



UNIVERSITE HASSIBA BENBOUALI DE CHLEF

Faculté de Technologie

Département d'Electronique

MEMOIRE DE MASTER

Domaine : SCIENCES ET TECHNOLOGIES

Filière : ELECTRONIQUE

Spécialité : AUTOMATIQUE ET CONTRÔLE INDUSTRIEL

**AUTOMATISATION ET REALISATION A PETITE ECHELLE
(MAQUETTE) D'UNE CHAINE TRANSPORTEUSE DE BRIQUES**

Par

Mohammed MAATOU

Abderrahman BELLAGH

Encadreur :

M. Ahmed AISSAOUI

Maître Assistant «A» à l'UHBC

Chlef, Juin 2016

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

قال تعالى: { يَرْفَعُ اللَّهُ الَّذِينَ آمَنُوا مِنْكُمْ وَالَّذِينَ آمَنُوا وَتُوُوا الْعِلْمَ دَرَجَاتٍ }

[المجادة : 11]

Remerciement

Louange à DIEU le très grand et miséricordieux, le seul et unique qui nous a donné la force et le courage pour terminer nos études et élaborer ce travail.

Avant de commencer la présentation de ce travail, Nous profitons de l'occasion pour remercier toutes les personnes qui ont contribué de près ou de loin à la réalisation de ce projet de fin d'études.

Nous tenons à exprimer nos vifs remerciements pour mon grand et respectueux, **M.Ahmed AISSAOUI**, d'avoir accepté de nous encadrer pour mon projet de fin d'études, ainsi que pour ses précieux conseils et surtout pour nous avoir laissé une grande liberté dans la conception et la rédaction de ce travail, ses remarques pertinentes et son encouragement.

Nous tenons à exprimer nos profondes reconnaissance et toutes nos pensées de gratitude à **M.Boualem FENGAL**, qui nous accompagne de près durant tout ce travail, pour sa disponibilité, pour la confiance qu'il a su nous accorder et les conseils précieux qu'il nous prodigués tout au long de la réalisation de ce projet.

Nos remerciements vont aussi à tous professeurs, enseignants et toutes les personnes qui nous soutenus jusqu'au bout, et qui ne cessent de nous donner des conseils très importants en signe de reconnaissance.

Dédicaces

A nos très chers parents...

Témoignage d'affection et de grande reconnaissance

Que dieu les guides

A toute notre famille

Nos très chers frères et sœurs

A tous nos amis

A tous ceux qu'on aime

RESUME

Ce travail vise à savoir, dans une branche de commande du sujet automatisé, Il est de connaître les divers procédés industriels intervenant dans la production de briques et de la façon de les coordonner. Pour ce faire, nous avons travaillé sur une maquette à échelle réduite pour réaliser une chaîne transporteuse de briques. Clairement à la façon dont la coordination entre le moteur et le travail intervenant mécanique dans le transfert de matériel d'une ligne à une autre. Pour organiser un mouvement à travers le programme LADDER montré que nous avons réalisé au sein de l'automate Schneider, qui à son tour complète la main de l'homme, et expliqué comment connecter les câbles électriques (le câblage) entre les deux zones. La partie de commande (API) et l'autre (partie d'opérative, il est les actionneurs), ceci est appelé le schéma de puissance. Et l'autre schéma de commande est aperçu le câblage de capteurs et détecteurs.

هو معرفة مختلف العمليات الصناعية المتدخلة في

الهدف من هذا العمل

الياجور، أوضحنا من خلاله كيفية

إنتاج الياجور وكيفية التنسيق بينها.

في نقل المواد من خط لآخر، و لتنظيم هذه الحركة جهاز تحكم

التنسيق بين المحركات والعمل الميكانيكي

(Schneider) , قمنا ببرمجته باستخدام لغة البرمجة (LADDER), قمنا بإيضاح مفصل عا

شكل مخطط لكيفية توصيل لكابلات الكهربائية بين المنطقتين (منطقة التحكم ومنطقة العمليات).

