République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université Dr. Yahia FARES de Médéa



Faculté des Sciences et de la Technologie

Département de Génie de la Matière

Mémoire de fin d'étude

Présenté pour l'obtention du Diplôme

De master en **Génie Mécanique**

Option : ***Mécatronique***

Thème:

Réalisation et commande d'un ascenseur pédagogique par API (Step-7 200)

Proposé et dirigé par :

Dr. BOUHEDDA Mounir

Réalisés et présentés par :

SEFRANI Anis

AYED Abelkader

Promotion 2011/2012

INTRUDUCTION GENERALE

Dans une société en plaine mutation ou les besoins d'un citoyen augmentent quotidiennement et ou il devient nécessaire de développer les moyens de transport tant horizontale que vertical en raison notamment de l'extension de l'habitat en hauteur, et ou l'automatisation devient un moyen indispensable au passage de la commande manuelle a la commande automatique

Nous avons opté pour la réalisation d'un prototype d'ascenseur, Ce choix nous permettra de traduire en pratique la somme de connaissances théorique accumulée durant notre cycle de formation tant du point de vue mécanique, électrique que de l'automatisation.

Pour ce faire, nous avons eu à utiliser les outils et moyens offerts par le laboratoire de l'université et d'autres moyens extérieurs, et pour l'automate siemens (STEP-7 200 CPU 216) pour l'automatisation de notre prototype.

Organisation de la mémoire :

Le chapitre I est consacré à l'étude théorique des ascenseurs.

Le chapitre II concerne l'étude sur les automates programmables industriels et la procédure de programmation.

Le chapitre III est scindé en trois parties, il s'agit de :

La partie mécanique

La partie électrique

Et la partie automatisation

Enfin, nous concluons notre mémoire par une conclusion générale et les perspectives du projet.

I.1. Introduction :[1]

Un ascenseur est un dispositif mobile ou semi-mobile assurant le déplacement des personnes (et des objets) en hauteur sur des niveaux définis d'une construction.

Les dimensions, la construction et le contrôle en temps réel pendant l'usage des ascenseurs permettent l'accès sécurisé des personnes.

L'ascenseur est installé la plupart du temps dans une cage d'ascenseur, une trémie verticale fermée (sorte de puits de la bâtisse) en général à l'intérieur de l'édifice.

Il peut aussi être construit sans cage fermée et être accessible en rebord de la dalle de chaque niveau pour des édifices ayant un noyau évidé de circulation centrale, sans séparation dans la vision des modes de déplacement verticaux, ou se situer en périphérie et même en façade sur l'extérieur avec une cage discernable. Le mouvement des cabines dans des cages en périphérie peut être vu parfois dans des cages transparentes.

Les ascenseurs visibles sont alors des éléments architecturaux identiques à ce que sont les escaliers architecturaux à volées visibles.

Des ascenseurs spéciaux pour personnes à mobilité réduite (en fauteuil roulant), dispositif à course verticale sans véritable cage permettent de parcourir les couloirs comportant des volées de marches sans rampes d'accès, voire d'accéder à un inter niveau ou un autre niveau desservi par un escalier sans qu'il y ait d'autres ascenseurs disponibles. Un dispositif qui suit une rampe garde-corps d'escalier est un monte-escalier.

Un ascenseur de charge est un appareil principalement destiné au transport de charges lourdes et de personnes en milieu commercial, culturel ou industriel. La cabine est fabriquée pour convenir à l'usage, l'aspect est cohérent avec le public qui l'utilise (appareil différent du monte-charge).

Le terme désigne aussi parfois la cabine de l'ascenseur, qui n'est qu'un élément du dispositif.

Il y aurait selon la Fédération des indépendants experts et bureaux de contrôle ascenseurs (Fiebca) environ 450 000 ascenseurs en 2010, dont 50 % avaient plus de 25 ans d'âge et 25 % plus de 40 ans.

I.2. Définition de l'ascenseur **:** [2]

Les ascenseurs sont des appareils de levage verticaux destinés au transport des personnes ou des marchandises on distingue techniquement trois types d'appareils :

- Les ascenseurs destinés à l'usage unique des personnes.
- Les monte-charges destinés aux personnes et marchandises,
- Les monte-charge industriels destinés aux marchandises uniquement.

I .3. Les types des ascenseurs **:** [3]

On distingue essentiellement deux types de familles d'ascenseur :

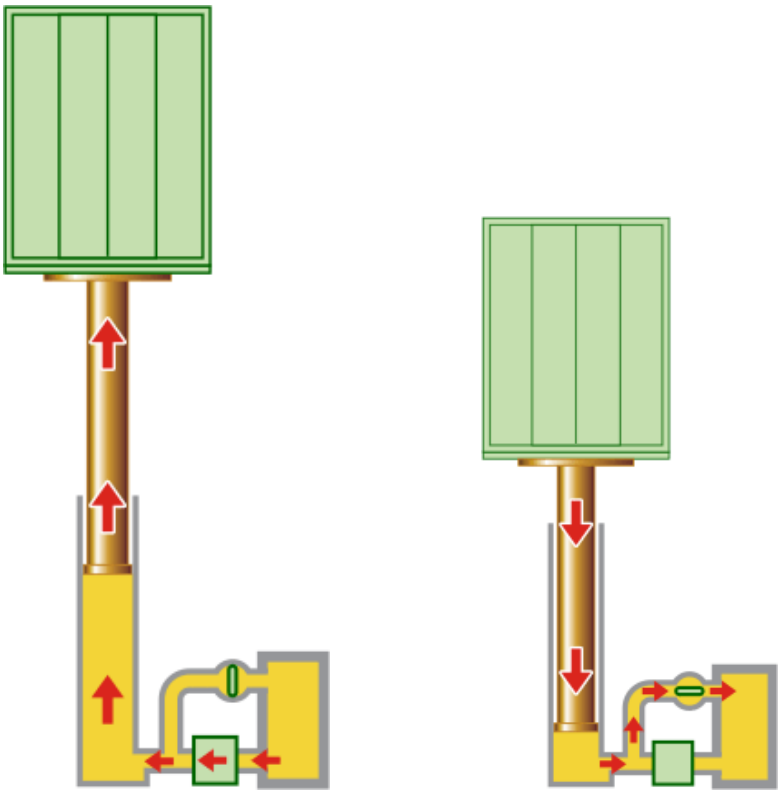
- les ascenseurs à traction à câble,
- les ascenseurs hydrauliques.

En règle générale, ces deux types utilisent l'énergie électrique pour déplacer les cabines verticalement (moteur électrique continu ou alternatif).

I .3.1. Les ascenseurs hydrauliques :

I .3.1.1. Principe de fonctionnement :

Comme toute machine hydraulique la pompe met sous pression l'huile qui pousse le piston hors du cylindre vers le haut. Lorsque la commande de descente est programmée, le bypass (vanne) de la pompe permet de laisser sortir l'huile du cylindre vers le réservoir. (**Figure I.1**)



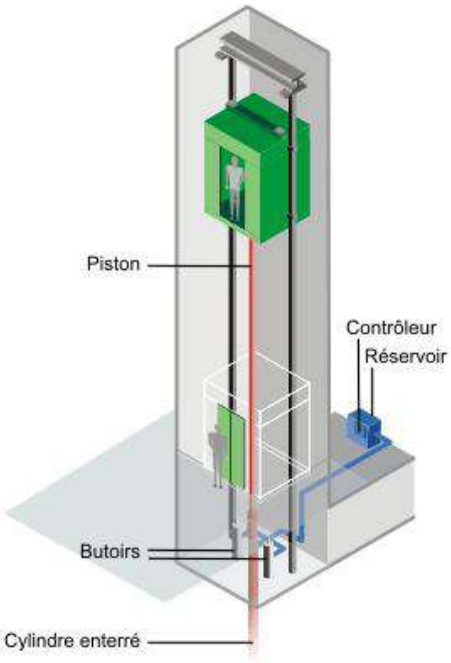
(**Figure I.1**) *Principe de fonctionnement*

I .3.1.2 Description :

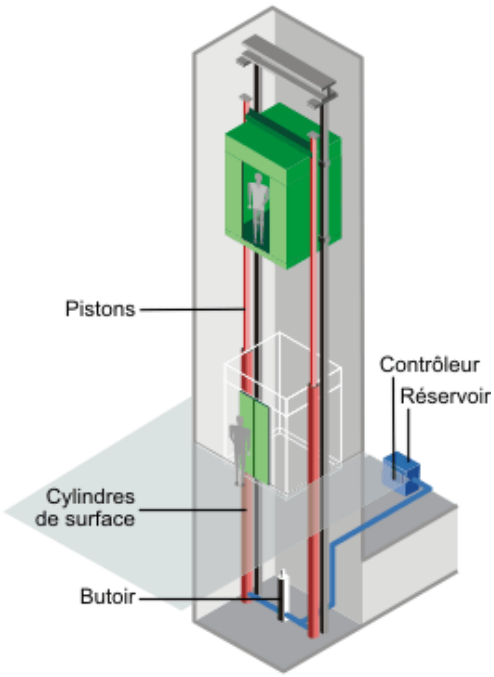
Les ascenseurs hydrauliques sont utilisés en général pour satisfaire des déplacements relativement courts de l'ordre de 15 à 18 m maximums.

Plusieurs modèles existent sur le marché. On citera les ascenseurs hydrauliques :

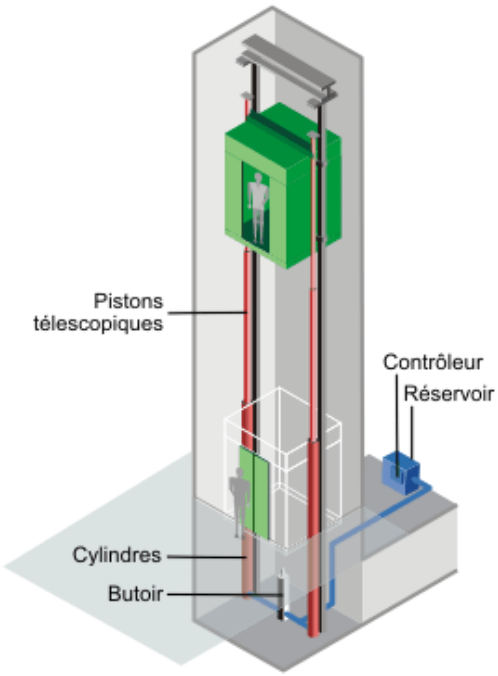
- A cylindre de surface. (**Figure I.3)**
- A cylindre enterré. (**Figure I.2)**
- Télescopiques à cylindre de surface. (**Figure I.4)**



(**Figure I.2**) *À cylindre enterré*



(**Figure I.3**) *À cylindre de surface*



(**Figure I.4**) *À cylindre de surface télescopique*

Les ascenseurs hydrauliques se composent principalement :

- d'une cabine.
- de guides.
- d'un ensemble pistons-cylindres hydrauliques placé sous la cabine de l'ascenseur.
- d'un réservoir d'huile.
- d'un moteur électrique accouplé à une pompe hydraulique.
- d'un contrôleur.

Les différents modèles permettent de tenir compte de critères :

- de place.
- de hauteur d'immeuble à desservir.
- de stabilité de sol et de sous-sol.

I .3.1.3 Avantages et inconvénients :

Ci-dessous, on trouvera les principaux avantages et inconvénients des ascenseurs hydrauliques :

I .3.1.3.1 Avantages :

- Précision au niveau du déplacement (mise à niveau).
- réglage facile de la vitesse de déplacement.
- ne nécessite pas de cabanon de machinerie.
- implantation facile dans un immeuble existant.

I .3.1.3.2 inconvénients :

- risque de pollution des sous-sols.
- course verticale limitée à une hauteur entre 15 et 18 m.
- consommation énergétique importante.
- nécessité de renforcer la dalle de sol.

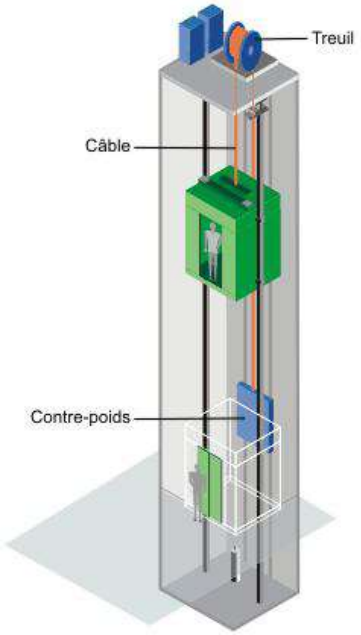
I .3.2. Les ascenseurs à traction à câbles :

I .3.2.1. Description :

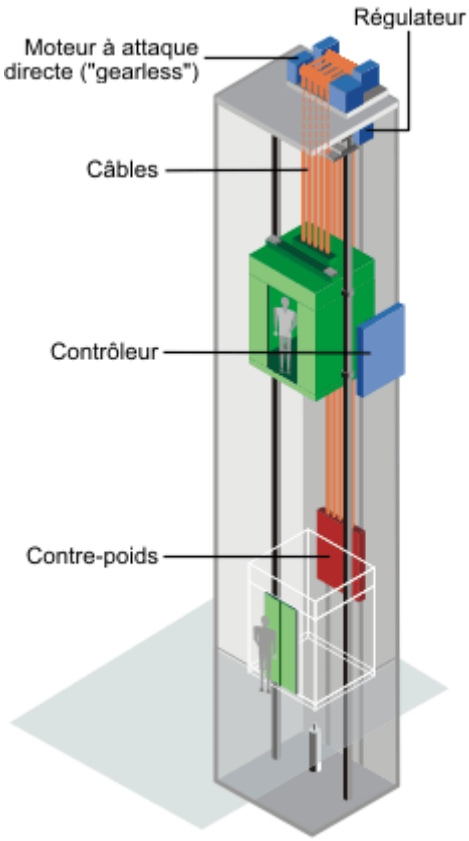
Les ascenseurs à traction à câbles sont les types d'ascenseurs que l'on rencontre le plus, notamment dans les bâtiments tertiaires.

Ils se différencient entre eux selon le type de motorisation :

- à moteur-treuil à vis sans fin. (**Figure I.5**)
- à moteur-treuil planétaire. (**Figure I.6**)
- à moteur à attaque directe (couramment appelé "Gearless" ou sans treuil).



(**Figure I.5**) *Ascenseur à moteur-treuil.*



(**Figure I.6**) *Ascenseur à moteur à attaque directe.*

Quel que soit le type, les ascenseurs à traction à câbles comprennent généralement :

- une cabine.

- un contrepoids.
- des câbles reliant la cabine au contrepoids.
- des guides.
- un système de traction au-dessus de la cage de l'ascenseur.

I.3.2.2. Avantages et inconvénients :

Ci-dessous, on trouvera les principaux avantages et inconvénients des ascenseurs hydrauliques :

I.3.2.2.1. Avantages:

- course verticale pas vraiment limitée.
- suivant le type de motorisation précision au niveau de la vitesse et du déplacement.
- rapidité de déplacement.
- efficacité énergétique importante.
- pas de souci de pollution.

I.3.2.2.2. inconvénients :

- en version standard, nécessite un cabanon technique en toiture.
- exigence très importante sur l'entretien.

I.4.Application : [2]

Le terme d'ascenseur est réservé aux appareils qui permettent de transporter verticalement des personnes entre différents niveaux. Lorsqu'il s'agit de déplacer des produits, des matériaux, des objets, etc.

