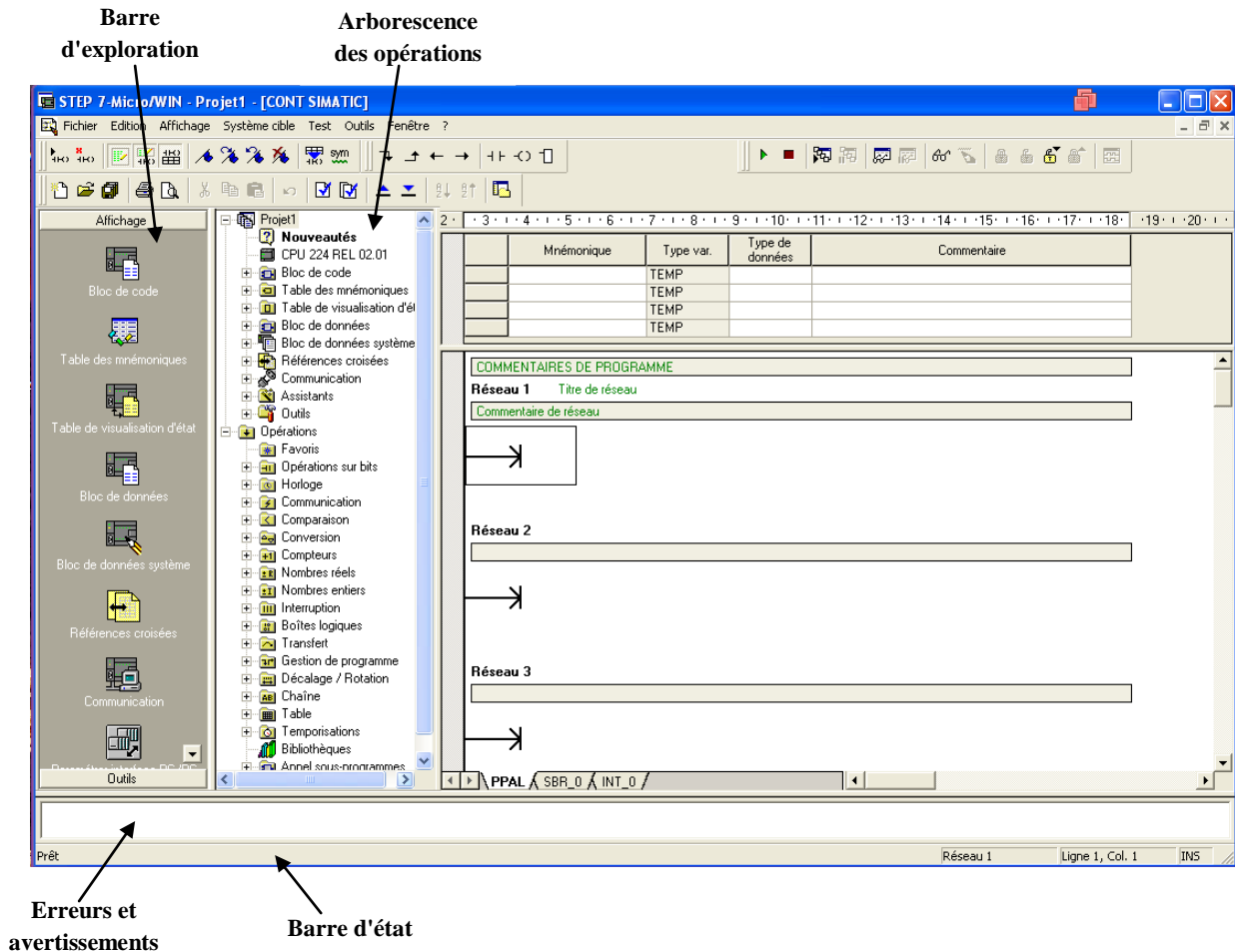


Figure 4. Page d'accueil du STEP7-MICRO/WIN.

- **Barre d'exploration :** présente des icônes sur lesquelles on clique pour accéder à des fonctions de programmation : affichage, outils.
- **Arborescence des opérations :** affiche une arborescence de tous les objets du projet et de toutes les opérations disponibles pour l'éditeur de programme en cours (CONT, LOG ou LIST).
- **Erreurs et avertissements :** affiche des messages d'information lorsque la compilation du programme ou de bibliothèque d'opérations.

- **Barre d'état** : affiche des informations sur l'état des commandes que vous exécutez dans STEP 7-Micro/WIN.
- **Barres d'outils** : permet d'accéder facilement par la souris aux commandes les plus fréquemment utilisées de STEP 7-Micro/WIN.

### VIII.2. Création et enregistrement d'un programme sous STEP7 MICRO /WIN

- Lancer le logiciel STEP7-MICRO/WIN.
- Cliquer sur nouveau projet.
- Cliquer sur Enregistrer projet.
- Donner le nom et Enregistrer dans l'emplacement de votre choix.

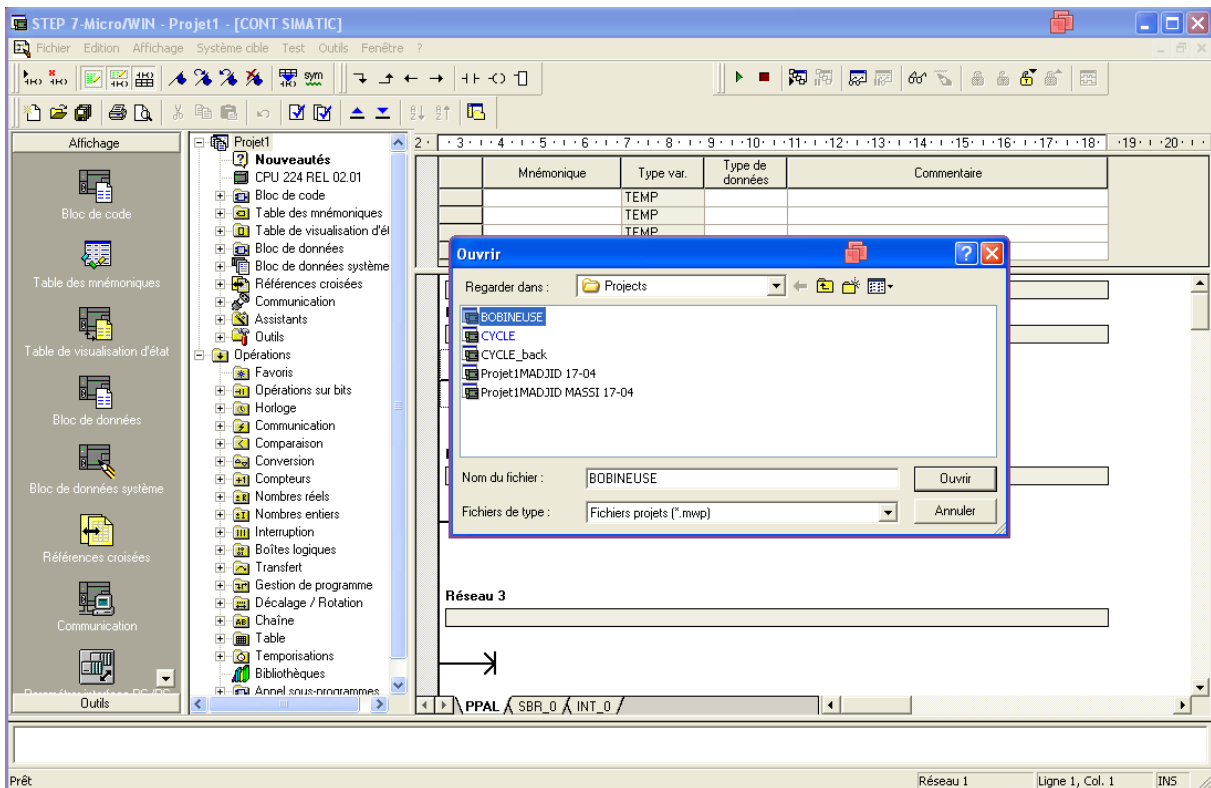


Figure 5. Fenêtre de création du projet et de l'enregistrer.

