

**République Algérienne Démocratique et Populaire**  
**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**  
**Université Mouloud MAMMERI, Tizi-Ouzou**



**Faculté de Génie Electrique et d'Informatique**  
**Département d'Automatique**

**Mémoire de Fin d'Etudes**

En vue de l'obtention du diplôme

*Master 2 Professionnel en Automatique*  
*Option: Automatique et Informatique Industrielles*

***Thème***

**Automatisation d'un palettiseur  
avec un automate programmable**

Encadré par :

M<sup>me</sup> Nadia DJEGHALI  
Co-encadré par :  
Lamia HAMACI

Présenté par :

Rabah TAAZIBT  
Abderrahim FELKAT

Soutenu le: 20/09/2015

**Promotion 2015**

Le travail a été réalisé au sein du centre d'enfûtage NAFTAL, Oued Aissi.

# Remerciements

Nous tenons tout d'abord à remercier Dieu le tout puissant, qui nous a donné la force et la patience d'accomplir ce modeste travail.

Nous voudrions également adresser toute notre gratitude à la promotrice de ce mémoire, Mme DJEGHALI Nadia, pour sa patience, sa disponibilité et surtout ses judicieux conseils, qui ont contribué à alimenter notre réflexion.

Nous tenons également à remercier Mr. CHARIF Moussa qui n'as pas hésité de nous donner de son temps et qui nous a été d'une grande aide en ce qui concerne la programmation.

Nous tenons aussi à remercier la société NAFTAL qui nous a ouvert ses portes pour notre stage, notre Co-promotrice sur le site Mme HAMACI Lamia et toute personne qui aurait participé de près ou de loin au bon déroulement de notre mémoire.

Nous remercions aussi les membres du jury pour l'honneur qu'ils nous ont fait en acceptant d'examiner notre travail.

## *Dédicaces*

*Je dédie ce travail à  
Mes très chers grands parents,  
À la mémoire de mon grand-père,  
Mes très chers parents,  
Mes très chères sœurs,  
Mes très chers frères,  
Mon binôme qui m'a supporté toute ces années,  
Tous mes amis et collègues d'études,  
Et à tous ceux qui me sont chers.*

*Rahim*

*Je dédie ce travail à  
Mes chers parents, jamais je ne saurais m'exprimer quant aux sacrifices et aux dévouements  
que vous avez consacrés à mon éducation et mes études,  
A mes chers frères Tarik et Anis,  
A tous mes chers amis,  
A mon cher ami mon binôme Rahim qui m'a supporté durant ces années et avec lequel j'ai  
trouvé l'entente dont j'avais besoin.*

*Rabah*

## Sommaire

<b>Introduction générale</b> .....	1
<b>Chapitre I: Système de palettisation de bouteilles de gaz.</b>	
I.1 Introduction .....	3
I.2 Présentation de l'entreprise Naftal .....	3
I.2.1 Historique .....	3
I.2.2 Organisation .....	3
I.2.3 Organigramme de L'entreprise .....	4
I.2.4 Description du centre d'enfûtage d'Oued-Aissi .....	5
I.2.4.1 Architecture du centre d'enfûtage d'Oued-Aissi .....	7
I.3 La production de bouteilles de gaz butane B13 .....	7
I.3.1 Déchargement de bouteilles .....	7
I.3.2 Acheminement des bouteilles .....	8
I.3.3 Pré pesage .....	8
I.3.4 Introduction des bouteilles .....	8
I.3.5 Remplissage de bouteilles .....	9
I.3.6 Ejection de bouteilles .....	9
I.3.7 Contrôleur de la tare .....	10
I.3.8 Contrôle de fuite .....	10
I.3.9 Chargement .....	10
I.4 Fonctionnement du palettiseur .....	11
I.4.1 La palettisation .....	11
I.4.1.1 Pose des casiers de bouteilles vides .....	12
I.4.1.2 Transfert des casiers .....	13
I.4.1.3 Déchargement des casiers .....	13
I.4.1.4 Chargement du casier .....	16
I.4.1.5 Enlèvement du casier .....	18
I.4.1.5.1 Poste d'enlèvement intermédiaire .....	18
I.4.1.5.2 Poste d'enlèvement de casiers pleins .....	18
I.5 Conclusion .....	18

## Chapitre II: Généralités sur les systèmes automatisés

II.1 Introduction .....	19
II.2 Structure d'un système automatisé .....	19
II.2.1 La Partie Opérative .....	20
II.2.2 La Partie commande .....	22
II.3 L'Automate Programmable Industriel (API) .....	22
II.3.1 Définition .....	23
II.4 Les avantages et les inconvénients d'un API .....	23
II.5 Structure d'un automate programmable industriel .....	24
II.5.1 Aspect extérieur .....	24
II.5.1 Remarque .....	25
II.5.2 Structure interne d'un automate programmable industriel .....	25
II.5.2.1 Le processeur .....	26
II.5.2.2 La zone mémoire .....	26
II.5.2.3 Les modules Entrées/Sorties .....	27
II.5.2.4 Le module d'Alimentation .....	27
II.5.2.5 Le bus interne .....	28
II.6 Principe de fonctionnement d'un automate programmable industriel .....	28
II.7 Les types des automates programmables industriels .....	29
II.8 Les langages de programmation d'un API .....	29
II.10 Critères de choix d'un API .....	31
II.11 Présentation de l'automate Siemens S7-300 .....	31
II.11.1 Caractéristique de S7-300 .....	31
II.11.2 Constitution de l'automate S7-300 .....	32
II.11.2.1 Modules d'alimentation (PS) .....	32
II.11.2.2 Unité centrale (CPU) .....	33
II.11.2.2.1 Interface (MPI) .....	34
II.11.2.2.2 Signalisation d'état .....	34
II.11.2.2.3 Commutateur de mode .....	34
II.11.2.2.4 Carte mémoire .....	35
II.11.2.2.5 La pile .....	35
II.11.3 Les modules d'entrées/Sorties.....	35
II.11.3.1 Modules d'entrées .....	35
II.11.3.2 Les modules de sorties .....	36

II.11.4 Coupleur (IM) .....	36
II.11.5 Module de signaux (SM) .....	36
II.11.6 Module de fonction (FM).....	36
II.11.7 Modules de communication (CP).....	36
II.12 Fonctionnement de l'automate S7-300.....	37
II.12.1 Réception des informations sur les états du système .....	37
II.12.2 Exécution du programme utilisateur .....	37
II.12.3 Commande du processus.....	37
II.13 Programmation de l'automate S7-300.....	37
II.14 Conclusion .....	37

### **Chapitre III: Modélisation et simulation du palettiseur**

III.1 Introduction .....	38
III.2 Généralités sur le GRAFCET.....	38
III.2.1 Grafcet niveau 1 .....	38
III.2.2 Grafcet niveau 2 .....	38
III.3 Eléments graphiques de base du GRAFCET .....	39
III.3.1 Les étapes .....	39
III.3.2 Les transitions .....	39
III.4 Convergence et divergence en « ET » et « OU » .....	40
III.4.1 Divergence simple en ET .....	40
III.4.2 Divergence en OU .....	40
III.4.3 Convergence en ET .....	40
III.4.4 Convergence en OU .....	40
III.5 Règles d'évolution du GRAFCET .....	41
III.5.1 Règle 1 : Situation initiale .....	41
III.5.2 Règle 2 : Franchissement d'une transition .....	41
III.5.3 Règle 3 : Evolution des étapes actives .....	41
III.5.4 Règle 4 : Evolution simultanée .....	41
III.5.5 Règle 5 : Activation et désactivation simultanée d'une étape .....	41
III.6 Modélisation avec du Grafcet niveau 1 .....	42
III.7 Présentation de l'outil Step 7 .....	46

III.8 Principe de la programmation en Step7 .....	47
III.8.1 Structure du programme .....	47
III.8.2 Créer son projet .....	48
III.8.3 Configuration de la station .....	51
III.8.4 Edition .....	52
III.8.5 Edition des mnémoniques .....	53
III.9 Le programme en step7 graph .....	54
III.10 Tableau des mnémoniques .....	58
III.11 Résultats de simulation .....	60
III.11.1 Introduction sur PLCSIM .....	60
III.11.2 Résultats de simulation de notre programme .....	61
III.12 Conclusion .....	64

## **Chapitre IV: Système de supervision du palettiseur**

IV.1 Introduction .....	65
IV.2 WinCC flexible .....	65
IV.2.1 Utilisation de SIMATIC WinCC flexible .....	65
IV.2.2 WinCC flexible Runtime .....	65
IV.2.3 La liaison WinCC avec PLCsim .....	66
IV.3 Vues .....	66
IV.3.1 Avantages de l'intégration dans Step7 .....	67
IV.4 Simulation .....	68
IV. 5 Conclusion .....	70
<b>Conclusion générale</b> .....	<b>71</b>



**Introduction générale**

L'automatisation des applications industrielles joue un rôle de plus en plus important dans l'économie mondiale et dans l'expérience quotidienne. La compétition technologique et commerciale entre les différents distributeurs mondiaux des composants d'automatisme a accéléré l'évolution des systèmes industriels et des applications automatisées bien développées. Actuellement, les automates programmables sont fonctionnels dans plusieurs domaines. De par la simplicité de leur mise en œuvre et de leur implémentation, ils occupent une place importante dans les processus d'automatisations.

Un Automate Programmable Industriel, API (en anglais Programmable Logic Controller, PLC) est un type particulier d'ordinateur, robuste et réactif, ayant des entrées et des sorties physiques, utilisé pour automatiser des processus comme la commande des machines sur une ligne de montage dans une usine, ou le pilotage de systèmes de manutention automatique. Là où les systèmes automatisés plus anciens employaient des centaines ou des milliers de relais et de cames, un simple automate suffit.

L'une des principales applications industrielles est le système de palettisation d'un produit. La palettisation consiste à charger un produit dans des palettes (casiers). Ces palettes seront après regroupées afin d'être démenagées de la salle de production puis stockées dans des dépôts, etc. Le système de palettisation joue un grand rôle dans la protection des produits au moment de leur livraison et de leur utilisation par le client.

L'automatisation du système de palettisation de bouteilles de gaz est le projet que nous avons en charge chez l'entreprise Naftal comme un projet de fin d'études. Le but est de parvenir à automatiser la palettiseuse de bouteilles de gaz qui fonctionne pour le moment en logique pneumatique en intégrant un automate programmable de type S7-300 tout en gardant le mode de fonctionnement actuel mais en réduisant le nombre d'opérateurs.

Pour rendre compte du travail que nous avons réalisé, nous organisons notre mémoire en quatre chapitres.

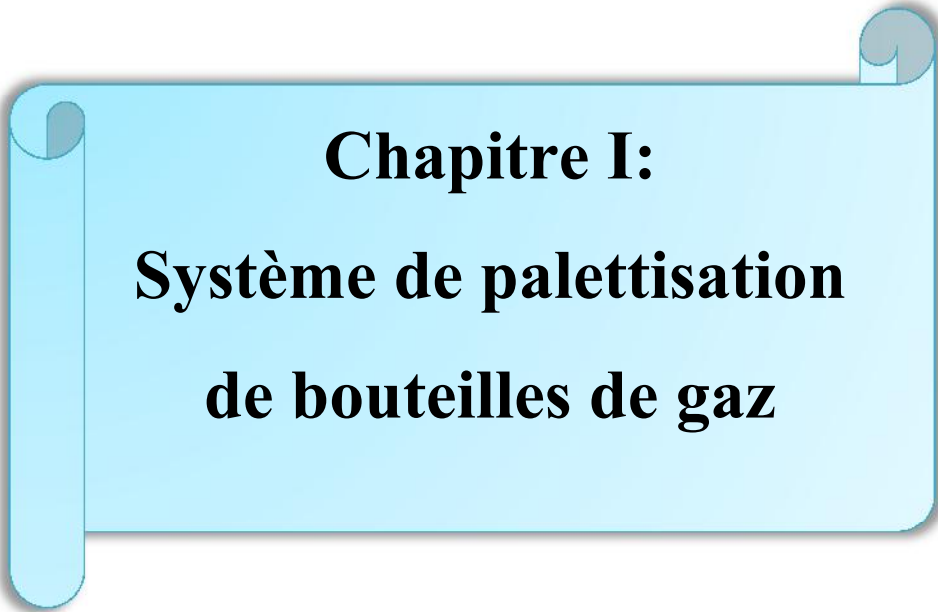
Dans le premier chapitre, après une brève présentation de l'entreprise Naftal, nous passons à la description du processus de production de bouteilles de gaz butane, puis nous présentons les différentes parties qui composent le palettiseur étudié et le principe de son fonctionnement.

Dans le deuxième chapitre, nous présentons les notions principales concernant les systèmes automatisés et les API en particulier. Avec une étude approfondie sur l'automate S7-300 que nous avons utilisé dans notre projet et de son langage de programmation Step 7.

Dans le troisième chapitre, nous avons modélisé le fonctionnement du palettiseur par un GRAFCET, puis nous l'avons traduit en langage S7-GRAPH pour l'implanter dans notre automate, et nous avons utilisé le PLCSIM pour simuler le fonctionnement du palettiseur.

Dans le quatrième chapitre, nous proposons de mettre en place un système de visualisation numérique par des écrans de contrôle digitaux. Nous l'avons réalisé avec le WinCC.

Enfin, ce travail sera complété par une conclusion générale à travers laquelle on exposera les principaux résultats obtenus.

A light blue scroll graphic with a dark blue border and a drop shadow. The scroll is unrolled in the center, with the top and bottom edges curled up. The text is centered on the unrolled portion.

**Chapitre I:**  
**Systeme de palettisation**  
**de bouteilles de gaz**

## I.1 Introduction

L'une des principales applications industrielles est le système de palettisation de bouteilles de gaz à l'entreprise Naftal située à Oued-Aissi. Ce système est responsable de chargement de bouteilles de gaz pleines dans des casiers (palettes). Ces casiers seront après regroupés afin d'être déménagés de la salle de production puis stockés dans des dépôts, etc. L'automatisation du système de palettisation est le projet que nous avons en charge chez l'entreprise Naftal comme un projet de fin d'études.

Dans ce chapitre, après une brève présentation de l'entreprise Naftal, nous passons à la description du processus de production de bouteilles de gaz butane, puis nous présentons les différentes parties qui composent le palettiseur étudié et le principe de son fonctionnement.

## I.2 Présentation de l'entreprise Naftal [1]

### I.2.1 Historique

Issue de SONATRACH, l'Entreprise ERDP (Entreprise nationale de Raffinage et de Distribution de produits Pétroliers) a été créée par le décret N° 80/101 du 06 avril 1981. Entrée en activité le 1<sup>er</sup> janvier 1982, elle est chargée de raffinage et de distribution des produits pétroliers.

En 1987, l'activité du raffinage est séparée de celle de distribution et attribuée à NAFTEC. NAFTAL est désormais chargée uniquement de la commercialisation et de la distribution des produits pétroliers et de leurs dérivés. En 1998, elle change de statut et devient une Société Par Actions (SPA) et filiale à 100% de SONATRACH.

NAFTAL a pour mission principale, la distribution et la commercialisation des produits pétroliers sur le marché national. Elle a aussi d'autres missions, telles que:

- Enfutage des GPL (Gaz de Pétrole Liquéfié).
- Formation de bitume.
- Distribution, stockage et commercialisation des carburants, GPL, lubrifiants, bitumes, pneumatiques, GPL carburant (GPL/c) et produits spéciaux.
- Transport des produits pétroliers.

### I.2.2 Organisation

NAFTAL SPA, filiale à 100% de SONATRACH est organisée en trois branches. Elle a pour mission principale, la distribution et la commercialisation des produits pétroliers sur le marché national.

En quelques chiffres : Avec un personnel de 30 000 travailleurs, NAFTAL est le premier distributeur de produits pétroliers en Algérie. Elle contribue à hauteur de 51% de l'énergie finale en fournissant 10 millions de tonnes de produits pétroliers par an sous forme de:

- Carburants (8 millions de TM (Tonne Métrique)).
- Gaz de pétrole liquéfié (plus de 1.6 million de TM).
- Bitumes (plus de 0.5 million de TM).
- Lubrifiants (plus de 70 000 TM).

Pour cela elle dispose de :

- 49 centres et dépôts de distribution et de stockage de carburants.
- 22 centres et 27 magasins lubrifiants et pneumatiques.
- 26 centres et dépôts aviation, 06 centres marine.
- 49 dépôts relais de stockage de GPL.
- 41 centres d'emplissage de GPL d'une capacité d'enfûtage de 1.2 millions tonnes/an.
- 03 centres vrac de GPL.
- 15 unités de bitumes d'une capacité de formulation de 360.000 tonnes/an.
- 3000 véhicules de distribution et 800 engins de manutention et de maintenance.
- 730 Km de canalisation.

### **I.2.3 Organigramme de L'entreprise**

L'organigramme de l'entreprise Naftal est illustré par la figure I.1 ci-dessous :

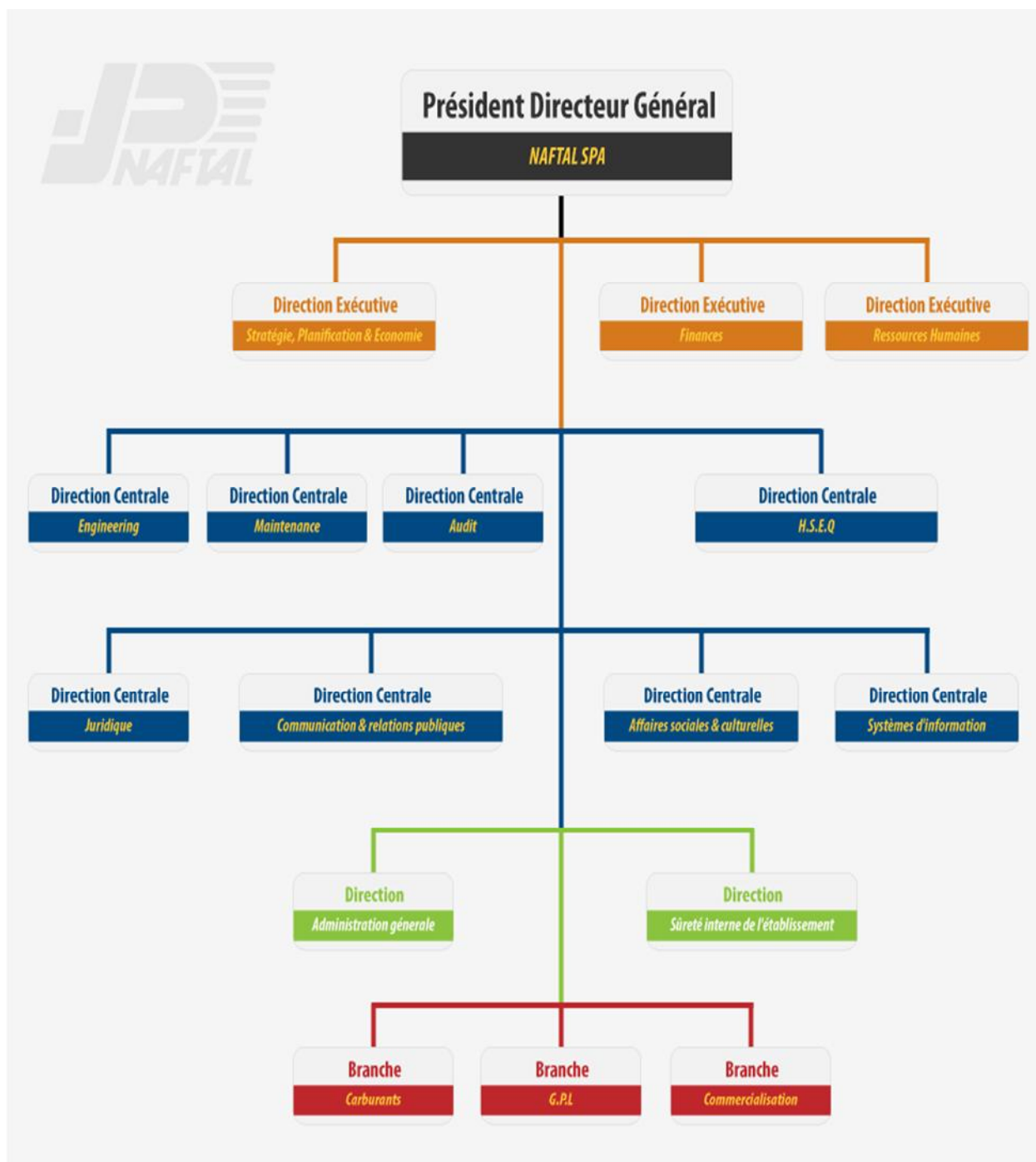


Fig I.1 Organigramme de l'entreprise

#### I.2.4 Description du centre d'enfûtage d'Oued-Aissi

Le district NAFTAL GPL de Tizi-Ouzou est composé des infrastructures suivantes:

- Un siège administratif et un atelier de maintenance du matériel roulant à Tala-Athmane.
- Un centre d'enfûtage de GPL à Freha.

- Un centre d'enfûtage de GPL à Oued-Aissi.
- Un micro-centre et un dépôt relais de GPL à Ouadhias.
- Un dépôt relais de GPL à Ain-El-Hammam.
- Un dépôt relais de GPL à Tizirt.



**Fig I.2** Centre d'enfûtage d'Oued-Aissi

Le centre d'enfûtage d'Oued-Aissi où s'est déroulé notre stage a pour mission l'approvisionnement de la wilaya de Tizi-Ouzou en produits GPL conditionnés (butane et propane) ainsi que d'une source de carburant issue des GPL, à savoir, le sirghaz (GPL/c). Pour assurer la production et la disponibilité de ces derniers, le centre d'enfûtage est ravitaillé en produit vrac (butane et propane) à partir de la raffinerie d'Alger avec des moyens propres (tracteurs, citernes) ainsi que des moyens tiers (tracteurs) pour faire face à la demande accrue en période hivernale.

Le centre d'enfûtage dispose de moyens de stockage divers pour les GPL: des sphères pour le butane au nombre de trois et des cages pour le propane au nombre de deux. Le conditionnement de ces produits GPL se fait dans des bouteilles B13 (13 kg net+13kg butane) et B3 (3 kg net + 3kg butane) ainsi qu'un emballage P11 (13kg net+11kg propane), P35 (35 kg net+35kg propane).

Le propane est aussi utilisé comme carburant et exploité tel qu'il est par NAFTAL. En hiver, le GPL/c est composé de 100% de propane en volume, par contre en été NAFTAL fait un mélange de 80% de propane et de 20% de butane. Cependant, les nouvelles lois en vigueur imposent à NAFTAL de produire un GPL/c en été constitué de 50% de propane et 50% de butane.

Le centre assure également l'approvisionnement en butane conditionné des dépôts relais sus cités ainsi que des stations essence à gestion direct GD et privée situées dans sa zone de distribution. Il assure aussi les livraisons en propane conditionné et sirghaz à travers toute la wilaya.

#### **I.2.4.1 Architecture du centre d'enfûtage d'Oued-Aïssi**

Le centre d'enfûtage d'Oued-Aïssi se compose des installations suivantes :

- Un poste de dépotage des camions citerne et d'une pomperie GPL.
- Une zone de stockage de GPL vrac, constituée de trois sphères de butane de 5000 tonnes chacune et de deux cigares de propane de 75 tonnes chacun.
- Un hall d'emplissage GPL abritant:
  - Deux chaînes de production du butane (B13) d'une capacité de 24 bascules chacune.
  - Une bascule de conditionnement du butane en B3.
  - Quatre bascules de conditionnement du propane en P35.
  - Une bascule de conditionnement du propane en P11.
  - Un atelier de rée preuve des différents emballages.
- Une centrale d'air comprimé composée de quatre compresseurs.
- Une centrale électrique.
- Un groupe électrogène d'une capacité de 600 KVA.
- Une salle de contrôle permettant le suivi du système de capteurs (feu, gaz, fumée et briseglace) ainsi qu'un système de télé jaugeage (contrôle des stocks vrac).
- Bâche à eau avec pompes d'incendie.
- Magasin de PDR (Pièces De Rechange), installations fixes et matériel roulant.
- Centrale du réseau anti-incendie (une motopompe et deux électropompes).

### **I.3 La production de bouteilles de gaz butane B13**

Cette partie a pour but de définir l'ensemble du fonctionnement actuel de la chaîne de production de bouteilles du gaz butane.

Deux chaînes de production assurent le conditionnement du gaz butane dans des bouteilles B13 (13 Kg net + 13 kg produit). Le processus de conditionnement du gaz butane est constitué des étapes suivantes :

#### **I.3.1 Déchargement de bouteilles**

Les bouteilles vides sont déchargées à partir des palettes au niveau du système de palettisation à l'aide d'un vérin hydraulique semi-automatique commandé par une manette possédant trois positions (P0: Arrêt, P1: Avance, P2: Reculer). Chaque palette est composée de 7 lignes de 5 bouteilles donc 35 bouteilles au total. L'opérateur décharge la palette ligne par ligne et la palette vidée est envoyée vers le poste de chargement.

### I.3.2 Acheminement des bouteilles

Les bouteilles déchargées sont acheminées vers les différents postes de la chaîne de production à l'aide d'un convoyeur à chaîne.

Deux chaînes de manutention entraînant en continu par traction les bouteilles présentes sur le convoyeur. Toutefois, lorsque cela s'avère nécessaire, les bouteilles peuvent être stoppées à partir d'un équipement complémentaire sans pour autant arrêter le fonctionnement des chaînes de manutention.



Fig I.3 Convoyeur à chaîne

### I.3.3 Pré pesage

Un opérateur se charge de relever la tare de la bouteille, puis encoder cette valeur à l'aide d'un clavier. Les bouteilles sont pesées sur la bascule en amont du carrousel (système de remplissage), pour vérifier s'il y a du gaz résiduel dans la bouteille et pour vérifier si la tare lue correspond à la tare pesée.

La bouteille arrive sur le convoyeur à chaîne activant ainsi la cellule photoélectrique B03, l'arrêt d'accumulation (C1/C1A) se ferme immédiatement devant la bouteille. L'opérateur introduit la tare de la bouteille, l'arrêt d'accumulation (C1/C1A) s'ouvre et la bouteille continue vers la bascule de pré pesage. Si la bouteille est en mauvaise condition, ou si le robinet et mauvais ou ne fonctionne pas, l'opérateur donne un signal et la bouteille est éjectée (l'éjecteur (C3) pousse la bouteille vers le convoyeur à rouleaux).

La table élévatrice (C2) s'élève et la bouteille est tarée. Après un certain temps prédéterminé, la table élévatrice (C2) redescend. S'il y a concordance entre la tare lue/encodée et la tare pesée, la bouteille avance à la position d'introduction. Si le contrôleur CUC (Crisplant Universal Controller) enregistre une bouteille avec une tare négative, la porte d'introduction (C6) se soulève pour empêcher la bouteille d'avancer vers l'entrée. La bouteille est arrêtée et l'éjecteur (C3) pousse la bouteille vers le convoyeur à rouleaux. Après l'éjection, l'éjecteur revient à sa place et la porte d'introduction (C6) descend.