On emploie le terme général de monte-charge. Suivant la nature de la charge transportée on rencontre :

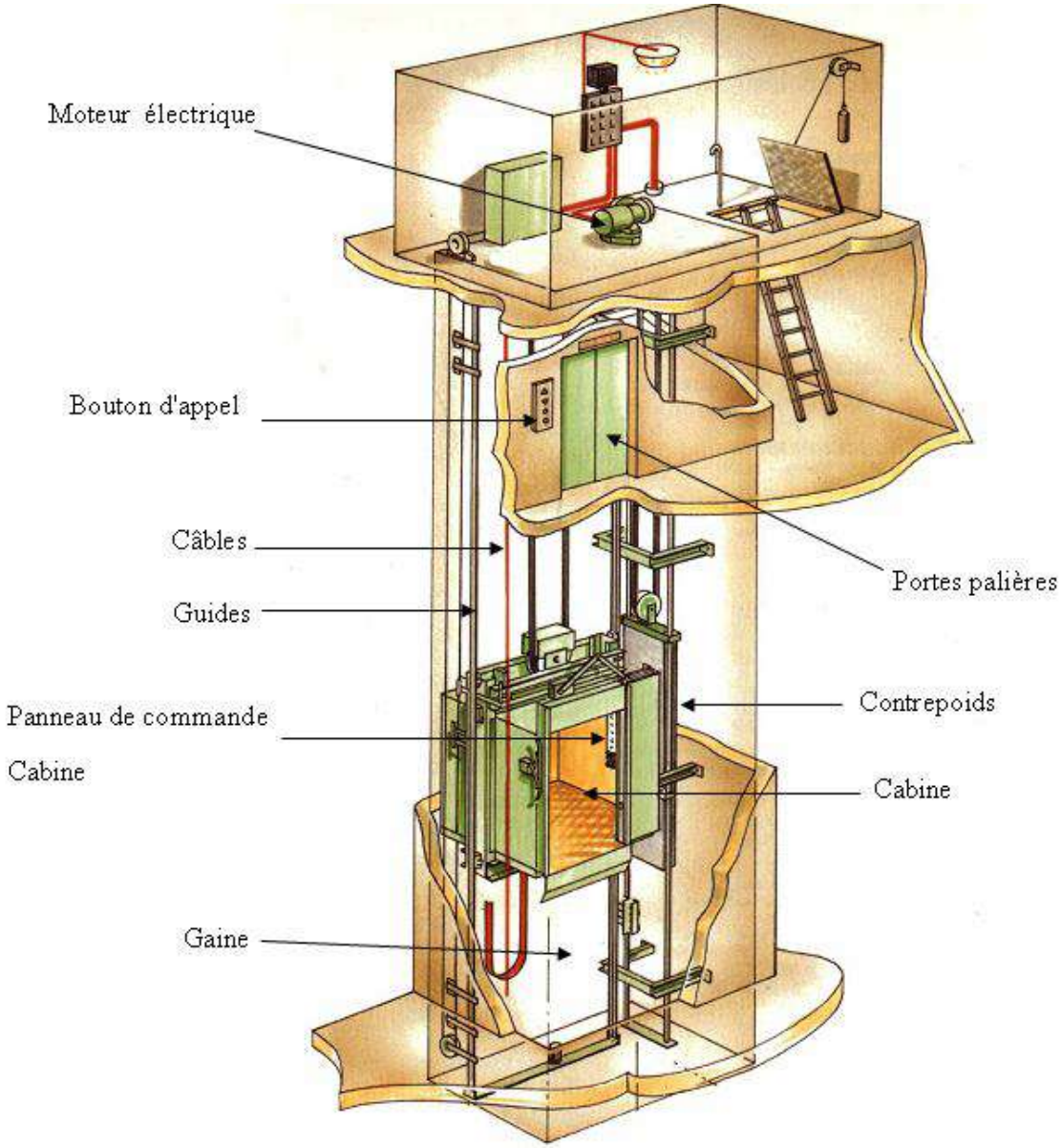
- les monte-voitures (garages).
- les monte-chariots (usines).
- les monte-décors (théâtres).
- les monte-scories (aciéries).
- les monte-fûts (magasins), etc.

Certains de ces appareils peuvent également transporter en même temps des personnes :

- les monte-charges accompagnés.
- les monte-malades.

I.5. Les composants principaux d'un ascenseur électrique : [2]

Un ascenseur électrique comporte toujours des éléments comme indique la (**figure I.7**) suivante :



(Figure I.7) *Structure d'un ascenseur électrique*

I.5.1 La gaine :

Gaine verticale dans laquelle se déplacent l'ascenseur et son contrepoids. Celle-ci est équipée de guides en acier destinés à guider la suspension de la cabine et le contrepoids profilée en acier, généralement en forme de T , destinés à guider la

cabine et le contrepoids dans la gaine, ou la gaine est un volume dans lequel se déplace la cabine et le contrepoids, s'il en existe un. Ce volume est matériellement délimité par le fond de la cuvette, les parois et le plafond

I.5.2. La cabine :

Organe de l'ascenseur ou de la monte charge, destiné à recevoir les personnes et/ou les charges à transporter. La cabine est composée :

- D'une porte a fermeture généralement automatique destinée à confiner l'utilisateur dans la cabine pendant le déplacement de celle-ci, lui interdisent tout contact avec les parties extérieures à la cabine.
- D'un dispositif installé sur une porte palière et destiné à interdire l'ouverture de celle- ci lorsque l'ascenseur n'est pas présent a l'étage, cet élément est inséré dans un cadre appelé suspension cabine.

I.5.3. Le contrepoids :

Elément destiné à contre balancer le poids de la suspension cabine augmenté de la moitié de la charge utile. Celui-ci est constitué d'une suspension métallique contenant des gueuzes en fonte destinées à l'alourdir. Lorsque la cabine d'ascenseur monte, le contrepoids descend. Les masses peuvent varier entre 25 a 120 kg, elles sont empilées sur les traverses inférieures de l'arcade et entre les ailes des montants.

I.5.4. Les câble à traction :

Les câbles assurent la liaison entre cabine et contrepoids pour les appareils actionnés par un treuil à adhérence, entre cabine et tambour du treuil pour les appareils non équilibrés, les câbles modernes son fabriqués à partir de fils en acier ou électronique qui subissent trois opérations distinctes : la trempe, la préparation des fils et le tréfilage, les câbles utilisés sont constitués de 6 ou 8 torons qui sont câblés selon deux dispositions :

- Soit le câblage dit croisé dans lequel si le toronnage est a gauche le câblage est effectué a droite ; la torsion des fils est ainsi inversée dans le câble par rapport au toron.

- Soit le câblage dit Lang dans lequel toronnage et câblage sont dans le même sens.

I.5.5. Les guide :

La cabine et le contrepoids circulent le long de guides en acier dont l'importance est telle pour le confort des usagers que les spécialistes n'hésitent pas à affirmer qu'il réside pour sa plus grande part dans la façon dont ils sont montés. Les guides de cabine étaient constitués par des barres d'acier de section ronde de 60 mm environ de diamètre. Ces barres comportaient deux rainures longitudinales qui permettaient leur propre fixation et également celle d'organes d'arrêt ou de sécurité disposés en gaine.

I.5.6. Le moteur :

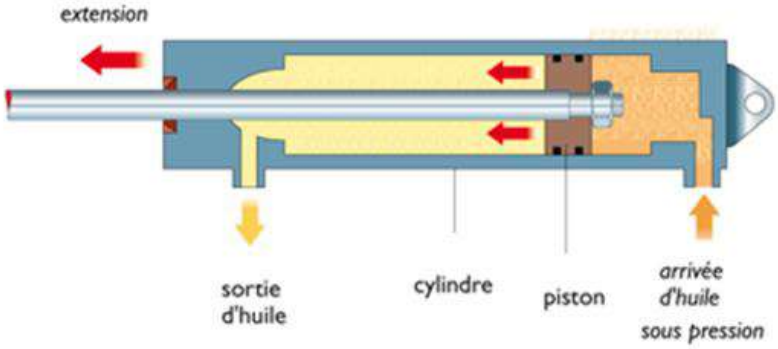
Le moteur le plus couramment employé est actuellement le moteur asynchrone à double cage, ce choix à des avantages de faible cout (y compris pour de fortes puissances), très robuste (pas de pièce de contact) et un bon rendement.

I.5.7. Les portes palières :

Portes obturant à chaque niveau (donc fixes) les ouvertures dans la gaine servant comme accès à la cabine d'ascenseur. Afin d'éviter les chutes de personne dans la gaine, leur ouverture n'est possible qu'en présence de la cabine.

Pour faire rentrer ou sortir (ouvertures ou fermeture de porte) un vérin double effet, on doit inverser l'alimentation en pression deux chambre (**Figure I.8**).

Moins couteux qu'un vérin simple effet, il est utilisé dans l'application pneumatique ne nécessitant pas de retour en position initiale en cas de coupure de pression.



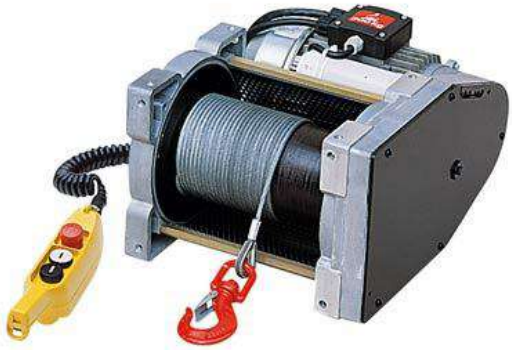
(**Figure I.8**) *vérin hydraulique à double effet*

I.5.8 Le treuil :

Le treuil entraine la cabine et le contrepoids dans leur mouvement comme l'indique la **figure I.9**. Nous ne décrirons ici que les treuils à adhérence. Le treuil à tambour sur lequel les câbles s'enroulent en spirale n'est plus utilisé que dans les anciens bâtiments, il n'est pas possible de disposer un contrepoids.

Le treuil est composé des éléments suivants :

- Un moteur électrique.
- Un réducteur roue et vis sans fin.
- Une poulie de traction.
- Un volant d'inertie.
- Un embrayage frein a tambour.



(**Figure I.9**) *le treuil*

I.5.9 Contact de surcharge :

Lorsque la surface de la cabine d'un ascenseur excède la valeur imposée, le contact de surcharge est nécessaire, car il peut éviter des incidents graves tels que la détérioration du moteur lorsque l'appareil est en montée ou à l'arrêt sur les amortisseurs du fond de cuvette pour la descente .ce contact est placé soit sous le plancher de la cabine ,soit sur le dispositif d'attache de l'arcade à la suspente , dans les deux cas , la surcharge provoque un déplacement excessif du plancher ou de l'attache et interdit le démarrage de l'appareil.

Il est indispensable de doubler l'action de ce contact par une signalisation optique et acoustique .un voyant (surcharge) sur la boite à boutons cabine et un ronfleur ou une sonnerie sont alors prévus .

I.5.10. Contact de présence de la cabine :

Le contact de présence de la cabine se fait a l'aide des capteurs de proximités ou « détecteurs de présences » qui sont des dispositifs autrefois mécaniques, et aujourd'hui de plus en plus caractérisés par l'absence de liaison mécanique entre le dispositif de mesure et l'objet ciblé (personne, animal, objet animé tel qu'un véhicule).

L'interaction entre le capteur et sa "cible" est alors réalisée par l'intermédiaire d'une caméra associée à un système d'analyse de l'image, ou plus souvent d'un champ (magnétique, électrique, électromagnétique).

Selon les capteurs, objets et situation, l'objet détecté doit être plus ou moins proche du capteur ou illuminé par une source rayonnante (éventuellement non-visible, par exemple dans l'infrarouge).

I.5.10.1. capteur de proximité :

Les capteurs de proximité (**Figure I.10**) sont utilisés soit en mode analogique, soit en mode binaire.

Dans le premier cas, l'amplitude du signal est une fonction de la position relative de l'objet ciblé.

Dans le second cas, le signal ne peut avoir que deux niveaux (haut et bas), selon que l'objet est présent à proximité ou non du capteur inductif



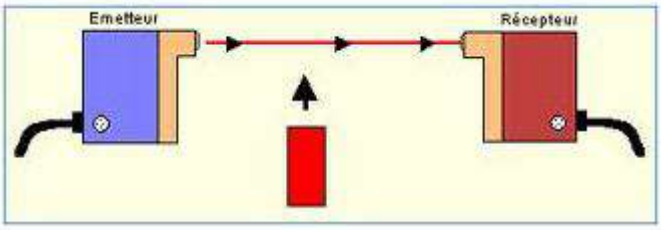
(**Figure I.10**) *Capteur de proximité*

I.5.10.2. Photocellules :

Elles sont destinées à la détection d'objets de toute nature. Le concepteur a le choix entre trois systèmes de base :

- Le système barrage. (**Figure I.11**)
- Le système reflex.
- Le système de proximité.

Dans les portes d'ascenseurs on utilise le système barrage (**Figure I-11**), pour les longues portes et objets réfléchissants. Ce système de détection contient un émetteur et récepteur séparés permet de détecter tout objet interrompant le faisceau lumineux (les personnes). Dans ce cas la porte ne ferme pas.



(**Figure I.11**) *Système barrage*

I.5.11. Les commande d'appel :

Les commandes d'appel (**figure I.12**) ont pour but de faire venir l'appareil aux étages d'où elles émanent respectivement .cette utilisation, a posteriori par rapport à la commande d'envoi, justifie la priorité que celle-ci doit avoir sur la commande d'appel.

Les appels peuvent eux-mêmes prendre quatre caractères :

- l'appel peut être commun .il intéresse alors un groupe d'appareils installés à proximité les uns des autres et constituant une batterie.
- l'appel est simple lorsqu'il provoque directement la mise en marche de la cabine si celle-ci est disponible.
- l'appel peut être enregistré, il provoque la venue de la cabine au niveau d'appel dés que les conditions d'arrêt à ce niveau sont satisfaisantes et sans qu'il ait été nécessaire de maintenir cette action d'appel, c'est-a-dire d'appuyer en permanence sur le bouton cette sorte d'appel peut être :
  - soit un appel d'ordre agissant sur une signalisation lumineuse ou sonore dans les appareils avec conducteur.
  - soit un appel direct, automatique, agissant sans intermédiaire dans les manœuvres collectives.

Enfin l’appel peut être spécialisé lorsqu’il est effectué sur des boutons différents suivant que l’usager désire monter ou descendre.



(**Figure I.12**) *bouton d’appel*

I.5.12. Les commande d'envoi

Leur rôle est d'envoyer la cabine du niveau où elle se trouve au niveau auquel la personne qui effectue l'envoi le désire. La commande d'envoi s'effectue par l'intermédiaire d'un bouton qui peut être placé soit en cabine dans la majorité des cas, soit pour certains monte –charge sur le palier.

Lorsque la commande d'envoi est faite en cabine, elle doit évidemment avoir priorité sur la commande extérieure. Pour les appareils du groupe 1(ascenseur et monte-charge accompagnés), l'usager qui est entré dans la cabine doit disposer, pour appuyer sur le bouton de son choix, d'un certain laps de temps après la fermeture des portes avant qu'une commande d'appel faite de l'extérieur puisse être exécutée

I.5.13. Les commande d'appel :

Elles ont pour but de faire venir l'appareil aux étages d'où elles émanent respectivement. Cette utilisation, à posteriori par rapport à la commande d'envoi, justifie la priorité que celle-ci doit avoir sur la commande d'appel.

Les appels peuvent eux-mêmes prendre quatre caractères :

- L'appel peut être commun. Il intéresse alors un groupe d'appareils installés à

proximité les uns des autres et constituant une batterie.

- L'appel est simple lorsqu'il provoque directement la mise en marche de la cabine si celle-ci est disponible.
- L'appel peut être enregistré. Il provoque la venue de la cabine au niveau d'appel dès que les conditions d'arrêt à ce niveau sont satisfaites et sans qu'il ait été nécessaire de maintenir cette action d'appel, c'est-à-dire d'appuyer en permanence sur le bouton. Cette sorte d'appel peut être :

1- soit un appel d'ordre agissant sur une signalisation lumineuse ou sonore dans les appareils avec conducteur.

2- soit un appel direct, automatique, agissant sans intermédiaire dans les manœuvres collectives.

Enfin l'appel peut être spécialisé lorsqu'il est effectué sur des boutons différents suivant que l'usager désire monter ou descendre.

I.6. Les systèmes de motorisation : [3]

I.6. 1.Généralités :

On peut ranger les types d'ascenseurs électriques dans deux catégories principales :

- Les ascenseurs à traction à câbles.
- Les ascenseurs hydrauliques.

Les ascenseurs à tractions à câbles sont de loin ceux le plus répandu au niveau du parc machine. C'est une des raisons pour laquelle on détaillera plus la motorisation des ascenseurs à traction à câbles. De plus, énergétiquement parlant, la motorisation des ascenseurs hydrauliques a des rendements assez faibles (de l'ordre de 20 %) et, de par l'absence entre autres de contre-poids, les moteurs hydrauliques consomment plus et ont des appels de puissance au démarrage beaucoup plus importants.