VII.3. Choix de la CPU

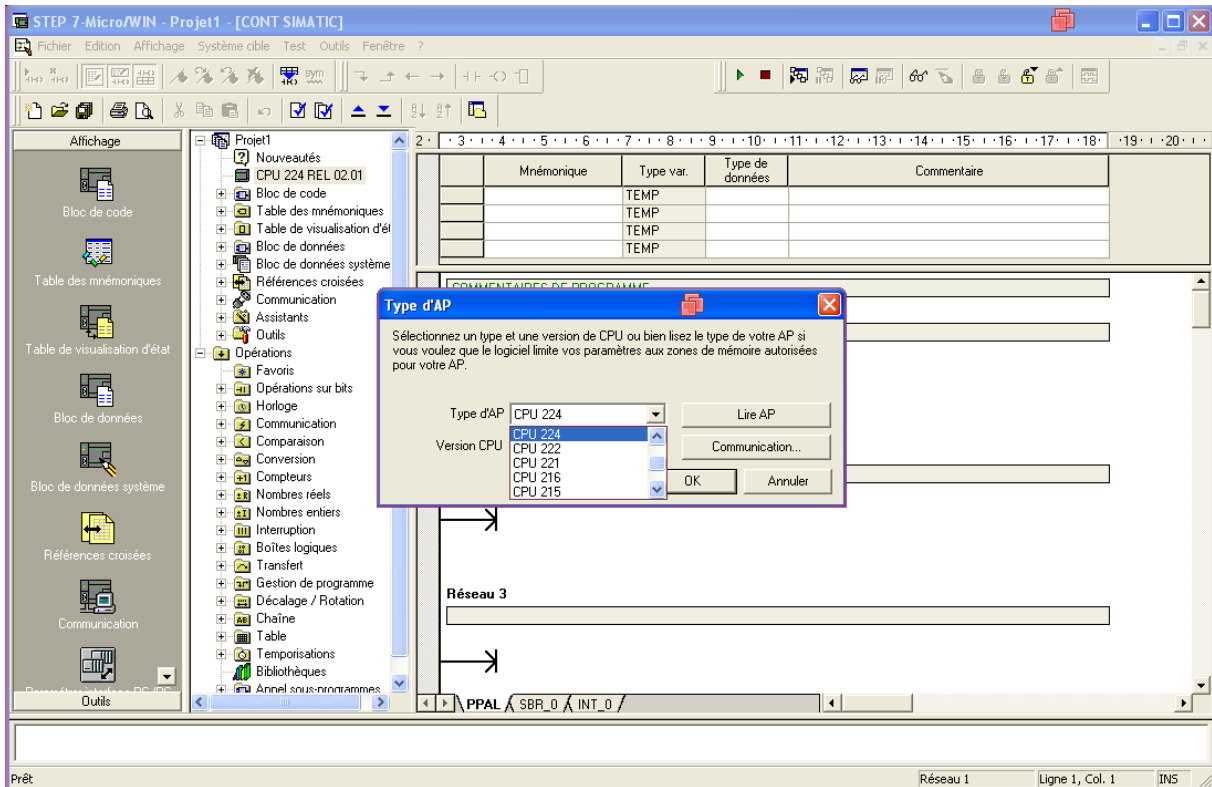



Figure 6. Choix de la CPU.

VIII.4. Transfert du programme dans la CPU de l'automate


Pour charger le programme dans la CPU, soit cliquer sur  ou bien aller sur « Fichier » puis « charger dans la CPU »

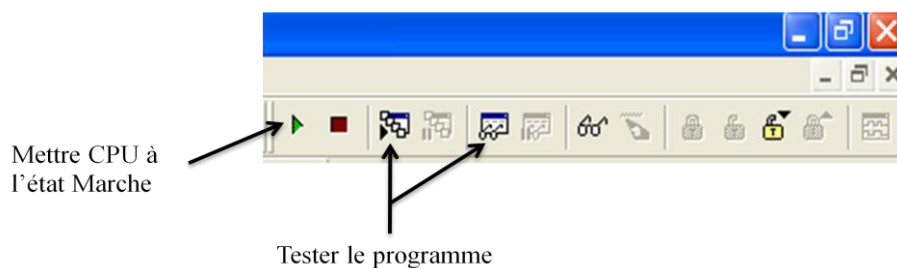


Charger dans la CPU

Figure 7. Barre d'outils.

### VIII.5. Mettre la CPU à l'état marche

Une fois le programme est chargé dans la CPU, on met la CPU à l'état Marche soit cliquer dans la Barre d'outils sur  ou bien sélectionner la commande « **Système cible** » et puis sur « **Marche** ».



**Figure 8.** Barre d'outils.

**Conclusion**

Après la mise en œuvre du programme de la bobineuse avec le logiciel STEP 7-MICRO/WIN, nous apporterons dans le chapitre qui suit des améliorations à la partie commande en introduisant un Panel tactile.

# Configuration du panel tactile avec WinCC Flexible



## **I. Introduction**

Lorsque la complexité des processus augmente et que les machines et installations doivent répondre à des spécifications de fonctionnalité toujours plus sévères, l'opérateur a besoin d'un maximum de transparence. Cette transparence s'obtient au moyen de l'Interface Homme-Machine (IHM).

Un système IHM constitue l'interface entre l'homme (opérateur) et le processus (machine/installation). Il existe par conséquent une interface entre l'opérateur et WinCC flexible (sur le pupitre opérateur) et une interface entre WinCC flexible et le système d'automatisation.

## **II. WinCC flexible**

SIMATIC WinCC flexible est le logiciel IHM innovant sous Windows, conçu pour la conduite et supervision au plus près des machines dans les processus automatisés. Le logiciel d'ingénierie permet une configuration homogène de tous les pupitres SIMATIC HMI sur base Windows – du plus petit Micro Panel jusqu'aux solutions sur PC.

### **II .1 Elements de WinCC flexible [3]**

#### **II.1.1.WinCC flexible Engineering System**

WinCC flexible Engineering System (Advanced) est le logiciel avec lequel on réalise toutes les tâches de configuration requise sous Windows, à partir de l'ordinateur de configuration.

#### **II.1.2. WinCC flexible Runtime**

Le logiciel de visualisation de processus WinCC flexible Runtime permet de faire fonctionner la configuration sous Windows et de visualiser le processus. WinCC flexible Runtime est également exécuté sur l'ordinateur de configuration pour tester et simuler le fichier projet compilé.

## **III. Critères de choix du panel tactile TP 177A**

La sélection du panel tactile TP 177A pour notre application s'est faite en prenant en considération plusieurs critères qui sont :

- La possibilité de créer plusieurs menus de commande, ce qui donne une commande plus performante et plus précise.
- La courte durée des temps de configuration et de mise en service dans WinCC.
- La possibilité de configuration en orientation verticale qui permet une utilisation dans des endroits exigus.

- La bibliothèque graphique qui donne une meilleure présentation pour l'opérateur de la machine.