**Remarque I.1:** Si l'arrêt d'accumulation ne s'ouvre pas et/ou la table élévatrice ne descend pas, c'est probablement dû au trop plein d'accumulation sur le convoyeur à rouleaux (la cellule photoélectrique B10 détecte les bouteilles) ou bien les bouteilles se sont accumulées après l'éjecteur (la cellule photoélectrique B09 détecte les bouteilles).

### I.3.4 Introduction des bouteilles

Une fois la bouteille arrivée au poste d'introduction, la cellule photoélectrique B13 est activée (détection de bouteille). La bouteille est à l'arrêt et n'attend que la première machine d'emplissage soit en position pour pouvoir entrer.

Un nombre de plaques codées est monté en dessous du bâti du carrousel. Chaque plaque codée correspond à une machine de remplissage. Sur chaque plaque codée, il ya deux codes binaires. Le code figurant en haut est une « horloge », il est identique pour chaque plaque codée et sert à indiquer la position de la machine d'emplissage. Le code en bas, nommé le code Manchester, est individuel pour chaque plaque codée et sert à identifier la machine de remplissage en question.

La plaque codée passe à travers le lecteur de code d'introduction. La fibre optique B01 lit l'horloge et la fibre optique B16 lit le code Manchester. Au même temps la disponibilité éventuelle d'une bouteille sur la machine de remplissage est constatée par la cellule photoélectrique B14, et un miroir réflecteur monté sur la machine d'emplissage est reconnu par la cellule photoélectrique B15 afin de constater si la tête de remplissage est dans sa position basse. Si une ou les deux conditions ne sont pas vérifiées et confirmées, aucune bouteille ne sera introduite sur la machine de remplissage. Si au contraire, la machine de remplissage est vide, et la tête de remplissage est à sa position initiale, une bouteille est introduite sur la machine de remplissage (C5). Après l'introduction, le bras d'introduction se renverse et le capteur inductif magnétique B11 est activé, et informe le contrôleur CUC que le dispositif d'introduction est libre pour recevoir une nouvelle bouteille.

### I.3.5 Remplissage de bouteilles

Une fois la bouteille introduite dans la machine de remplissage qui lui correspond, l'opérateur fixe la tête de remplissage. En tout, il y a 24 machines de remplissage et chacune d'elles est équipée d'un débitmètre massique qui permet de fournir la bouteille en la quantité du gaz qui lui correspond.

Une fois le remplissage fini, le débitmètre massique affiche un message sur son afficheur LCD (Ecran à Cristaux Liquides) pour indiquer que le remplissage est fini, afin que l'opérateur puisse enlever la tête de remplissage.



**Fig I.4** Carrousel de remplissage

### I.3.6 Ejection de bouteilles

L'éjection des bouteilles tangentiellement sur le convoyeur à chaîne se réalise de manière automatique.

A l'arrivée de la machine de remplissage au poste d'éjection, la plaque codée passe à travers le lecteur de code de l'éjection, la fibre optique B01 lit l'horloge et la fibre optique B16 lit le code Manchester. Au même temps, la disponibilité éventuelle d'une bouteille sur la machine de remplissage est constatée par la cellule photoélectrique B14, et un miroir réflecteur monté sur la machine de remplissage est reconnu par la cellule photoélectrique B15 afin de

constater si la tête de remplissage est dans sa position basse. Si le rapport de B14 dit qu'il n'y a aucune bouteille disponible sur la machine d'emplissage, et/ou que le rapport de B15 dit que la tête d'emplissage est dans sa position basse, l'unité de commande restera dans sa position initiale, et aucune bouteille ne sera éjectée de la machine d'emplissage. Au cas contraire, si une bouteille disponible est observée sur la machine d'emplissage, et la tête d'emplissage est située dans sa position initiale, la plaque supérieure de l'unité de commande (C10) sera levée et touchera une vanne à rouleaux, montée sur le bâti du carrousel (une par machine d'emplissage). Au retour de la plaque supérieure, le capteur inductif magnétique B11 (monté sur C10) est activé et signale au contrôleur CUC que le dispositif de l'éjection est de retour dans sa position initiale.

### I.3.7 Contrôleur de la tare

La bouteille arrive sur le convoyeur à chaîne, la cellule photoélectrique B03 (détection de bouteille) est activée, et l'arrêt d'accumulation s'ouvre (C7/C7A). La bouteille est pesée tout en étant levée par la table élévatrice (C8) de la bascule. En cas de sur ou sous emplissage de la bouteille, celle-ci est poussée sur le convoyeur à rouleaux par l'éjecteur (C9). Après l'éjection, l'éjecteur retourne à sa position initiale (la poussée en avant est détectée par un capteur inductif magnétique B07, la poussée en arrière est détectée par le capteur inductif magnétique B08).

**Remarque I.2 :** Une absence éventuelle d'ouverture de l'arrêt d'accumulation est probablement due à une accumulation totale sur le convoyeur à rouleaux (les bouteilles sont détectées par la cellule photoélectrique B10) ou une accumulation de bouteille après l'éjecteur (détecté par B09).

### I.3.8 Contrôle de fuite

En aval de la bascule du contrôle de la tare, on trouve un dispositif qui permet de contrôler l'étanchéité de la bouteille. Une cellule photoélectrique détecte l'arrivée d'une bouteille et provoque l'arrêt d'accumulation. Une fois l'arrêt d'accumulation fermé, le contrôle de fuite s'effectue et la bouteille sera éjectée si elle est fuyarde.

### I.3.9 Chargement

Les bouteilles arrivent au poste de chargement au niveau du palettiseur, un palpeur pneumatique permet de compter l'arrivée de 5 bouteilles. Une fois sur place, un arrêt d'accumulation empêche plus de 5 bouteilles à passer et un vérin pneumatique est commandé pour pousser les 5 bouteilles dans la palette. Cette opération est répétée 7 fois pour que la palette soit remplie, et c'est l'opérateur qui commande l'arrêt du chargement et le reprendre après avoir amené au poste de chargement une nouvelle palette vide.

Le tableau ci-dessous résume le sens de tous les éléments cités dans les paragraphes précédents:

B03	Capteur qui detecte la presence de bouteilles
C1/C1A	Arrêt d'acumulation sur le convoyeur à chaine
B01	Fibre optique de la lecture de l'horloge
B07 ,B08,B11	Capteurs inductifs magnétiques
B9,B10,B13,B14,B15	Cellules photoélectriques de détection de bouteilles
B16	Fibre optique de la lecture du code Manchester
C7/C7A	Arrêt d'accumulation au niveau du centre de la tare
C2	Table elevatrice dans la zone de la tare
C3	Ejecteur qui pousse les bouteilles vers le convoyeur
C6	Porte d'introduction
C5	Machine de remplissage
C8	Table elevatrice
C10	Plaque supérieure de l'unité de commande

#### I.4 Fonctionnement du palettiseur [2]

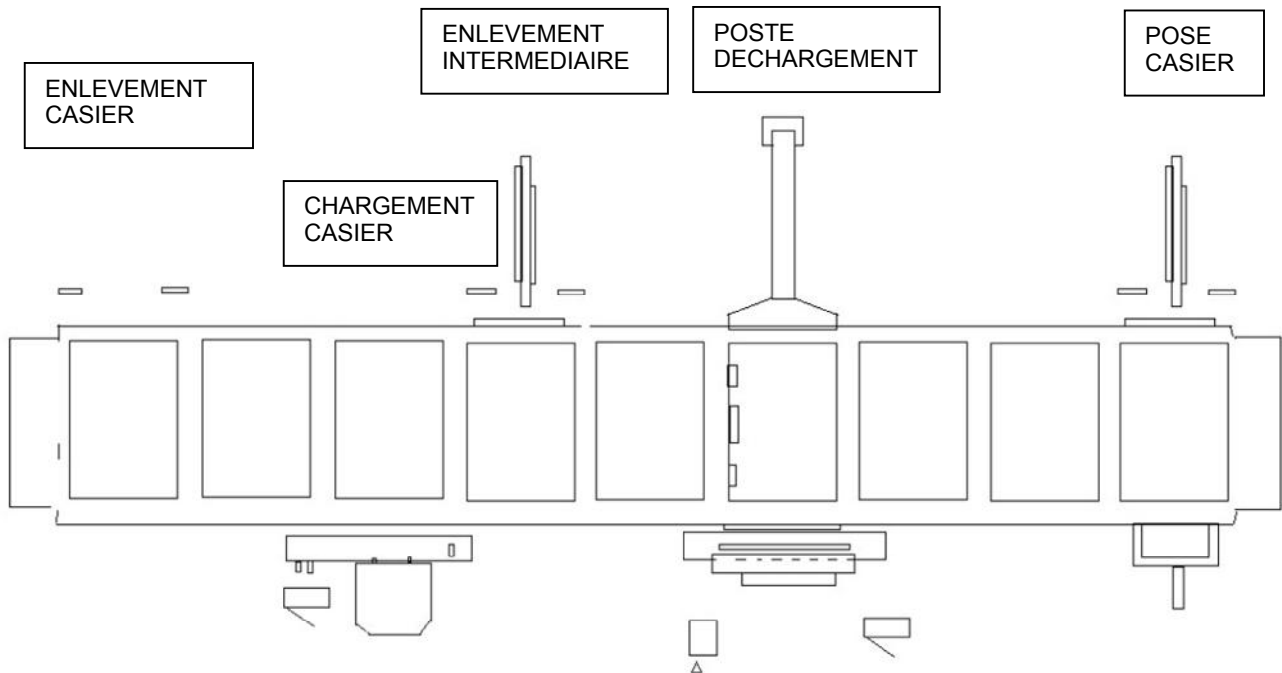
Cette partie a pour but de définir le principe de fonctionnement actuel du palettiseur (le système qui fera l'objet de notre étude).

**I.4.1 La palettisation:** permet de charger les bouteilles pleines dans les casiers (palettes) (voir figure I.5). Elle comprend quatre étapes (voir figure I.6) :

- Pose des casiers de bouteilles vides ;
- Déchargement des casiers ;
- Chargement des casiers ;
- Enlèvement des casiers.



**Fig I.5** Casiers (Palettes) chargés de bouteilles de gaz



**Fig I.6** Palettiseur à 9 casiers

Le fonctionnement de chaque étape du processus de palettisation des bouteilles de gaz est comme suit :

#### **I.4.1.1 Pose des casiers de bouteilles vides**

Le poste de pose de casiers de bouteilles vides est situé à l'extrémité du chemin de la chaîne de transfert, il comprend :

- Un heurtoir de pose fixé au sol ;
- Une trémie de guidage, constituée de tôles épaisses fixées à l'extrémité et à l'avant du chemin de la chaîne de transfert, facilitant la mise en place des casiers ;
- Un témoin de présence de chariot fixé au sol ;
- Un témoin de présence de casier en position haute ;
- Un dispositif de soulèvement des casiers

La procédure de pose de casiers de bouteilles vides se fait comme suit:

L'arrivée du chariot élévateur au poste commande la montée du dispositif de soulèvement (par le contact ch.1), ce qui permet de poser les casiers sur une partie fixe. En cas de pose de casiers par paire, le levage du casier supérieur jusqu'au palpeur (casier position haute : ph. 1) donne l'autorisation de transfert du premier casier. La pose du deuxième casier est effectuée après échappement du chariot élévateur (relâchement de ch.1).

### I.4.1.2 Transfert des casiers

Le chemin de transfert comprend :

- Deux chaînes situées aux extrémités du chemin, entraînées par un groupe motoréducteur. Une protection mécanique est assurée par une roue à fusible. De plus, le démarrage du moteur est asservi à la pression d'air sur la palettisation.

- Des ensembles de relevage de casier par casier, situés au dessous des postes de traitement et intermédiaires. Ces relevages sont constitués d'enceintes gonflables alimentées par des distributeurs, commandés par des palpeurs de présence de casiers

- Des vérins d'arrêt qui assurent aux casiers une position précise vers les postes de chargement et de déchargement.

Le transfert des casiers se fait comme suit:

Le premier casier va du poste de pose jusqu'au poste de déchargement sans arrêt intermédiaire. Au poste de déchargement, les palpeurs CS3 et PC3 émettent un signal qui commande les deux arrêts des casiers et les relevages. Ce même signal CS3 active le relevage du poste intermédiaire précédent, et par les contacts CS2 et PC2, le casier suivant sera à son tour soulevé, le relevage du poste précédent activé, et ainsi de suite jusqu'au poste de pose.

L'autorisation de départ de premier casier sera donnée après le retour du vérin de déchargement. Son départ entrainera celui du casier suivant.

Au poste d'enlèvement intermédiaire, la présence du chariot sur le contact ch. 2, interdit le transfert vers ce poste et les postes suivants. La régularisation du transfert au poste de chargement est effectuée de la même façon qu'au poste de déchargement.

### I.4.1.3 Déchargement des casiers

Le poste de déchargement est composé des équipements suivants :

➤ Un ensemble poussoir de déchargement comprenant :

- Un châssis métallique supportant une barre de poussée en forme de T, garnie d'une plaque en ertalon et guidée pendant toute la traversée du casier par des rouleaux à billes.

- Un ensemble hydraulique comprenant :

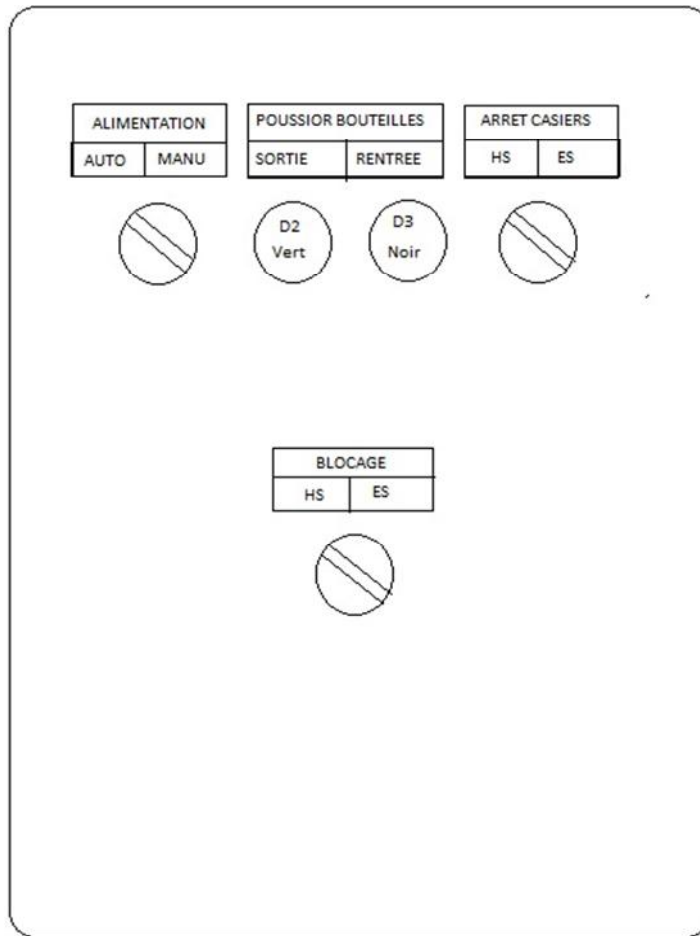
- Un groupe motopompe, constitué d'un moteur antidéflagrant, d'une pompe à engrenage à haute pression et d'une cuve.

- Un vérin.

- Un circuit distributeur, soupape tarée, manomètre et tuyauteries de liaisons.
- Deux palpeurs de fin de course avant et arrière du poussoir de déchargement (d1, do).
  - Un ensemble de réception de bouteilles sur la chaîne comprenant:
    - Un butoir de déchargement actionné par un vérin (B),
    - Un contact de présence de bouteilles sur la chaîne au droit du poste de déchargement (pb) ;
    - Un dispositif de maintien (mpb) du contact de présence de bouteilles ;
    - Un ensemble de désolidarisation des bouteilles de la chaîne au poste (P).
  - Un coffret de commande pneumatique permettant :
    - Le fonctionnement automatique de l'ensemble (**Fig I.7**);
    - Le fonctionnement manuel grâce aux commandes situées sur la porte avant du coffret (**Fig I.7**).

Les commandes figurant sur la porte avant du coffret sont:

- **Blocage** : La position de fonctionnement normal est "Hors Service (HS) ", l'autre position (En Service (ES)) est à utiliser quand on désire mettre des casiers de bouteilles vides sur le palettiseur sans provoquer le déchargement ;
- **Arrêt casier** : La position de fonctionnement normal est "En Service (ES) ", la position Hors Service (HS) permet de laisser passer le casier sans le décharger. Toutefois, le casier est soumis à la régulation normale (arrêt, soulèvement) lors de son transfert ;
- Manuel-automatique**: Ce bouton offre le choix du type de fonctionnement. Le mode manuel est utilisé particulièrement en cas d'incidents ;
- Les poussoirs bouteilles D2 et D3**: L'action sur ces deux commandes provoque la sortie et l'entrée du vérin de déchargement.



**Fig I.7** Coffret de commande pneumatique

L'opération de déchargement d'un casier est composée des étapes suivantes:

1. Transfert du casier: positionnement du casier au poste par les arrêts et le dispositif de soulèvement ;
2. Déchargement par rangée de cinq bouteilles. Simultanément au positionnement, un ordre autorise la sortie du poussoir de déchargement. L'arrêt du poussoir après chaque déchargement de cinq bouteilles est obtenu par l'action du palpeur (pb). Celui-ci commande également le retrait du butoir et la descente du plateau permettant ainsi l'évacuation des bouteilles. Le relâchement du palpeur (pb) donne l'ordre inverse: sortie du butoir, soulèvement du plateau et autorisation au poussoir de décharger une nouvelle rangée de cinq bouteilles.

Ce cycle se renouvelle sept fois par casier. Le capteur de fin de course (dl) du poussoir commande alors sa rentrée. Le poussoir en position rentrée (contact do) et l'absence de casier au poste suivant ( $\overline{CS4}$ ), autorise le départ du casier: descente de celui-ci et retrait des arrêts casiers.

#### I.4.1.4 Chargement du casier

Le poste de chargement se compose des équipements suivants :

➤ Un ensemble poussoir de chargement comprenant:

- Un châssis métallique supportant une barre de poussée guidée dans son mouvement par un ensemble de barres et douilles à billes, ;
- Un vérin pneumatique (C) actionnant directement la barre ;
- Un dispositif d'arrêt de bouteilles (AB), (;
- Un compteur à présélection (coffret pneumatique). Un palpeur de passage de bouteilles (cb) ;
- Un palpeur indiquant que le chargement du casier est terminé (cp).

➤ Un coffret de logique pneumatique, ce coffret :

- Assure le fonctionnement automatique du poste ;  
- Donne la possibilité de fonctionnement manuel à l'aide des commandes situées sur la porte avant du coffret (voir la figure I.8). Les commandes sont:

-**Arrêt casier**: Fonctionnement normal: position En Service (ES). La position Hors Service est utilisée lorsque l'on veut laisser passer le casier sans effectuer de chargement. Le casier reste toutefois, soumis aux arrêts et relevages, en cas de bourrage en aval.

-**Blocage**: Fonctionnement normal : position Hors Service (HS). La position En Service est utilisée quand on veut garder un casier vide au poste sans provoquer son chargement.

-**Fin de chargement - Commande manuelle**: Cette commande permet de libérer le casier même non rempli entièrement. Cette manœuvre peut être utilisée lors de changement de production de type de bouteilles.

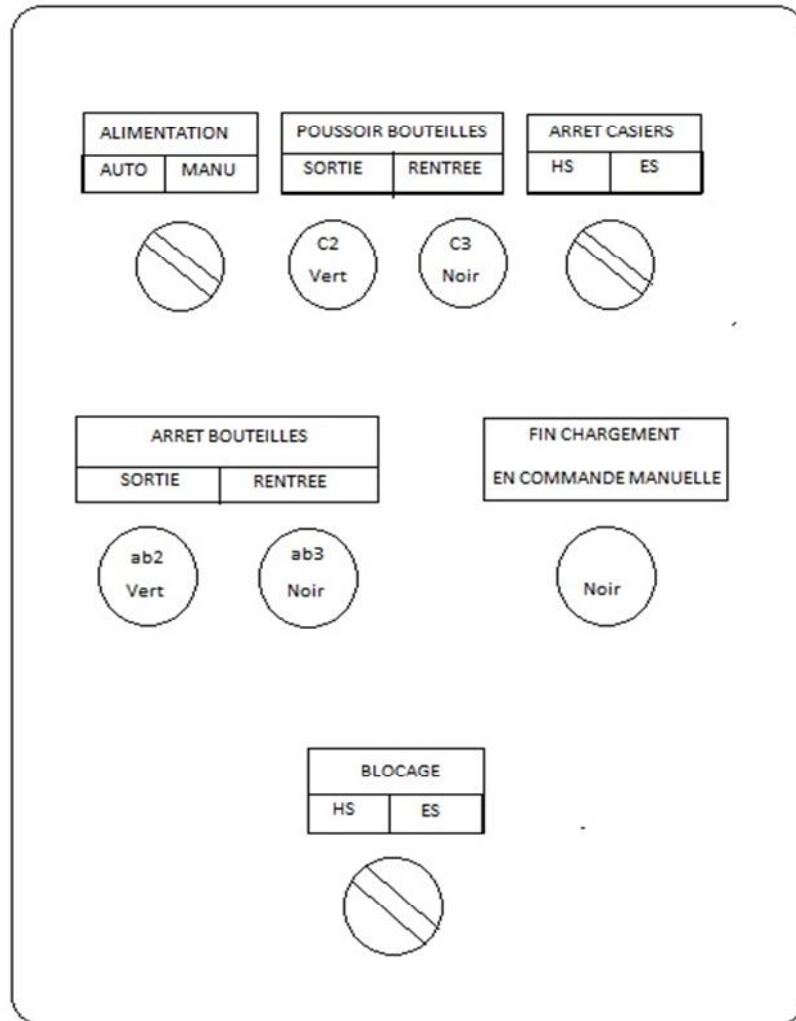
- **Manuel -automatique** : Cette commande offre le choix du type de fonctionnement. Le mode manuel est utilisé particulièrement en cas d'incidents. Dans cette position, on obtient les commandes manuelles suivantes :

- **Poussoir bouteilles**

Les boutons poussoirs C2 et C3 provoquent respectivement la sortie et la rentrée du vérin de chargement.

- **Arrêt bouteilles**

Les boutons poussoirs ab2, ab3 commandent respectivement la sortie et la rentrée du vérin d'arrêt bouteilles.



**Fig I.8** Coffret de logique pneumatique

L'opération de chargement d'un casier est composée des étapes suivantes:

1. Transfert du casier: positionnement de celui-ci par les arrêts et le dispositif de soulèvement.
2. Comptage et chargement: Cinq bouteilles sont admises au poste de chargement, le compteur à présélection donne, alors, l'ordre de fermer l'arrêt bouteilles et la sortie du vérin de chargement. Le capteur de fin de course (C1) détecte la sortie du vérin de chargement et commande le retour. La rentrée du vérin, par le capteur (Co) permet l'ouverture de l'arrêt bouteille et l'admission d'une autre série de cinq bouteilles.

Ce cycle se renouvelle sept fois, jusqu'à l'obtention de l'emplissage complet du casier détecté par le palpeur de fin de chargement (cp).

3. L'absence de casier au poste suivant autorise le départ du casier (descente du casier, retrait des arrêts).

### **I.4.1.5 Enlèvement du casier**

#### **I.4.1.5.1 Poste d'enlèvement intermédiaire**

Celui-ci est situé entre les postes de déchargement et de chargement. La présence du chariot élévateur sur le palpeur de présence de chariot (ch.2) immobilise le casier sur ce poste, ainsi que la progression des casiers suivants. Le casier en poste peut être enlevé facilement et la progression des casiers suivants sera assurée après le retour du chariot.

#### **I.4.1.5.2 Poste d'enlèvement de casiers pleins**

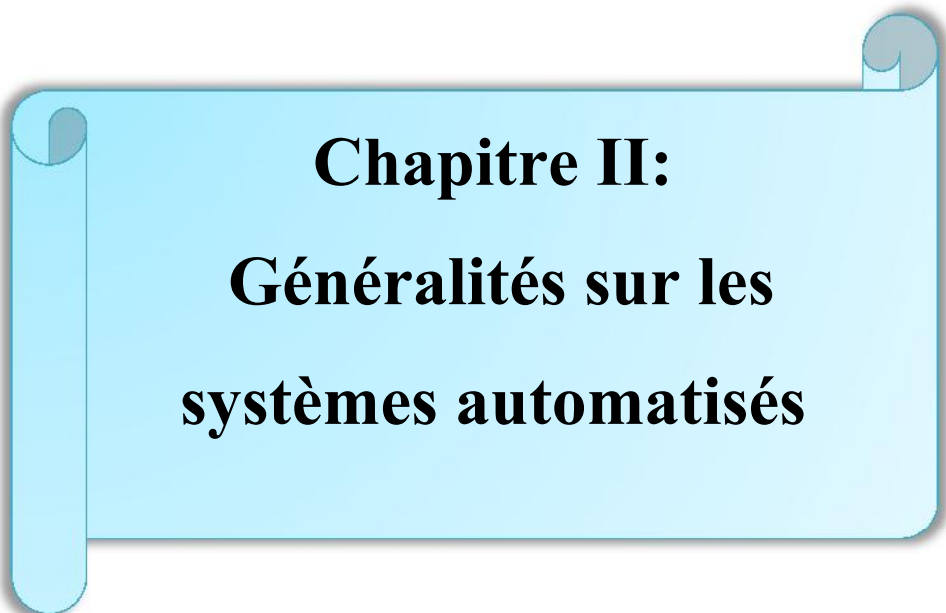
Le chariot procède à l'enlèvement du casier lorsque celui-ci est en position et soulevé. La progression du casier suivant ne sera permise que lorsque le casier soulevé par le chariot aura totalement relâché le palpeur (pc 9). Le chariot peut alors évacuer le premier casier, ou attendre le second pour procéder à l'enlèvement des deux casiers.

## **I.5 Conclusion**

A travers l'étude que nous avons menée sur le fonctionnement du processus de remplissage de bouteilles de gaz butane, il ressort les insuffisances suivantes de l'installation actuelle:

- Le palettiseur demande l'intervention de plus de cinq opérateurs pour fonctionner. C'est eux qui contrôlent l'avancement des palettes, et la sortie du bras de chargement et un agent est obligé de surveiller le flux de bouteilles car il arrive qu'une bouteille fuit ou mal fermée ou qu'une bouteille coince sur la chaîne.
- La technologie installée sur le palettiseur qui est du type pneumatique est vétuste. Cette technologie datant du dix-neuvième siècle est dépassée sur tous les plans économiques, sûreté, fiabilité, qualité et rendement. Pleine, de défections, elle provoque de nombreuses pannes répétées, ce qui conduit à des performances très très médiocres.

Afin d'améliorer l'installation existante, nous proposons une conception de l'automatisation du palettiseur et de mettre en place un système de visualisation numérique par des écrans de contrôle digitaux afin d'améliorer l'interface homme-machine et permettre aux opérateurs un suivi continu et efficace du fonctionnement du processus et d'intervenir dans le cas de besoin sur le procédé dans des délais exigés.