Le (**tableau I.1**) ci-dessous montre un comparatif des performances énergétiques de différents types de motorisation :

(**Tableau I.1**) *Performances énergétiques de différents types de motorisation*

Type de motorisation	rendement
Moteur hydraulique	De l'ordre de 20%

Moteur-treuil a vis sans fin	Ancienne génération	De l'ordre de 45%
	Nouvelle génération	60 à 65%
Moteur sans treuil (gearless)		De l'ordre 78%
Moteur-treuil planétaire		De l'ordre 97%

Comme nous le verrons ci-dessous, l'efficacité énergétique des systèmes de motorisation des ascenseurs dépend surtout du type de moteur d'entraînement accouplé :

- au treuil pour les ascenseurs à traction,
- à la pompe pour les ascenseurs hydrauliques.

Un réducteur planétaire, par exemple, peut être accouplé à :

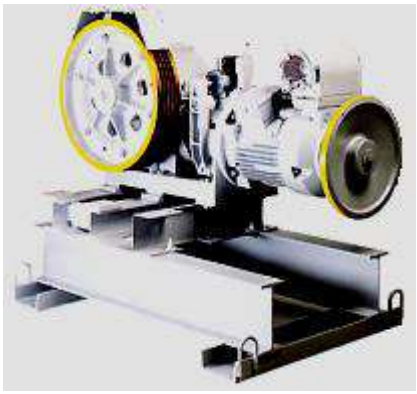
- un moteur à courant continu à excitation indépendante ou shunt,
- un groupe Ward Léonard,
- un moteur à courant alternatif asynchrone à démarrage :
  - étoile-triangle,
  - à deux vitesses,
  - commandé par un variateur de fréquence,
- un moteur à courant alternatif synchrone à démarrage par variateur de fréquence;

Suivant la combinaison du moteur et du réducteur planétaire, les consommations peuvent être sensiblement différente. Dans un souci de trouver la meilleure adéquation, il sera nécessaire de prendre en compte :

- le rendement mécanique de l'ensemble,
- le rendement électrique du circuit puissance (alimentation et moteur).

I.6.2.Les moteurs-treuils ou moteur à traction :

I.6.2.1.Les moteurs-treuils à vis sans fin à une ou deux vitesses :



*(***Figure I.13***) Moteur-treuil à vis sans fin.*

A l'heure actuelle, les moteurs-treuils avec vis sans fin (**figure I.13)** sont abandonnés au profit des moteurs à attaque directe (sans réducteur ou "gearless).

Dans ce type de motorisation, la vis sans fin entraîne beaucoup de pertes mécaniques et, par conséquent, des consommations électriques plus importantes.

I.6.2.1.1. Avantages:

- couple élevé.
- grande plage de variation de vitesse.
- précision dans les déplacements et sur la régulation de vitesse.

I.6.2.1.2. inconvénients :

- entretien important.
- efficacité énergétique faible.
- consommation électrique non négligeable.

I.6.2.2.Les moteurs-treuils planétaires :



(**Figure I.14**) *Moteur-treuil planétaire.*

Les appareils à treuil planétaire **(Figure I.14)** utilisent le système de réduction de vitesse par engrenages planétaires. Accouplés à un moteur électrique, ils permettent d'avoir un rapport de réduction appréciable pour obtenir une plage de vitesse compatible avec le confort et l'efficacité de déplacement souhaitée.

Ce système a un rendement mécanique de l'ordre de 97 à 98 % permettant, pour autant que les moteurs d'entraînement soient performants, d'obtenir des rendements énergétiques globaux intéressants au niveau du moteur-treuil (de l'ordre de 80 %).

Les réducteurs planétaires peuvent être accouplés à des moteurs électriques :

- à courant continu (grande plage de variation de vitesse),
- à courant alternatif asynchrone à deux vitesses,
- à courant alternatif asynchrone commandé par un variateur de fréquence.

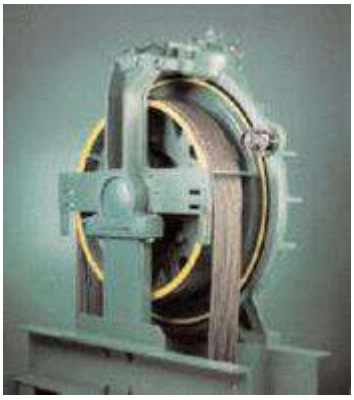
I.6.2.2.1 Avantages:

- Couple important.
- en fonction du type de moteur accouplé :
  - grande plage de variation de vitesse (courant continu ou alternatif à variation de fréquence).
  - rendement énergétique important entraînant une diminution des coûts à la conception (puissance installée plus faible) et des consommations moindres à l'exploitation.
- précision dans les déplacements et sur la régulation de vitesse.

I.6.2.2.2. inconvénients :

- Entretien nécessitant une main d'œuvre qualifiée.

I.6.2.3.Les moteurs à attaque directe ("Gearless" ou sans treuil) :



(**Figure I.15**) *Moteur gearless classique.*

Les moteurs à attaque directe (**Figure I.15**) sans réducteur ont fait leur apparition avec la venue des variateurs de fréquence. Les installations deviennent tellement compactes qu'ils sont possible à l'heure actuelle de se passer de local des machines sur le toit des immeubles.

Ce système est énergétiquement performant principalement de part la présence d'un variateur de fréquence qui optimise la consommation énergétique; la réduction des pertes mécaniques vu l'absence de réducteur contribuant aussi à l'optimisation de l'efficacité énergétique.

Certains constructeurs annoncent des rendements énergétiques de l'ordre de 80 %.

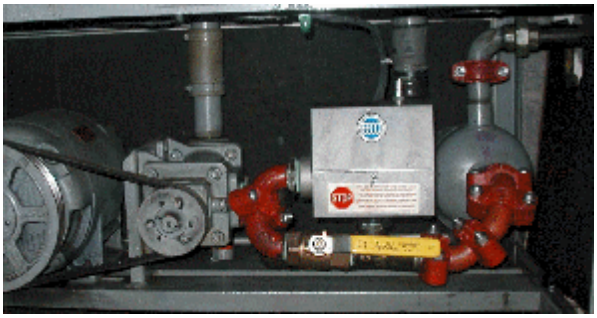
I.6.2.3.1. Avantages:

- vitesse optimisée par le variateur de fréquence.
- compacité du système.
- pas de cabanon technique nécessaire pour les ascenseurs.
- précision dans le déplacement et sur la régulation de vitesse.
- pertes mécaniques réduites.
- efficacité énergétique intéressante.
- faible niveau de bruit.
- poids réduit.

I.6.2.3.2. inconvénients :

- la compacité peut entraîner des difficultés de maintenance.
- difficulté d'intervention dans la cage d'ascenseur.

I.6.3.Les motopompes hydrauliques :



(**Figure I.16**) *Les motopompes hydrauliques*

Les groupes motopompes comme indique la **(Figure I.16)** utilisés dans les ascenseurs hydrauliques sont composés essentiellement :

- d'une pompe hydraulique de mise en pression de l'huile dans le cylindre,
- d'un moteur électrique d'entraînement de la pompe,
- d'une vanne de réglage de débit pour la descente de l'ascenseur,
- d'un réservoir à huile,
- d'une régulation,

La consommation électrique du moteur d'entraînement est due en grande partie au fait que, dans le principe même d'un ascenseur hydraulique, il n'y a pas de contre-poids. De plus, dans une moindre mesure et pour une même charge, les consommations :

- des motopompes centrifuges, par exemple, varient avec le cube de la vitesse,
- des moteurs-treuils, par exemple, varient proportionnellement à la vitesse.

I.7. Conclusion :

Dans ce chapitre, nous avons développé l'étude théorique de l'ascenseur notamment les différents type d'ascenseurs, les avantage et inconvénients que présente chaque

type, leur principe de fonctionnement, et leur principaux composants. Nous avons également passé en revue leur système de motorisation.

II.1. Introduction : [4]

Les Automates Programmables Industriels (API) sont apparus aux Etats-Unis vers 1969 où ils répondaient aux désirs des industries de l'automobile de développer des chaînes de fabrication automatisées qui pourraient suivre l'évolution des techniques et des modèles fabriqués.

Un Automate Programmable Industriel (API) est une machine électronique programmable qui est destiné à piloter en ambiance industrielle et en temps réel des procédés industriels. Un automate programmable est adaptable à un maximum d'application, d'un point de vue traitement, composants, langage. C'est pour cela qu'il est de construction modulaire.

Le développement de l'industrie à entraîner une augmentation constante des fonctions électroniques présentes dans un automatisme c'est pour ça que l'API s'est substitué aux armoires à relais en raison de sa souplesse dans la mise en œuvre, mais aussi parce que dans les coûts de câblage et de maintenance devenaient trop élevés.

II.2. Pourquoi l'automatisation ? [4]

L'automatisation permet d'apporter des éléments supplémentaires à la valeur ajoutée par le système. Ces éléments sont exprimables en termes d'objectifs par :

- Accroître la productivité (rentabilité, compétitivité) du système.
- Améliorer la flexibilité de production.
- Améliorer la qualité du produit.
- Adaptation à des contextes particuliers tel que les environnements hostiles pour l'homme (milieu toxique, dangereux, nucléaire...), adaptation à des tâches physiques ou intellectuelles pénibles pour l'homme (manipulation de lourdes charges, tâches répétitives parallélisées...).
- simplicité de mise en œuvre
- Fiabilité
- Augmenter la sécurité, etc...

II.3. Structure interne d'un automate programmable industriel (API): [4]

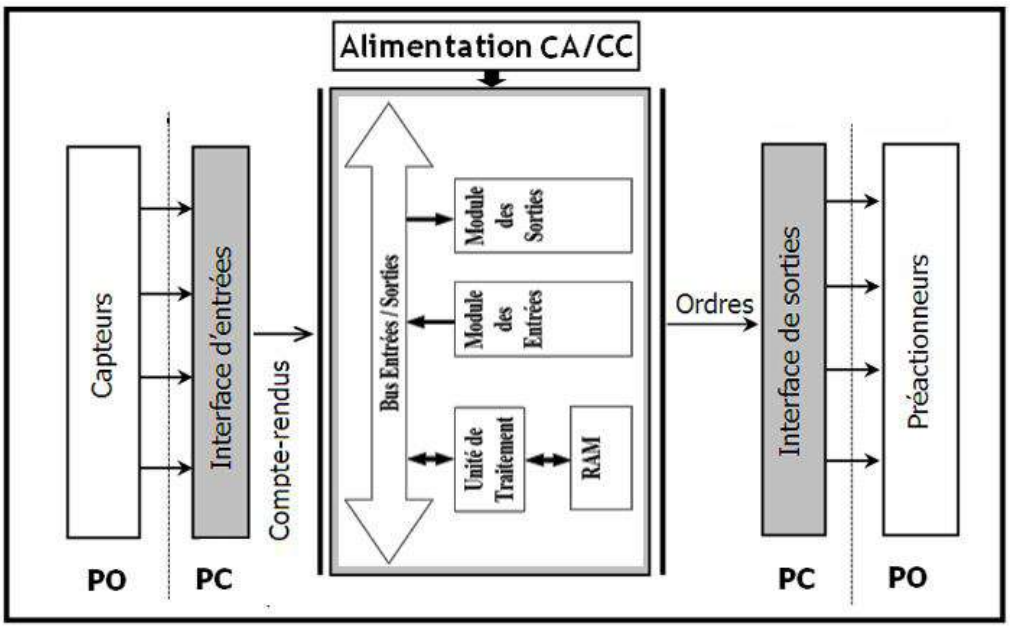
Les API comportent quatre principales parties:

- Une unité de traitement (un processeur CPU);
- Une mémoire ;
- Des modules d'entrées-sorties ;
- Des interfaces d'entrées-sorties ;
- Une alimentation 230 V, 50/60 Hz (AC) - 24 V (DC) ;

La structure interne d'un automate programmable industriel (API) est assez voisine de celle d'un système informatique simple (**figure II.1**), L'unité centrale est le regroupement du processeur et de la mémoire centrale. Elle commande l'interprétation et l'exécution des instructions programme. Les instructions sont effectuées les unes après les autres, séquencées par une horloge.

Deux types de mémoire cohabitent :

- La mémoire Programme où est stocké le langage de programmation. Elle est en général figée, c'est à dire en lecture seulement. (ROM : mémoire morte)
- La mémoire de données utilisable en lecture-écriture pendant le fonctionnement c'est la RAM (mémoire vive). Elle fait partie du système entrées-sorties. Elle fige les valeurs (0 ou 1) présentes sur les lignes d'entrées, à chaque prise en compte cyclique de celle-ci, elle mémorise les valeurs calculées à placer sur les sorties.

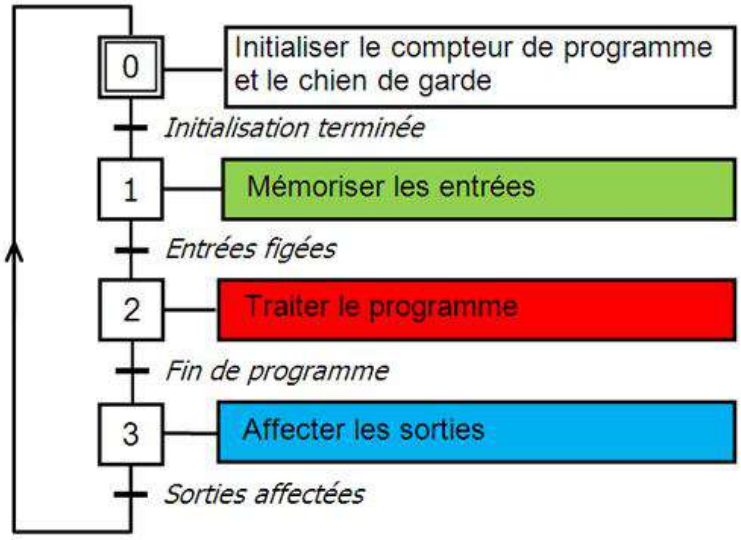


(**Figure II.1**) *schémas de fonctionnement d'un API*

II.4. Fonctionnement : [4]

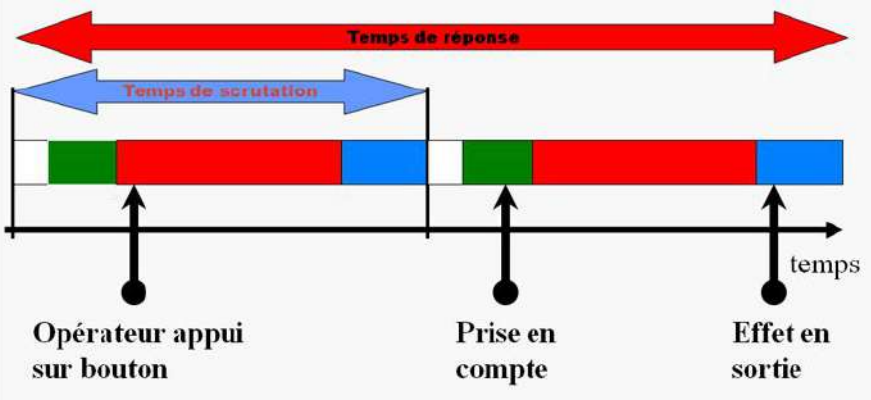
L'automate programmable reçoit les informations relatives à l'état du système et puis commande les pré-actionneurs suivant le programme inscrit dans sa mémoire.

Généralement les automates programmables industriels ont un fonctionnement cyclique (**figure II.2**). Le microprocesseur réalise toutes les fonctions logiques ET, OU, les fonctions de temporisation, de comptage, de calcul... Il est connecté aux autres éléments (mémoire et interface E/S) par des liaisons parallèles appelées ' BUS ' qui véhiculent les informations sous forme binaire. Lorsque le fonctionnement est dit synchrone par rapport aux entrées et aux sorties, le cycle de traitement commence par la prise en compte des entrées qui sont figées en mémoire pour tout le cycle.