## IV. Etapes de configuration

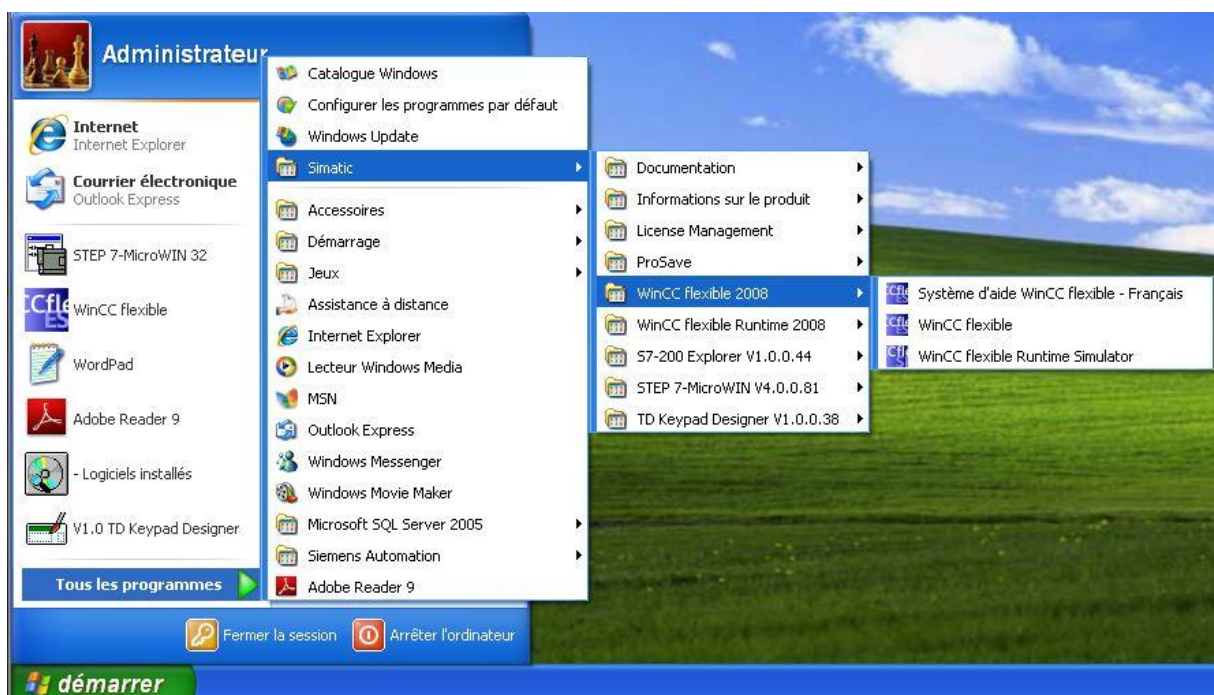
### IV.1. Créer un projet

L'élément de base de la configuration de l'interface graphique est le projet. Il sert à créer et à configurer tous les objets indispensables à la commande et au contrôle de notre machine (vues, variables, alarmes...).

Pour créer un projet on suit les étapes suivantes :

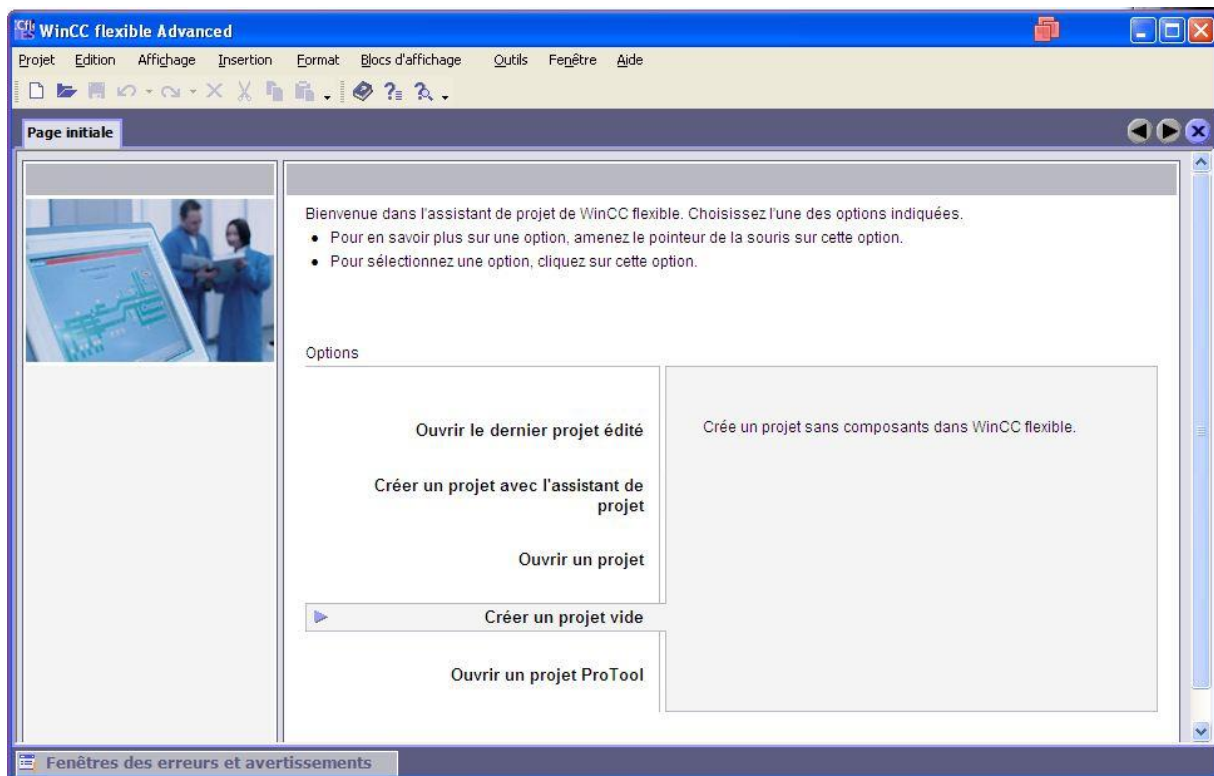
#### IV.1.1. Démarrer WinCC flexible

A partir du menu démarrer on ouvre le SIMATIC WINCC Flexible comme suit :