**Chapitre II:**  
**Généralités sur les**  
**systemes automatisés**

## II.1 Introduction

Les systèmes automatisés sont de plus en plus présents dans notre environnement. En effet, ils accomplissent les tâches pénibles et répétitives à notre place. Dans l'industrie, ils remplacent les ouvriers et effectuent des tâches de production, de manutention, de contrôle, de montage, etc. Ce qui a pour effet de diminuer les coûts de production. Ils interviennent aussi dans des lieux inaccessibles ou dangereux, etc.

Parmi l'ensemble des systèmes automatisés, nous nous intéressons, dans ce mémoire, plus particulièrement aux systèmes automatisés pilotés par des Automates Programmables Industriels (API). Un API est une machine électronique programmable et destiné à commander en temps réel des procédés industriels. Ils sont apparus aux Etats-Unis vers 1969, où ils répondaient aux désirs des industries de l'automobile en développant des chaînes de fabrication automatisées qui pourraient suivre l'évolution des techniques et des modèles fabriqués.

Dans ce chapitre, nous présentons les notions principales concernant les systèmes automatisés et les API en particulier.

## II.2 Structure d'un système automatisé [05][09][11]

Un système est dit automatisé lorsque le processus qui permet de passer d'une situation initiale à la situation finale, se fait sans intervention humaine et que ce comportement est répétitif. Un système automatisé réalise un certain nombre d'actions appelées « tâches ». Il accomplit une suite d'opérations, appelées « cycle », depuis un état initial jusqu'à un état final. Deux types de cycle :

Cycle	Ouvert	Fermé
Définition	Les tâches s'enchaînent et sans vérification	Les tâches ne se déclenchent que lorsque c'est nécessaire
Exemple	Feu rouge : fonctionnement identique le jour et la nuit)	Régulation de température

**Tab II.1** Types du cycle

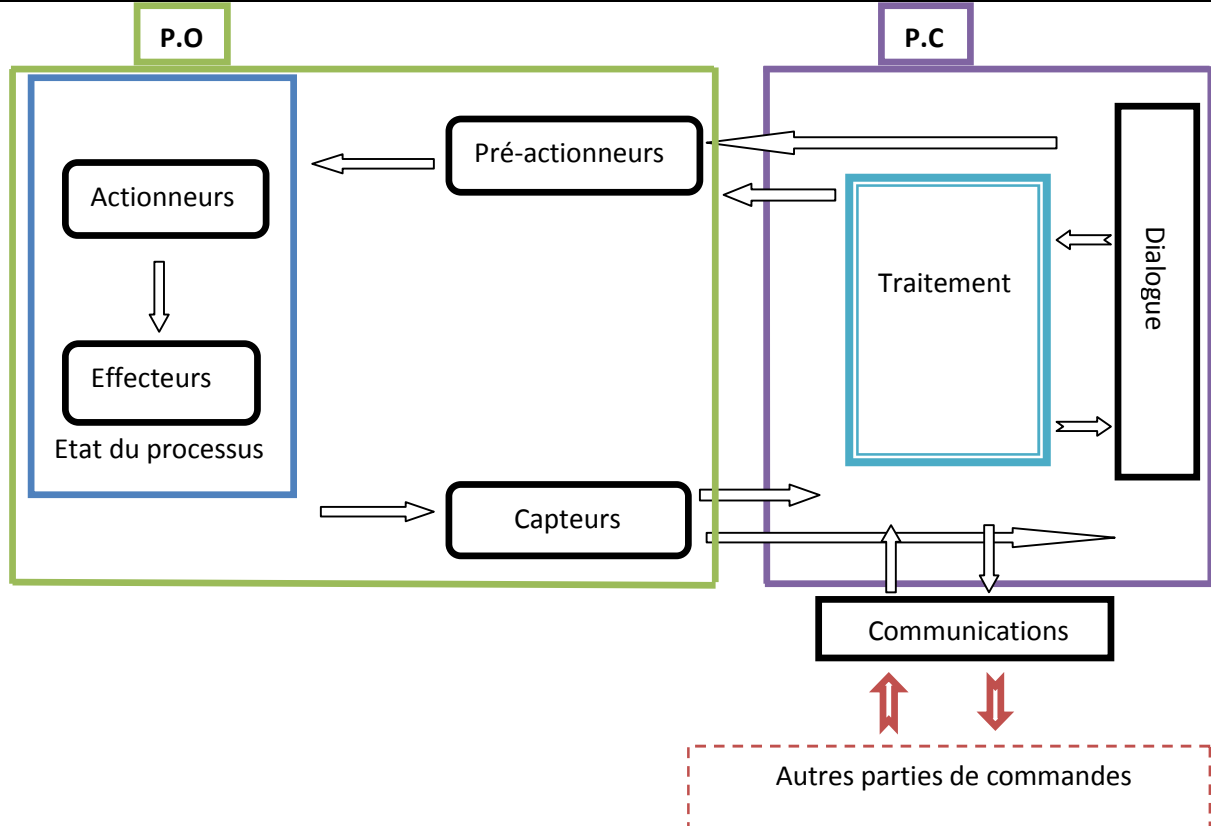


Fig II.1 Structure d'un système automatisé

La figure II.1 montre la structure d'un système automatisé. Elle comporte deux parties importantes:

### II.2.1 La Partie Opérative (PO)

Cette partie exécute les ordres qu'elle reçoit de la partie commande. Elle comporte généralement les éléments suivants:

✓ **Pré-actionneur**: son rôle est de distribuer, sur ordre de la partie commande, l'énergie utile aux actionneurs pour fonctionner. Les pré-actionneurs les plus utilisés sont les contacteurs (pour les moteurs électriques) et les distributeurs (pour les vérins pneumatiques).



Fig II.2 Pré-actionneurs

✓ **Actionneur**: transforme l'énergie d'entrée qui lui est appliquée en une énergie de sortie (généralement mécanique) utilisable par un effecteur pour fournir une action définie.



Moteur électrique





Moteur pneumatique



Vérin pneumatique

**Fig II.3** Actionneurs

✓ **Effecteurs**: Ils reçoivent leur énergie des actionneurs et agissent sur la matière d'œuvre. Les effecteurs sont multiples et variés (pales de ventilateurs, etc).

Système	Effecteur
 <p data-bbox="300 1295 428 1323">ventilateur</p>	 <p data-bbox="874 1356 1102 1384">pale de ventilateur</p>

**Fig II.4** Exemple d'effecteurs

✓ **Capteur**: est un élément de prélèvement et de codage d'informations sur un processus ou sur l'environnement du système. Il convertit une grandeur physique (position, vitesse,...) en une information appelée compte-rendu et compréhensible par la Partie Commande.



Capteur de pression



Capteur de position



Détecteur de fumée

Fig II.5 Capteurs

### II.2.2 La Partie commande (PC)

Cette partie est le cerveau du système, elle pilote la partie opérative et reçoit des informations venant des capteurs de la partie opérative, et les transmet vers cette même partie opérative en direction des pré-actionneurs et actionneurs. Dans les systèmes automatisés à base d'un API, l'unité de traitement de la partie commande est un **Automate Programmable Industriel**.

### II.3 L'Automate Programmable Industriel (API) [03]

A la fin des années 60, GENERAL MOTORS a passé un appel d'offre pour la conception d'un système pour remplacer les armoires à relais et lui permettre de faire des modifications de cycle de fabrication à moindre coût, plus rapidement, et en conservant une possibilité d'évolution.

C'est la société BEDFORD ASSOCIATES, et plus particulièrement Richard E. MORLEY qui, en créant le concept d'automate programmable, emporta le marché.

Richard MORLEY et son équipe, créèrent la société MODICON (MODular Digital CONTROL). Le premier automate fut baptisé Modicon 084 car il concrétisait le 84<sup>ième</sup> projet de la société, et fut présenté à la fin du 1969.

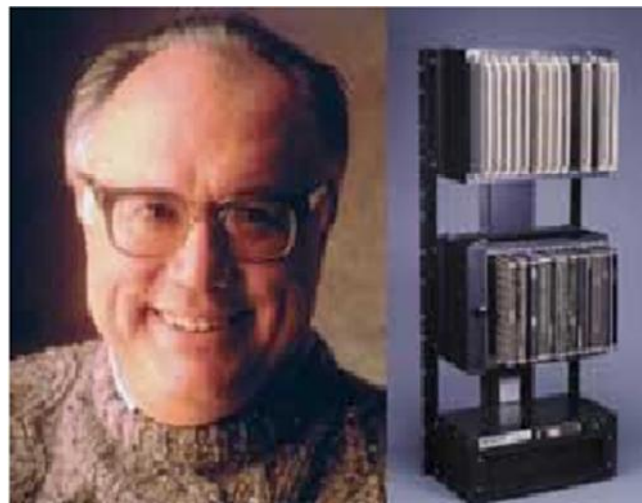


Fig II.6 Richard E. MORLEY

Modicon 084: 255 E/S, mémoire 4 Ko, programmation LADDER, dimension L 500 x H 1200 x P 340, poids 46 kg.

### II.3.1 Définition [04]

Un API (Automate Programmable Industriel) ou en anglais PLC (Programmable Logic Controller) est un appareil électronique destiné à la commande de processus industriels par un traitement séquentiel (il contrôle les actionneurs grâce à un programme informatique qui traite les données d'entrée recueillies par des capteurs). Il comporte une mémoire programmable par un utilisateur à l'aide d'un langage adapté pour le stockage interne des instructions données pour satisfaire un objectif donné. L'automate permet de contrôler, de coordonner et d'agir sur les actionneurs, alors on peut dire que l'API est utilisé pour l'automatisation des processus. L'API est constitué d'une unité de calcul (processeur), de cartes d'entrées-sorties, de bus de communication et de modules d'interface et d'alimentation. Il existe sur le marché de nombreuses marques d'automates: Télémécanique, Schneider, Siemens, Omron, Allen Bradley, Cegetel, etc.

### II.4 Les avantages et les inconvénients d'un API [11]

Ses avantages sont :

- ✓ Améliorer les conditions de travail en éliminant les travaux répétitifs.
- ✓ Améliorer la productivité en augmentant la production.
- ✓ Améliorer la qualité des produits tout en réduisant les coûts de production.
- ✓ Les automates programmables sont programmés facilement et ont un langage de programmation facile à comprendre (logique programmée) alors la modification du programme facile par rapport à la logique câblée.
- ✓ Simplification du câblage.
- ✓ Puissance et rapidité.
- ✓ Facilité de maintenance (l'API par lui même est relativement fiable et peut aider l'homme dans la recherche de défauts).
- ✓ Augmenter la sécurité.
- ✓ Possibilités de communication avec l'extérieur (ordinateur, autre API).
- ✓ Enorme possibilité d'exploitation.
- ✓ Plus économique

Ses inconvénients sont:

- ✓ Plantage.
- ✓ Il y a trop de travail requis dans les fils de connexion.
- ✓ Besoin de formation.

## II.5 Structure d'un automate programmable industriel

### II.5.1 Aspect extérieur

Les automates peuvent être de type compact ou modulaire.

✓ **Automate de type compact:** Il intègre le processeur, l'alimentation, les entrées et les sorties. Selon les modèles et les fabricants, il pourra réaliser certaines fonctions supplémentaires (comptage rapide, E/S analogiques, etc) et recevoir des extensions en nombre limité. Ces automates, de fonctionnement simple, sont généralement destinés à la commande de petits automatismes (micro automate).

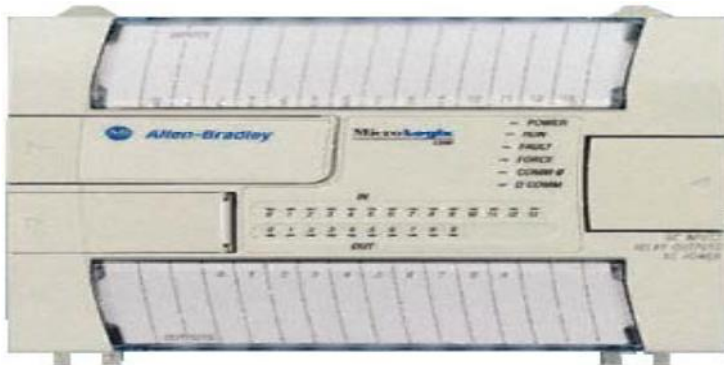


Fig II.7 API (type compact)

✓ **Automate De type modulaire:** le processeur, l'alimentation et les interfaces d'entrées /sorties résident dans des unités séparées (modules) et sont fixées sur un ou plusieurs racks contenant le "fond de panier" (bus plus connecteurs). Ces automates sont intégrés dans les automatismes complexes où la puissance, la capacité de traitement et la flexibilité sont nécessaires.



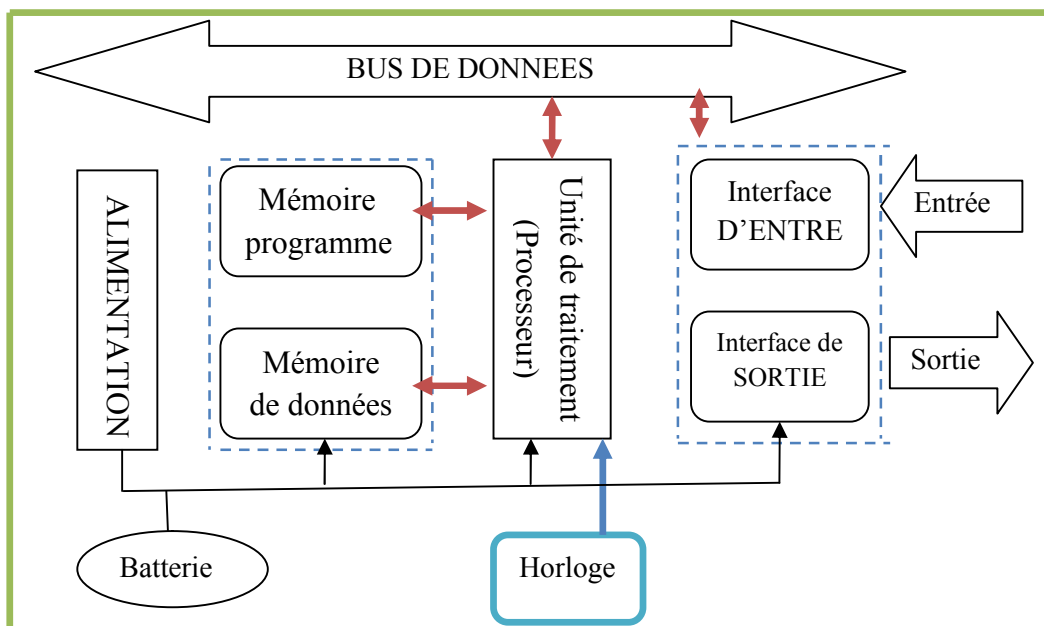
Fig II.8 API d'un type modulaire

### II.5.1 Remarque

Les automates compacts permettent de commander des sorties en tout ou rien et gèrent parfois des fonctions de comptage et de traitement analogique mais les automates modulaires permettent de réaliser de nombreuses autres fonctions grâce à des modules intelligents dont ils disposent sur un ou plusieurs racks. Ces modules ont l'avantage de ne pas surcharger le travail de l'unité de calcul car ils disposent bien souvent de leur propre processeur.

### II.5.2 Structure interne d'un automate programmable industriel [04],[05],[08]

La structure interne d'un API est montrée par la figure II.9 ci-dessous.



**Fig II.9** Structure interne d'un automate programmable industriel

La compacité, la robustesse et la facilité d'emplois des automates programmables industriels font qu'ils sont très utilisés dans la partie commande des systèmes industriels automatisés. L'automate programmable reçoit et envoie les informations par les modules d'entrées et de sorties (logiques, numériques ou analogiques) et puis commande par l'unité de calcul ou processeur (en anglais Central Processing Unit, CPU) suivant le programme inscrit dans sa mémoire.

Un API se compose des éléments suivants :

- ✓ Le processeur.
- ✓ La zone mémoire.
- ✓ Le module Entrée/Sortie.

- ✓ Le module d'alimentation.
- ✓ Le bus de données (bus de communication)

### II.5.2.1 Le processeur [04],[05]

Le processeur est connecté aux autres éléments (mémoire et interface E/S) par des liaisons parallèles appelées ' BUS ' qui véhiculent les informations sous forme binaire. Le processeur gère l'ensemble des échanges informationnels en assurant :

- ✓ L'exécution des instructions (toutes les fonctions logiques ET, OU, les fonctions de temporisation, de comptage, de calcul...) du programme contenu dans la mémoire.
- ✓ La lecture des informations d'entrée.
- ✓ La commande ou l'écriture des sorties.

### II.5.2.2 La zone mémoire

La mémoire de l'API est l'élément fonctionnel qui peut recevoir, conserver et restituer des informations. Les mémoires utilisées dans un API peuvent être des types suivants:

- ✓ **R.A.M. (Random Access Memory):** mémoire à accès aléatoire. Cette mémoire doit être alimentée électriquement pour pouvoir conserver les informations. On l'appelle également la mémoire vive. Avant son exécution, le programme est transféré dans cette mémoire qui permet d'atteindre des vitesses en lecture et écriture très rapides.
- ✓ **R.O.M. (Read Only Memory):** mémoire à lecture uniquement, appelée également mémoire morte, elle permet de stocker des informations indéfiniment sans aucune alimentation électrique.
- ✓ **P.R.O.M. (Programmable Read Only Memory):** mémoire de type ROM mais programmable. C'est une ROM que l'on peut programmer une seule fois.
- ✓ **E.P.R.O.M. (Erasable Programmable Read Only Memory):** mémoire de type PROM que l'on peut effacer par exposition du circuit aux rayons ultra-violets.
- ✓ **E.E.P.R.O.M. (Electrical Erasable Programmable Read Only Memory):** mémoire de type PROM que l'on peut effacer électriquement en écrivant à nouveau sur le contenu de la mémoire. Ce type de mémoire par sa simplicité de mise en œuvre tend à remplacer de plus en plus la mémoire EPROM.

L'espace mémoire peut être divisé en deux parties :

- ✓ **La mémoire programme** qui permet le stockage des instructions à exécuter par l'API et cette mémoire est de type RAM contient les instructions à exécuter par le processeur afin de déterminer les ordres à envoyer aux pré-actionneurs reliés à l'interface de sortie en fonction

des informations recueillies par les capteurs reliés à l'interface d'entrée.

✓ **La mémoire de données** qui permet le stockage de:

- Les états forcés ou non des E/S.
- Les variables internes utilisées par le programme (résultats de calculs, états intermédiaires, etc).
- L'état des sorties élaborées par le processeur.
- L'image des entrées reliées à l'interface d'entrée.

Cette mémoire est de type ROM, PROM, EPROM, EEPROM.

### II.5.2.3 Les modules Entrées/Sorties

✓ **Interfaces d'entrée:** Ce sont des circuits spécialisés capables de recevoir en toute sécurité les signaux issus des capteurs ou de l'opérateur. Elles peuvent être :

- Logiques ou Tout Ou Rien: l'information ne peut prendre que deux états (vrai/faux, 0 ou 1, etc). C'est le type d'information délivré par un détecteur, un bouton poussoir, etc.
- Numériques: l'information est contenue dans des mots codés sous forme binaire ou bien hexadécimale. C'est le type d'information délivré par un ordinateur ou un module intelligent.
- Analogiques: 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, etc).

Ces différentes entrées sont mises en forme par l'interface d'entrée avant d'être stockées dans la mémoire de données.

✓ **Interfaces de sortie:** Ce sont des circuits spécialisés capables de commander en toute sécurité les circuits extérieurs. Elles peuvent être :

- Logiques ou Tout Ou Rien
- Numériques
- Analogiques

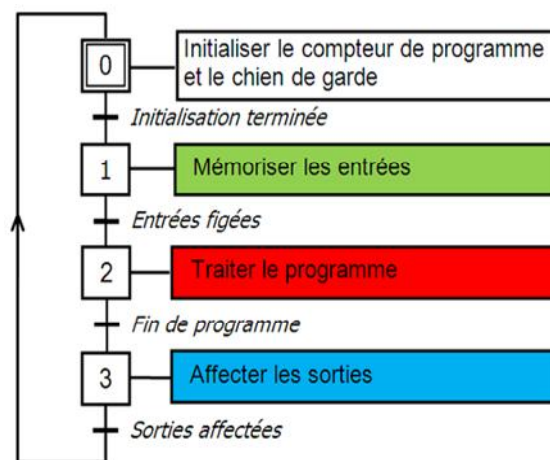
### II.5.2.4 Le module d'Alimentation

Le module d'alimentation transforme l'énergie externe provenant du réseau en la mettant en forme, afin de fournir aux différents modules de l'API, les niveaux de tension nécessaires à leur bon fonctionnement. Plusieurs niveaux de tension peuvent être utilisés par les circuits internes (3v, 5v, 12v, 24v, etc). Il sera dimensionné en fonction des consommations des différentes parties.

**II.5.2.5 Le bus interne** permet la communication de l'ensemble des blocs de l'automate et des éventuelles extensions

## II.6 Principe de fonctionnement d'un automate programmable industriel [04],[05],[06]

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 (figure II.10). Le microprocesseur réalise toutes les fonctions logiques ET, OU, les fonctions de temporisation, de comptage, de calcul, etc. 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.



**Fig II.10** Fonctionnement cyclique d'un API

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, figure II.11).

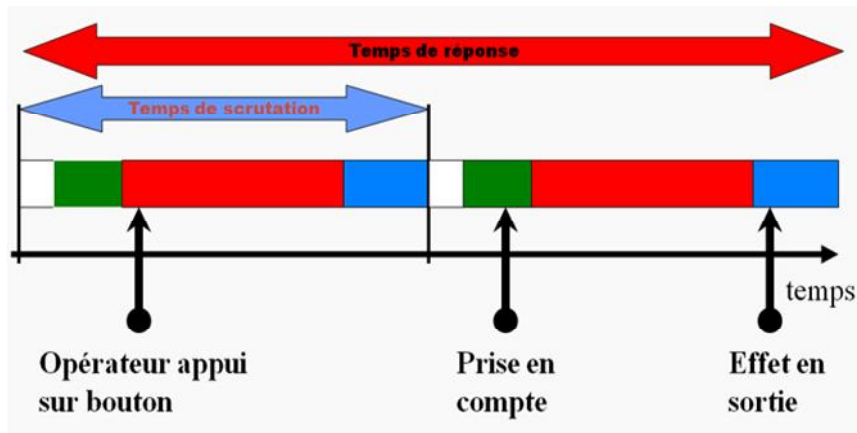


Fig II.11 Temps de scrutation et temps de réponse

## II.7 Les types des automates programmables industriels

✓ **Les automates de petite gamme:** ces automates sont destinés pour de petite application. Le nombre d'entrées/sorties ne dépasse pas 48. Ils se présentent dans les boîtiers compacts où tous les modules (CPU, Alimentation, Module d'E/S, interface de communication) sont intégrés dans un boîtier. Ils ne disposent d'aucune possibilité d'extension.

✓ **Les automates de moyenne gamme:** dans cette gamme, le nombre d'E/S peut atteindre 400, ces automates ont une structure modulaire extensible.

✓ **Les automates de haute gamme:** ce sont des automates très puissants dont les performances permettent de gérer jusqu'à 2024 E/S et plus. Ils disposent d'une structure modulaire.

## II.8 Les langages de programmation d'un API [07],[09],[10]

Chaque automate possède son propre langage. Mais par contre, les constructeurs proposent tous une interface logicielle répondant à la norme CEI 1131-3 (CEI : Commission Electrotechnique Internationale). Cette norme définit cinq langages de programmation utilisables, qui sont :

### ➤ Langage liste d'instructions «IL»

(Instruction List): est très proche du langage assembleur, on travaille au plus près du processeur en utilisant l'unité arithmétique et logique, ses registres et ses accumulateurs. Ce langage textuel est de bas niveau.

```
(***** Langage IL *****)
%L0:
LD    %I1.0
ANDN  %M12
OR(   %TM4.Q
AND   %M17
)
AND   %I3.7
ST    %Q2.5

%L2:
LD    %I3.5
```

➤ **Langage texte structuré «ST»**

(Structured Text): Ce langage est un langage textuel de haut niveau. Il permet la programmation de tout type d'algorithme plus ou moins complexe.

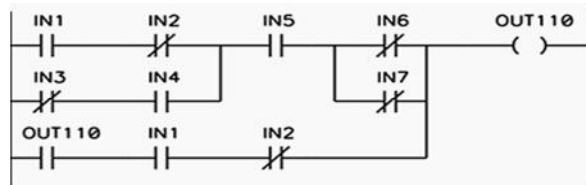
```

(***** Langage ST *****)
IF %M0 THEN
  FOR %MW99:=0 TO 31 DO
    IF %MW100 [%MW99]<> 0 THEN
      %MW10:= %MW100 [%MW99];
      %MW11:= %MW99;
      %M1:=TRUE;
      EXIT;
    ELSE
      %M1:=FALSE;
    END_FOR;
  ELSE
    %M1:=FALSE;
  END_IF;

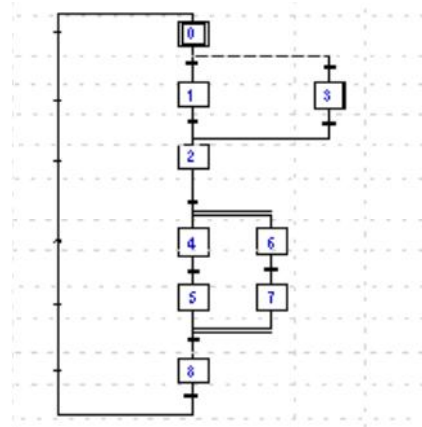
```

➤ **Le langage à contacts ou diagramme en échelle (LD : Ladder Diagram) :**

Ce langage graphique est essentiellement dédié à la programmation d'équations booléennes (true/false). C'est le plus utilisé.

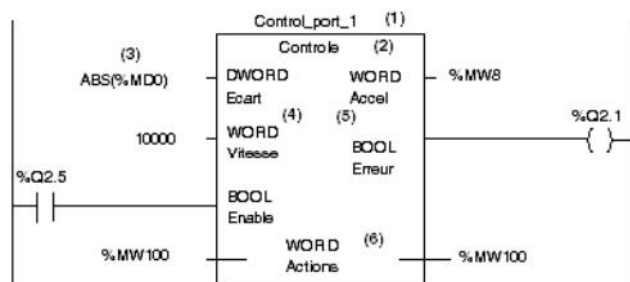


➤ **Le GRAFCET (Graphe Fonctionnel de Commande par Etapes et Transitions) ou SFC (Sequential Function Chart):** C'est un outil graphique qui décrit les différents comportements de l'évolution d'un automate. C'est un mode de représentation et d'analyse d'un automate, particulièrement bien adapté aux systèmes à évolution séquentielle, c'est à dire décomposable en étapes.



➤ **Blocs Fonctionnels (FBD: Function Bloc Diagram):**

Ce langage permet de programmer graphiquement à l'aide de blocs, représentant des variables, des opérateurs ou des fonctions. Il permet de manipuler tous les types de variables.