(**Figure II.2**) *fonctionnement de API*

Le processeur exécute alors le programme instruction par instruction en rangeant à chaque fois les résultats en mémoire **(figure II.3)**. 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.3) *exécution du programme*

II.5. Jeu d'instructions : [5]

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

Chaque automate possède ses propres jeux d'instructions. Mais par contre, les constructeurs proposent tous une interface logicielle de programmation répondant à la norme CEI1131-3. Cette norme définit cinq langages de programmation utilisables, qui sont :

Les langages graphiques :

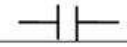
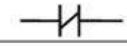
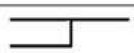
- LD : Ladder Diagramme (Diagrammes échelles)
- FBD : Function Block Diagram (Logigrammes)
- SFC : Sequential Function Chart (Grafcet)

Les langages textuels :

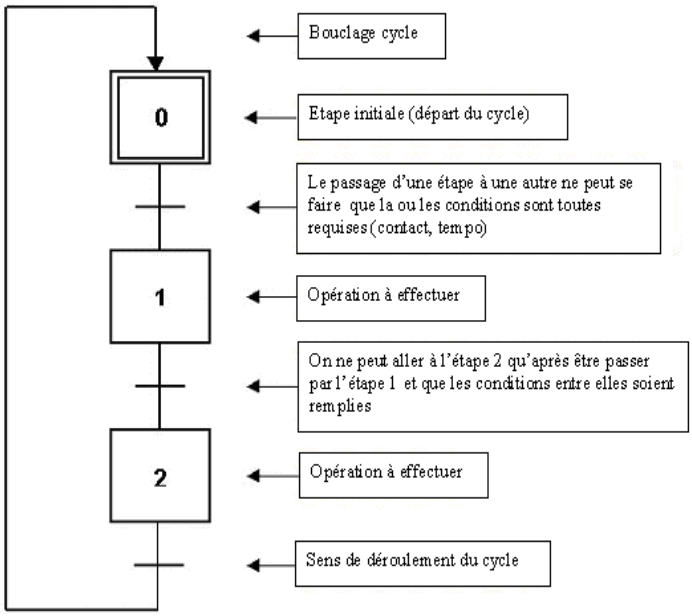
- IL : Instruction List (Liste d'instructions).
- ST: Structured Text (Text structure).

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

(**Tableau II.1**) *symboles et fonction des instructions du langage ladder*

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

Le GRAFCET et très fonctionnel comporte un nombre restreint de symboles conventionnels et permet la correction d'une partie de séquence sans remettre en cause les autres facilitant ainsi les modifications **(figure II.4**). De plus ce système étant conventionnel à l'avantage d'être facilement interprété par n'importe qu'elle personne le connaissant



(Figure II.4) *symboles et fonction des instructions du Grafcet*

II.6.Sécurité : [4]

Les systèmes automatisés sont, par nature, source de nombreux dangers (tensions utilisées, déplacements mécaniques, jets de matière sous pression ...).

Placé au cœur du système automatisé, l'automate se doit d'être un élément fiable car un dysfonctionnement de celui-ci pourrait avoir de graves répercussions sur la sécurité des personnes, de plus les coûts de réparation et un arrêt de la production peuvent avoir de lourdes conséquences sur le plan financier.

Aussi, l'automate fait l'objet de nombreuses dispositions pour assurer la sécurité :

- Contraintes extérieures : l'automate est conçu pour supporter les différentes contraintes du monde industriel et à fait l'objet de nombreux tests normalisés.
- Coupures d'alimentation : l'automate est conçu pour supporter les coupures d'alimentation et permet, par programme, d'assurer un fonctionnement correct lors de la réalimentation (reprises à froid ou à chaud).
- Mode RUN/STOP : Seul un technicien peut mettre en marche ou arrêter un automate et la remise en marche se fait par une procédure d'initialisation (programmée)
- Contrôles cycliques :

- Procédures d'autocontrôle des mémoires, de l'horloge, de la batterie, des tensions d'alimentation et des entrées / sorties.
- Vérification du temps de scrutation à chaque cycle appelée Watch dog (chien de garde), et enclenchement d'une procédure d'alarme en cas de dépassement de celui-ci (réglé par l'utilisateur)
- Visualisation : Les automates offrent un écran de visualisation où l'on peut voir l'évolution des entrées / sorties.

Les normes interdisent la gestion des arrêts d'urgence par l'automate ; celle-ci doit être réalisée en technologie câblée.

II.7.Critères de choix d'un automate :

Le choix d'un automate programmable est généralement basé sur :

- Nombre d'entrées / sorties : le nombre de cartes peut avoir une incidence sur le nombre de racks dès que le nombre d'entrées / sorties nécessaires devient élevé.
- Type de processeur : la taille mémoire, la vitesse de traitement et les fonctions spéciales offertes par le processeur permettront le choix dans la gamme souvent très étendue.
- Fonctions ou modules spéciaux : certaines cartes (commande d'axe, pesage ...) permettront de "soulager" le processeur et devront offrir les caractéristiques souhaitées (résolution, ...).
- Fonctions de communication : l'automate doit pouvoir communiquer avec les autres systèmes de commande (API, supervision ...) et offrir des possibilités de communication avec des standards normalisés (Profibus ...).

II.8. Mise en œuvre et diagnostic d'un API :

II.8.1. Vérification du fonctionnement :

Lors de sa première mise en œuvre il faut réaliser la mise au point du système.

Prendre connaissance du système (dossier technique, des programmes, affectation des entrées / sorties, Les schémas de commande et de puissance des entrées et des sorties).

- Lancer l'exécution du programme (RUN ou MARCHE).
- Visualiser l'état des programmes, des variables..

Il existe deux façons de vérifier le fonctionnement :

- En simulation (sans Partie Opérative).

  Le fonctionnement sera vérifié en simulant le composent de la partie opérative. C'est-à-dire l'état des capteurs, en validant uniquement des entrées.

  - Valider les entrées correspondant à l'état initial (position) de la partie opérative.
  - Valider les entrées correspondant aux conditions de marche de cycle.
  - Vérifier l'évolution des programmes (étape actives).
  - Vérifier les ordres émis (LED de sortie)
  - Modifier l'état des entrées en fonction des ordres émis (état transitoire de la P.O)
  - Modifier l'état des entrées en fonction des ordres émis (état final de la P.O.)

  Toutes les évolutions du programme doivent être vérifiées

- En condition réelle (avec Partie Opérative).

  Le fonctionnement sera vérifié en suivant le comportement de la P.O.

  - Positionner la P.O. dans sa position initiale.
  - Valider les conditions da marche du cycle.
  - Vérifier l'évolution des programmes et le comportement de la P.O.

Toutes les évolutions du programme doivent être vérifiées

II.8.2. Recherche des dysfonctionnements :

Un dysfonctionnement peut avoir pour origine :

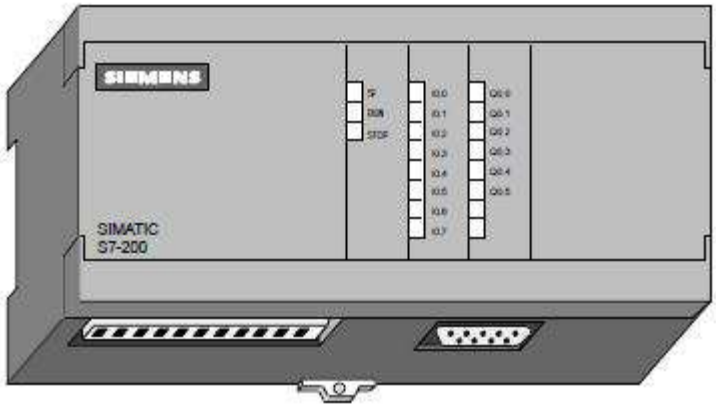
- Un composant mécanique défaillant (pré actionneur, actionneur, détecteur,...).
- Un câblage incorrect ou défaillant (entrées, sorties).
- Un composant électrique ou électronique défectueux (interface d'entrée ou de sortie).

- Une erreur de programmation (affectation d'entrées-sorties, ou d'écriture).
- Un système non initialisé (étape, conditions initiales...).

II.9. Présentation du micro-automate S7-200 : [6]

Dans notre projet on a eu la possibilité d'utilisée le S7-200 (CPU).

La famille S7-200 est constituée de micro-automates programmables utilisables dans des applications d'automatisation variées. La **(figure II.5)** présente un micro-automate S7-200. Son dessin compact, ses possibilités d'expansion, son faible prix et son important jeu d'opérations en font une solution idéale pour la commande de petites applications. En outre, le large choix de tailles et de tensions de CPU nous offre la souplesse nécessaire pour résoudre les problèmes d'automatisation.



(Figure II.5) *Automate programmable S7-200*

II.9.1 Comparaison des caractéristiques des micro-automates S7-200 :

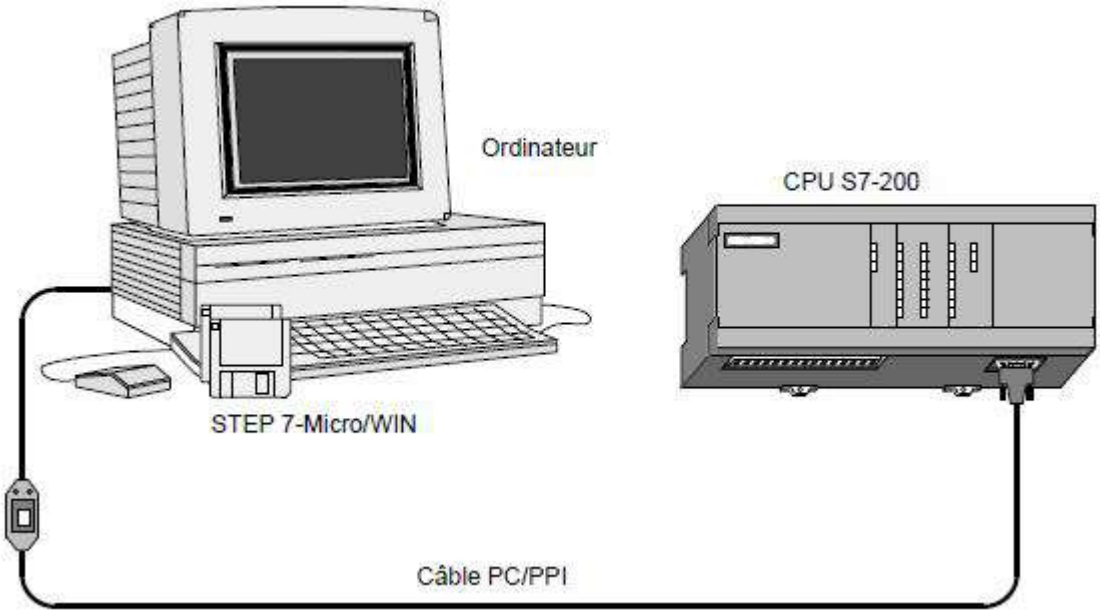
II.9.1.1.Equipement :

La **(figure. II.6)** montre un système d'automate programmable (AP) S7-200 de base, comprenant une unité centrale (CPU) S7-200, un ordinateur personnel, le logiciel de programmation STEP 7-Micro/WIN et un câble de communication.

Pour pouvoir utiliser un ordinateur personnel (PC), on doit disposer :

- d'un câble PC/PPI,

- ou d'une carte de processeur de communication (CP) et d'un câble d'interface multipoint (MPI)
- ou d'une carte d'interface multipoint (MPI). Un câble de communication est fourni avec la carte MPI.



(**Figure. II.6)** *Composantes d'un système d'automate programmable S7-200*

II.9.1.2. Possibilités des CPU S7-200 :

La famille S7-200 comporte un large choix de CPU. On dispose ainsi de différentes fonctions permettant de concevoir une solution d'automatisation au meilleur prix possible. Le (**Tableau II.1)** résume les principales caractéristiques de chaque CPU S7-200.

(**Tableau II.2)** *les principales caractéristiques de chaque CPU S7-200*

Caractéristique	CPU 212	CPU 214	CPU 215	CPU 216
Taille physique de l'unité	160 mm x 80 mm x 62 mm	197 mm x 80 mm x 62 mm	218 mm x 80 mm x 62 mm	218 mm x 80 mm x 62 mm
Mémoire				
Programme (EEPROM)	512 mots	2 kilo-mots	4 kilo-mots	4 kilo-mots
Données utilisateur	512 mots	2 kilo-mots	2,5 kilo-mots	2,5 kilo-mots
Mémentos internes	128	256	256	256
Cartouche mémoire	Néant	oui, EEPROM	oui, EEPROM	oui, EEPROM
Cartouche pile optionnelle	Néant	200 jours, typique	200 jours, typique	200 jours, typique
Sauvegarde (supercondensateur)	50 heures, typique	190 heures, typique	190 heures, typique	190 heures, typique

Entrées/sorties (E/S)				
E/S locales	8 E TOR/6 S TOR	14 E TOR/10 S TOR	14 E TOR/10 S TOR	24 E TOR/16 S TOR
Modules d'extension (max.)	2 modules	7 modules	7 modules	7 modules
Mémoire image des E/S	64 E TOR/64 S TOR	64 E TOR/64 S TOR	64 E TOR/64 S TOR	64 E TOR/64 S TOR
E/S analogiques (extension)	16 E ANA/16 S ANA	16 E ANA/16 S ANA	16 E ANA/16 S ANA	16 E ANA/16 S ANA
Filtres d'entrée sélectionnables	Non	Oui	Oui	Oui

Opérations				
Vitesse d'exécution booléenne	1,2 µs/opération	0,8 µs/opération	0,8 µs/opération	0,8 µs/opération
Compteurs / temporisations	64/64	128/128	256/256	256/256
Boucles FOR/NEXT	Non	Oui	Oui	Oui
arithmétiques, entiers	Oui	Oui	Oui	Oui
arithmétiques, réels	Non	Oui	Oui	Oui
PID	Non	Non	Oui	Oui

Fonctions supplémentaires				
Compteur rapide	1 logiciel	1 logiciel, 2 matériels	1 logiciel, 2 matériels	1 logiciel, 2 matériels
Potentiomètres analogiques	1	2	2	2
Sorties d'impulsions	Néant	2	2	2
Evénement d'interruption de communication	1 pour émission/ 1 pour réception	1 pour émission/ 1 pour réception	1 pour émission/ 2 pour réception	2 pour émission/ 4 pour réception
Interruptions cycliques	1	2	2	2
Interruptions d'entrée matérielles	1	4	4	4
Horloge temps réel	Néant	Oui	Oui	Oui

Communication				
Nombre d'interfaces	1 (RS-485)	1 (RS-485)	2 (RS-485)	2 (RS-485)
Protocoles acceptés Interface 0 : Interface 1 :	PPI, communication programmable Non disponible	PPI, communication programmable Non disponible	PPI, communication programmable, MPI DP, MPI	PPI, communication programmable, MPI PPI, communication programmable, MPI
Egal à égal	esclave uniquement	Oui	Oui	Oui

II.10. Composantes principales de l'automate programmable S7-200 : [6]

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

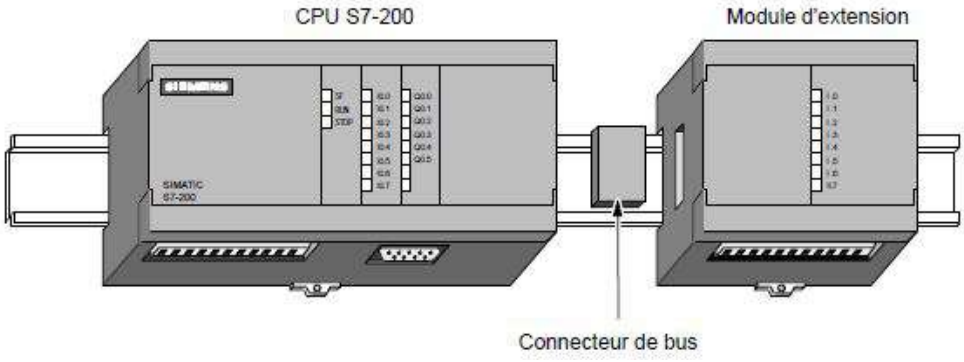
II.10.1.CPU S7-200 :

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

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

II.10.2.Modules d'extension :

La CPU S7-200 comporte des entrées/sorties locales. Les modules d'extension permettent d'ajouter des entrées/sorties à l'appareil de base. Comme illustré à la (**figure II.7**), un connecteur de bus fourni avec le module d'extension permet de connecter ce dernier à l'appareil de base.