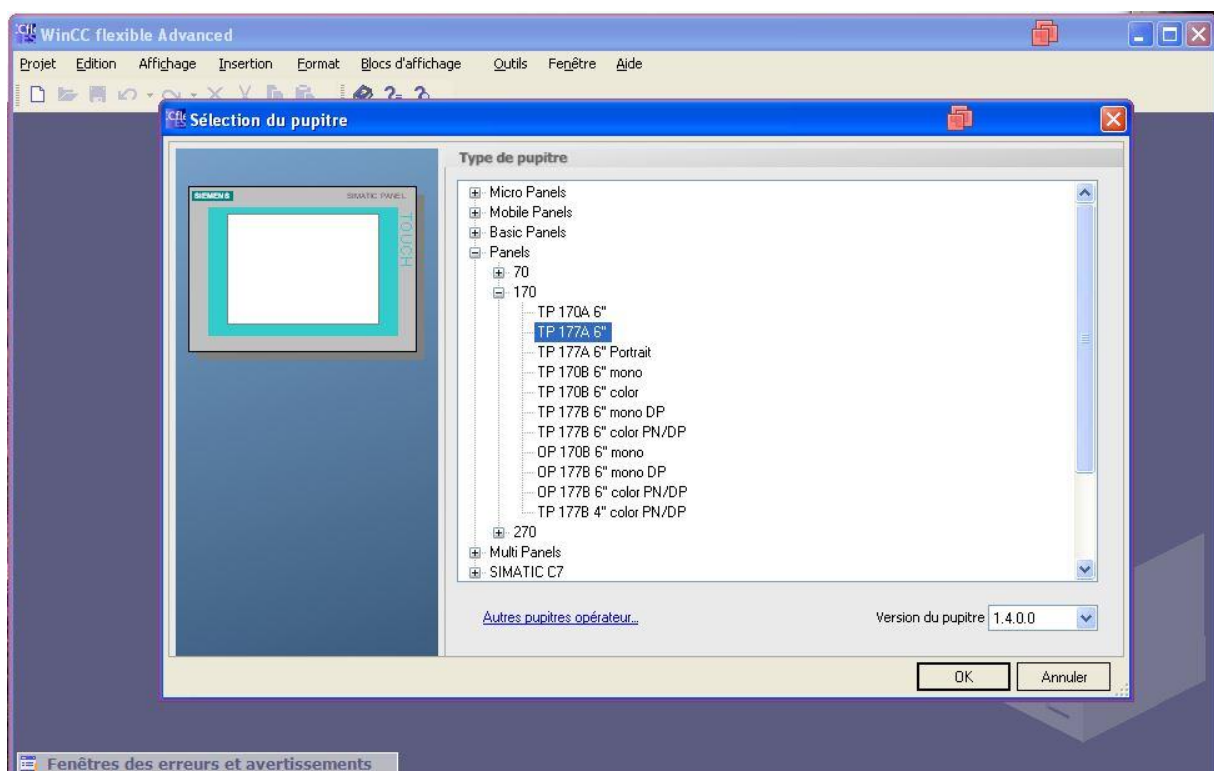
#### IV.1.2. Créer un projet vide

Après avoir lancé WinCC, la fenêtre suivante apparaît et on appuie sur « *Créer Projet Vide* ».

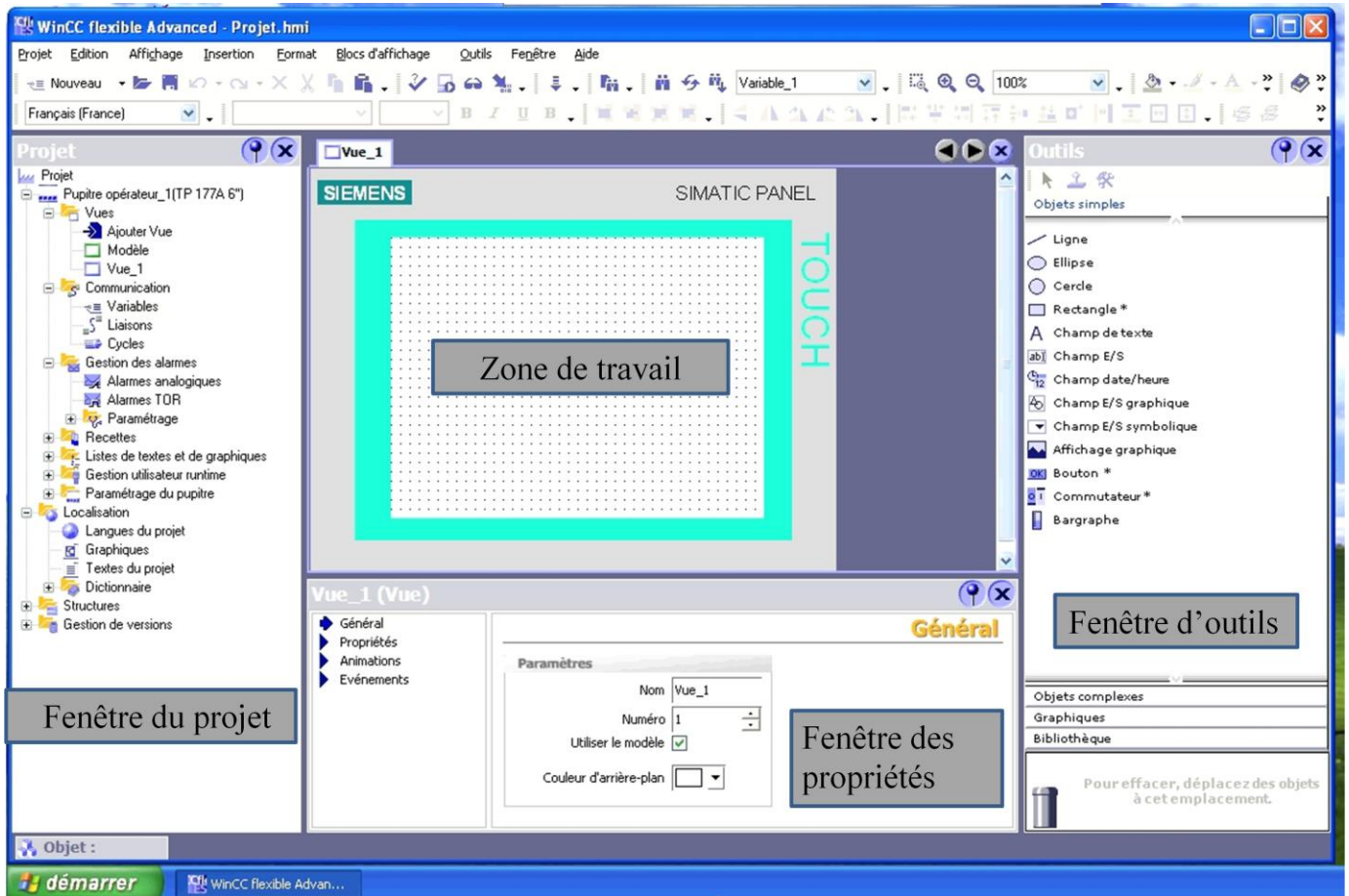


### IV.1.3 .Sélectionner le pupitre

Après avoir appuyé sur 'Créer Projet Vide', la fenêtre suivante apparaît pour la sélection du pupitre, et on choisit dans 'Panels' le pupitre «177A ».



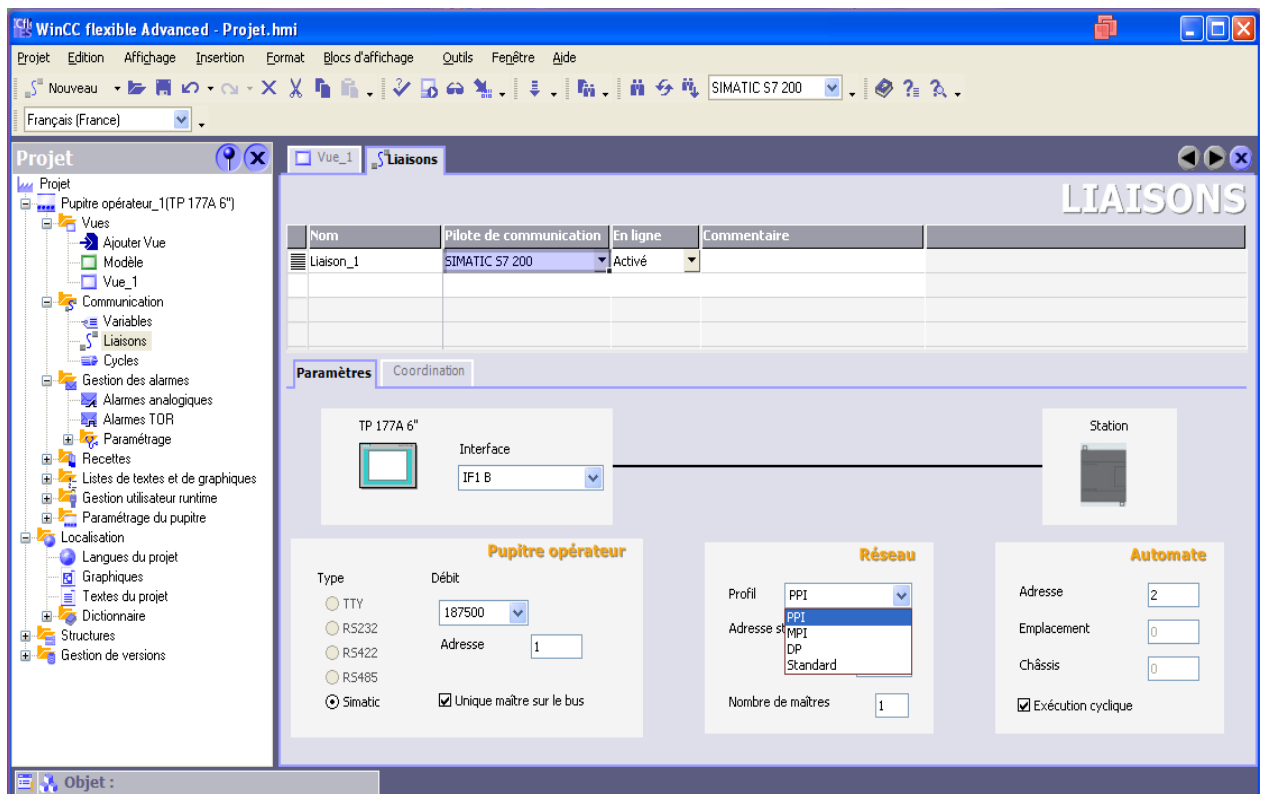
Une fois le projet créé, la fenêtre suivante s’affiche :