## II.10 Critères de choix d'un API [08]

Le choix d'un API se base sur les critères suivants :

- ✓ Nombre d'entrées / sorties.
- ✓ Le temps de traitement.
- ✓ La capacité de la mémoire.
- ✓ Le nombre d'étapes ou d'instructions.
- ✓ Le nombre de temporisateurs.
- ✓ Le langage de programmation.

Dans notre cas, nous avons porté notre choix sur l'automate programmable industriel « SIMATIC S7-300 ». Ce choix est justifié par les capacités de SIMATIC S7-300 de gérer un grand nombre d'entrées/sorties et d'exécuter des instructions à grande vitesse.

## II.11 Présentation de l'automate Siemens S7-300

L'automate S7-300 est constitué d'une alimentation, d'une CPU, de modules d'entrées et de modules de sorties. A ceux-ci peuvent s'ajouter des processeurs de communication et des modules de fonction qui se chargeront de fonctions spéciales, telle que la commande d'un moteur pas à pas, par exemple.

Comme tout API, l'automate S7-300 est un système d'automatisation modulaire offrant la gamme de modules suivants :

- Modules d'alimentation (PS: Power Supply) avec 2A, 5A ou 10A.
- Unités centrales (CPU315) travaillant avec une mémoire de 48 ko, sa vitesse d'exécution est 0.3ms/1K instructions.
- Modules de signaux (SM) pour entrées et sorties TOR et analogiques.
- Modules d'extension pour configuration multi-rangée de S7-300.
- Modules de fonction (FM) pour fonctions spéciales (par exemple l'activation d'un moteur pas à pas).
- Processeurs de communication (CP) pour la connexion en réseau, etc.

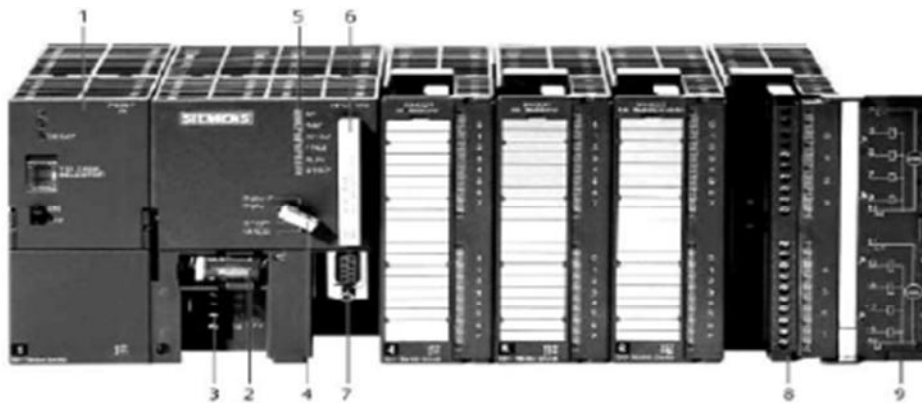
### II.11.1 Caractéristique de S7-300

L'automate S7-300 offre les caractéristiques suivantes :

1. Gamme diversifiée de CPU.
2. Gamme complète de modules.
3. Possibilité d'exécution jusqu'à 32 modules.

4. Bus de fond de panier intégré au module.
5. Possibilité de mise en réseau avec MPI, PROFIBUS ou INDUSTRIAL ETHERNET.
6. Raccordement central de la PG avec accès à tous les modules.
7. Liberté de montage aux différents emplacements.
8. Configuration et paramétrage à l'aide de l'outil configuration matérielle.

**II.11.2 Constitution de l'automate S7-300**



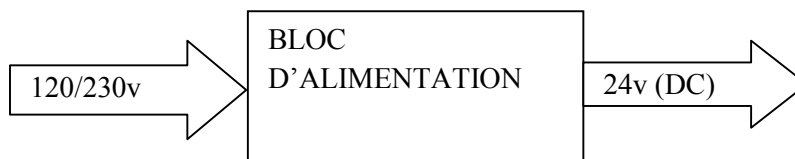
1	Module de l'alimentation	6	Carte mémoire
2	Pile de sauvegarde	7	Interface Multipoint
3	Connexion au 24 V	8	Connecteur frontal
4	Commutateur de mode (à clé)	9	Volet en face avant
5	LED de signalisation d'état et de défauts		

**Fig II.12** Constitution de l'automate S7-300

**II.11.2.1 Modules d'alimentation (PS)**

Tout réseau industriel de 24 volts peut être utilisé pour alimenter la CPU du S7-300. Les modules d'alimentation suivants, de la gamme S7 sont prévus pour être utilisés. Ces derniers sont résumés dans le tableau ci-après.

Désignation	Courant	Tension à la sortie	Tension à l'entrée
PS 307	2A	DC 24v	AC 120v/230v
PS 307	5A	DC 24v	AC 120v/230v
PS 307	10A	DC 24v	AC 120v/230v



### II.11.2.2 Unité centrale (CPU)

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

La gamme S7-300 offre une grande variété de CPU telles que la CPU312, 314, 314IFM, 315. 315- 2DP, etc. Chaque CPU possède certaines caractéristiques différentes des autres. Par conséquent le choix de la CPU pour un problème d'automatisation donné est conditionné par les caractéristiques offertes par la CPU choisie.

La CPU 315-2DP peut aussi être utilisée dans un sous réseau PROFIBUS:

- Comme maître DP ;
- Comme esclave DP raccordé à un maître DP/M7 ou à un maître DP. Un exemple de la CPU est représenté sur la figure II.13.

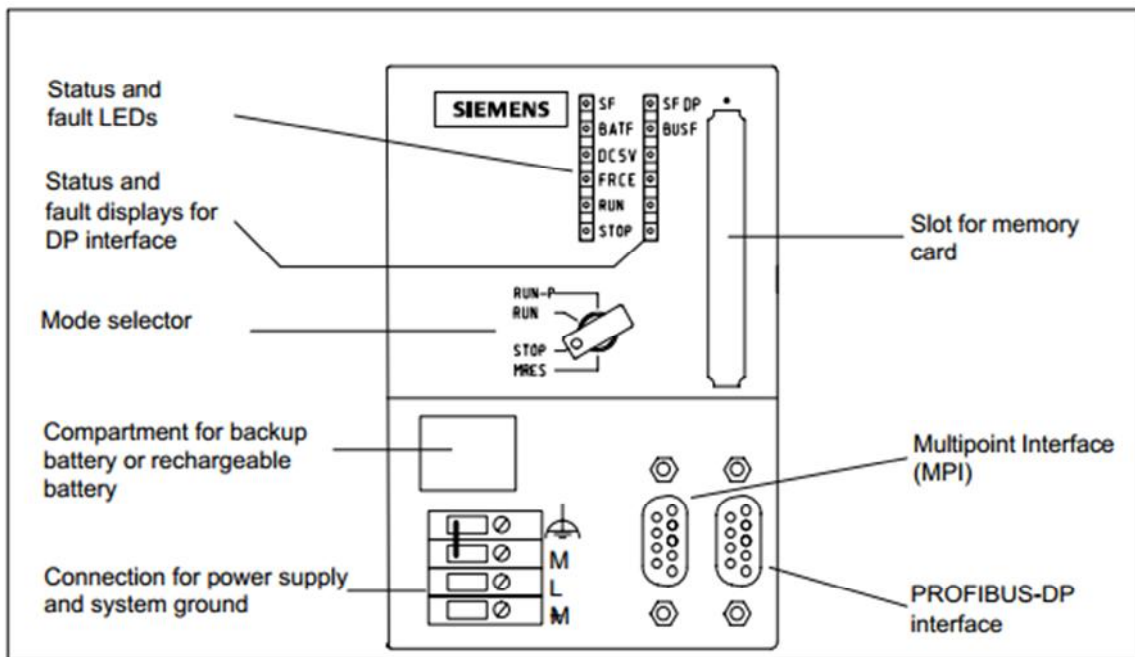


Fig II.13 Représentation d'un exemple de la CPU d'un S7-300.

#### II.11.2.2.1 Interface (MPI)

MPI (Interface Multipoint) est l'interface de la CPU vers les PG/OP ou pour la communication avec plusieurs stations au sein d'un sous-réseau MPI.

#### II.11.2.2.2 Signalisation d'état

La CPU comporte des LED de signalisation suivante :

- SF (rouge) : signalisation groupée de défauts.
- BATF (rouge): défaut de la pile.
- DC 5v (verte) : alimentation 5Vcc pour la CPU et le bus S7-300.
- FRCE (jaune) : forçage permanent.
- RUN (verte) : état de fonctionnement RUN.
- STOP (jaune) : état de fonctionnement stop.

#### II.11.2.2.3 Commutateur de mode

Le commutateur de mode et les éléments d'affichage de toutes les CPU sont identiques. Leurs rôles et leurs fonctions sont également identiques. On définit ici les quatre positions principales de la CPU de S7-300 :

- RUN-P (mode de fonctionnement RUN programme): la CPU traite le programme utilisateur et la clé ne peut pas être retirée.
- RUN (mode de fonctionnement RUN): la CPU traite le programme utilisateur. Dans cette position la clé peut être retirée.

- STOP (mode fonctionnement STOP): la CPU ne traite aucun programme utilisateur, la clé peut être retirée pour éviter le changement de mode inattendu mais on peut lire et écrire dans la CPU.
  - MRES (Machine RESET): effacement général. L'arrêt est réalisé pour STOP/MRES et la marche pour RUN/RUN-P.

#### II.11.2.2.4 Carte mémoire :

Une carte mémoire conserve le contenu du programme en cas de coupure de courant, même en l'absence de pile.



Figure II.14 Carte mémoire de la CPU.

#### II.11.2.2.5 La pile

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

### II.11.3 Les modules d'entrées/Sorties

#### II.11.3.1 Modules d'entrées

Il existe deux types de module d'entrées : les modules d'entrées Tout Ou Rien (TOR) et les modules d'entrées Analogiques.

##### ❖ Les modules d'entrées TOR

Un module d'entrée Tout ou Rien, permet à l'unité centrale d'effectuer une lecture de l'état logique des capteurs qui lui sont reliés et de le matérialiser par un bit image de l'état du capteur.

##### ❖ Les modules d'entrées analogiques

Les modules d'entrées analogiques permettent de :

- Sélectionner la gamme d'entrées de chaque voie.
- Scruter des voies d'entrées par multiplexage et l'acquisition des valeurs.
- Faire une conversion analogique/numérique des mesures d'entrées.

### II.11.3.2 Les modules de sorties

Il existe deux types de modules de sorties : les modules de sorties Tout ou Rien et les modules de sorties analogiques.

#### ❖ Les modules de sorties Tout ou Rien

Les modules de sorties Tout ou Rien permettent à l'automate programmable d'agir sur les pré-actionneurs ou de signaler à l'opérateur.

#### ❖ Les modules de sorties analogiques

Les différentes fonctions du module de sortie analogique sont :

- La sélection de la gamme pour chaque sortie ;
- La conversion numérique/analogique.

### II.11.4 Coupleur (IM)

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

### II.11.5 Module de signaux (SM)

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

### II.11.6 Module de fonction (FM)

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

### II.11.7 Modules de communication (CP)

Ils permettent d'établir des liaisons homme-machine et machine-homme qui sont effectuées à l'aide des interfaces de communication :

- Point à point.
- PROFIBUS.
- Industrial Ethernet.

## II.12 Fonctionnement de l'automate S7-300

L'automate lors de son fonctionnement exécute le programme cyclique, qui commence par l'acquisition des entrées issues des capteurs sur l'état du processus et finit par l'envoi des sorties aux actionneurs.

### II.12.1 Réception des informations sur les états du système

Le S7-300 reçoit des informations sur l'état du processus via les capteurs de signaux reliés aux entrées. Le S7-300 va mettre à jour la mémoire image des entrées au début de chaque cycle de programme, en transférant le nouveau état des signaux d'entrées des modules vers la mémoire image des entrées ce qui permet à la CPU de savoir l'état du processus.

### II.12.2 Exécution du programme utilisateur

Après avoir acquis les informations d'entrée et exécuté le système d'exploitation, la CPU passe à l'exécution du programme utilisateur, qui contient la liste d'instructions à exécuter pour faire fonctionner le processeur. Il est composé essentiellement de bloc de données, de code et de bloc d'organisation.

### II.12.3 Commande du processus

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


## II.13 Programmation de l'automate S7-300

La programmation des automates de la famille S7 se fait sur la console de programmation qui est actuellement le PC et sous un environnement WINDOWS. Le langage de programmation est le STEP 7 que nous allons décrire dans le chapitre 4.

## II.14 Conclusion

D'après ce qui précède, le développement scientifique a laissé sa trace sur les systèmes de production donnant naissance aux systèmes automatisés de production, qui s'avèrent être plus ou moins un remède au paradoxe des paramètres coût-qualité visés généralement par la gestion de production (optimisation du coût, qualité et délai).

Le rôle de l'automatisme industriel est prépondérant puisque les systèmes automatisés occupent et contrôlent l'ensemble des secteurs de l'économie, il a comme objectif l'amélioration de la productivité, la qualité, la sécurité et autres variables qui peuvent influencer sur les objectifs de l'entreprise.



**Chapitre III:  
Modélisation et  
simulation du palettiseur**

### III.1 Introduction

En général, la modélisation est une représentation analytique (mathématique) d'un système physique sous forme d'équations mathématiques qui reflètent au mieux le comportement du système considéré. Le modèle mathématique obtenu permet de réaliser des objectifs tels que la conception, la compréhension, la prévention, la simulation et la commande des systèmes.

Aussi, il existe une modélisation par des outils graphiques lorsqu'on a affaire à des systèmes ou automatismes séquentielles ou cycliques. Parmi ces outils, on trouve le GRAFCET (GRAPhe Fonctionnel de Commande des Etapes et Transitions), les Organigrammes, les réseaux de Pétri, et le GEMMA.

Pour notre travail, nous avons opté pour le GRAFCET qui est un outil permettant d'exprimer de façon simple le rapport de causalité entre les entrées et les sorties du système à modéliser.

### III.2 Généralités sur le GRAFCET

Le Grafcet exprime le rapport de causalité entre les entrées et les sorties d'un système à l'aide des réceptivités, des actions associées aux étapes, et des règles d'évolutions. Notons également qu'un nombre quelconque d'entrées/sorties peut être pris en compte.

Une des lacunes du Grafcet, relativement à la commande d'un système automatisé, est l'absence de possibilité de représentation d'interactions de types analogiques avec la partie opérative. Par contre, il faut savoir que certaines équipes de recherche travaillent à l'intégration de ce type d'entrée/sortie.

En général, un Grafcet est réalisé en deux niveaux:

**III.2.1 Grafcet niveau 1:** appelé aussi niveau de la partie commande (spécifications fonctionnelles). Il décrit l'enchaînement des étapes sans préjuger de la technologie.

**III.2.2 Grafcet niveau 2:** appelé aussi niveau de la partie opérative. Il tient compte des détails sur les spécifications technologiques et opérationnelles (la technologie des actionneurs, des pré-actionneurs et des capteurs).

En 1977, le premier rapport de l'AFCE (Association Française pour la Cybernétique Economique et Technique) présente le GRAFCET (Graphe Fonctionnel de Commande Etapes-Transitions). Cet outil fait son entrée dans les programmes français de formation technique en 1979. Il est devenu une norme internationale en 1987.

Dans le langage GRAFCET on trouve :

- Des éléments graphiques de base: étapes, transitions, liaisons orientées, etc.
- Une interprétation associant des expressions logiques :
  - Les actions associées aux étapes.
  - Les réceptivités associées aux transitions.

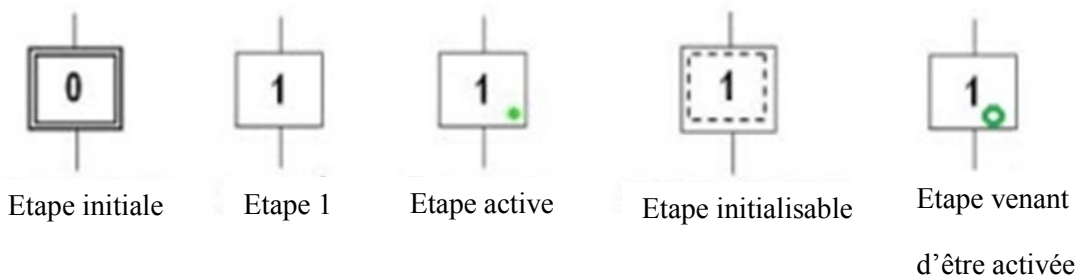
- Des règles d'évolution définissant formellement le comportement dynamique de la partie commande ainsi décrite.

Dans le langage GRAFCET, à chaque étape sont associées une ou plusieurs actions et à chaque transition sera associée une réceptivité.

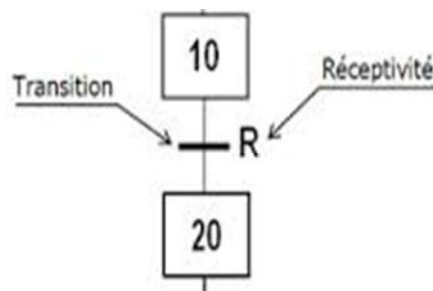
### III.3 Eléments graphiques de base du GRAFCET

**III.3.1 Les étapes:** une étape caractérise un état du système. Elle est représentée par un carré et un nombre « i ». Elle est associée à la variable binaire  $X_i$ , dite variable d'étape. Cette variable a pour valeur le 1 logique lorsque l'étape associée est active, et pour valeur le 0 logique lorsque celle-ci est inactive. Le nombre de l'étape est inscrit dans le carré. Un point peut caractériser une étape active, et un cercle peut indiquer qu'elle vient d'être activée. Une étape peut être initiale, et est alors active au début du processus de commande (les étapes non initiales sont alors inactives). On repère une étape initiale grâce à un doublement du symbole d'étape. L'ensemble de ces étapes caractérise le comportement initial de la partie commande.

On peut associer une interprétation à une étape grâce à une action. Ces actions servent à émettre des ordres vers la partie opérative du système. Lorsqu'une étape est active alors l'action associée est exécutée, lorsque l'étape est inactive l'action associée ne s'exécute pas.



**III.3.2 Les transitions:** une transition est représentée par un petit trait horizontal coupant une liaison verticale. La réceptivité de cette transition est placée à droite. Une transition indique une et une seule possibilité d'évolution entre deux ou plusieurs étapes. Cette évolution s'accomplit par le franchissement de la transition si la réceptivité est vraie. Une réceptivité est associée à chaque transition (l'absence de réceptivité est en fait la réceptivité toujours vrai). C'est une condition qui détermine la possibilité ou non d'évolution du système par cette transition.



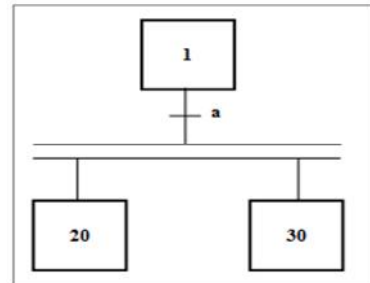
**III.4 Convergence et divergence en « ET » et « OU »**

Deux étapes ne doivent jamais être reliées directement, elles doivent être séparées par une transition.

Deux transitions ne doivent jamais être reliées directement, elles doivent être séparées par une étape.

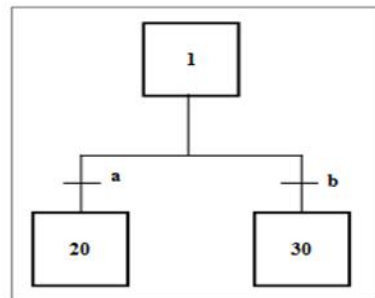
**III.4.1 Divergence simple en ET**

Quand la transition est franchissable, il suffit d'activer deux étapes au lieu d'une. Le seul problème est la désactivation de l'étape précédente: il faut être sûr que les deux étapes suivantes ont eu le temps de prendre l'information d'activation avant de désactiver la précédente.



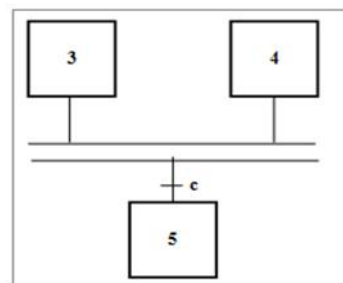
**III.4.2 Divergence en OU**

Quand l'étape 1 est active Et la réceptivité a (ou b) est vraie, l'étape 20 (ou 30) devient active et l'étape 1 est désactivée. Il est possible que l'évolution devienne simultanée si les deux réceptivités sont vraies.



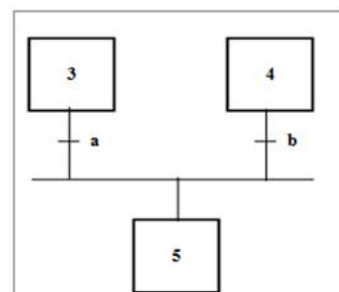
**III.4.3 Convergence en ET**

Quand les deux étapes (3 et 4) sont actives ET la réceptivité c est vraie alors l'étape 5 devient active et les deux étapes (3 et 4) sont désactivées.



**III.4.4 Convergence en OU**

Quand l'étape 3 (ou 4) est active ET la réceptivité a (ou b) est vraie alors l'étape 5 devient active et l'étape 3 (ou 4) est désactivée.



### III.5 Règles d'évolution du GRAFCET [12]

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

#### III.5.1 Règle 1: Situation initiale

La situation initiale d'un Grafcet caractérise le comportement initial de la partie commande vis-à-vis de la partie opérative, de l'opérateur et/ou des éléments extérieurs. Elle correspond aux étapes actives au début du fonctionnement: ces étapes sont les étapes initiales (doublement du symbole d'étape).

#### III.5.2 Règle 2: Franchissement d'une transition

Une transition est dite validée lorsque toutes les étapes amont (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.

#### III.5.3 Règle 3: Evolution des étapes actives

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

#### III.5.4 Règle 4: Evolution simultanée

Plusieurs transitions simultanément franchissables sont simultanément franchies.

#### III.5.5 Règle5: Activation et désactivation simultanée d'une étape

Si au cours du fonctionnement la même étape est simultanément activée et désactivée elle reste active. On évite ainsi des commandes transitoires (néfastes au procédé) non désirées.

Il est important de noter que:

La durée de franchissement d'une transition est très petite (temps de cycle automate) mais non nulle. Ainsi si deux transitions successives (séparée par une étape) ont pour réceptivité le même front d'une variable, alors il faudra deux fronts de cette variable pour franchir les deux transitions.

Le GRAFCET fait l'hypothèse d'un monde asynchrone: deux événements non corrélés ne peuvent survenir simultanément.

III.6 Modélisation avec du Grafcet niveau 1

Pour rendre notre travail plus facile et plus simple pour des raisons de maintenance, nous avons scindé la modélisation à des sous-modèles. De cette façon, nous pouvons identifier et résoudre séparément les problèmes dans les programmes correspondant à chaque modèle.