(**Figure II.7**) *Unité centrale avec module d'extension*

II.11. Mode de représentation du langage de programmation STEP 7 : [6]

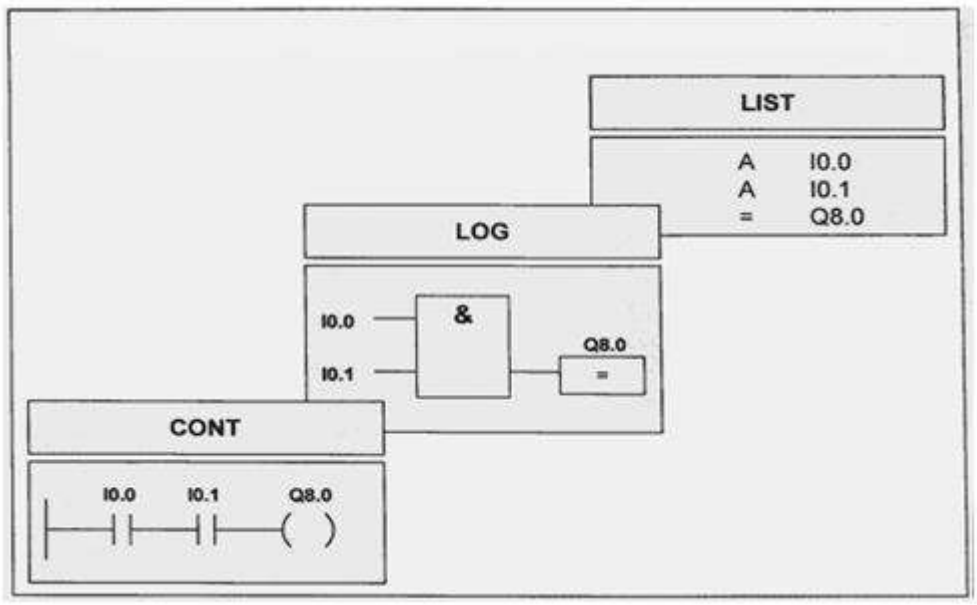
Le langage de programmation STEP 7 dispose de plusieurs modes de représentation, selon les goûts et l'état des connaissances, le programme peut être conçu sous forme de liste d'instructions puis converti en un autre mode de représentation (**figure. II.8)**. En représentant certaines règles sur le schéma suivant:

CONT : le schéma à contacts s'apparente aux schémas de circuits électriques, il fait appel à des symboles, tels que les contacts et bobines. Ce mode de représentation convient particulièrement à tous les familiers de technologie des contacteurs.

LIST *:* la liste d'instructions se compose d'opérations STEP7. **List** permet de programmer librement (et parfois de manière complexe). Ce mode de représentation est plus particulièrement destiné aux programmeurs qui maîtrisent déjà d'autres langages de programmation.

LOG : le programme est un langage de programmation graphique qui utilise des boites logiques. Le signe placé dans la boite symbolise la fonction (par exemple &

opération logique ET). Ce mode de représentation permet à des non programmeurs, par exemple des spécialistes des procédés industriels, d'accéder à la programmation.



(**Figure. II.8**) *modes de représentation de programme*

II.12. Saisie d'un programme « CONT » : [6]

II.12.1. Création d'un projet :

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

II.12.2. Ouvrir un projet existant :

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

- ✓ Ouvrir permet de sélectionner un projet existant et de l'ouvrir.
- ✓ Nom de fichier : Si on a travaillé récemment sur un projet , il apparaît dans le menu

Fichier où on peut le sélectionner directement sans passer par la boîte de dialogue "Ouvrir".

On peut également faire appel à l'explorateur Windows pour sélectionner le répertoire approprié et ouvrir le projet sans avoir à lancer STEP 7-Micro/WIN 32 lors d'une

Étape précédente. Dans STEP 7-Micro/WIN 32, version 3.0 ou plus, un projet est contenu dans un fichier unique, d'extension.

II.12.3. Compilation en CONT :

Marche à suivre pour la compilation on peut effectuer la compilation à l'aide de boutons dans la barre d'outils ou de Commandes du menu Système cible **(Figure. II.9)**.



(Figure. II.9) *Système cible.*

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

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

II.12.4. Correction des erreurs à l'aide de la fenêtre :

Lors de la compilation, la fenêtre "Erreurs et avertissements" énumère toutes les erreurs qui apparaissent. Ces erreurs sont identifiées par leur emplacement (réseau,

ligne ou colonne), ainsi que par leur type. Double-cliquez sur une ligne d'erreur pour afficher le réseau contenant cette erreur dans l'éditeur de programme.

II.12.5. Enregistrement du travail :

Pour enregistrer le travail, cliquer sur le bouton "Enregistrer"  dans la barre d'outils, permet de sauvegarder rapidement toutes les modifications apportées à votre travail. Lors du premier enregistrement d'un projet, le système nous demande de confirmer ou de modifier les choix de nom et de répertoire par défaut pour le projet en cours ou exécuter la commande Enregistrer ou Enregistrer sous du menu Fichier, permet de modifier le nom ou le répertoire du projet en cours. Par défaut, STEP 7-Micro/WIN donne le nom "Projet1.mwp" au projet lors de sa création. On peut accepter ce nom ou le modifier ; s'il est accepté, le nom par défaut pour le nouveau projet sera automatiquement modifié en "Projet2.mwp ".

Par défaut, les projets STEP 7-Micro/WIN 32 sont enregistrés dans le dossier "Project" situé dans le répertoire "STEP 7-Micro WIN". On n'est pas obligé d'utiliser ce dossier pris par défaut

II.13.Etablissement et communication et chargement d'un programme :

II.13.1. Sélection d'un type d'AP :

On peut sélectionner un AP en choisissant un modèle dans la liste de la boîte de dialogue "Type d'AP" comme indique la **(Figure. II.10)**. Si on dispose, par exemple, d'une CPU 224.la liste affiche :



(Figure. II.10) *Type d'AP.*

Lorsqu'on effectue cette sélection, les restrictions de plages et de fonctions pour la dernière version de microprogramme de la CPU 224sont appliquées. Mais, on peut

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

Remarque

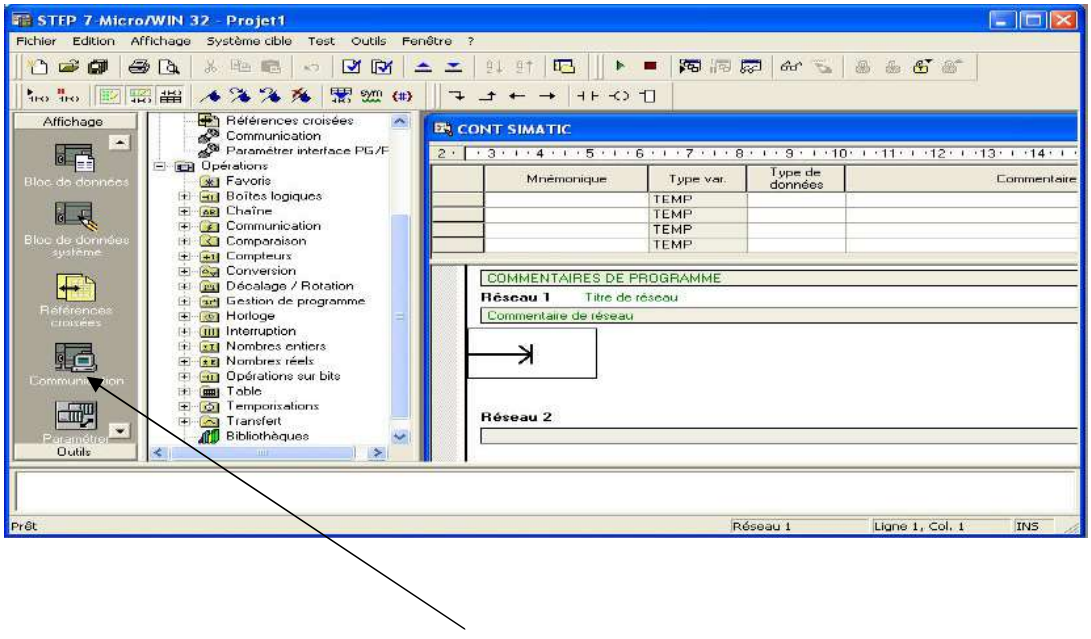
On n'a pas besoin de configurer la communication pour sélectionner un type d'AP. La communication ne doit être établie que si nous voulons que STEP 7-Micro/WIN 32 lise le type d'AP éloigné.

II.13.2. Lecture du type d'AP éloigné par STEP 7-Micro/WIN 32 :

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

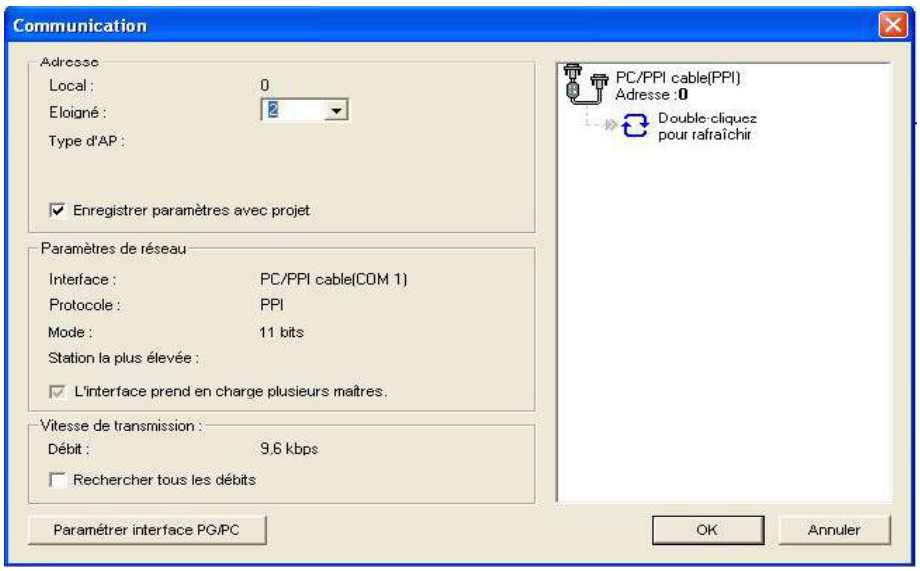
II.13.3. Test du réseau de communication :

1- Dans STEP 7-Micro/WIN 32, cliquer sur l'icône Communication dans la barre d'exploration (**Figure. II.11)** ou sélectionner Affichage > Composante > Communication dans la barre des menus



(Figure. II.11) *l'icône Communication.*

2- Dans la partie droite de la boîte de dialogue "Communication", cliqué sur le texte "Double clique pour rafraîchir". **(Figure. II.12)**



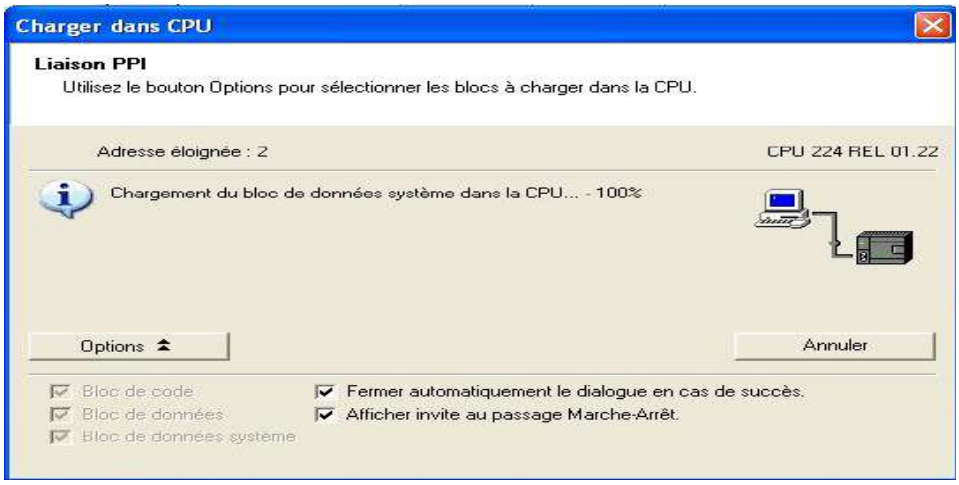
(Figure. II.12) *Communication.*

Si on a réussi à établir la communication entre notre ordinateur personnel et les unités dans notre réseau, la liste de ces unités (avec leur type de modèle et leur adresse de station) s'affiche.

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

II.13.4. Chargement d'un programme dans la CPU :

Si on a établit la communication entre l'ordinateur personnel sur lequel s'exécute STEP 7-Micro/WIN 32 et un AP, on peut charger le programme dans cet AP. En procédant comme suit:



(Figure. II.13) *Charger dans CPU.*

Remarque

Lorsqu' on charge un bloc de code, un bloc de données ou un bloc de données système de l'ordinateur personnel dans l'AP, le contenu du bloc que nous avons chargé à partir de l'ordinateur personnel écrase le contenu du bloc se trouvant éventuellement dans l'AP. S'assurer qu'on veut bien écraser le bloc dans l'AP avant de procéder à un chargement.

1) Vérifiez que l'AP est à l'état de fonctionnement "Arrêt" (STOP) avant d'effectuer un Chargement dans l'AP. Examiner à cet effet la DEL d'état sur l'AP. Si l'AP n'est pas à l'état

"Arrêt", cliquer sur le bouton Arrêt dans la barre d'outils ou sélectionner la commande système cible > Arrêt

2) Cliquer sur le bouton de chargement dans la barre d'outils ou sélectionne la commande Fichier > Charger dans CPU

. La boîte de dialogue "Charger dans la CPU" apparaît alors.

3) Par défaut, les options "Bloc de code", "Bloc de données" et "Configuration de la CPU" (Bloc de données système) sont activés la première fois qu'on lance la commande de chargement. Si le chargement d'un bloc particulier est inutile, cliquer sur la case à cocher correspondante pour la désactiver...

4) Cliquez sur le bouton OK pour déclencher la procédure de chargement.

5) Si le chargement réussit, une boîte de confirmation affiche le message suivant :

"Chargement réussi". Passez alors à l'étape 12.

6) Si la valeur dans STEP 7-Micro/WIN 32 pour le type du CPU ne correspond pas à notre AP réel, une boîte d'avertissement affiche le message suivant :

"Le type d'AP sélectionné pour le projet ne correspond pas au type d'AP éloigné. Poursuivre le chargement ?"

7) Pour corriger le type d'AP, choisir "Non" pour interrompre le chargement.

8) Dans la barre des menus, sélectionner Système cible > Type afin d'afficher la boîte de

Dialogue "Type d'AP".

9) On peut y sélectionner le type d'AP correct dans la liste déroulante ou cliquer sur le bouton "Lire AP" afin que STEP 7-Micro/WIN 32 lise automatiquement la valeur correcte.

10) Cliquer sur le bouton "OK" pour confirmer le type d'AP et quitter la boîte de dialogue.

11) Relancer la procédure de chargement en cliquant sur le bouton de chargement  dans la barre d'outils ou en sélectionnant la commande Fichier > Charger dans CPU dans la barre Des menus.

12) Une fois le chargement dans la CPU réussi, on doit faire passer l'automate Programmable de l'état "Arrêt" à l'état "Marche" pour pouvoir exécuter le programme dans

L'AP. Cliquer sur le bouton Marche  dans la barre d'outils ou sélectionner la commande système cible > Marche pour remettre l'AP à l'état "Marche" (RUN).

II.14. Conclusion :

Dans le chapitre II, nous avons développé l'étude générale sur les automates et mis en évidence leur histoire leur structure, leur fonctionnement et les critères de choix d'un API.

Après examen, nous avons opté pour le micro automate S7-200 que nous avons adapté à notre projet. Nous avons donné une idée sur le langage de programmation tout en exposant les différentes étapes de saisie du programme en langage ladder.