- La zone de travail sert à éditer les objets d’un projet.
- Dans la fenêtre du projet s’affichent tous les éléments et tous les éditeurs disponibles d'un projet, et qui peuvent être ouverts.
- Dans la fenêtre des propriétés on édite les propriétés des objets.
- Dans la fenêtre d'outils se trouve les objets qu’on peut insérer dans les vues (objets simples et complexes, objets graphiques).

#### IV.2. Configuration de la liaison

Après la création du projet, la configuration de la liaison entre le pupitre et l’API est nécessaire. Pour ce faire, on appuie dans la fenêtre « *Projet* » sur « *Liaison* », et la fenêtre ci-dessous apparait. On sélectionne dans la case « *Pilote de communication* » SIMATIC S7-200 et dans « *Paramètres* » on choisit le réseau PPI.

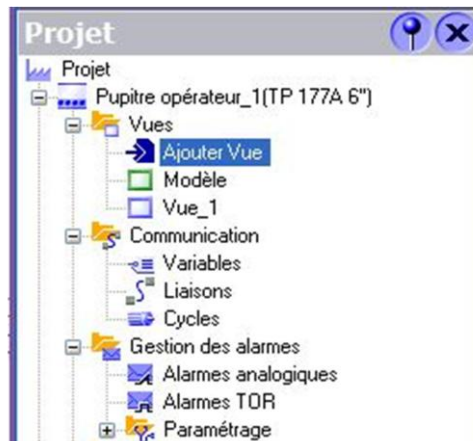


### IV.3. Création de vues

Les vues sont les éléments principaux du projet. Elles permettent de commander et de contrôler la bobineuse.

Les vues contiennent des objets comme les champs de sortie, les zones de texte et d'affichage permettant de représenter les différents constituants du pupitre.

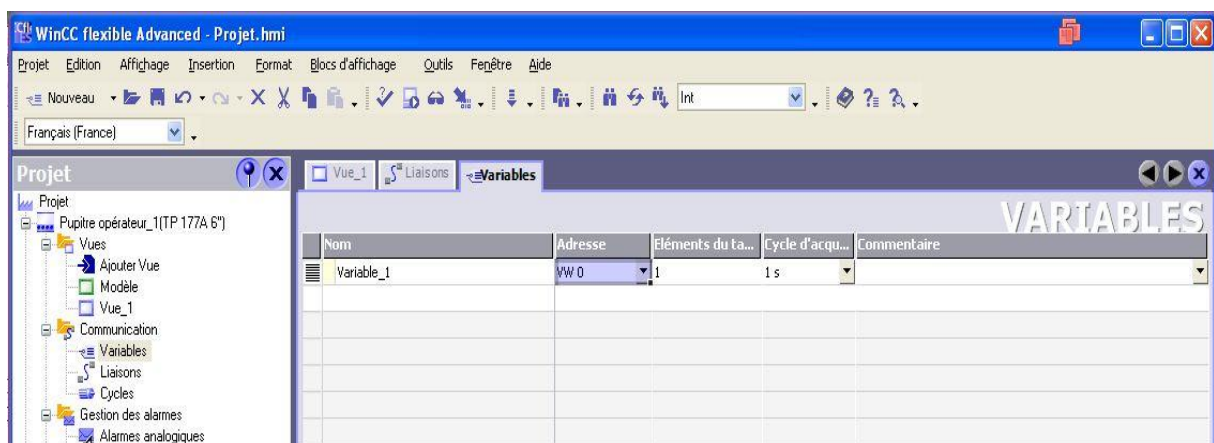
Le nombre de vues à créer dépend du nombre de menu à utiliser. Pour créer une vue, on double clic sur « *Ajouter vue* » dans la fenêtre de « *Projet* ».



#### IV.4. Création de variables

Les variables permettent de communiquer, c.-à-d. d'échanger des données entre le pupitre opérateur et l'automate programmable (champs d'entrée ou champs de sortie).

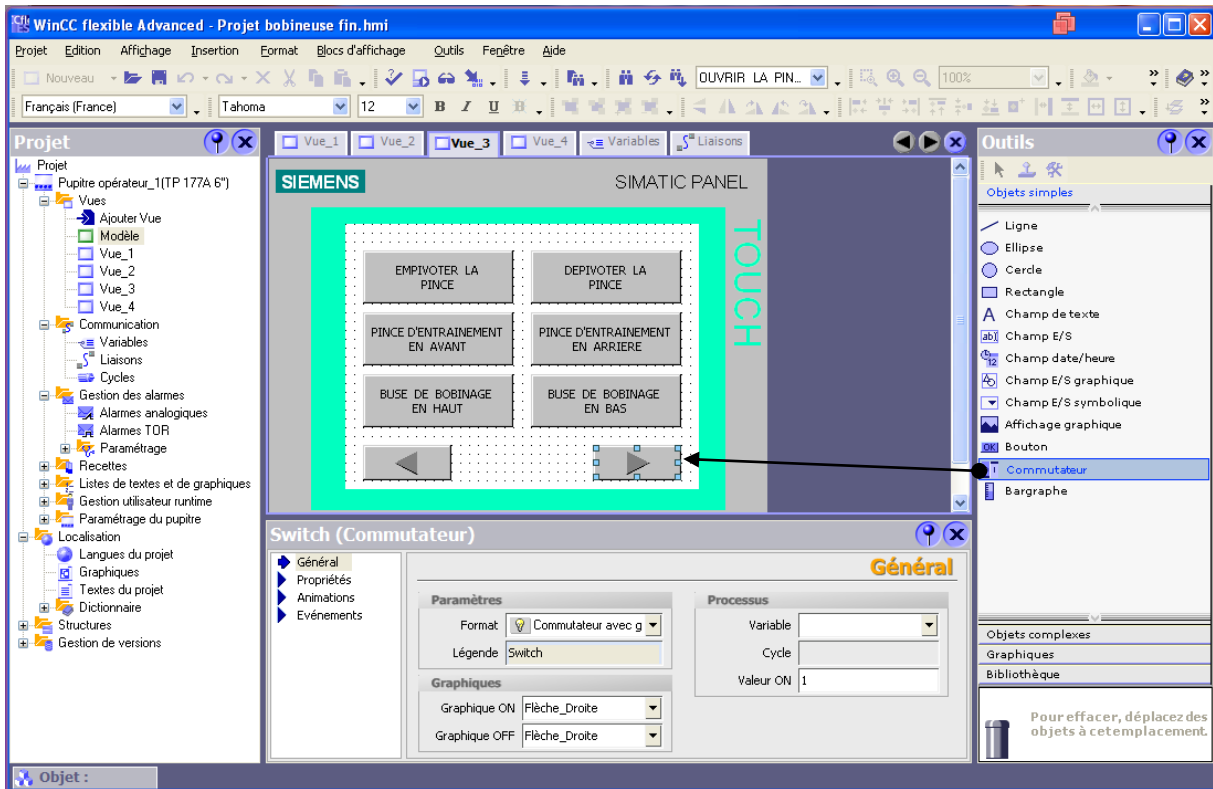
Pour créer une variable, on double clic sur « Variable » dans la fenêtre de « Projet ».