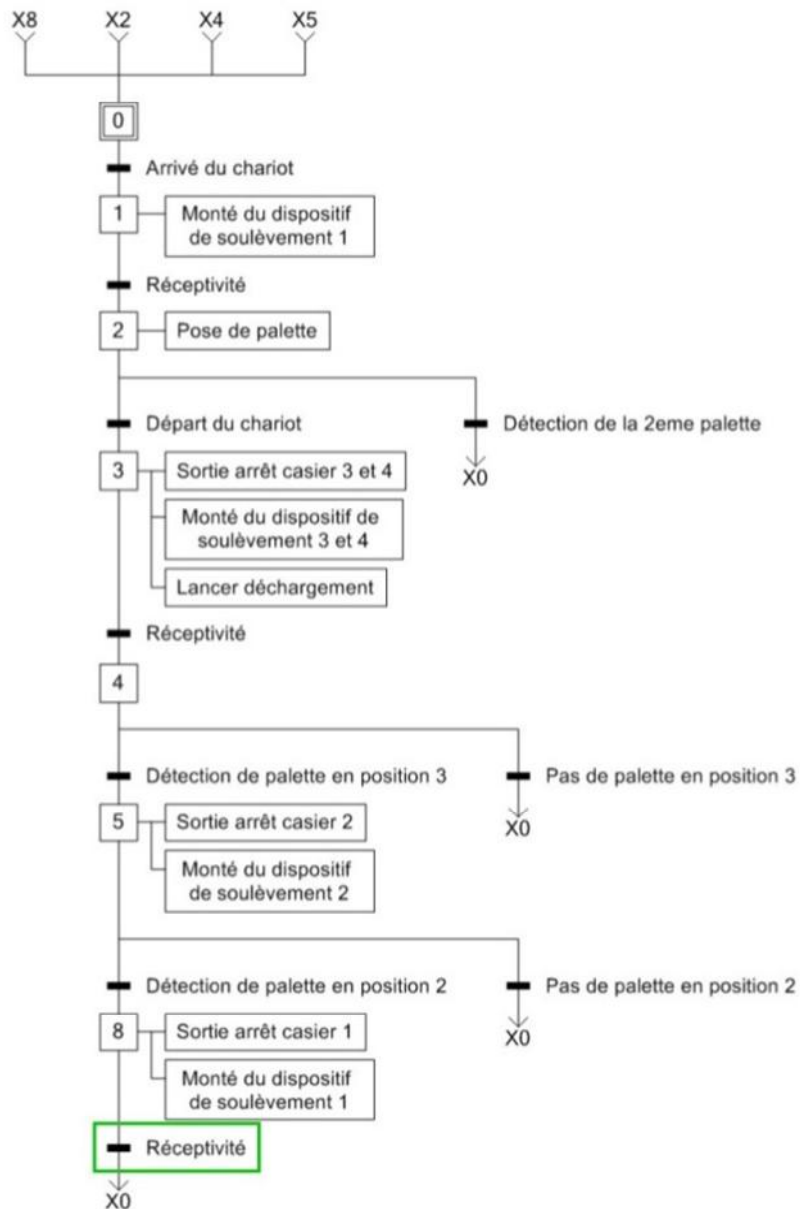


Fig III.1 Arrivée du chariot en grafcet niveau 1



Fig III.2 Déchargement de palettes en grafcet niveau 1

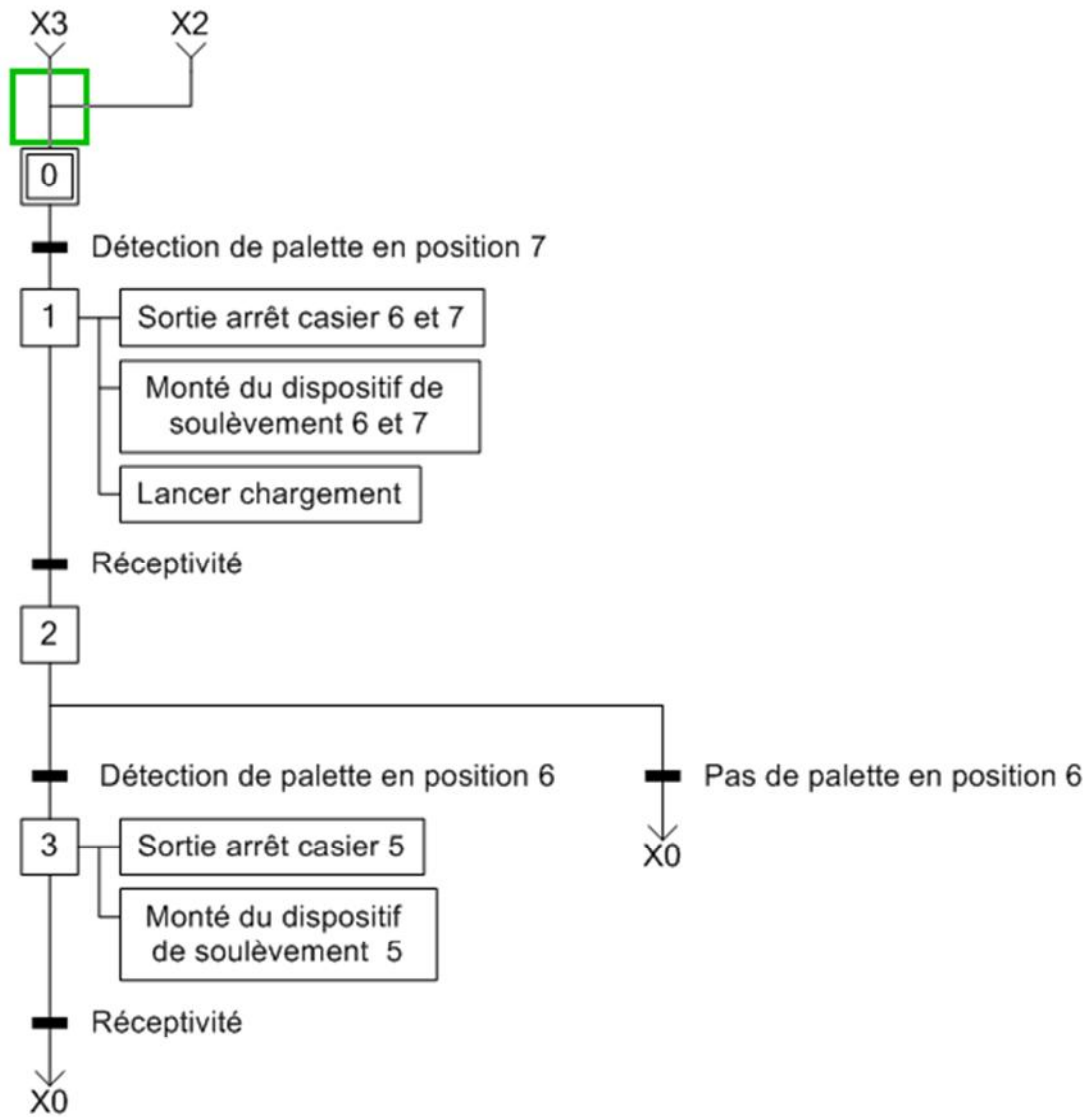


Fig III.3 Déplacement de palettes en grafcet niveau 1

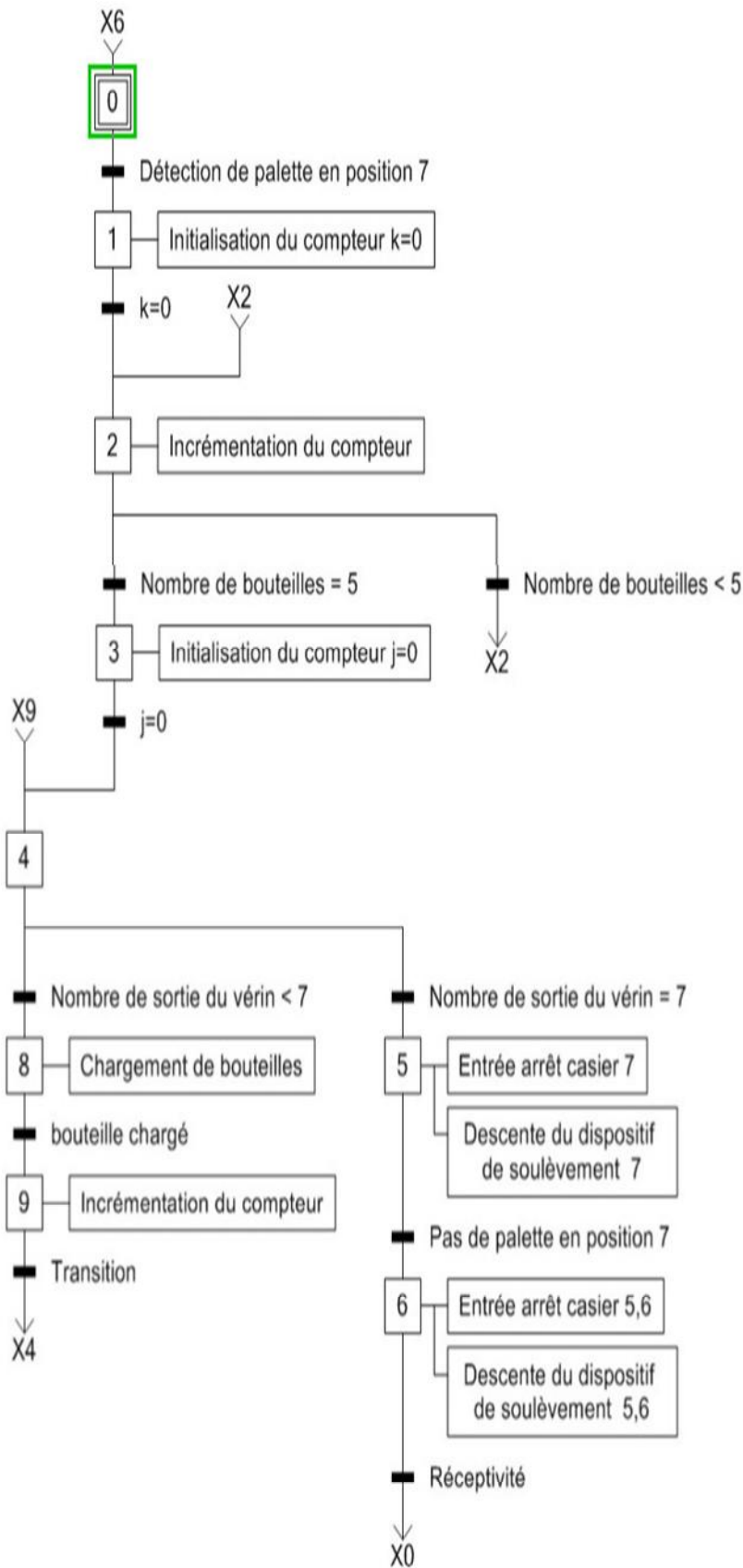
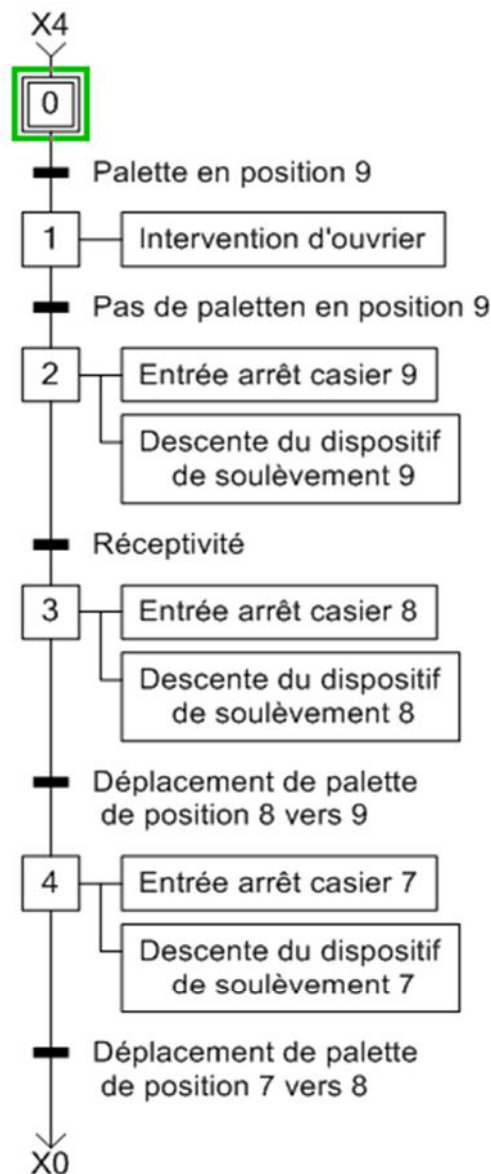


Fig III.4 Chargement de palettes en grafcet niveau 1



**Figure III.5** Enlèvement de palettes pleines en grafcet niveau 1

### III.7 Présentation de l'outil Step 7

Le logiciel STEP 7 est un outil de base pour la configuration et la programmation des systèmes d'automatisation SIMATIC. Le logiciel Step7 permet l'accès "de base" aux automates Siemens. Il permet de programmer individuellement un automate (en différents langages). Il prend également en compte le réseau, ce qui permet d'accéder à tous les automates du réseau (pour les programmer), et il permet éventuellement aux automates d'échanger des messages. Mais il ne permet pas de faire participer les ordinateurs à l'automatisme.

L'écriture d'un programme consiste à créer une liste d'instructions permettant l'exécution des opérations nécessaires au fonctionnement du système.

Le logiciel Step 7 propose 5 langages de programmation:

Le langage booléen (**Logigramme**).

Le langage à contact (**Ladder**)

Le langage LIST

Le langage Littéral structuré

Le langage Step7 graph

### III.8 Principe de la programmation en Step7

#### III.8.1 Structure du programme

La programmation structurée permet la rédaction claire et transparente de programmes. Elle permet la construction d'un programme complet à l'aide de modules qui peuvent être échangés et/ou modifiés à volonté.

Pour permettre une programmation structurée confortable, il faut prévoir plusieurs types de modules: les modules d'organisation (OB), de programmes (FB), fonctionnels (FC), de pas de séquences (SB), de données (DB).

Les modules de programmes (FC) servent à subdiviser le programme en parties fonctionnelles et/ou orientées vers le "processus".

Les modules de données (DB) contiennent des données variables, textes, valeurs de temporisations ou de comptage, résultats de calculs, etc. Ils sont accessibles et actualisables à tout moment.

Les modules séquentiels (SB) sont spécialement utilisés pour effectuer des séquences selon Grafcet. Les paramètres d'entrées y seront les conditions d'avancement d'un pas de séquence et les paramètres de sorties, les ordres à exécuter lorsque ces conditions seront vérifiées.

Les modules d'organisation (OB) sont, comme leur nom l'indique, utilisés pour l'organisation interne du programme et forment ainsi un moyen puissant et essentiel pour la programmation structurée. Ils servent par exemple au déroulement cyclique du programme principal, à l'exécution de programmes d'interruption par des fonctions d'alarmes ou de temps, ou par des fonctions de diagnostic interne autant du point de vue hardware que software du système complet

Les modules fonctionnels (FB) sont librement paramétrables spécialement conçus pour la standardisation de fonctions complexes.

### III.8.2 Créer son projet

Un projet contient la description complète de votre automatisme. Il comporte donc deux grandes parties: la description du matériel, et la description du fonctionnement (le programme).

En entrant dans Step7, il peut y avoir un assistant qui vous propose de créer un nouveau projet comme le montre la figure ci-dessous.

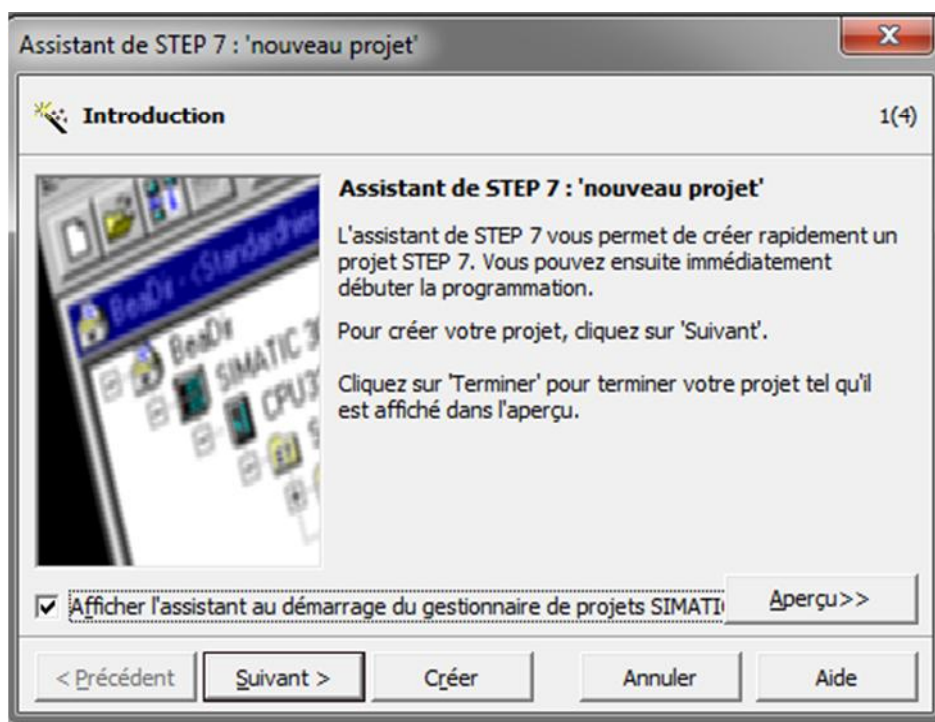


Fig III.6 Assistant de step7

L'assistant de STEP 7 nous permet de créer rapidement un projet STEP 7. Nous pouvons choisir la configuration CPU, les blocs et le langage pour les blocs choisis.

Les programmes sont mémorisés et regroupés en PROJETS. Dans un projet, il y a plusieurs niveaux:

Au niveau 1, on trouve le nom de projet (exemple: mémoire 2). Le projet contient d'une part les stations connectées (Station SIMATIC 300) et les réseaux configurés (réseau MPI (1)).

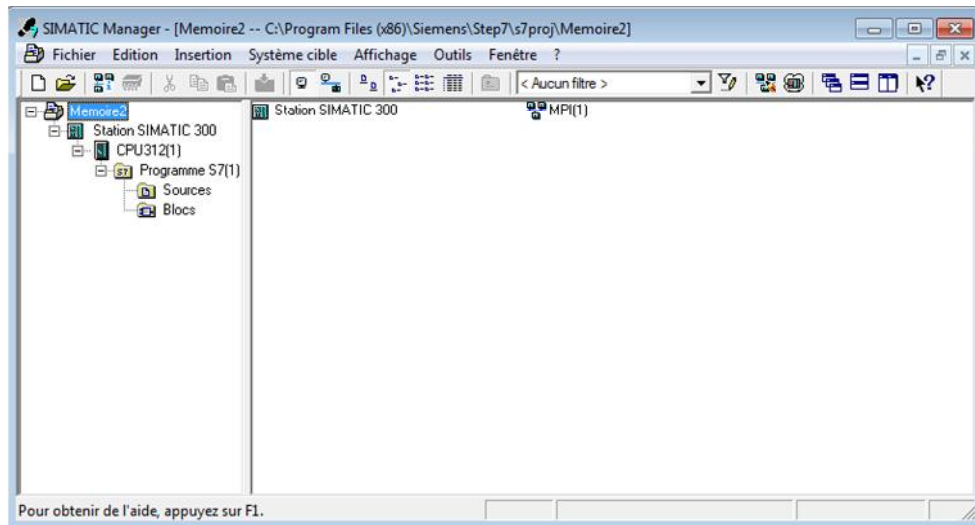


Fig III.7 Simatic S7 manager

Au niveau 2, on accède à la définition de la station: le type de CPU (CPU 312) et la station de configuration (Matériel).

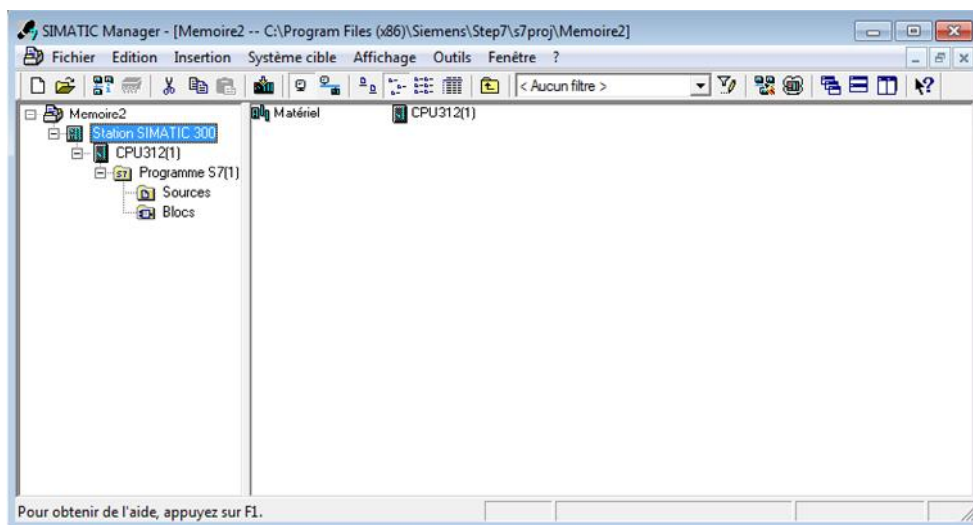


Fig III.8 Niveau 2, configuration de la station

Au niveau 3, on accède aux différents programmes (Programmes S7) et à la table des connexions définies pour le réseau (Liaisons).

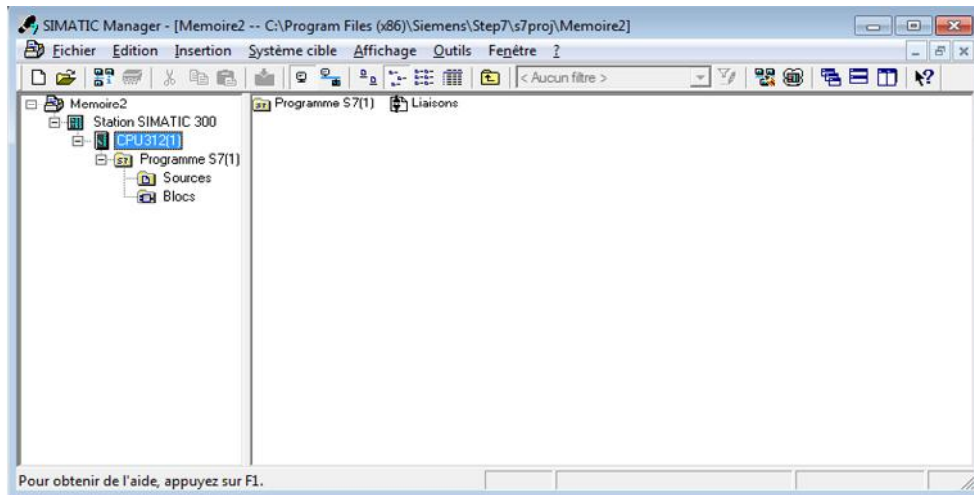


Fig. III.9 Niveau 3, niveau de la CPU

Au niveau 4, on accède :

- Aux sources externes éventuelles (répertoire Sources) comme des fichiers textes en listes d'instructions.
- A la table des mnémoniques (fichiers Mnémoniques).

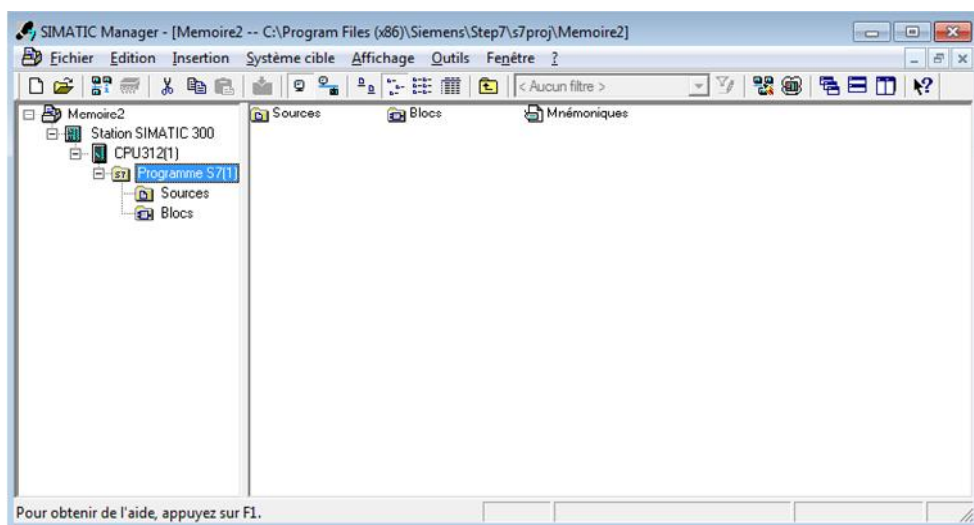


Fig III.10 Niveau 4, niveau des programmes

Au niveau 5, on accède au programme proprement dit (blocs d'instructions OBs, FBs..., blocs de données DBs...)

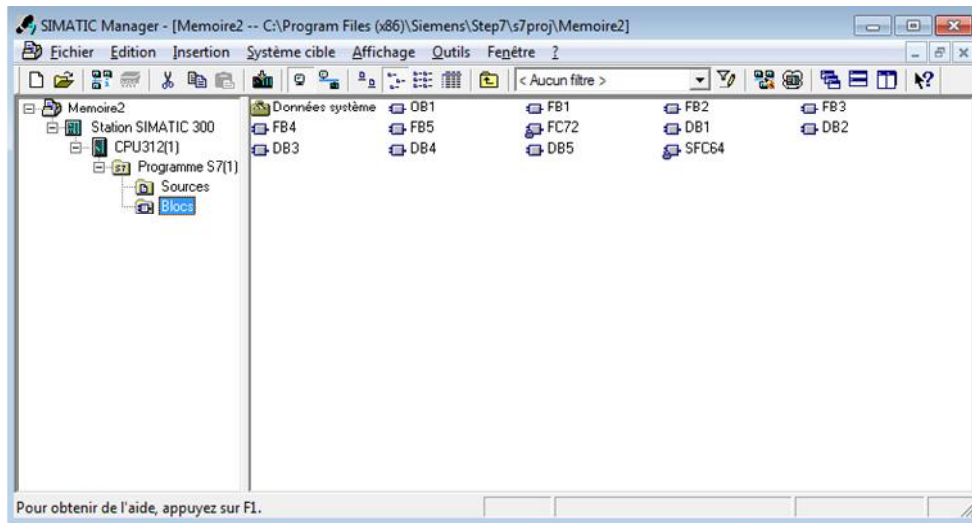


Fig III.11 Niveau 5, les blocs de programme

### III.8.3 Configuration de la station

Quand on crée un nouveau projet, il faut définir la configuration matérielle du niveau 2 qui permet de configurer l'automate lors du transfert du programme dans celui-ci. Cette configuration doit correspondre évidemment à la configuration matérielle réelle qui consiste à détailler la position et le type des différentes cartes enfichées sur le bus fond de panier.

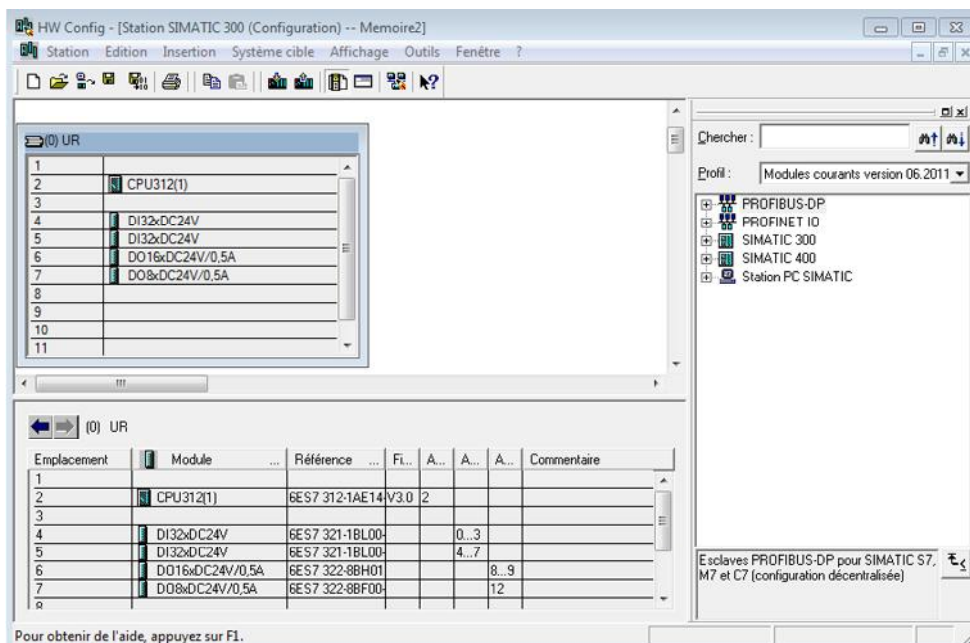


Fig III.12 Configuration matérielle

### III.8.4 Edition

On peut maintenant passer en mode édition dans le fichier programme (niveau 5). Il faut sélectionner un bloc et double-cliquer sur l'icône du fichier, du bloc de données, etc. On accède alors à l'éditeur de programme.

Pour créer un module fonctionnel qui nous permet de programmer en Step7 graphe, il faut cliquer avec le bouton droit sur OB1 → Insérer un nouvel objet → bloc fonctionnel.

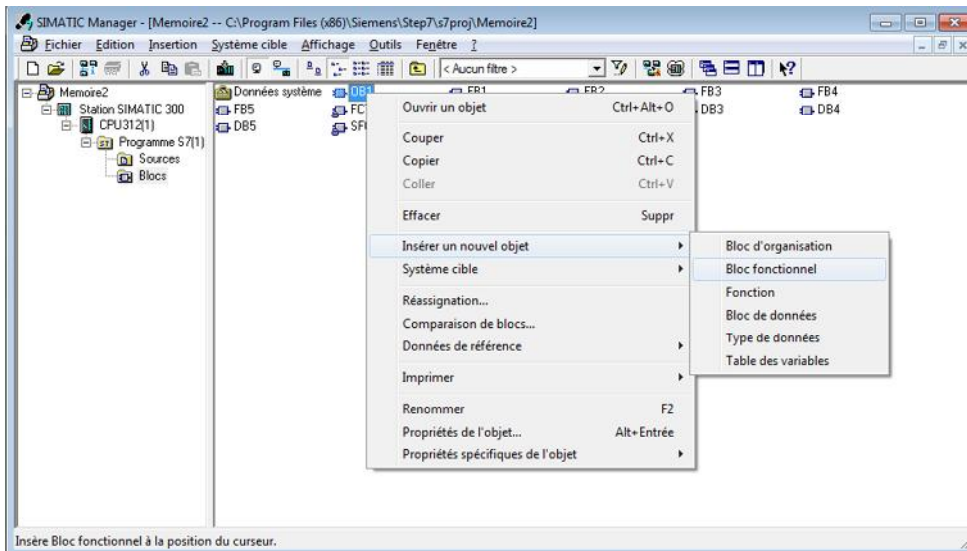


Fig III.13 Création d'un bloc fonctionnel

Ensuite on double-clique sur notre bloc FB qui nous permet d'accéder à l'outil de programmation en step7 graph.