III.1. la partie mécanique :

Dans le cadre de la concrétisation de notre projet nous avons été amenés à retenir un processus de réalisation dont les détails sont définis ci après :

III.1.1. La structure extérieure :

Nous avons utilisée des plaques de résine que nous avons perforée pour permettre la pose de boutons de commande et de LED et une armature en aluminium que nous avons assemblé a l'aide de vis et de bouchon de fermeture **(figure III.1).**



(Figure III.1) *construction de structure externe*

III.1.2. La structure intérieure :

La structure intérieure comprend :

- Deux plaques de fixation situées sur les parties inférieures et supérieures de la structure, retenue par une plaque métallique posée verticalement comme démontre la **figure III.2.**



(**Figure III.2)** *plaques de fixation*

- Une fixation de moteur posée au sommet de la structure **(Figure III.3).**



(Figure III.3) *Fixation de moteur*

- Deux guides cylindriques posées verticalement et en parallèle **(Figure III.4).**



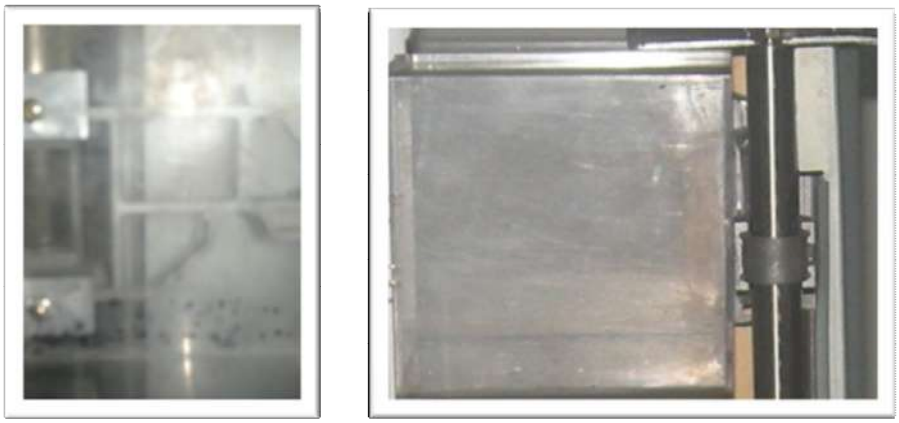
(Figure III.4) *les guides cylindriques*

- Deux poulies posées à la base et au sommet de la structure et une couroi d'entrainement cranté **(Figure III.5).**



(Figure III.5) *poulies/couroi.*

- Une cabine réalisée avec une boite en plastique posée sur un coulisseau en aluminium réaménagée coulissant le long de deux guides cylindriques et fixée sur une courroie d'entrainement crantée **(Figure III.6).**



(Figure III.6) *La cabine.*

- Un contre poids fixé sur la couroi d'entrainement **(Figure III.7).**



(Figure III.7) *Le contre poids*

- Une planche posée le long de la couroi pour assurer le guidage du contre poids **(Figure III.8).**



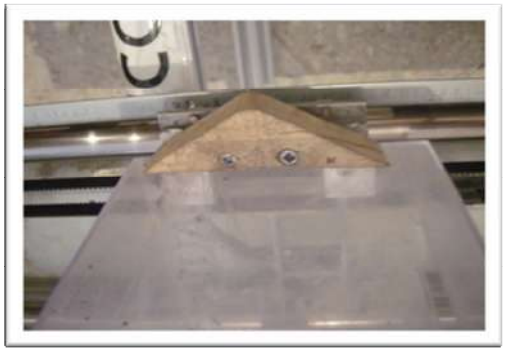
(Figure III.8) *le guide du contre poids*

- Une planche en bois posée verticalement et comprenant quatre capteurs de la position de la cabine **(Figure III.9)**



(Figure III.9) *porte capteurs*

- Une pièce en bois en forme de triangle fixé sur la partie extérieur de la cabine pour assurer le contact avec les capteurs de position **(Figure III.10).**



(Figure III.10) *triangle de contacte*

III.1.3. Transformation d'un mouvement de rotation en mouvement de translation :

Pour déplacer la cabine il faut transformer le mouvement de rotation du moteur en translation. Il existe plusieurs types de transformation :

- poulie / courroie
- Le système pignon/crémaillère.
- Le système vis/écrou.
- Le système bielle/manivelle.
- Système roue/vis sans fin
- treuil

Nous avons adopté le système de deux **Poulies crantées** (roues dentées) mobiles en rotation autour de deux axes fixes. Ces roues sont caractérisées par leur diamètre et le nombre de dents qu'elles possèdent, et aménagées pour recevoir une courroie dentée comme illustré dans la **Figure III.11.**



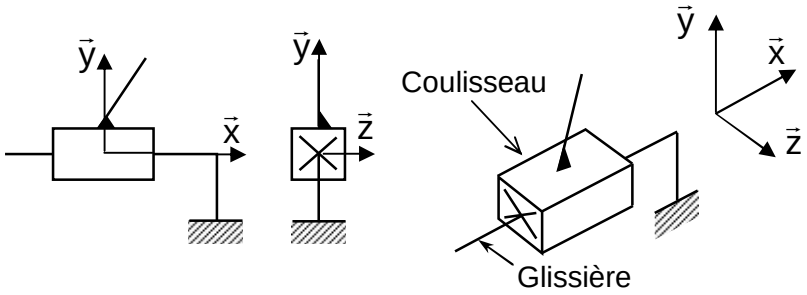
(Figure III.11) *Poulies/couroi (crantées)*

L'avantage de l'utilisation des poulies crantées est :

- Qu'elles permettent une coordination sans dérapage et sans glissement entre la courroie et les poulies qui leurs sont associées
- Une souplesse par rapport à la chaine.
- Elles supportent bien les basses vitesses et exigent une tension initiale plus faible.

III.1.4. Guidage en translation :

Le guidage en translation est la solution constructive qui réalise une liaison glissière entre deux pièces ou ensembles de pièces. Le seul mouvement relatif possible entre les deux pièces ou ensembles de pièces est une translation rectiligne.



(Figure III.11) *Schématisation cinématique normalisée de la liaison glissière*

La partie mobile est appelée **coulisseau** et la partie fixe (en général liée au bâti) est appelée **glissière** (ou **guide)**.

Nous avons utilisé le guidage par deux cylindres pour éliminer 5 degrés de liberté : 2Translations et 3 Rotations et la conservation d'un seul mouvement prismatique pour la montée et la décente de la cabine.

Cette solution est cependant classique elle nécessite un parfait parallélisme, ce qui nous a posé un problème de blocage de la cabine lorsqu' elle déplace.

Pour résoudre ce problème on a élargit un trou pour éviter le blocage de la cabine **(Figure III.12).**



(Figure III.12) *solution proposé*

III.1.5. Le contre poids

Le contrepoids existe seulement sur les ascenseurs électriques à adhérence. Son rôle est de contre balancer le poids de la cabine. En général, son poids équivaut au poids de la cabine plus la moitié de sa capacité. Donc à vide, le contrepoids est plus lourd que la cabine.

En adaptant ce principe à notre projet nous avons eu des difficultés à assurer l'équilibre de notre cabine en raison de la faiblesse du moteur et des difficultés de freinage de la cabine ce qui a influencé sur les arrêts de la cabine par rapport aux capteurs de position que nous avons installé. Autrement la cabine ne s'arrête pas à la position exacte.

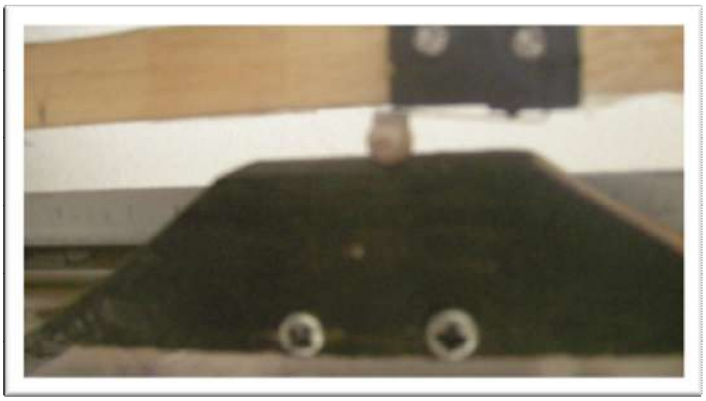
En conséquence on a pensé à équilibrer le poids entre la cabine et le contre poids pour assurer la souplesse du déplacement.

Pour ce faire nous avons opté pour un système de moulage, nous avons donc utilisé du plomb comme matière. **(Figure III.13)**



(Figure III.13) *moule utilisée*

Et comme précaution afin d'assurer le contact lorsque la cabine s'arrête nous avons agrandir l'intervalle d'erreur en remplacent le triangle de contact par un trapèze isocèle comme indique la **figure.III.14.**



(figure.III.14) *trapèze de contacte*

III.1.6. Identification des éléments intervenant pour le déplacement de la cabine :

- moteur à courant continu
- poulies
- courroie
- deux guides cylindriques fixes
- Fixation guides, poulie et moteur
- Fixation cabine
- Contre poids

III.2. Partie électronique :

Après avoir décrit la partie mécanique, nous allons maintenant entamer la partie électronique afin de commander tout le système. Commençons tout d'abord par une description théorique des différents composants qu'on a utilisés pour pouvoir réaliser notre prototype d'ascenseur.

III.2.1 Le moteur électrique :



(Figure III.15) *Moteur à courant continu*

Pour notre projet nous avons opté pour un moteur à courant continu **(figure III.15)** parce qu'il présente des facilités d'utilisation qui se traduisent par les caractéristiques suivantes : [Annexe 1]

- Couple nominal 1,6 Ncm
- Puissance utile de 4,9 à 5,3 W
- Tension nominale 12 ou 24 Vdc
- Vitesse nominale de 3150 à 2900 tr/mn
- Courant nominal de 0,45 à 0,95 A
- Constante de vitesse de 1281 à 1375
- Constante de couple de 2 à 4,2 (Ncm / A)
- Inertie (gcm^2) 14
- Masse 0,1 Kg
- Longueur L 50 (mm)

III.2.2. Les capteurs :

Nous avons utilisés des capteurs mécaniques de position, appelés aussi interrupteurs de position comme illustré a la **figure III.16**.Ces capteurs sont employés essentiellement dans les systèmes automatisés pour assurer la fonction de détection des positions de la cabine.

Ils sont réalisés à base de microcontacts placés dans un corps de protection avec 3 sorties cosses a câbler (C, NC, NO)



(Figure III.16) *Capteurs mécaniques de position*

III.2.3. Les LED :

Nous avons utilisé deux couleurs de LED standards pour indiquer la position de la cabine d'une part et pour matérialiser l'appel d'autre part. **(Figure III.17)**



(Figure III.17) *LED standard*

III.2.4. Les boutons poussoirs:

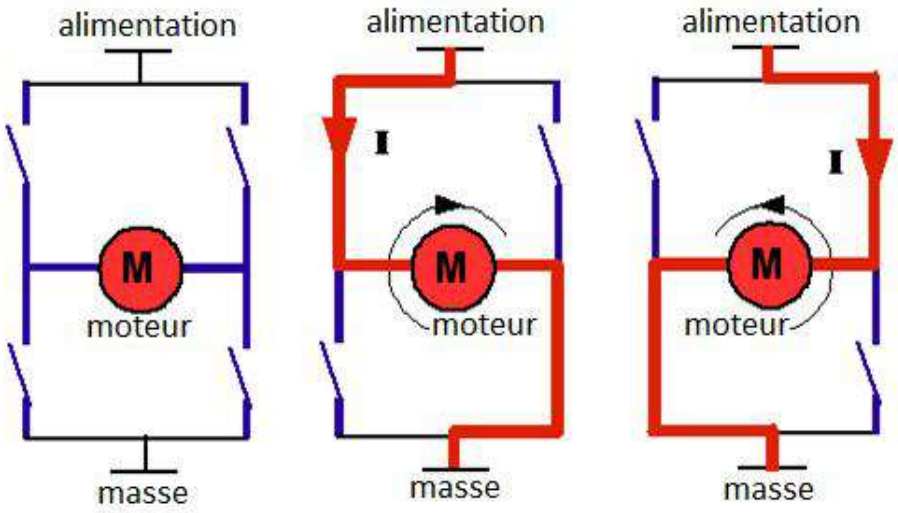
Nous avons utilisé des boutons poussoirs momentanés normalement ouvert, rond, vert, Corps noir, comme indique la **figure III.18**, supportant un courant nominal de 3A et une tension de 250V au max.



(Figure III.18) *Boutons poussoirs*

III.2.5. La commande du moteur :

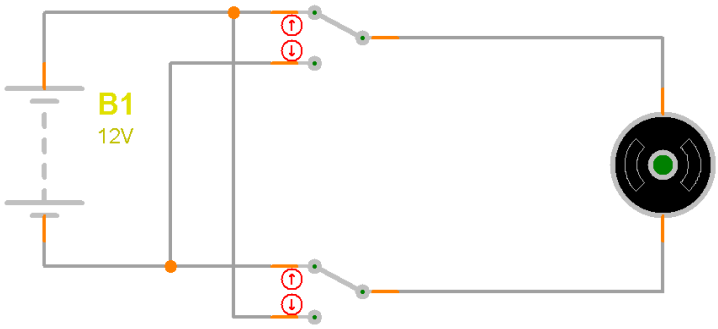
Lorsqu'on veut faire tourner un moteur à courant continu dans les deux sens de rotation, il suffit d'inverser la polarité de l'alimentation sur ses bornes. Le montage est très simple, avec 4 interrupteurs comme illustré à la **figure III.19**, ou avec 2 inverseurs.



(Figure III.19) *Montage avec 4 interrupteurs*

On comprendrait que le montage s'appelle un Pont en H.

Le montage avec des interrupteurs est dangereux, on risque de mettre l'alimentation en court-circuit!, nous avons alors pensé à mettre des inverseurs double comme indique la figure.III.20. pour avoir un seul bouton de manipulation.



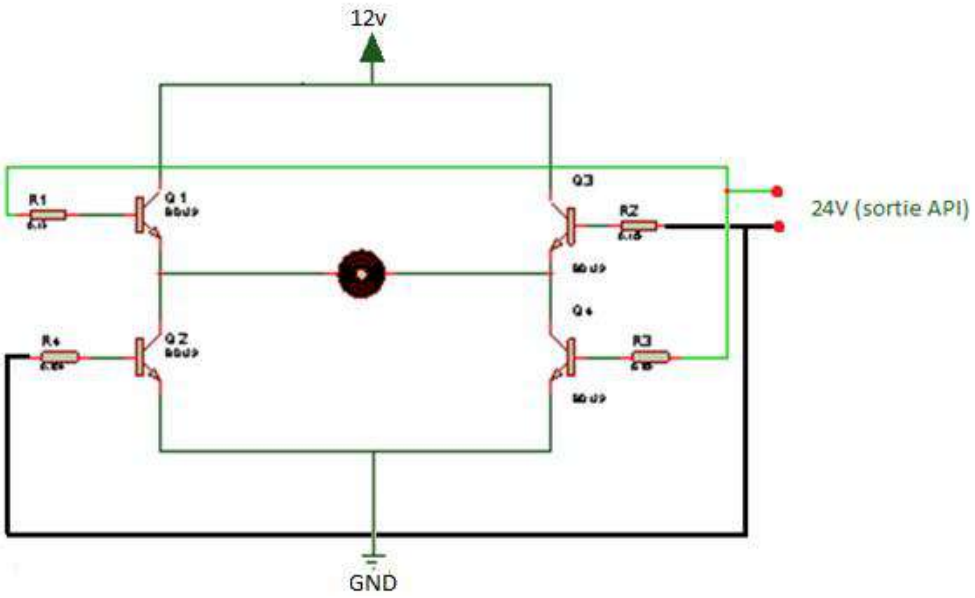
(Figure III.20) *montage avec inverseur double*

Cette solution ne nous convient pas puisque c'est une commande manuelle. Pour pouvoir commander électroniquement le moteur, on a pensé à utiliser un relais (inverseur double).

Mais compte tenu de sa lenteur, il provoque des rebonds sur les contacts et présente un inconvénient majeur par l'usure en raison de son contact mécanique.

Nous avons en conséquence opté pour des transistors. On a remplacé les interrupteurs de la **figure III.19.** Par des transistors fonctionnant en mode "bloqué-saturé"

La **figure III.21.** Illustre le montage réalisé :



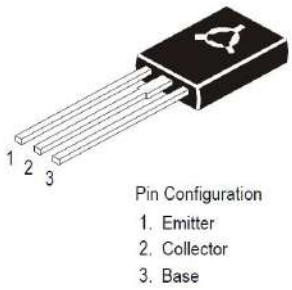
(Figure III.21) *Montage en H avec 4 transistors*

III.2.6. Réalisation de la carte :

Nous présentons ci après une description détaillée pour la réalisation de la carte de commande de notre moteur.