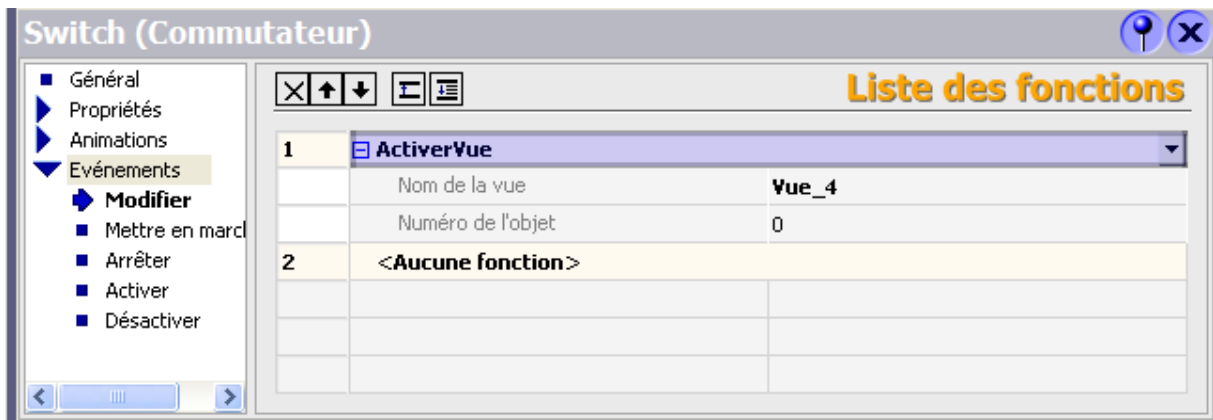
#### IV.5. Ajouter des changements de vues

Afin de pouvoir passer entre les différentes vues sur le panel en cours d'exploitation, on ajoute des changements de vues en créant des commutateurs. Pour cela, on procède comme suit :

- Créer un commutateur et lui associer une image en mode graphique



- Associer le commutateur à une vue :

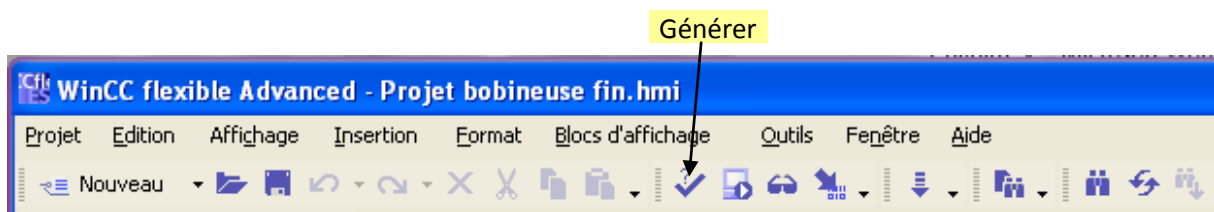


## IV.6. Test et simulation du projet

Avant de finir la configuration, on doit compiler et simuler le projet avec le simulateur de WinCC flexible (WinCC flexible Runtime), et cela en suivant ces étapes :

### Compilation du projet :

- ✚ Démarrer le contrôle de cohérence en cliquant sur « Générer » comme suit :



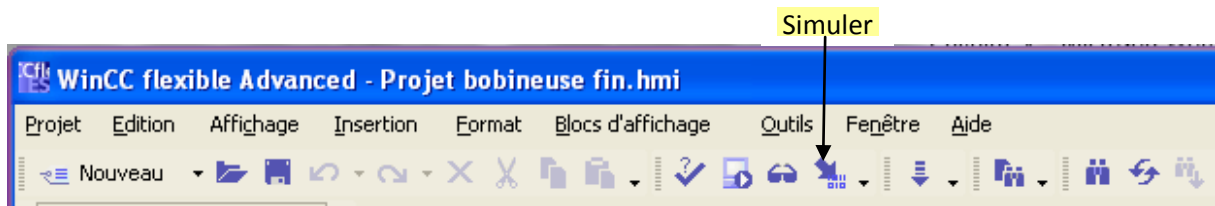
Le résultat du test est affiché dans la fenêtre des erreurs et avertissements

Heure	Catégorie	Description
03:56:34.87	Compilateur	Polices en cours de conversion ...
03:56:34.87	Compilateur	Vérification des résultats en cours ...
03:56:34.87	Compilateur	Aucune variable de process définie p...
03:56:34.88	Compilateur	Aucune variable de process définie p...
03:56:34.88	Compilateur	Aucune variable de process définie p...
03:56:34.88	Compilateur	Aucune variable de process définie p...
03:56:34.88	Compilateur	Aucune variable de process définie p...
03:56:34.88	Compilateur	Aucune variable de process définie p...
03:56:34.90	Compilateur	Aucune variable de process définie p...
03:56:34.90	Compilateur	Ecriture des fichiers de sortie en co...
03:56:34.90	Compilateur	Nombre de Power Tags utilisés : 23
03:56:34.90	Compilateur	Terminé avec 0 erreur(s), 7 avertisse...
03:56:34.90	Compilateur	Horodatage : 22/06/2011 03:56:05 - ...
03:56:34.90	Compilateur	Compilation terminée !

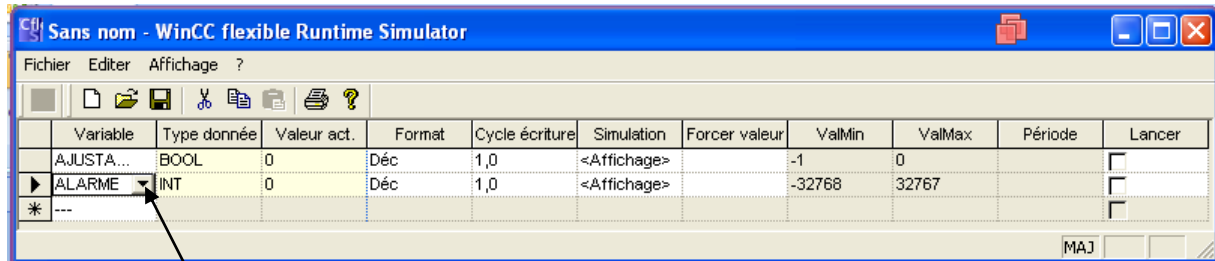
### Simulation du projet :

Avec le simulateur, on simule le projet directement sur le PC de configuration. La simulation permet de détecter des erreurs logiques de configuration avant de mettre le projet en marche.