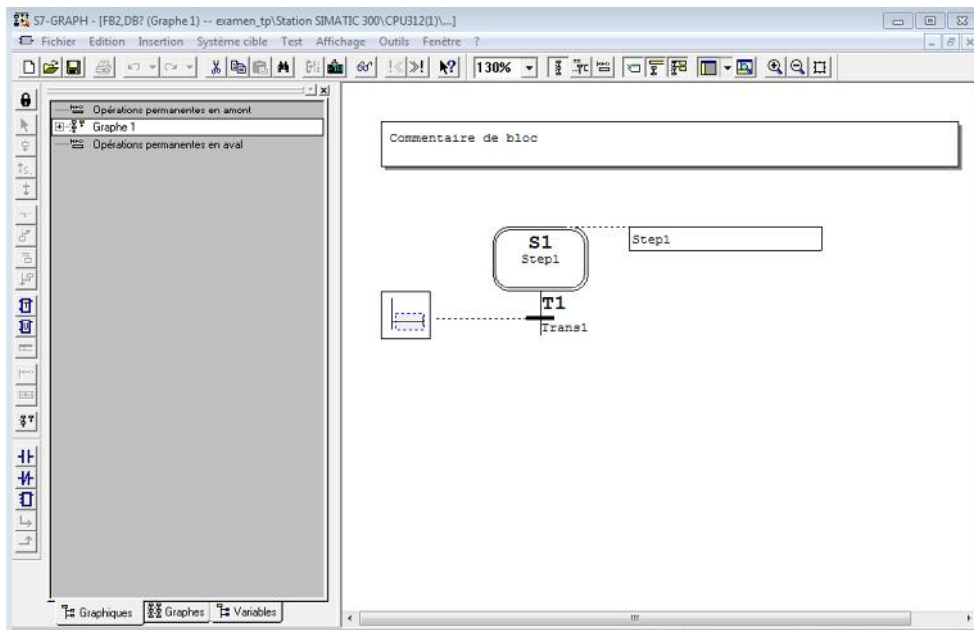


Fig III.14 Outil de programmation en step7 graph

A l'aide de la barre d'éléments GRAPH située sur le côté gauche de l'écran, on peut commencer notre programme par ajout des étapes et transitions, la branche OU et le saut, etc.

### III.8.5 Edition des mnémoniques

Les mnémoniques sont des noms que l'on peut donner aux variables afin de faciliter la programmation. Ils améliorent la lisibilité du programme et servent également de documentation.

Les mnémoniques se définissent au niveau 4 en double-cliquant sur l'icône "Mnémonique". On accède alors à l'éditeur des mnémoniques.

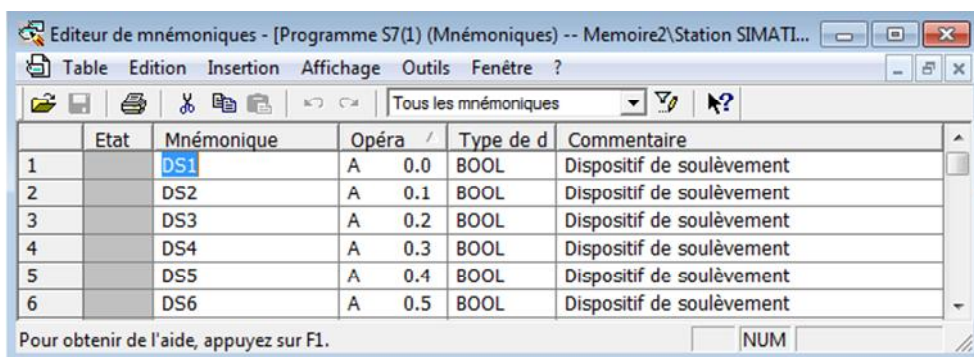


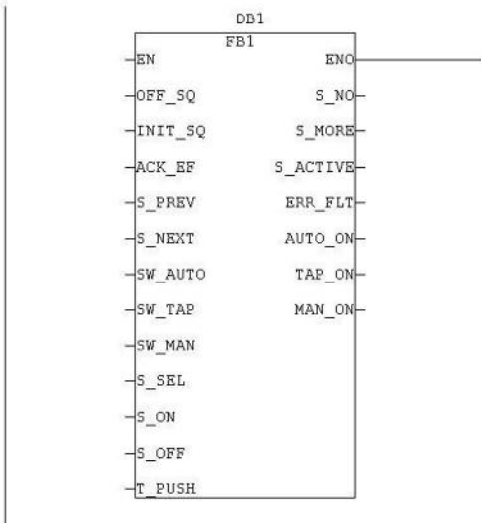
Fig III.15 Editeur de mnémoniques

### III.9 Le programme en step7 graph

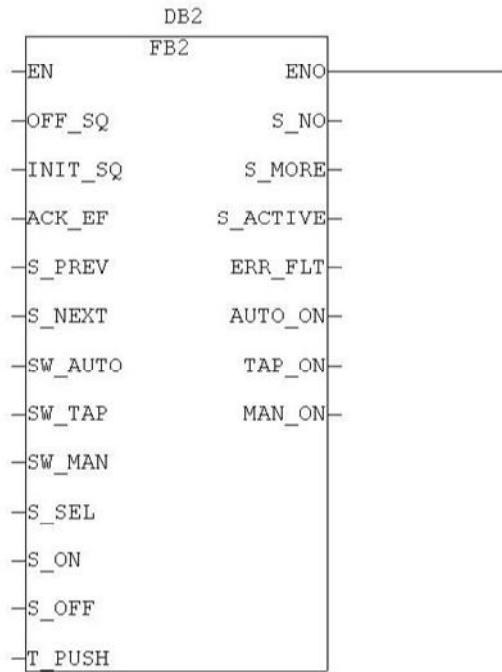
Dans notre programme, on a un bloc d'organisation OB1 qui contient les compteurs et les blocs DB.

Bloc : OB1 "Main Program Sweep (Cycle)"

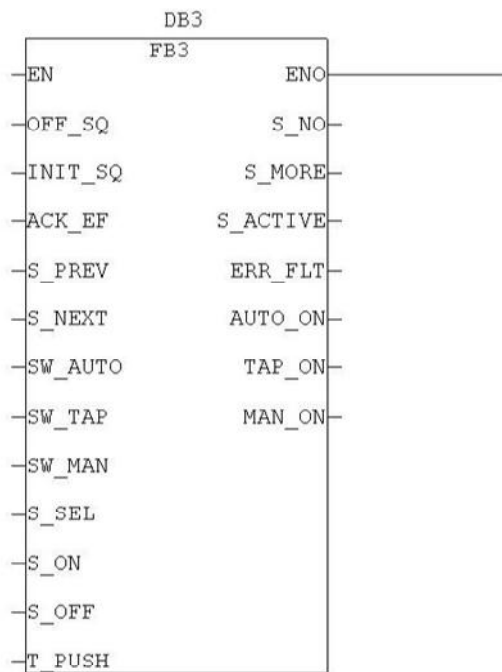
Réseau : 1 Arrivé palette



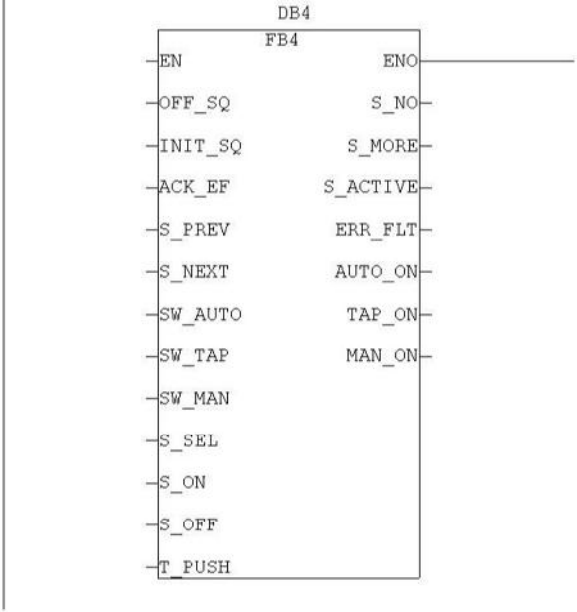
Réseau : 2 Déchargement



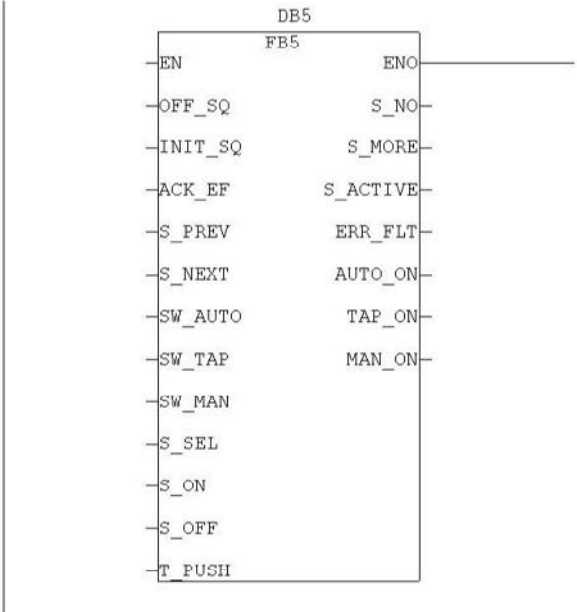
Réseau : 3 Déplacement de palette vers chargement



Réseau : 4      Chargement



Réseau : 5      Déplacement de palette vers position 9



Réseau : 6 Initialisation de compteur k = 5



Réseau : 7 Décrémentation de compteur k



Réseau : 8 Initialisation de compteur j = 7



Réseau : 9 Décrémentation de compteur j



Le programme contient aussi 5 blocs fonctionnels FB donnés dans l'annexe III.1.

III.10 Tableau des mnémoniques

	Etat	Mnémonique	Opéra /	Type de d	Commentaire
1		DS1	A 0.0	BOOL	Dispositif de soulèvement
2		DS2	A 0.1	BOOL	Dispositif de soulèvement
3		DS3	A 0.2	BOOL	Dispositif de soulèvement
4		DS4	A 0.3	BOOL	Dispositif de soulèvement
5		DS5	A 0.4	BOOL	Dispositif de soulèvement
6		DS6	A 0.5	BOOL	Dispositif de soulèvement
7		DS7	A 0.6	BOOL	Dispositif de soulèvement
8		DS8	A 0.7	BOOL	Dispositif de soulèvement
9		DS9	A 1.0	BOOL	Dispositif de soulèvement
1		AC1	A 1.1	BOOL	Arrêt casier
1		AC2	A 1.2	BOOL	Arrêt casier
1		AC3	A 1.3	BOOL	Arrêt casier
1		AC4	A 1.4	BOOL	Arrêt casier
1		AC5	A 1.5	BOOL	Arrêt casier
1		AC6	A 1.6	BOOL	Arrêt casier
1		AC7	A 1.7	BOOL	Arrêt casier
1		AC8	A 2.0	BOOL	Arrêt casier
1		AC9	A 2.1	BOOL	Arrêt casier
1		DECH1	A 2.2	BOOL	vérin dechargement palette
2		DECH2	A 2.3	BOOL	vérin dechargement palette
2		PL	A 2.4	BOOL	vérin dechargement bouteilles
2		CHAR	A 2.5	BOOL	
2		VAB	A 2.6	BOOL	vérin chargement
2		BP	E 0.0	BOOL	
2		DS1E	E 0.1	BOOL	Dispositif de soulèvement
2		DS1S	E 0.2	BOOL	Dispositif de soulèvement
2		DS2E	E 0.3	BOOL	Dispositif de soulèvement
2		DS2S	E 0.4	BOOL	Dispositif de soulèvement
2		DS3E	E 0.5	BOOL	Dispositif de soulèvement
3		DS3S	E 0.6	BOOL	Dispositif de soulèvement
3		DS4E	E 0.7	BOOL	Dispositif de soulèvement
3		DS4S	E 1.0	BOOL	Dispositif de soulèvement
3		DS5E	E 1.1	BOOL	Dispositif de soulèvement
3		DS5S	E 1.2	BOOL	Dispositif de soulèvement

	Etat	Mnémonique	Opéra /	Type de d	Commentaire
3		DS6E	E 1.3	BOOL	Dispositif de soulèvement
3		DS6S	E 1.4	BOOL	Dispositif de soulèvement
3		DS7E	E 1.5	BOOL	Dispositif de soulèvement
3		DS7S	E 1.6	BOOL	Dispositif de soulèvement
3		DS8E	E 1.7	BOOL	Dispositif de soulèvement
4		DS8S	E 2.0	BOOL	Dispositif de soulèvement
4		DS9E	E 2.1	BOOL	Dispositif de soulèvement
4		DS9S	E 2.2	BOOL	Dispositif de soulèvement
4		AC1E	E 2.3	BOOL	Arrêt casier
4		AC1S	E 2.4	BOOL	Arrêt casier
4		AC2E	E 2.5	BOOL	Arrêt casier
4		AC2S	E 2.6	BOOL	Arrêt casier
4		AC3E	E 2.7	BOOL	Arrêt casier
4		AC3S	E 3.0	BOOL	Arrêt casier
4		AC4E	E 3.1	BOOL	Arrêt casier
5		AC4S	E 3.2	BOOL	Arrêt casier
5		AC5E	E 3.3	BOOL	Arrêt casier
5		AC5S	E 3.4	BOOL	Arrêt casier
5		AC6E	E 3.5	BOOL	Arrêt casier
5		AC6S	E 3.6	BOOL	Arrêt casier
5		AC7E	E 3.7	BOOL	Arrêt casier
5		AC7S	E 4.0	BOOL	Arrêt casier
5		AC8E	E 4.1	BOOL	Arrêt casier
5		AC8S	E 4.2	BOOL	Arrêt casier
5		AC9E	E 4.3	BOOL	Arrêt casier
6		AC9S	E 4.4	BOOL	Arrêt casier
6		P1	E 4.5	BOOL	Capteur palette 1
6		P2	E 4.6	BOOL	Capteur palette 2
6		P3	E 4.7	BOOL	Capteur palette 3
6		P4	E 5.0	BOOL	Capteur palette 4
6		P5	E 5.1	BOOL	Capteur palette 5
6		P6	E 5.2	BOOL	Capteur palette 6
6		P7	E 5.3	BOOL	Capteur palette 7
6		P8	E 5.4	BOOL	Capteur palette 8

6		P9	E	5.5	BOOL	Capteur palette 9
7		CD0	E	5.6	BOOL	Capteur vérin dechargement
7		CD1	E	5.7	BOOL	Capteur vérin dechargement
7		CD2	E	6.0	BOOL	Capteur vérin dechargement
7		CD3	E	6.1	BOOL	Capteur vérin dechargement
7		CD4	E	6.2	BOOL	Capteur vérin dechargement
7		CD5	E	6.3	BOOL	Capteur vérin dechargement
7		CD6	E	6.4	BOOL	Capteur vérin dechargement
7		CD7	E	6.5	BOOL	Capteur vérin dechargement
7		pb	E	6.6	BOOL	Capteur bouteilles dechargement
7		plh	E	6.7	BOOL	
8		plb	E	7.0	BOOL	
8		cp	E	7.1	BOOL	Capteur bouteilles chargement
8		C0	E	7.2	BOOL	Capteur vérin chargement
8		C1	E	7.3	BOOL	Capteur vérin chargement
8		vabe	E	7.4	BOOL	capteur vérin arrêt bouteilles-entrée
8		vabs	E	7.5	BOOL	capteur vérin arrêt bouteilles-sortie
8		CH1	E	7.6	BOOL	Capteur chariot 1
8		CH2	E	7.7	BOOL	Capteur chariot 2
8		PH1	E	8.0	BOOL	Capteur 2 palette
8		G7_STD_3	FC	72	FC 72	
9		Cycle Execution	OB	1	OB 1	
9		TIME_TCK	SFC	64	SFC 64	Read the System Time

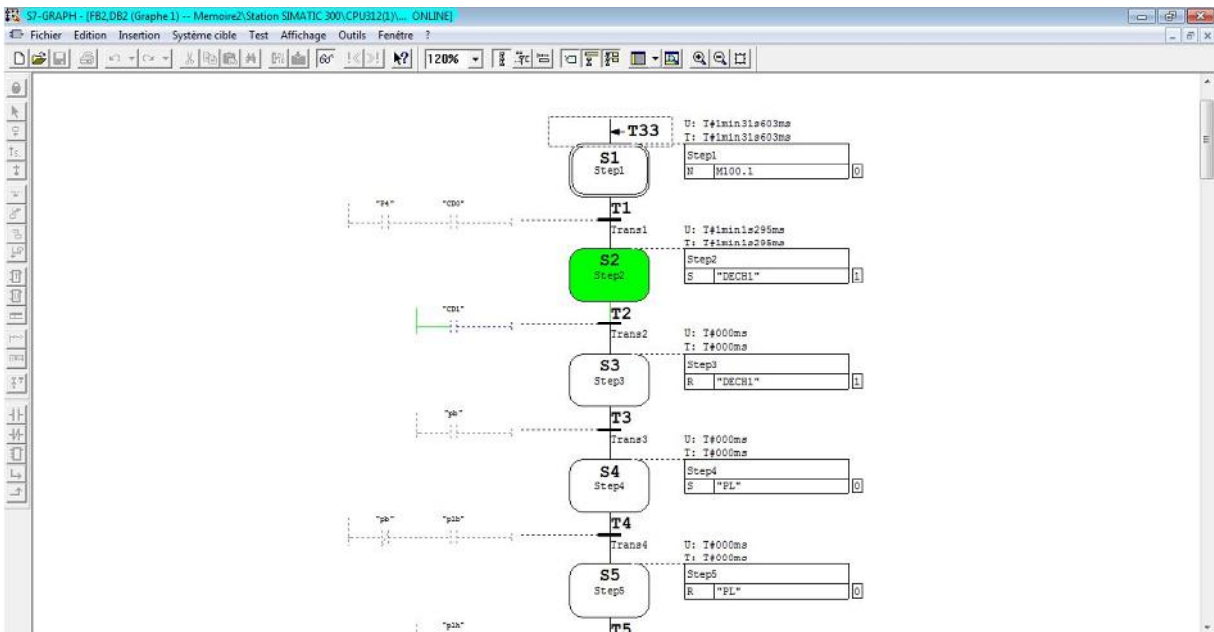
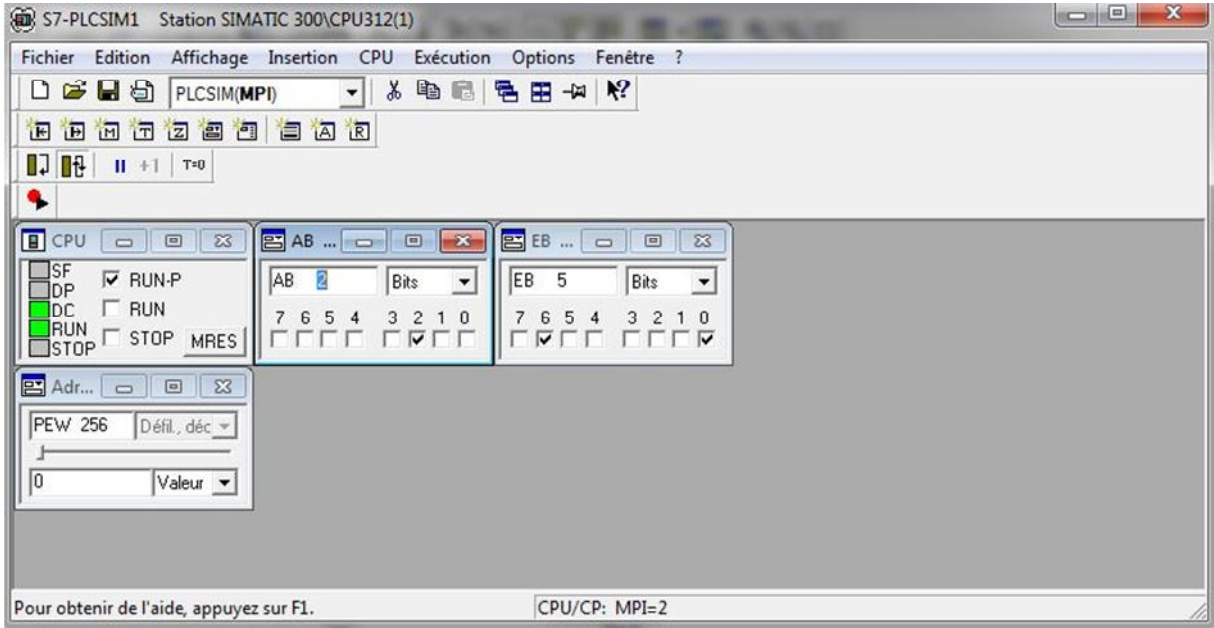
### III.11 Résultats de simulation

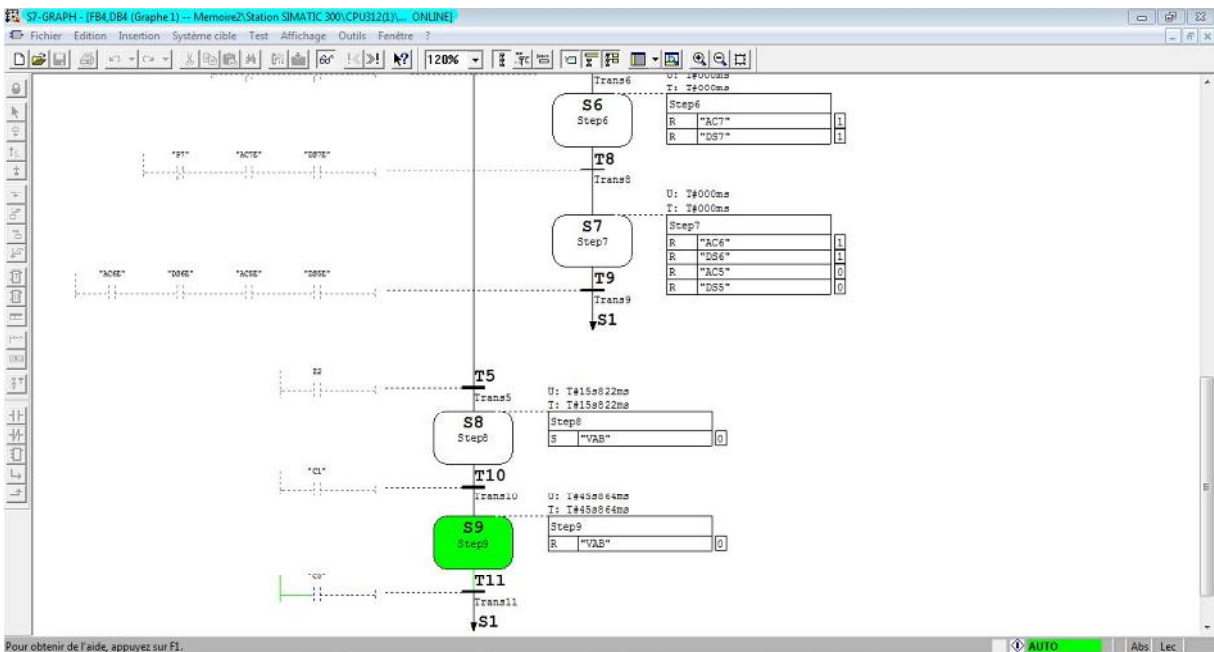
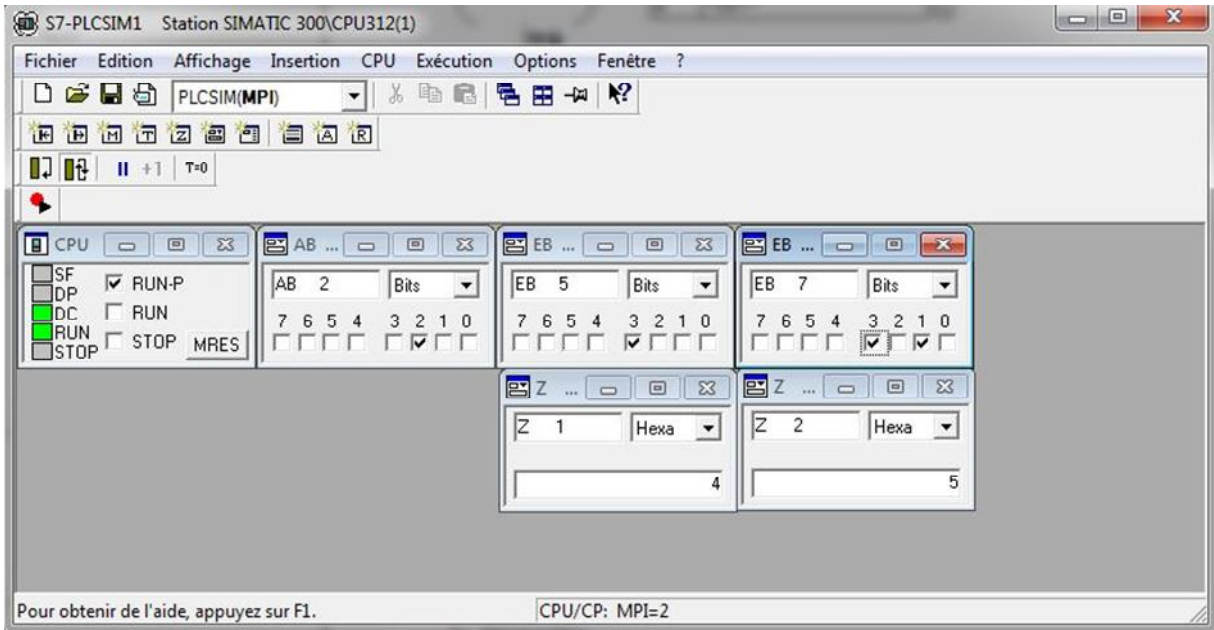
#### III.11.1 Introduction sur PLCSIM [17]

S7-PLCSIM nous permet d'exécuter et de tester notre programme utilisateur STEP7 dans un automate programmable (AP) que nous simulons. Cette simulation s'exécute dans notre ordinateur ou dans notre console de programmation, par exemple une PG 740, une PG Power ou une PG Field. La simulation étant complètement réalisée au sein du logiciel STEP 7, il n'est pas nécessaire qu'une liaison soit établie avec un matériel S7 quelconque (CPU ou module de signaux). Nous pouvons utiliser S7-PLCSIM pour simuler des programmes utilisateur STEP 7 qui ont été conçus pour des S7-300, des S7-400 et des contrôleurs WinAC.

S7-PLCSIM fournit une interface simple avec le programme utilisateur STEP7 pour visualiser et forcer différents objets comme des entrées et des sorties. Tout en exécutant votre programme dans l'AP de simulation, on a également la possibilité de mettre en œuvre les diverses applications du logiciel STEP7 comme, par exemple, la table des variables (VAT) afin d'y visualiser et d'y forcer des variables.