III.2.6.1. Composants de la carte :

III.2.6.1.1. Transistors :



(Figure III.22) *transistor BD 177*

Nous avons utilisé des transistors de type BD 177**(Figure III.22)** dont les caractéristiques suivantes : [annexe 2]

- Désignation Type: BD177
- Matériel de transistor: Si
- Polarité: npn
- Maximum de dissipation de puissance collecteur (Pc): 30W
- Maximum de tension collecteur-base (UCB): 60V
- Maximum tension collecteur-émetteur (UCE): 60V
- Maximum de tension émetteur-base (UEB): 5V
- Collecteur de courant maximum (Ic max): 3A
- Température de jonction maximum (Tj): 135 ° C
- fréquence de transition (pi): 3MHz
- Ratio de transfert avant de courant (HFE), min / max: 40min
- Paquet de BD177 transistor: TO126
- Application: alimentation, usage général

III.2.6.1.2. Résistance :

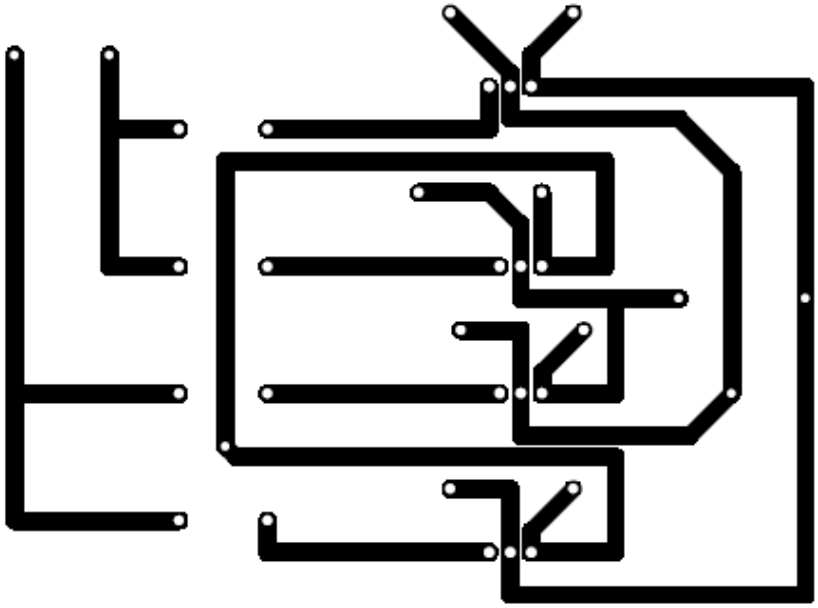
La valeur de la résistance doit être calculée selon la tension d'alimentation (U) et le type de transistor, en utilisant dans ce cas la formule R = U / I, dérivée de la célèbre formule U=RxI.

Sachant que l'alimentation est de 24V (la sortie de API) et **I**B=0.1A. La valeur de la résistance serait égale à **240Ω**

Par précaution On a choisi de prendre **R=680 Ω** pour notre carte.

III.2.6.2. Elaboration du typon :

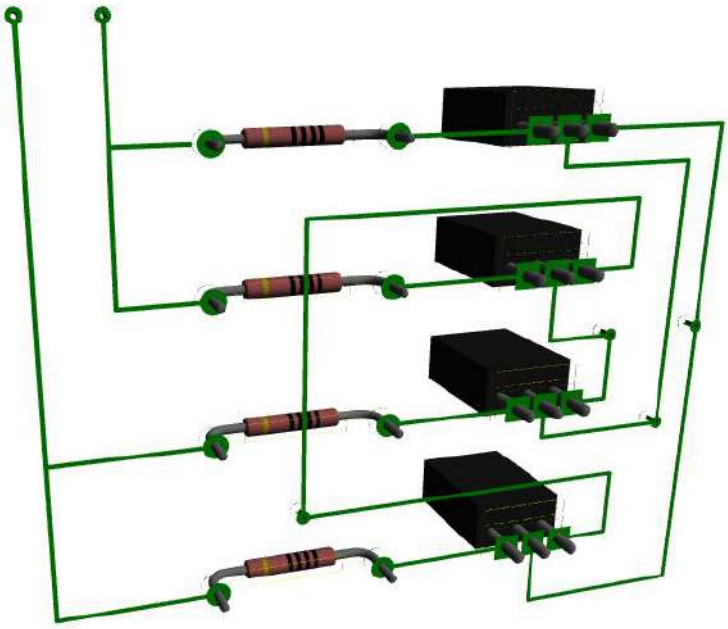
Nous avons utilisé le logiciel ARES pour confectionner le typon démontré à **la figure III.23** :



(Figure III.23) *Elaboration du typon.*

III.2.6.3. Visualisation en 3D :

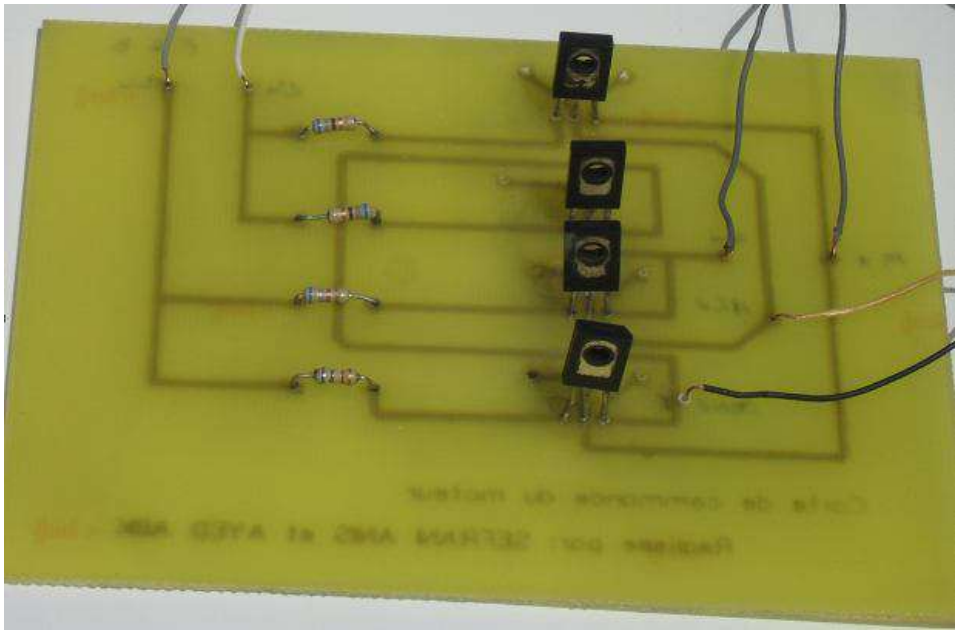
La **figure III.24.** Donne la carte en trois dimensions



(Figure III.24) *Visualisation en 3D.*

III.2.6.4. Vue en plan de la carte réalisée :

La **figure III.25**.démontre la carte dans son étape de finalisation

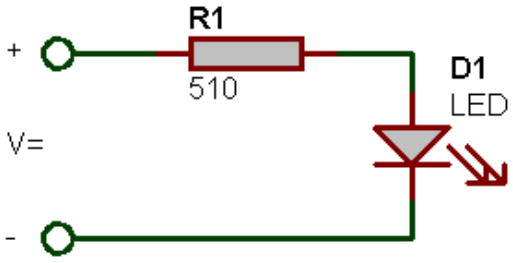


(Figure III.25) *Vue en plan de la carte réalisée.*

III.2.7. Les LED :

Si on alimente la LED directement et sans autre forme à une source de tension continue ou à une source de tension alternative, nous auront 99% de chances pour que la LED soit grillée

Le câblage minimal d'une LED consiste donc à lui ajouter une résistance en série afin de limiter le courant qui va la parcourir, ce que montre la **figure III.26**, où la LED est raccordée à une source de tension continue



(Figure III.26) *résistance en série avec une LED*

La valeur de la résistance doit être calculée selon la tension d'alimentation (U) et selon le type de la LED, en utilisant la formule R = U / I, dérivée de la célèbre formule U = R x I

R = (Ualim - Uled) / Iled

Où :

- **R :** est la valeur (en ohms) de la résistance qui limite le courant dans la led,
- **Ualim :** est la tension disponible (en volts) qui servira à alimenter la led,
- **Uled :** est la chute de tension (en volts) à la borne de la led (on parle aussi de tension nominale de la led),
- **Iled :** le courant (en ampères) devant parcourir la led pour qu'elle s'allume "normalement".

Dans notre prototype on a utilisé deux types de LED (rouge et bleu) dans les caractéristiques illustré dans le **tableau III.1.** :[annexe 3]

(Tableau III.1) *caractéristiques des LED*

Type de LED	**ROUGE**	**BLEU**
Tension (Uled)	1.6 V	3.6 V
Courant (Iled)	20 mA	20mA

On aura à utiliser un courant de 20 mA (0,02 A) dans les LED, avec une tension d'alim de 24 V (la sortie de API)

- Pour la LED rouge : R = (Ualim - Uled) / Iled = (24 - 1,6) / 0,02 = **1120 Ω**
- Pour la LED bleu : R = (Ualim - Uled) / Iled = (24 - 3,6) / 0,02 = **1020 Ω**

Par précaution nous avons choisi de prendre **R=1800 Ω** pour les deux LED

III.2.8. Le câblage :

A l'occasion de la réalisation de notre projet, nous avons eu à utiliser dans un premier temps un câble de réseaux renfermant différents câbles de plusieurs couleurs et ce pour permettre une utilisation uniforme.

Ces câbles ont été raccordés aux différents composants par soudage à l'étain. **(Figure III.27)**



(Figure III.27.) *Soudage à étain*

Lors de cette opération, un des boutons a été abimé par cause de chaleur ce qui nous a amené à dessouder la totalité des points de soudure. C'est alors que nous avons pensé à utiliser des fiches de raccordement simple dans leur manipulation **(figure III.28.)** Lors d'une éventuelle opération de maintenance et éviter de ce fait la détérioration des composants.



(Figure III.28.) *Utilisation des fiche de raccorde*

III.3 Partie automatisation :

Après avoir finalisé la partie mécanique et la partie électrique, on se consacre maintenant a la partie automatisation.

III.3.1 Objectif :

L'API doit être installée pour le contrôle des ascenseurs à n étages.

Le problème concerne le programme nécessaire pour déplacer les ascenseurs entre les étages en fonction des contraintes suivantes:

- Au démarrage du système la cabine doit être à l'étage 1
- Chaque étage dispose d'un bouton pour demander le transfert vers le haut ou vers le bas.
- Une LED est allumée si a l'étage correspondent il ya une requête non servi
- Lorsque l'ascenseur n'a pas de demande, il reste à son état actuel.
- Si plusieurs requêtes sont données ils seront servis selon le principe du premier arrivé-premier servi. (first-in first-out, FIFO)
- La position de la cabine est indiquée par une LED de position
- La cabine dispose de quatre boutons poussoirs pour le choix de la destination par le passager.

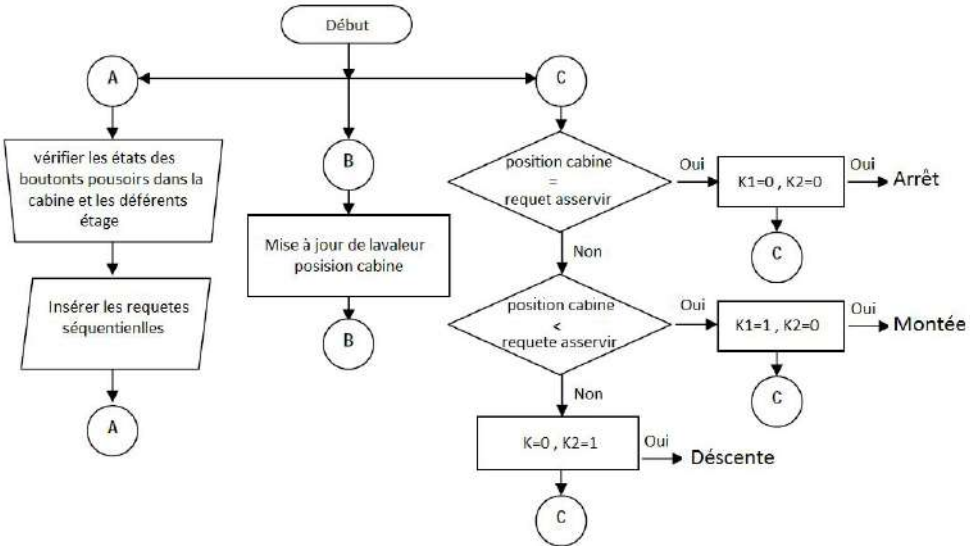
III.3.2 Organigramme de fonctionnement :

On expose l'organigramme du programme (**figure.III.29)** qui se compose de trois parties (A, B, et C).

Partie A: vérifie en permanence l'état de boutons-poussoirs placés dans les différents étages et dans l'ascenseur. Il insère aussi les demandes dans une file d'attente qui est définie dans la mémoire de l'automate.

Partie B: Toujours vérifier la position de levage actuel à l'aide de capteurs de position placés dans des étages différents.

Partie C: Prend soin de mouvement à la monté ou à la décente de la cabine



III.3.3 Description de ladder :

Le diagramme ladder a été conçue à STEP-7 MicroWIN un logiciel à 32bit (fabriqué par Siemens). Il a été réalisé pour commander plusieurs niveaux ou multi étages.

Le programme ladder a été réalisé de telle sorte qu'il peut être facilement appliqué sur un nombre N d'étages. Le programme ladder a principalement trois parties (ou, fils) qui sont en cours d'exécution simultanément. Le premier fils est en cours d'exécution pour vérifier l'état des boutons poussoirs placés dans les différents étages et les stocker dans une file d'attente. Le second est utilisé pour suivre la position actuelle de l'ascenseur et les stocker dans une mémoire temporaire. Et le troisième est utilisé pour servir les requêtes stockées dans la file d'attente comme ils ont été stockées (premier-entré-premier sorti). Pour effectuer ce genre d'opération une structure de donnée intégrée et certains blocs prêts à l'emploi ont été utilisés.

Siemens prend en charge une structure de données intégré comme FIFO, LIFO, etc.

Le tableau FIFO est utilisé pour mettre en œuvre la technique du premier arrivé, premier servi. En outre, certains autres blocs importants sont utilisés pour effectuer l'opération Queuing et d'effacement dans la file d'attente.

III.3.4 Tableau de mnémonique :

Tout les entré, sortie et adresse mémoire utilisé dans notre programme sont citer dans le **tableau.III.2.**

(Tableau III.2) *Table de mnémonique*

	Mnémonique	Adresse	commentaire
1	Cap_1	I1.5	Capteur de position de la cabine 1 étage
2	Cap_2	I1.6	Capteur de position de la cabine 2 étage
3	Cap_3	I1.7	Capteur de position de la cabine 3 étage
4	Cap_4	I2.0	Capteur de position de la cabine 4 étage
5	Bp_h_1	I0.0	Bouton d'appel haut (1 étage)
6	Bp_h_2	I0.1	Bouton d'appel haut (2 étage)
7	Bp_b_2	I0.2	Bouton d'appel haut (2 étage)
8	Bp_h_3	I0.3	Bouton d'appel haut (3 étage)
9	Bp_b_3	I0.4	Bouton d'appel haut (3 étage)
10	Bp_h_4	I0.5	Bouton d'appel haut (4 étage)
11	Bp_c_1	I0.6	BP cabine 1
12	Bp_c_2	I0.7	BP cabine 2
13	Bp_c_3	I1.0	BP cabine 3
14	Bp_c_4	I1.1	BP cabine 4

15	Led_1	Q0.0	Led 1
16	Led_2	Q0.1	Led 2
17	Led_3	Q0.2	Led 3
18	Led_4	Q0.3	Led 4
19	Led_c_1	Q0.4	Led position de la cabine (étage1)
20	Led_c_2	Q0.5	Led position de la cabine (étage2)
21	Led_c_3	Q0.6	Led position de la cabine (étage3)
22	Led_c_4	Q0.7	Led position de la cabine (étage4)
23	M_h	Q1.0	Moteur position haut
24	M_b	Q1.1	Moteur position bas
25	Requête_ asservir	Vw52	La requête asservir
26	Position_cabine	Vw50	La position de la cabine

III3.5 Composants attachés avec notre prototype :

III.3.5.1 Les entrées :

- Deux boutons poussoirs à chaque étage, sauf pour le premier et le dernier ou on utilise un seul bouton poussoir
- quatre boutons poussoirs a l'intérieure de la cabine
- quatre capteurs de position pour chaque étage

III.3.5.2Les sorties :

- quatre LED pour indiquer la position de la cabine
- quatre LED pour matérialiser l'appel
- moteur a courant continu

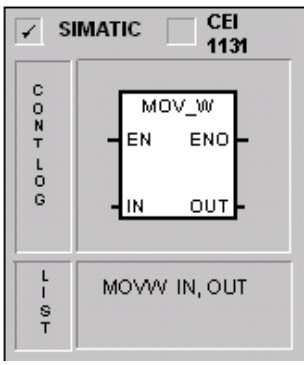
III.3.6 Spécification de l'automate

- Type de l'API: Simatic S7-200 CPU-216 de Siemens
- Nombre d'entrée disponible dans l'API : 24
- Nombre de sortie disponible dans l'API: 16
- Logiciel utilisée: STEP-7 MicroWin32 de Siemens

III.3.7 Les opération utilisée dans notre programme : [7]

III.3.7.1 Opération MOV_W :

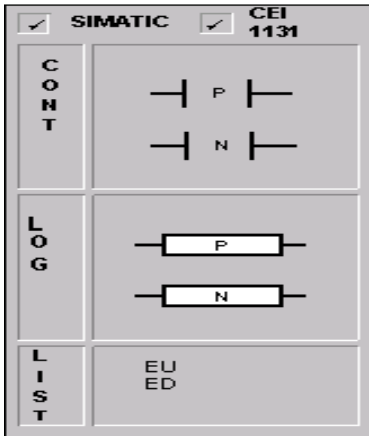
L'opération Transférer mot (MOVB) figure transfère le mot d'entrée (IN) dans le mot de sortie (OUT) sans modifier la valeur d'origine. **(Figure.III.30)**



(Figure.III.30) *Opération MOV_W*

III.3.7.2 Opération P front montant :

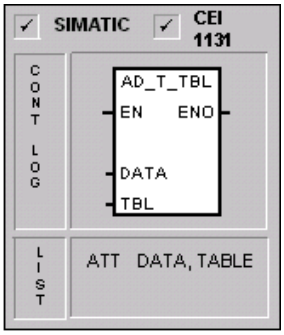
Le contact Front montant (**figureIII.31)** permet au courant de circuler pour un cycle, à chaque transition de 0 à 1.