1. Démarrer le simulateur :

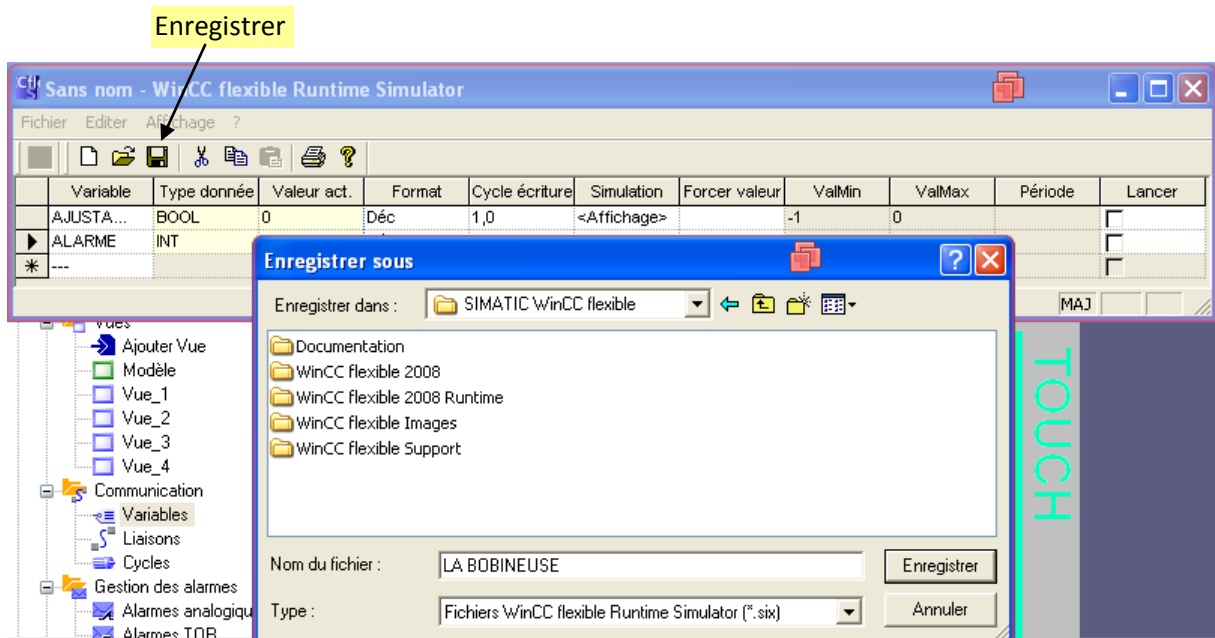


2. Sélectionner les variables :



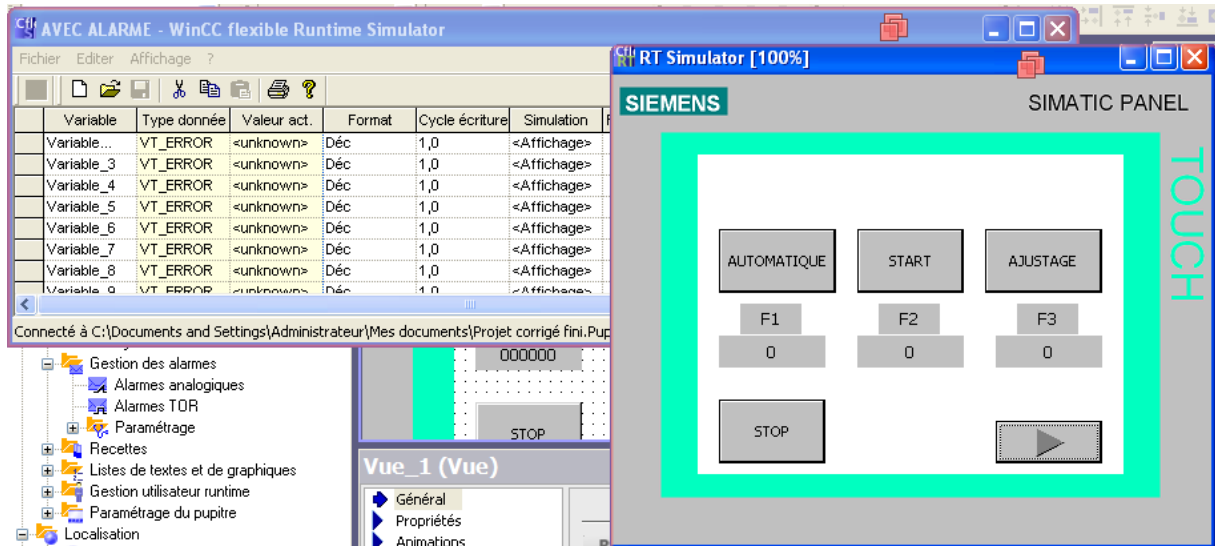
Sélectionner

3. Enregistrer la simulation :



Enregistrer

 Simulation :



### **Conclusion :**

Dans ce dernier chapitre, nous avons procédé à l'amélioration de la partie commande de la bobineuse en remplaçant le pupitre à entrée physique par un panel tactile.

Les différentes étapes de configuration du panel tactile offrent à l'opérateur de la bobineuse une meilleure utilisation, et un meilleur suivi de l'évolution du processus.

Ce stage pratique nous a permis de découvrir la réalité du milieu industriel. Il nous a non seulement permis de mettre en œuvre et d'approfondir nos connaissances théoriques acquises sur les automatismes au cours de notre cursus de graduation, mais aussi de développer un esprit de travail collectif.

Ce projet qui nous a été confié par l'entreprise algérienne Electro-Industries d'Azazga, consiste à l'amélioration de la machine bobineuse en l'automatisant par un automate programmable S7-200 (CPU 224) de la firme SIEMENS et améliorer la commande et la supervision du fonctionnement de la machine à l'aide d'un écran tactile 177A.

Après l'étude de la machine, nous avons pu comprendre son fonctionnement et détecter ses différentes insuffisances ce qui nous a permis de compléter le cahier des charges et nous fixer des objectifs.

Le fonctionnement automatique de la machine a d'abord été modélisé par GRAFCET, devenu plus qu'un outil graphique. Le modèle est ensuite traduit en langage à contact sur l'éditeur du Micro/Win puis transféré dans l'automate S7-200.

Pour la commande de la bobineuse, un écran tactile a été configuré à l'aide du logiciel WinCC Flexible. Le panel sert à commander et visualiser l'état de la bobineuse, ceci a réduit considérablement le volume de l'unité de commande.

Pour vérifier le bon fonctionnement du programme, nous avons eu recours au simulateur S7-200. Le programme fonctionne bien néanmoins une application n'est pas reconnue par la version du simulateur utilisée. Cette application est HDEF qui nécessite une autre version du simulateur, ainsi le programme serait complètement validé.

# Bibliographie

## ❖ Manuels

- [1] Manuel d'utilisation de la bobineuse (Electro-Industries).
- [2] Manuel d'utilisation de STEP 7 Micro/Win v4.0.
- [3] Manuel d'utilisation de Win CC Flexible 2008.

## ❖ Ouvrages

- [4] G. Michel. - Les A.P.I. : Architecture et applications des automates programmables industriels, 1988.
- [5] Automatisation et supervision de la machine à garnir les encoches " Encocheuse" à base d'un API S7-200 et une HMI TP 177A "6" présenté par : ZEDEK Razika et MEGHNEZ Samia, promotion 2010.
- [6] TOUDJINE Dyhia Meriem et MOUAZER Salah « Modernisation d'une machine à fabriquer les cales d'encoche par un API S7-200 et un Panel tactile TP177 A », promotion 2010.
- [7] AZZOUG Younes et BOUDJEMAI Nassima « Régulation et supervision d'un procédé d'étalonnage de débitmètre », promotion 2009.