III.11.2 Résultats de simulation de notre programme





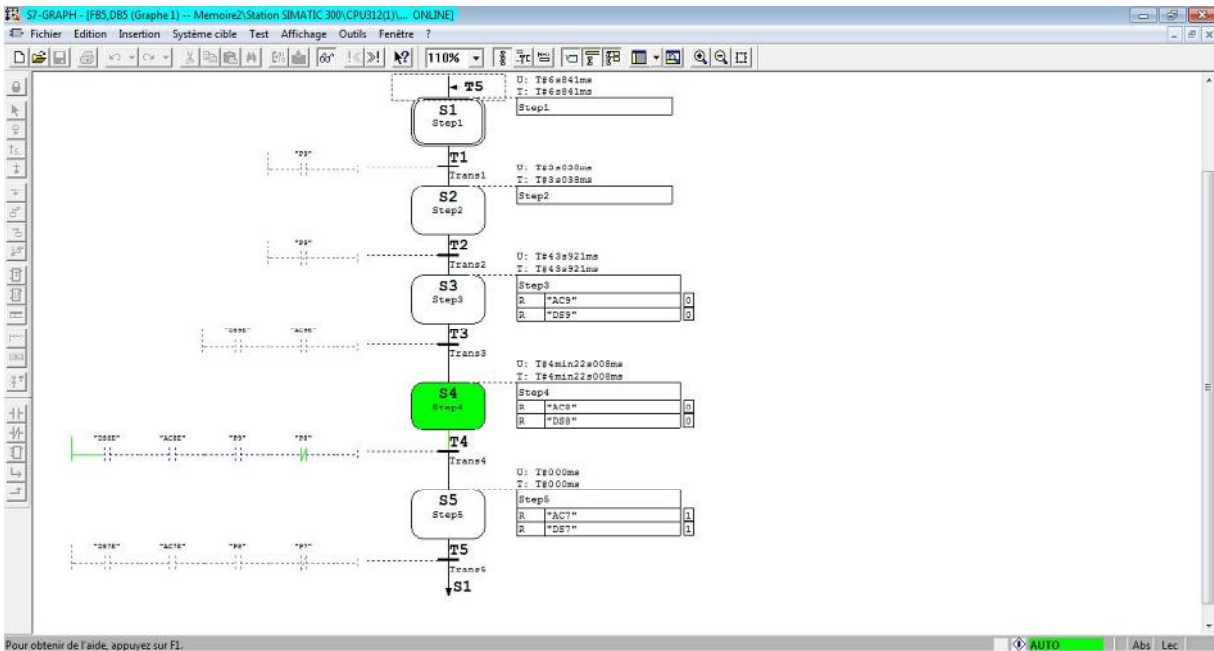
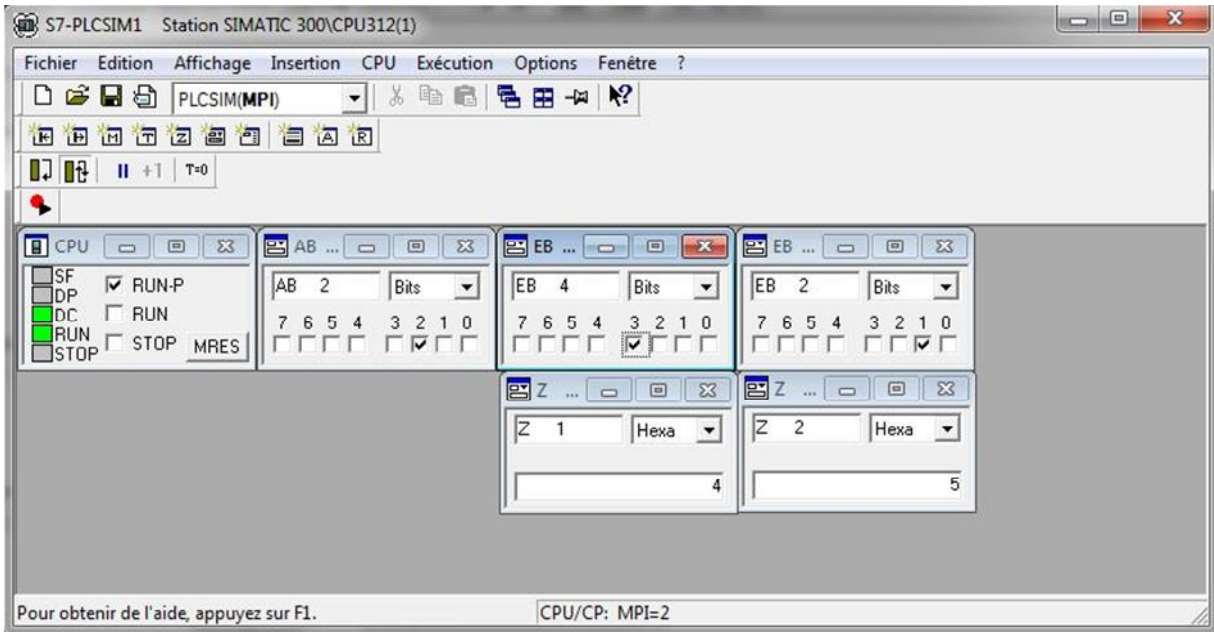


Fig.III.16 Résultats de simulation

**III.12 Conclusion**

Dans ce chapitre, nous avons modélisé le fonctionnement du palettiseur en utilisant le Grafcet niveau 1. La programmation et la simulation a été faite en utilisant le logiciel STEP 7. Nous avons vu comment utiliser les outils fournis par Siemens pour faire une solution simple et valide à l'aide de Grafcet. Nous avons divisé le programme principal à des mini-programmes pour rendre l'analyse et les tests plus faciles.



**Chapitre IV:  
Système de supervision  
du palettiseur**

## IV.1 Introduction

Lorsque la complexité des processus augmente et que les machines et les 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).

Dans ce chapitre, nous réalisons un système de supervision en utilisant le logiciel WinCC flexible.

## IV.2 WinCC flexible [13]

Un système IHM constitue l'interface entre l'homme (opérateur) et le processus (machine/installation). Le contrôle proprement dit du processus est assuré par le système d'automatisation. 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.

### IV.2.1 Utilisation de SIMATIC WinCC flexible

WinCC flexible est le logiciel IHM pour la réalisation, par des moyens d'ingénierie simples et efficaces, des concepts d'automatisation évolutifs, au niveau machine. WinCC flexible réunit les avantages suivants:

- Simplicité
- Ouverture
- Flexibilité

### IV.2.2 WinCC flexible Runtime

**Principe:** Au Runtime, l'opérateur peut réaliser le contrôle-commande du processus. Les tâches suivantes sont alors exécutées:

- Communication avec les automates.
- Affichage des vues à l'écran.
- Commande du processus, par exemple spécification de consignes ou ouverture et fermeture de vannes.
- Archivage des données de runtime actuelles, des valeurs de processus et événements d'alarme.

### IV.2.3 La liaison WinCC avec PLCsim

Concernant la communication dans notre application, nous avons deux types de réseaux :

- Communication SIMATIC 300-PC déporté via un réseau MPI;
- Communication SIMATIC 300-Pupitre opérateur via un réseau PROFIBUS-DP.

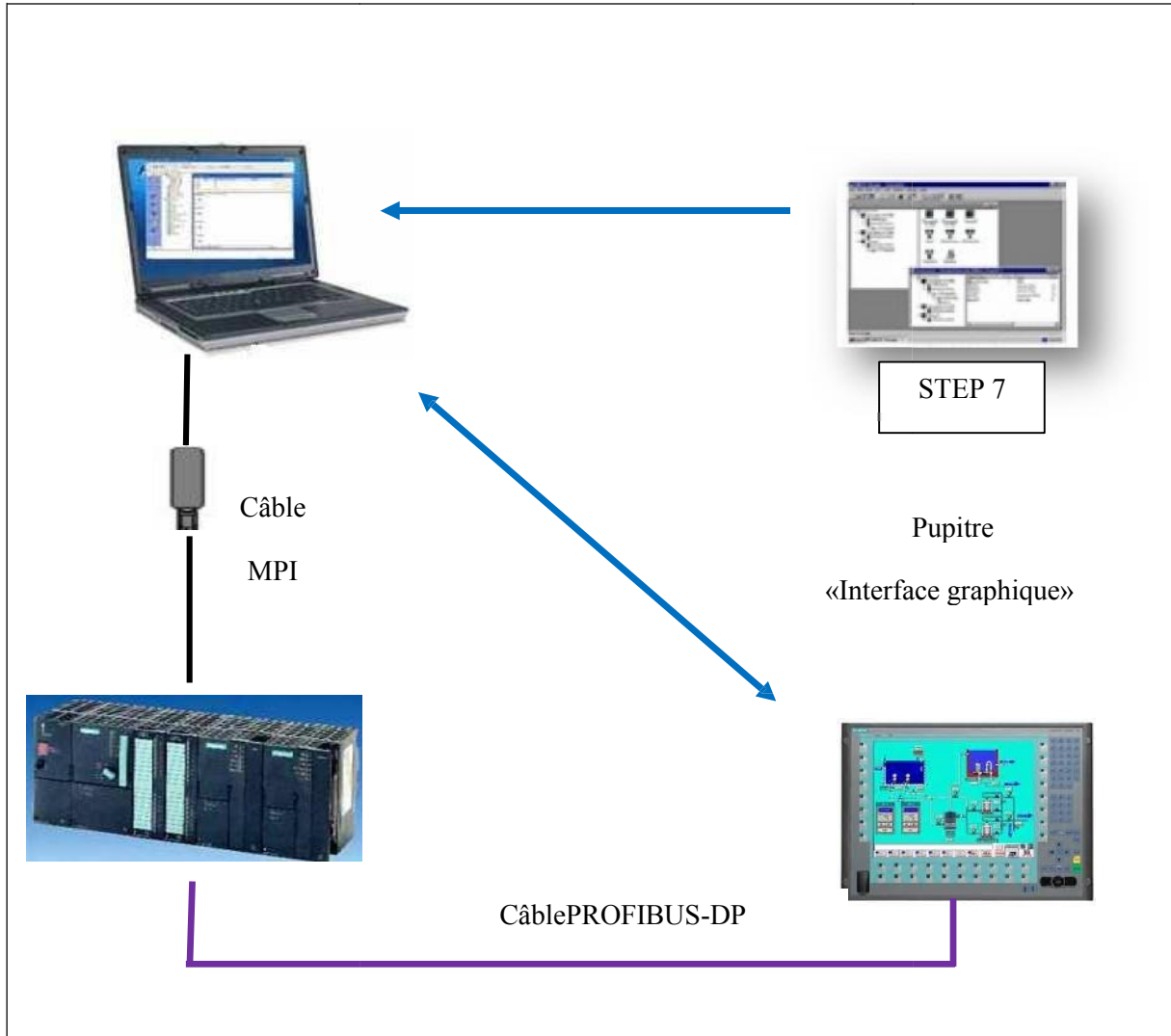


Fig. IV.1 Schéma de principe de l'application

### IV.3 Vues

Dans WinCC flexible, chaque projet créé contient principalement des vues que l'on crée pour le contrôle-commande de machines et d'installations. Lors de la création des vues, vous disposez d'objets prédéfinis permettant d'afficher des procédures et de définir des valeurs du processus.

Les différents outils et barres de l'éditeur des vues sont représentés dans la figure qui suit:

- **Barre des menus:** La barre des menus contient toutes les commandes nécessaires à l'utilisation de WinCC flexible. Les raccourcis disponibles sont indiqués en regard de la commande du menu.
- **Barres d'outils:** La barre d'outils permet d'afficher tout dont le programmeur a besoin.
- **Zone de travail:** La zone de travail sert à configurer des vues, de façon qu'elles soient plus compréhensibles par l'utilisateur, et très faciles à manipuler et consulter les résultats.
- **Boîte à outils:** La fenêtre des outils propose un choix d'objets simples ou complexes qu'on insère dans les vues, par exemple des objets graphiques et éléments de commande.
- **Fenêtre des propriétés:** Le contenu de la fenêtre des propriétés dépend de la sélection actuelle dans la zone de travail, lorsqu'un objet est sélectionné, on peut éditer les propriétés de l'objet en question dans la fenêtre des propriétés.

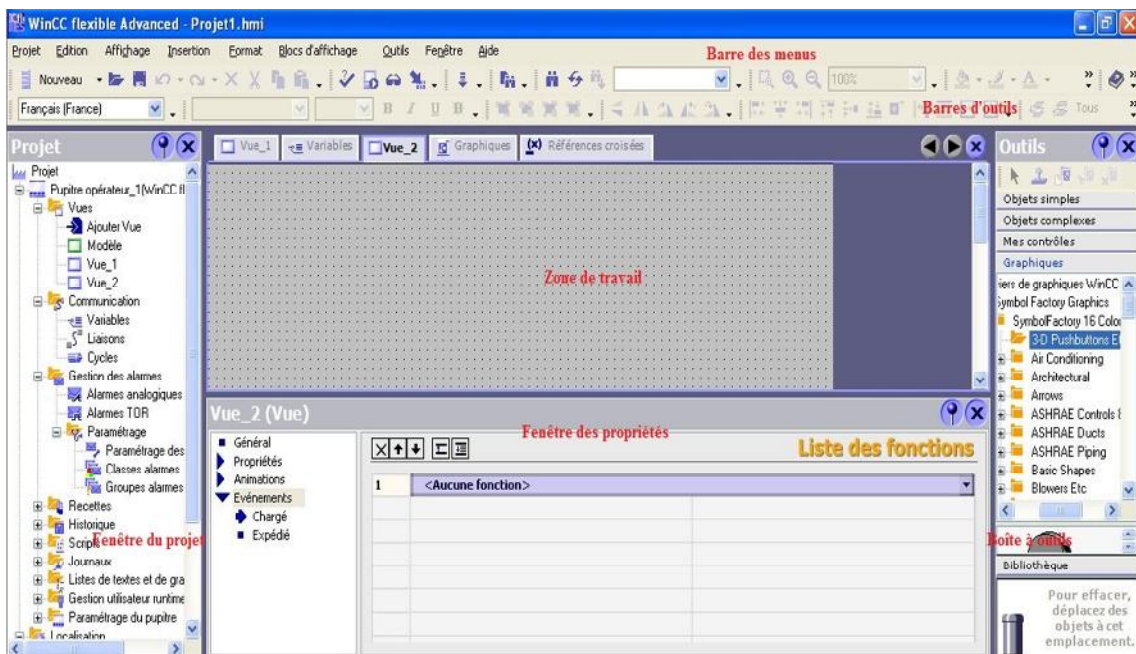


Fig. IV.2 Vue d'ensemble du logiciel WinCC flexible

### IV.3.1 Avantages de l'intégration dans Step7

Lors de la configuration intégrée, nous avons accès aux données de configuration que nous avons créées lors de la configuration de l'automate avec Step 7. Les avantages sont les suivants:

- ✓ Nous pouvons utiliser le gestionnaire SIMATIC Manager comme poste central de création, d'édition et de gestion des automates SIMATIC et des projets WinCC flexible.

- ✓ Les paramètres de communication de l'automate sont entrés par défaut lors de la création du projet WinCC flexible. Toute modification sous Step 7 se traduit par une mise à jour des paramètres de communication sous WinCC flexible.
- ✓ Lors de la configuration de variables et de pointeurs de zone, nous pouvons accéder sous WinCC flexible directement aux mnémoniques de Step 7. Sélectionnez simplement sous WinCC flexible le mnémonique Step 7 auquel vous voulez affecter une variable. Les modifications de mnémonique sous Step 7 sont mises à jour sous WinCC flexible.
- ✓ Il nous suffit de définir les mnémoniques une seule fois sous Step 7 pour pouvoir les utiliser sous Step 7 et sous WinCC flexible.
- ✓ Nous pouvons créer un projet WinCC flexible sans intégration dans Step 7 et intégrer ce projet ultérieurement dans Step 7.
- ✓ Dans un multiprojet Step 7, nous pouvons configurer des liaisons de communication sur plusieurs projets.

#### IV.4 Simulation

Les résultats de simulations sont donnés par les figures ci-dessous :

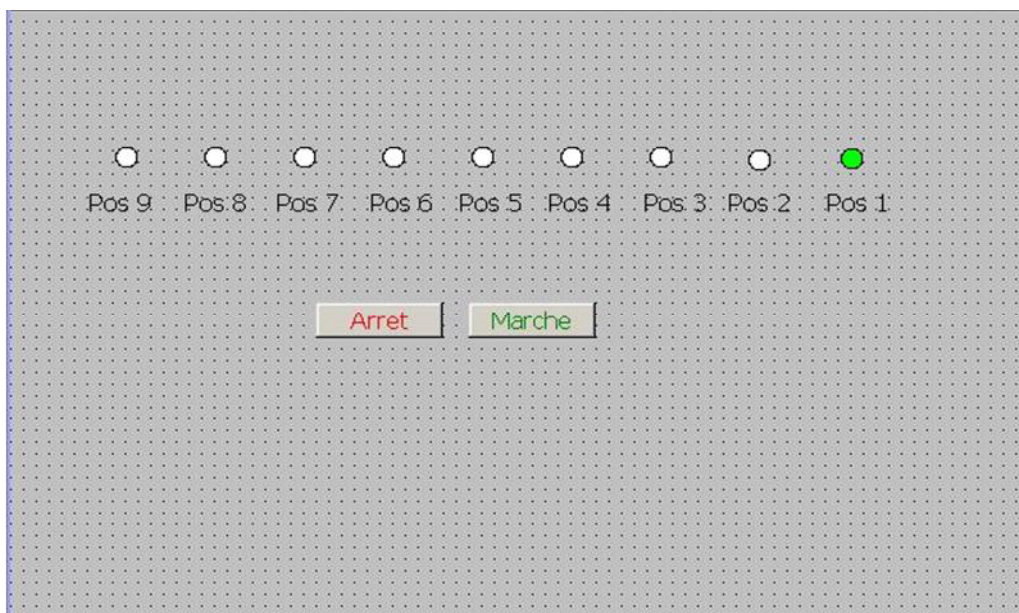


Fig. IV.3 Ecran de contrôle

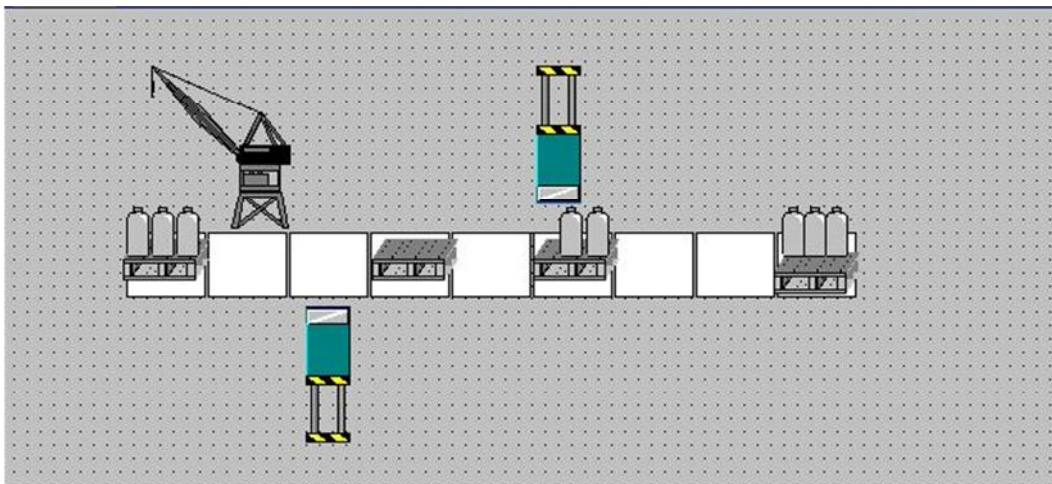
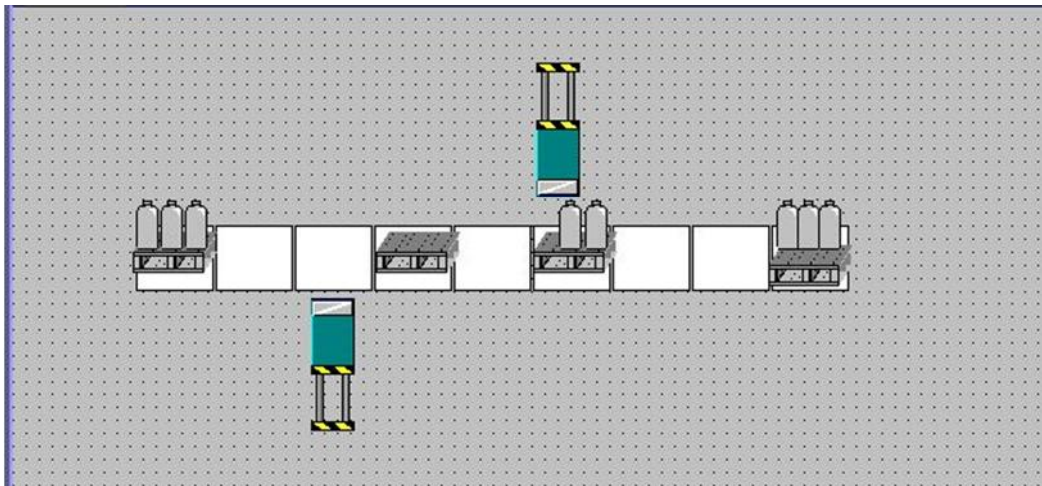
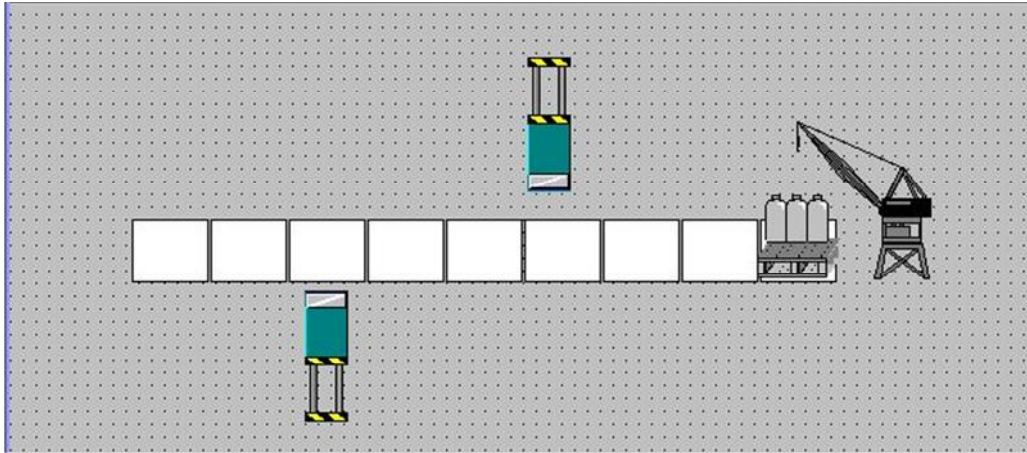


Fig. IV.4 Résultats de simulation

**IV. 5 Conclusion**

La mise en place de notre plateforme de supervision s'est faite grâce au logiciel WinCC qui fait partie de la panoplie des logiciels SIMATIC HMI. Grâce à WinCC, on peut avoir une vision globale de l'architecture matérielle du procédé en temps réel, ce qui permet à l'opérateur de suivre le fonctionnement du procédé et d'effectuer des tâches de routine (vérification des paramètres, contrôle de l'installation, etc). Le suivi de l'évolution du processus permettra d'améliorer la qualité de production.



**Conclusion générale**

Notre travail a été effectué au sein de l'entreprise Naftal. L'objectif est d'automatiser une palettiseuse de bouteilles de gaz par le nouvel automate S7-300.

La technologie qui a été installée sur le palettiseur est du type pneumatique. Cette technologie datant du dix-neuvième siècle est dépassée sur tous les plans économiques, sûreté, fiabilité, qualité et rendement. Pleine de défections, elle provoque de nombreuses pannes répétées, ce qui conduit à des performances très très médiocres.

Du point de vue méthodologie, la collecte des données techniques nécessaires pour notre travail sur le terrain, nous a permis de toucher plusieurs domaines: l'automatique, l'électrotechnique, l'électronique la mécanique et l'informatique industrielle. Nous avons pris connaissance de la chaîne de production puis approfondi sur le fonctionnement de la machine et de ses éléments constitutifs ainsi que leurs rôles. Nous avons ensuite étudié les automates programmables industriels de la gamme SIEMENS, leurs caractéristiques et leurs domaines d'utilisation.

Pour la programmation, on a commencé par la modélisation en Grafset niveau 1 puis nous l'avons transcrit en langage Step 7 en utilisant S7 graph. Nous l'avons simulé avec le PLCSIM, ce qui nous a permis de toucher à tous les logiciels de base qui sont utilisés dans l'automatisation.

Ce travail nous a permis aussi de développer un sens de communication avec les différents ingénieurs et techniciens de l'industrie et d'acquérir le sens de travail en équipe. Avec ce modeste travail, les techniciens et les ingénieurs peuvent s'en servir pour bien comprendre le fonctionnement de la machine et d'optimiser le temps des opérations de maintenance. Nous espérons que l'entreprise NAFTAL prendra notre travail comme référence.

Afin d'améliorer le fonctionnement de l'installation existante, nous proposons de changer les chariots par des bras mécaniques automatisés qui vont se charger directement du chargement et du déchargement des palettes.