(Figure.III.31) *Opération P front montant*

III.3.7.3 Opération AD_T_TBL:

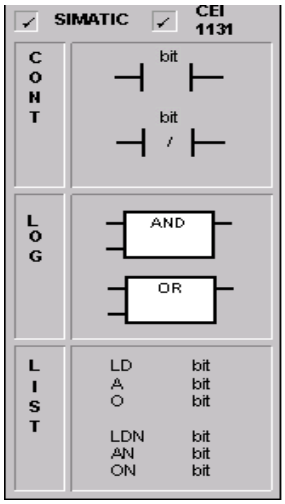
L'opération Inscrire dans table (ATT) (**figure.III.32)** inscrit des valeurs de mot (DATA) dans la table TBL. La première valeur dans la table (LT) correspond à la longueur maximale de la table et la seconde valeur (DE) au décompte des entrées effectivement dans la table. Les nouvelles données sont ajoutées après la dernière entrée de la table. Le décompte des entrées est incrémenté à chaque inscription de nouvelles données. Une table peut comporter jusqu'à 100 entrées, les deux paramètres indiquant respectivement le nombre maximal d'entrées et le nombre réel d'entrées étant exclus.



(Figure.III.32) *Opération AD_T_TBL*

III.3.7.4 OPERATION M0.2 :

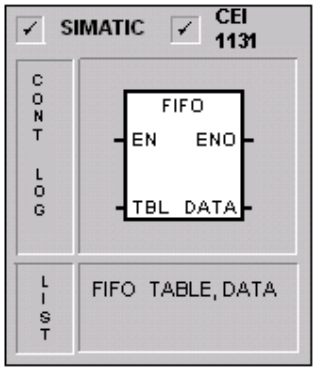
Ces opérations obtiennent la valeur référencée dans la mémoire ou dans la mémoire image si le type de données est I ou Q.



(Figure.III.33) *OPERATION M0.2*

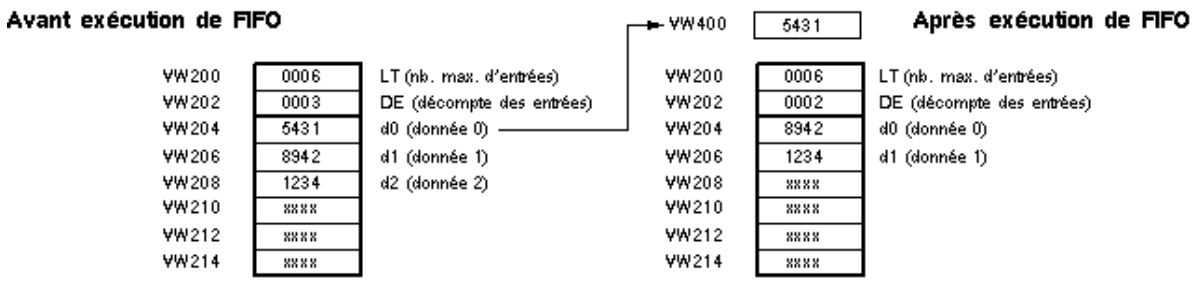
III.3.7.5 Opération FIFO :

L'opération Premier entré, premier sorti (FIFO) transfère l'entrée la plus ancienne d'une table dans l'adresse de mémoire de sortie en extrayant la première entrée dans la table (TBL) et en l'écrivant dans l'adresse indiquée par DATA. Les entrées restant dans la table sont décalées d'une position vers le haut. Le décompte des entrées (DE) est décrémenté à chaque exécution de cette opération.



(Figure.III.34) *Opération FIFO*

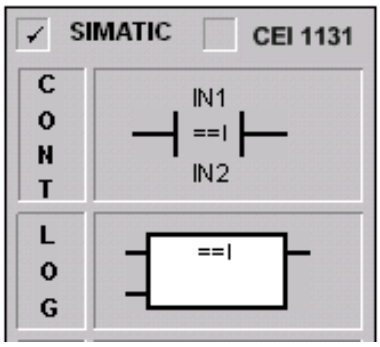
III.3.7.5.1 Exemple d'exécution de l'opération FIFO :



III.3.7.6 Comparer entiers de 16 bits :

L'opération Comparer entiers de 16 bits **(Figure.III.35)** permet de comparer les valeurs IN1 et IN2. Les comparisons incluent :

IN1 = IN2, IN1 >= IN2, IN1 <= IN2, IN1 > IN2, IN1 < IN2 ET IN1 <> IN2.

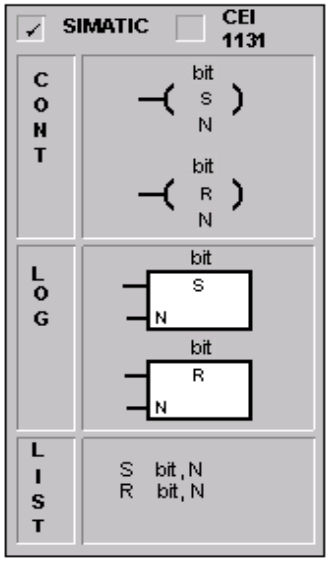


(Figure.III.35) *Comparer entiers de 16 bits*

En CONT, le contact est activé lorsque la comparaison est vraie.

III.3.7.7 Operation set (S) ET reset (R):

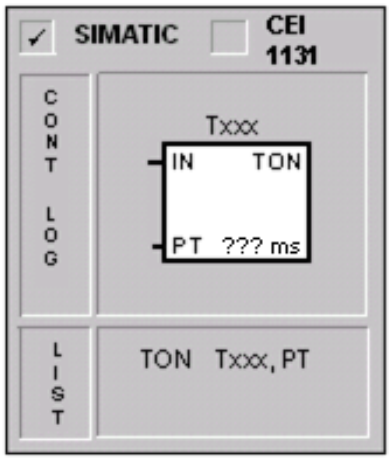
Les opérations Mettre à 1 (S) et Mettre à 0 (R) mettent à 1 (activent) ou mettent à 0 (désactivent) le nombre N de sorties indiquées, en commençant à l'adresse Bit indiquée. Vous pouvez mettre 1 à 255 sorties à 1 ou à 0.Si, pour l'opération Mettre à 0, le bit correspond à un bit de temporisation (T) ou de compteur (C), l'opération met le bit de temporisation ou de compteur à 0 et efface la valeur de comptage ou de temporisation en cours. **(Figure.III.36)**



(Figure.III.36) *Operation set (S) ET reset (R)*

III.3.7.8 Opération de temporisation T37 :

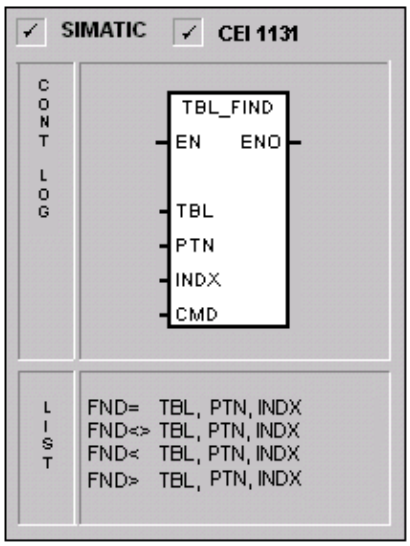
L'opération Démarrer temporisation sous forme de retard à la montée (TON) compte le temps qui s'écoule lorsque l'entrée de validation est activée. Lorsque la valeur en cours (Txxx) est supérieure ou égale au temps prédéfini (PT), le bit de temporisation est activé. Lorsque l'entrée de validation est désactivée, la valeur en cours de la temporisation TON est effacée. La temporisation TON poursuit le comptage une fois la valeur prédéfinie atteinte ; son exécution s'arrête à la valeur maximale (32767). **(Figure.III.37)**



(Figure.III.37) *Opération de temporisation T37*

III.3.7.9 Opération Chercher dans table (FND) :

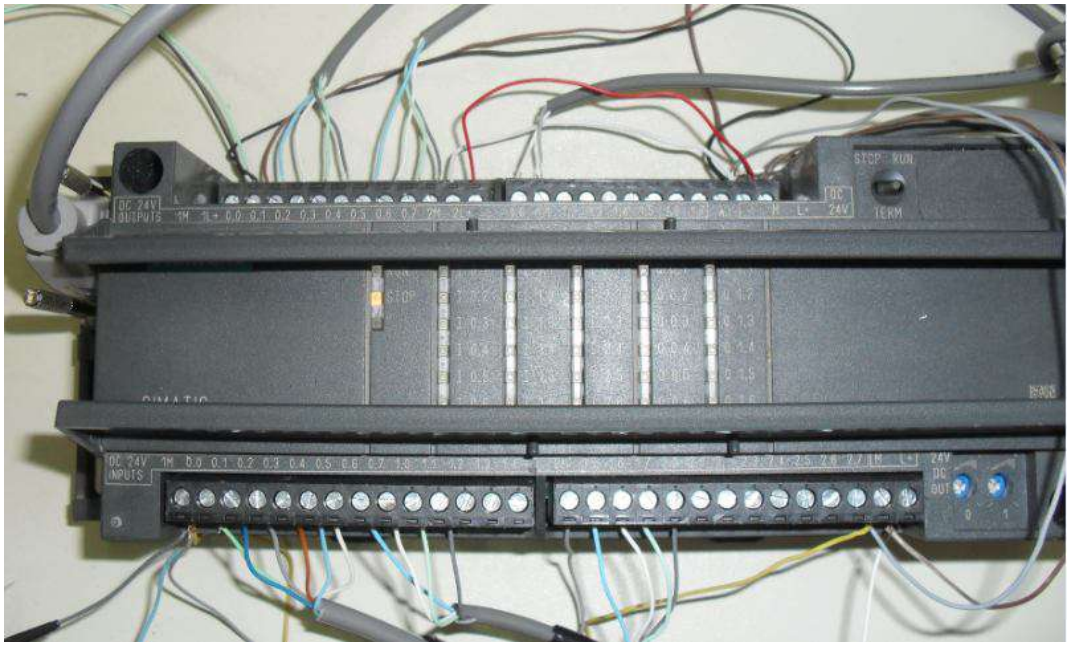
L'opération Chercher dans table (FND) recherche des données correspondant à certains critères dans une table (TBL). L'opération Chercher dans table recherche, dans la table en commençant par l'entrée de table indiquée par INDX, le profil de données (PTN) correspondant au critère défini par CMD. Le paramètre CMD contient une valeur de 1 à 4 associée à =, <>, <, et >, respectivement. Si une entrée correspondant au critère est trouvée, INDX pointe sur cette entrée. Pour chercher l'entrée suivante correspondant au critère, il faut incrémenter INDX avant de rappeler l'opération de recherche. Si une telle entrée s'avère introuvable, la valeur de INDX est égale au décompte des entrées. Une table peut comporter jusqu'à 100 entrées. Les entrées de données (zone de recherche) sont numérotées de 0 à la valeur maximale égale à 99. **(Figure.III.38)**



(Figure.III.38) *Opération Chercher dans table (FND)*

III.3.8 Câblage:

Le câblage a été réalisé comme illustré dans la figure.III.39 ci-dessous :



(Figure.III.39) *Câblage de l'ascenseur avec l'automate*

III.3.9 Description du programme :

Le programme qui a été développé règle les procédures de marche de l'ascenseur

Les différentes positions de la cabine sont stockées dans un registre dans une première étape

Les requêtes sont également identifiées et stockées dans l'ordre dans une table de registre

Le programme procède par la suite a la sélection des différentes requête et effectue une comparaison entre la première requête stockée dans la table et la position de la cabine.

Si la requête est supérieure à la position de la cabine, le moteur prends alors le chemin ascendant (vers le haut).

Si la requête est inférieure à la position de la cabine, le moteur prends alors le chemin descendant (vers le bas).

Si la requête est égale à la position de la cabine, le moteur prend une position statique et le temporisateur commence alors à décompter le temps (5s) au terme duquel il exécute la prochaine requête qui va reprendre le même cycle que la précédente.

- Les lampes de requêtes :

Toutes les requêtes sont matérialisées par l'allumage de lampes qui s'éteignent au fur et à mesure que les requêtes sont exécutées

- Les lampes de position :

Chacune des lampes s'allume pour indiquer la position de la cabine et s'éteint quand la cabine passe à la prochaine position

III.3.10 Scénario proposé :

Pour mieux appréhender le fonctionnement de notre prototype nous proposons le scénario suivant :

Si on appui sur le bouton du 4 étage la lampe 4 s'allume le moteur se met en marche et entraine la cabine jusqu'au 4 étage.

Si on appui sur les boutons 2,3et 1 les lampe 2,3et 1 s'allumes et la cabine est entrainée au niveau 2 (requête 1) marque un arrêt de 5 seconde et reprends son chemin au niveau 3(requête 2) marque un arrêt de 5 seconde et reprends son chemin vers le niveau 1(requête 3).

A noter qu'à chaque fois que la cabine se mets eu mouvement vers une requête la lampe de cette requête s'éteint de manière automatique.

III.4 Conclusion :

Dans le chapitre III nous avons démontré toutes les étapes que nous avons traversées pour la réalisation et la commande de notre ascenseur pédagogique.

Les étapes qu'on a eu à réaliser concernent principalement :

- La partie mécanique
- La partie électrique et
- La partie automatisation

Conclusion générale

L'objectif de ce travail était de faire l'étude, la programmation et la réalisation d'un ascenseur pédagogique sur la base du langage de programmation ladder. Pour arriver à ce but nous avons commencé par la réalisation du prototype d'ascenseur avec ses parties mécanique est électrique. Nous sommes passé ensuite à l'étude du fonctionnement de l'automate et de son logiciel de programmation (STEP-7 200)

L'API est un équipement spécialement conçu pour l'industrie et destiné au pilotage des chaines de montages, robots industriels, machines outils, transport vertical…..ils présentent les avantages suivants :

- Simplification du câblage.
- Modification du programme facile à effectuer par rapport a une logique câblée.
- Enormes possibilités d'exploitation.
- Fiabilité professionnelle.

Ce travail nous a permis d'avoir une idée générale sur le domaine de l'automatisation des systèmes, et de mieux comprendre la réalisation et le fonctionnement des véhicules à transport vertical.

A la fin de notre projet et en dépit de quelques difficultés rencontrées notamment dans la disponibilité de certains outils nous avons eu beaucoup de plaisir à réussir la réalisation de notre projet dans sa totalité.

Nous tenons en outre à signaler que le fruit du travail que nous avons accompli avec beaucoup de soin sera mis à la disposition et au service des futurs étudiants à qui nous souhaitons une totale réussite.