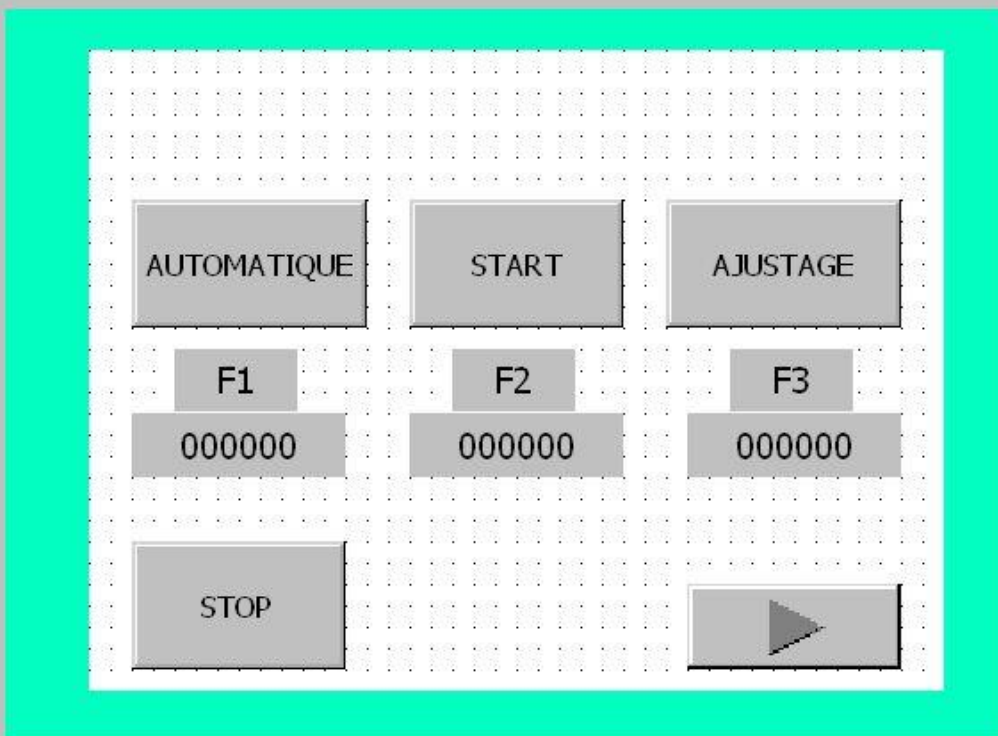
## ❖ Liens internet :

- [8] <http://www.technologuepro.com/cours-automate-programmable-industriel/Cours-Grafcet-notions-de-base.htm>
- [9] [http://fr.wikipedia.org \"transformateur d'isolement\"](http://fr.wikipedia.org \)
- [10] <http://abdellah.chelouah.perso.neuf.fr/>
- [11] <http://www.mei-latecoere.com/>
- [12] <http://www.dpulsa.fr/>
- [13] <http://www.audin.fr/pdf/documentations/omron/capteurs-de-proximite/E2FM.pdf>

SIEMENS

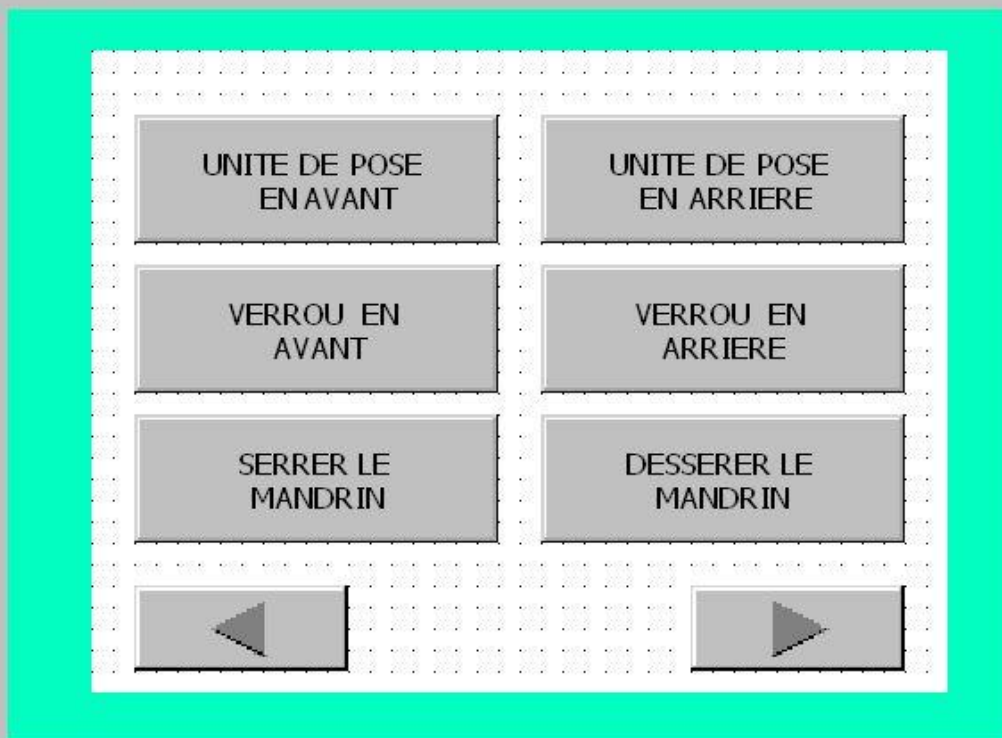
SIMATIC PANEL

TOUCH



SIEMENS

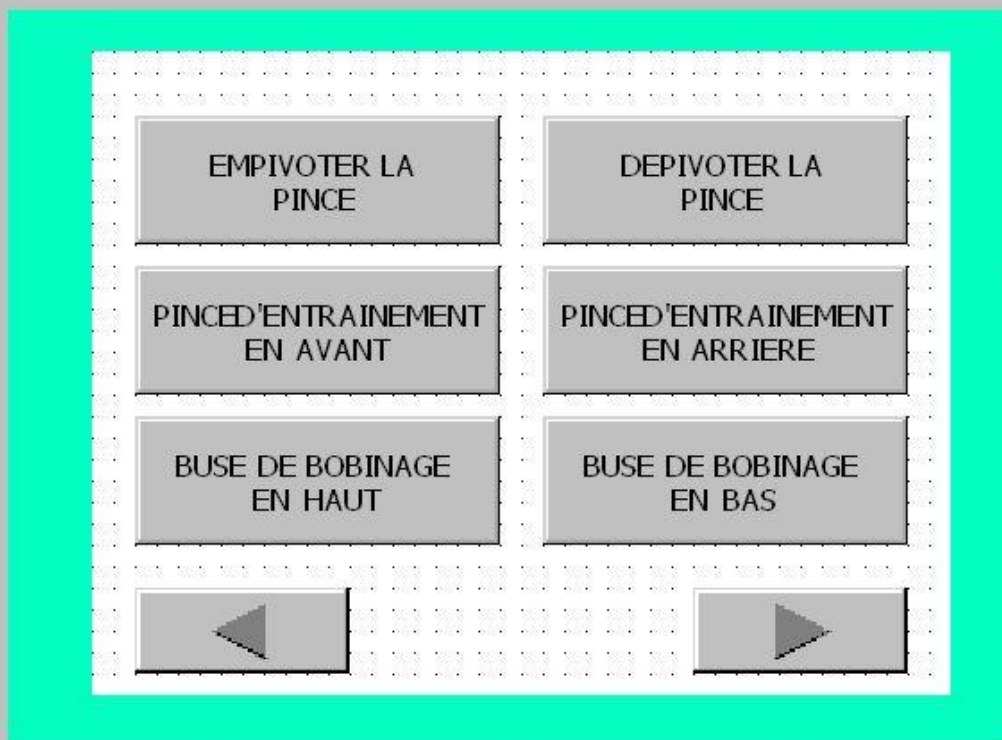
SIMATIC PANEL



TOUCH

SIEMENS

SIMATIC PANEL



TOUCH

SIEMENS

SIMATIC PANEL

TOUCH

OUVRIR LA PORTE  
DE PROTECTION

FERMER LA PORTE  
DE PROTECTION

OUVRIR LA PINCE  
D'ENTRAINEMENT

FERMER LA PINCE  
D'ENTRAINEMENT