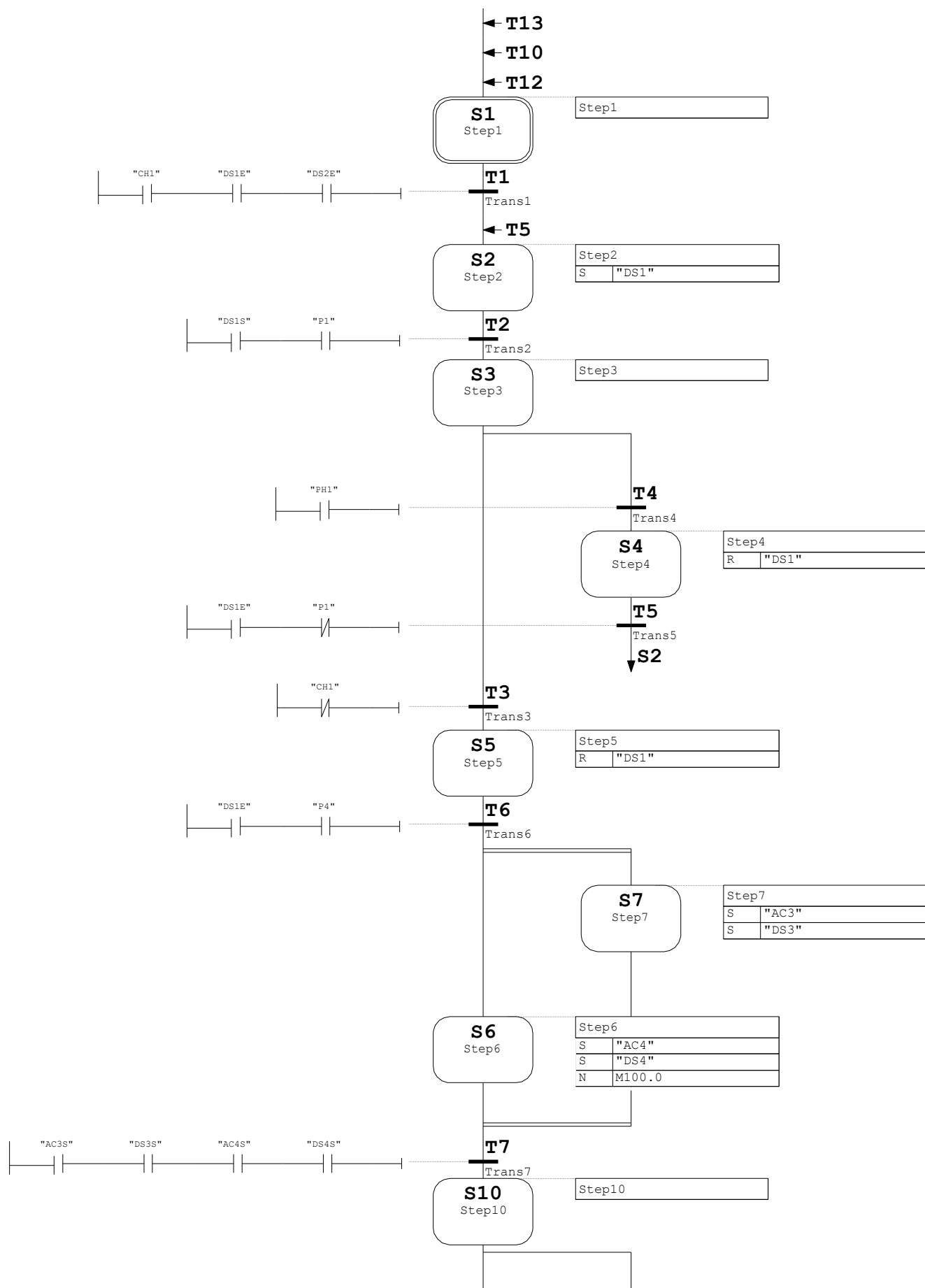
- [1] Documentation interne NAFTAL.
- [2] Documentation sur le palettiseur PAM.
- [3] Les automates programmables,  
[http://www.groupeisf.net/Automatismes/Automatesprogrammables/API\\_ATTOL/Bases\\_automatismes/an9\\_seq1\\_Place\\_et\\_role\\_de\\_l\\_API.ppt](http://www.groupeisf.net/Automatismes/Automatesprogrammables/API_ATTOL/Bases_automatismes/an9_seq1_Place_et_role_de_l_API.ppt).
- [4] L'Automate Programmable Industriel, <http://www.fichier-pdf.fr/2011/03/17/api/api.pdf>.
- [5] Philippe HOARAU, L'Automate Programmable Industriel, TS MAI,  
<http://bannaladi.fr/cours/Traitement/API.pdf>
- [6] Houssam AZAIEZ, Introduction aux A.P.I, Septembre 2007.
- [7] SCHNEIDER ELECTRIC, Chapitre 10: Traitement de données et logiciel,  
[http://www2.schneider-electric.com/documents/automation-control/pdf/Guide\\_des\\_Solutions\\_Automatisme\\_2008-FR\\_web.pdf](http://www2.schneider-electric.com/documents/automation-control/pdf/Guide_des_Solutions_Automatisme_2008-FR_web.pdf)
- [8] Philippe LE BRUN, Automates programmables industriels, Lycée Louis ARMAND, Décembre 1999.
- [9] L. BERGOUGNOUX, Automates Programmables Industriels, POLYTECH' Marseille Département de Mécanique Energétique, 2004–2005.
- [10] Philippe HOARAU, Langages orientés applications,  
[http://bannaladi.fr/TP/langages\\_orientes\\_applications.pdf](http://bannaladi.fr/TP/langages_orientes_applications.pdf).
- [11] Alain GONZAGA, Les automates programmables industriels,  
[http://www.geea.org/IMG/pdf/LES\\_AUTOMATES\\_PROGRAMMABLES\\_INDUSTRIELS\\_pour\\_GEEA.pdf](http://www.geea.org/IMG/pdf/LES_AUTOMATES_PROGRAMMABLES_INDUSTRIELS_pour_GEEA.pdf).
- [12] Philippe LE BRUN, Programmation des automates, Lycée Louis ARMAND, Décembre 1999.
- [13] SIEMENS, WinCC flexible 2008, SIMATIC HMI  
[https://w3.siemens.com/automation/jp/Documents/7\\_SIMATIC\\_HMI\\_renewal\\_manual\\_en.pdf](https://w3.siemens.com/automation/jp/Documents/7_SIMATIC_HMI_renewal_manual_en.pdf).
- [14] C.T. JONES, Step7 in 7 steps, A practical Guide to Implementing S7- 300/S7-400 Programmable Controllers, first Edition, 2006.
- [15] SIEMENS, Programmation avec STEP 7, SIMATIC, 2008.
- [16] SIEMENS, STEP 7, Getting started, SIMATIC, 2007.
- [17] SIEMENS, S7PLCSIM, Testez vos programmes, SIMATIC, 2008.



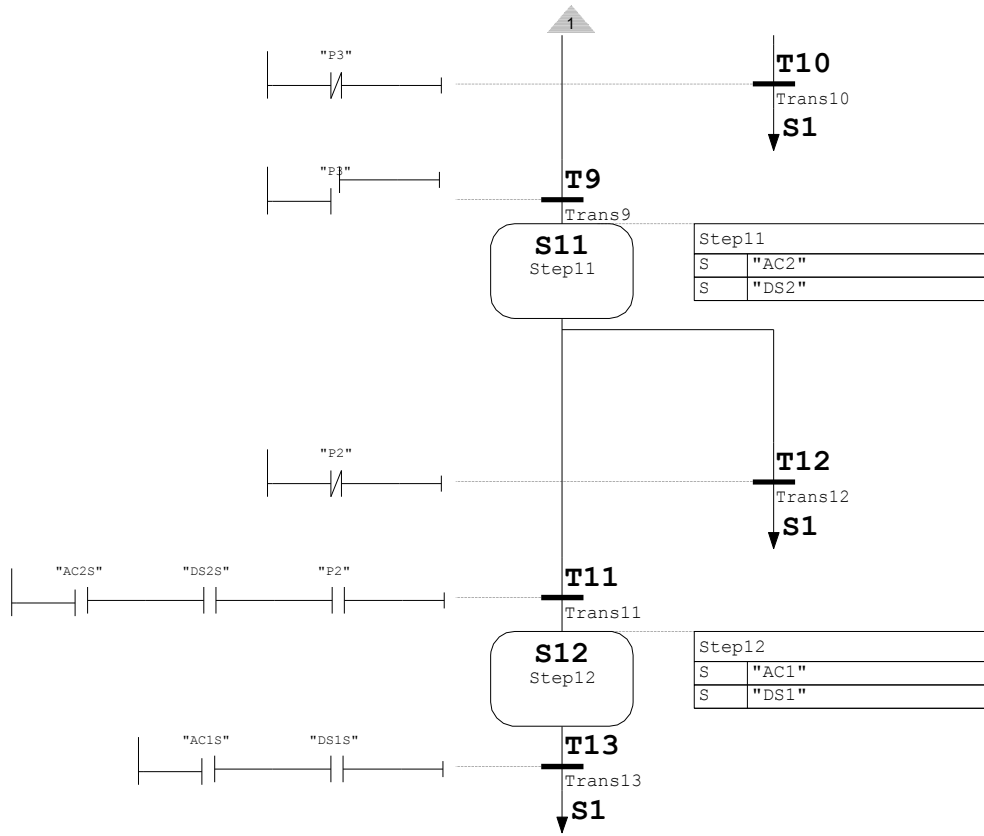
**Annexes**

# Annexe III.1

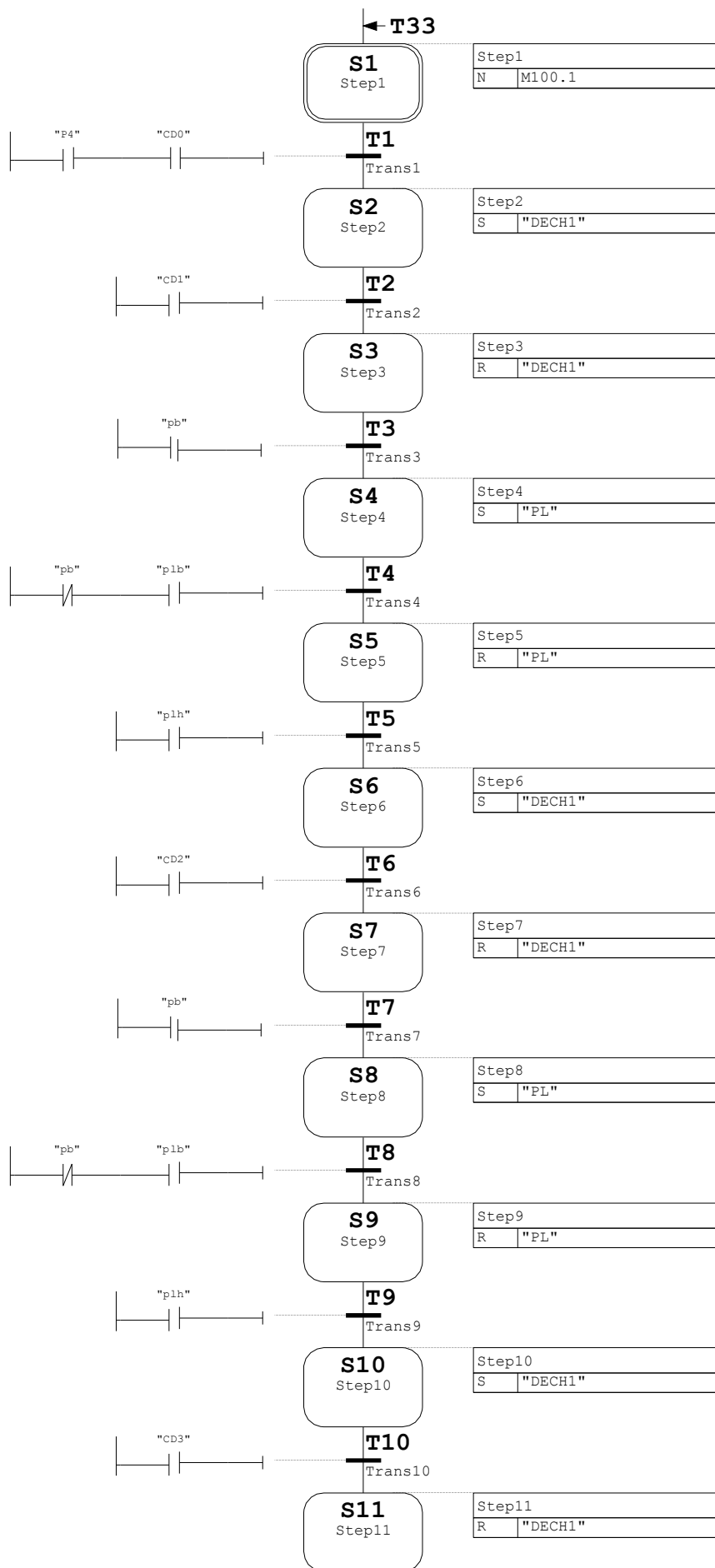
Arrivé de palette

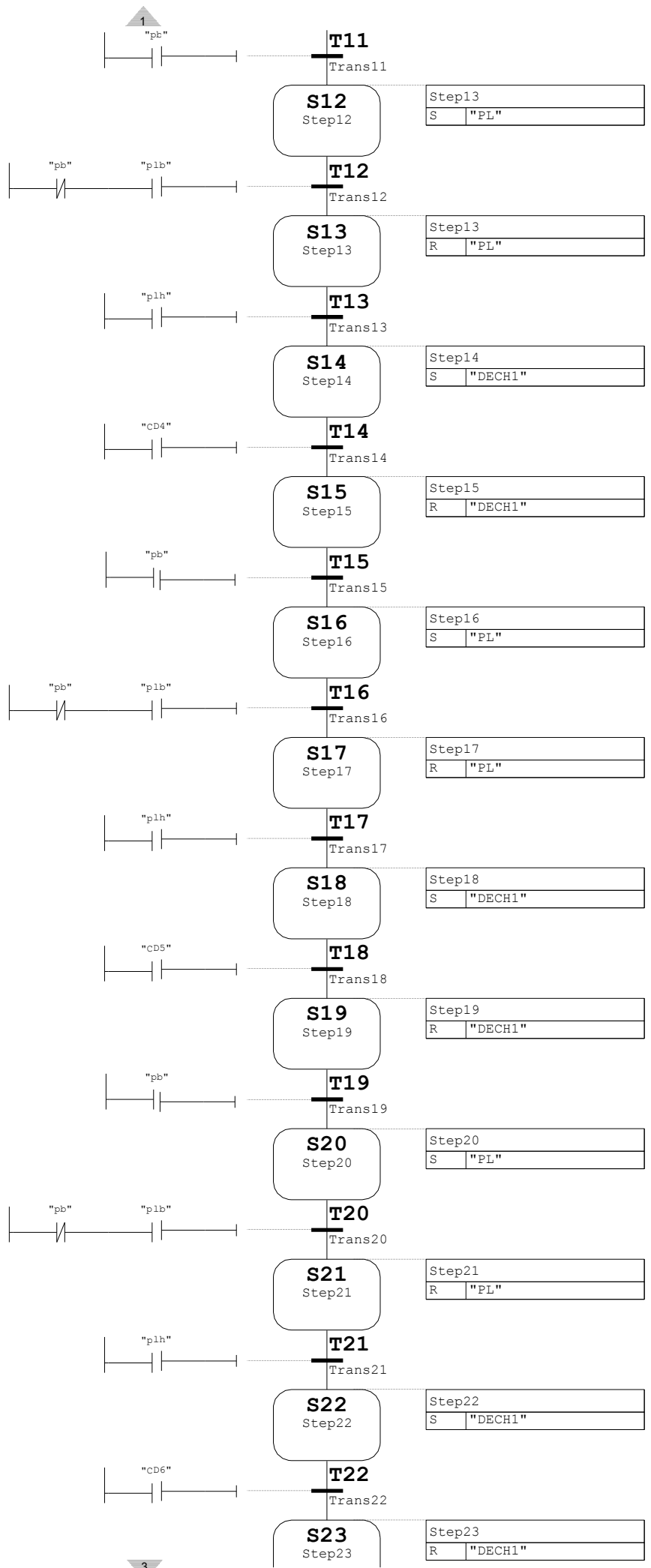


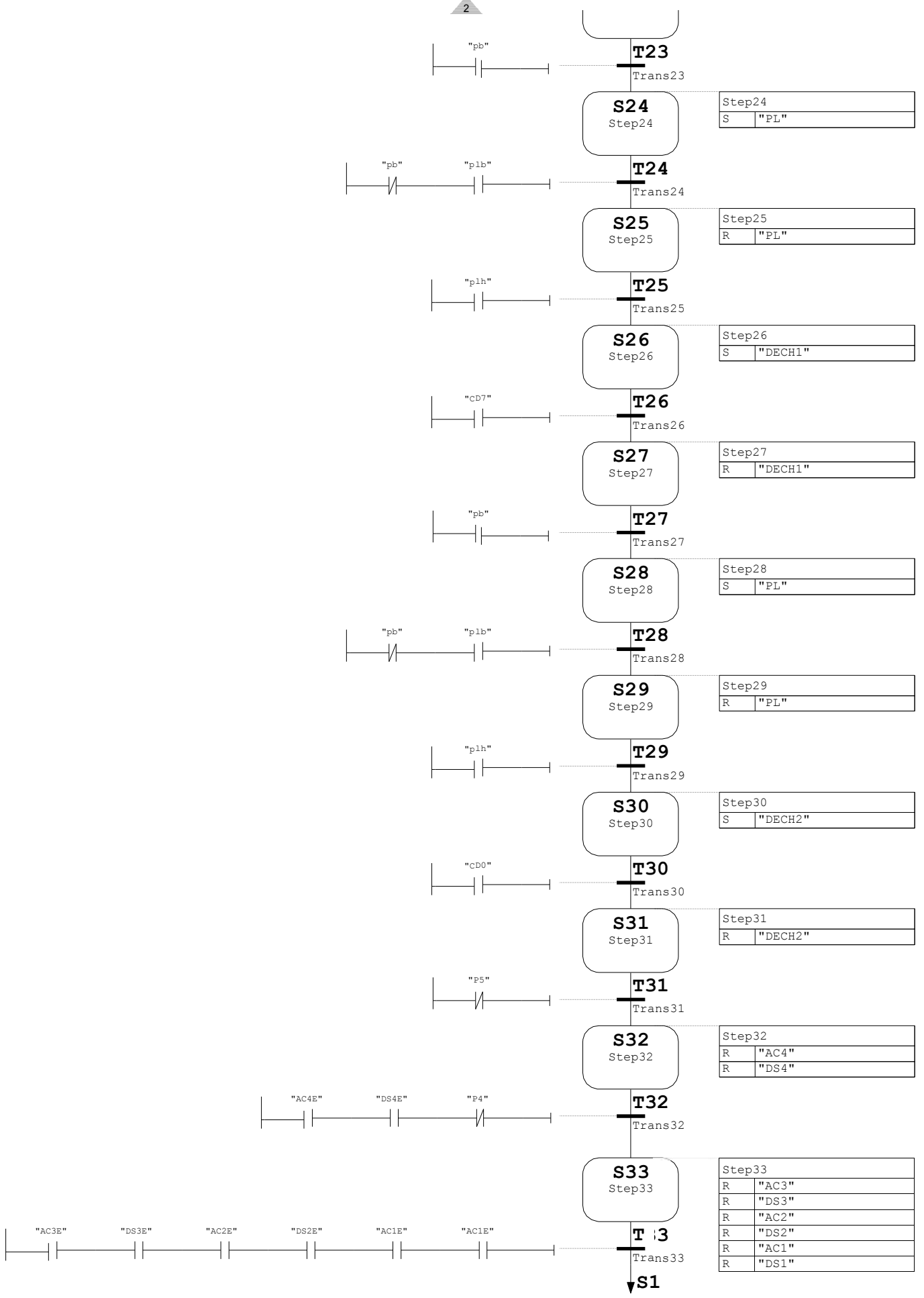
# Annexe III.1



DECHARGEMENT

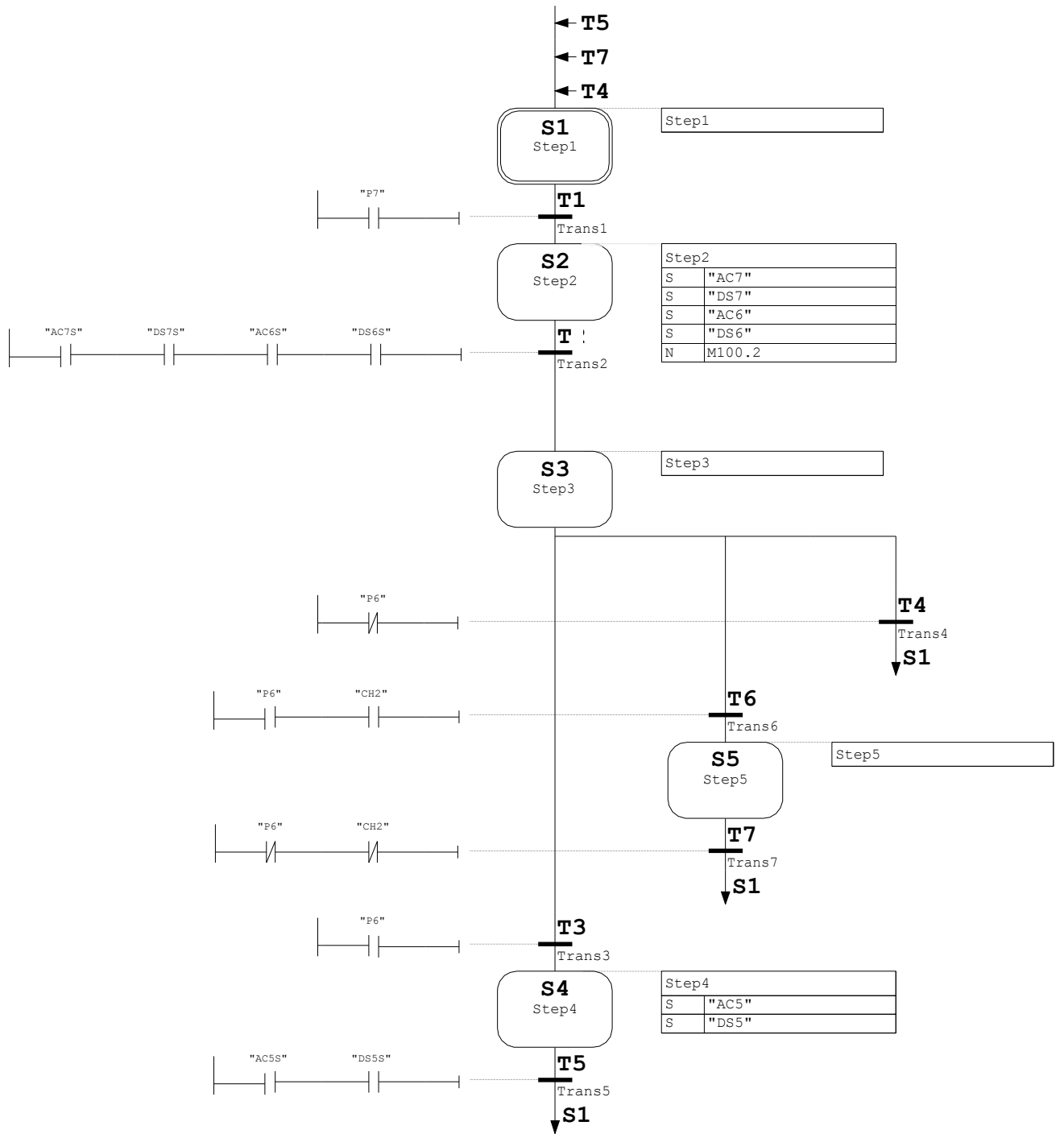




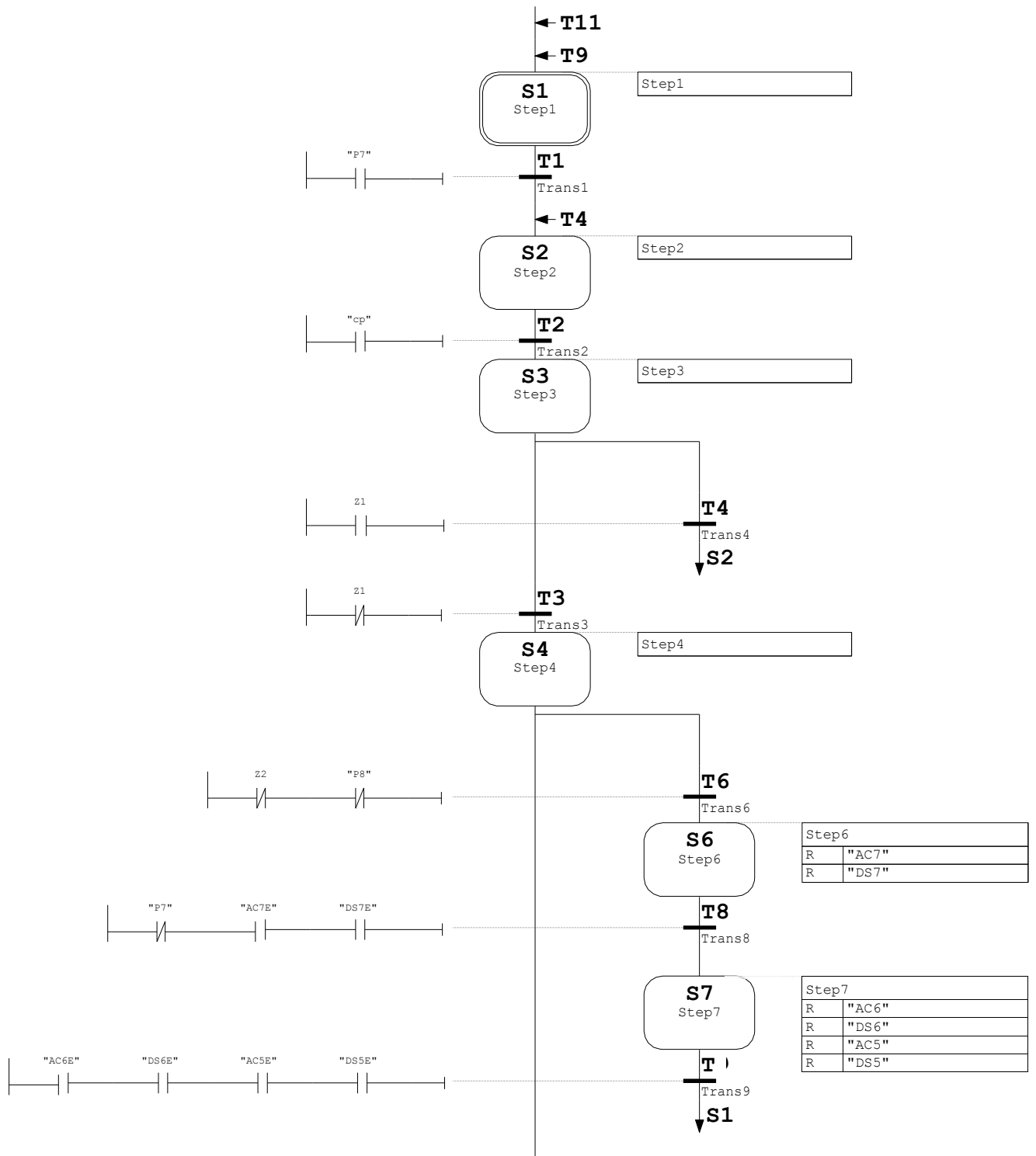


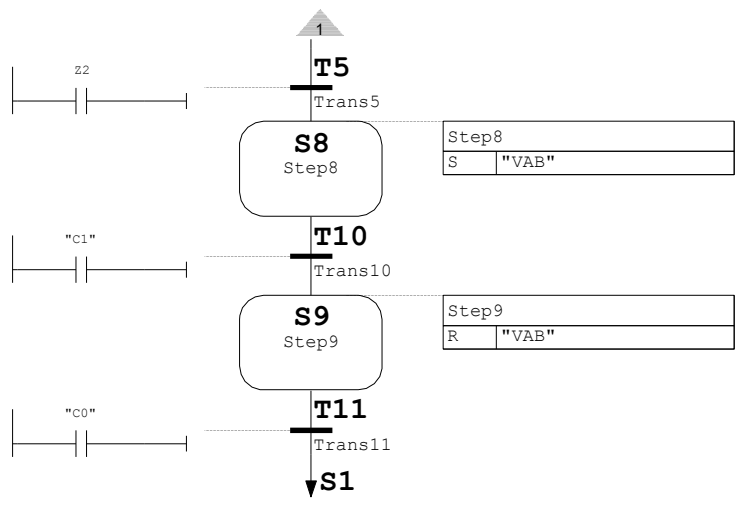
# Annexe III.1

Commentaire de bloc



Commentaire de bloc





Commentaire de bloc