Liste des Figures

- Figure 1.1 :** Structure d'un système automatisé.
- Figure 1.2 :** Structure de pré-actionneur électrique.
- Figure 1.3 :** Contacteur.
- Figure 1.4 :** Structure de pré-actionneur pneumatique
- Figure 1.5 :** Fonctionnement de distributeur.
- Figure 1.6 :** Image d'un relais.
- Figure 1.7:** Structure d'un relais.
- Figure 1.8 :** Disjoncteur magnétothermique et leur symbole.
- Figure 1.9 :** Disjoncteur moteurs magnétique.
- Figure 1.10:** Fusible.
- Figure 1.11:** Symbole d'un moteur courant continu.
- Figure 1.12 :** Moteur a courant continue.
- Figure 1.13 :** Représentation d'un moteur asynchrone.
- Figure 1.14:** Stator et rotor d'un moteur.
- Figure 1.15 :** Principe de fonctionnement d'un moteur.
- Figure 1.16 :** Coupe d'un moteur à aimants permanents.
- Figure 1.17 :** Stator.
- Figure 1.18 :** Variation de la vitesse en fonction du nombre de paires de pôles.
- Figure 1.19 :** rotor.
- Figure 1.20 :** Constituants d'un moteur pas à pas.
- Figure 1.21 :** Vérin pneumatique.
- Figure 1.22 :** Vérin simple effet avec son distributeur
- Figure 1.23 :** Vérin double effet avec son distributeur
- Figure 1.24 :** Principe de fonctionnement d'un capteur.
- Figure 1.25 :** Schéma interne de capteur.
- Figure 1.26 :** Fonction logique d'un capteur TOR.
- Figure 1.27 :** Fonction d'un capteur analogique.
- Figure 1.28 :** Fonction d'un capteur numérique.

Figure.1.29: Méthode de choix d'un capteur.

Figure.2.1: Automate programmable industriel.

Figure.2.2: Architecteur d'un API.

Figure.2.3: Exemple d'une carte d'entrées typique d'un API

Figure.2.4: Exemple d'une carte de sortie typique d'un API

Figure.2.5: Alimentation de l'automate.

Figure.2.6: Alimentations des entrées d'automate.

Figure.2.7: Alimentations des sorties de automate.

Figure.2.8: Fonctionnement de l'automate.

Figure.3.1: Exemple d'un programme en Ladder.

Figure 4.1. Schéma synoptique de la maquette

Figure 4.2. Projet de réalisation (maquette).

Figure 4.3 : début du cycle de fonctionnement

Figure 4.4 : Action 1 du cycle de fonctionnement

Figure 4.5 : Action 2 du cycle de fonctionnement

Figure 4.6 : Action 3 du cycle de fonctionnement

Figure 4.7 : Action 4 du cycle de fonctionnement

Figure 4.8 : Action 5 du cycle de fonctionnement

Figure 4.9 : Action 6 du cycle de fonctionnement

Figure 4.10 : Action 7 du cycle de fonctionnement

Figure 4.11 : Action 8 du cycle de fonctionnement

Figure 4.12 : Action 9 du cycle de fonctionnement

Figure 4.13 : Action 10 du cycle de fonctionnement

Figure 4.14 : Action 11 du cycle de fonctionnement

Figure 4.15 : Action 12 du cycle de fonctionnement

Figure 4.16 : Action 13 du cycle de fonctionnement

Figure 4.17 : Schéma de commande

Figure 4.18 : Schéma de puissance

- Figure 4.19** : Grafcet de niveaux 1 (partie 1)
Figure 4.20 : Grafcet de niveaux 1 (partie 2)
Figure 4.21 : Grafcet de niveaux 2 (partie 1)
Figure 4.22 : Grafcet de niveaux 2 (partie 2)
Figure 4.23 : Les temporisations des actions T1 et T2
Figure 4.24 Les temporisations des actions T4 et T5
Figure 4.25 : Les temporisations des actions T6 et T7
Figure 4.26 : Temporisation de l'action T8
Figure 4.27 : Temporisation de l'action T9
Figure 4.28 : Temporisation de l'action TA

Liste des Tableaux

Tableau 1.1 : la vitesse du moteur peut varier en fonction du nombre de paires de pôles.

Tableau. 3.1:les principaux éléments d'un réseau LD.

Tableau 4.1 : Liste du matériel utilisé.

Tableau.4.2: les transitions.

Tableau.4.3: les actions en niveau 1 et niveau 2

Tableau 4.4 : Entrées physiques de l'API

Tableau 4.5 : Sorties physiques de l'API

Sommaire

Introduction Générale	2
CHAPITRE 1: Structure Des Systèmes Automatisés De Production	
1.1. Introduction.....	4
1.2. Objectif d'automatisation	4
1.3. Organisation d'un système automatisé.....	4
1.4. Partie commande.....	5
1.4.1. Automate Programmable Industriel	6
1.4.2. Pré-actionneurs.....	6
1.4.2.1. Pré-actionneurs électriques	6
1.4.2.2 Pré-actionneurs pneumatiques: les distributeurs	8
1.4.2.3. Relais.....	9
1.4.2.4. Protection	11
1.5. Partie opérative	12
1.5.1. Les actionneurs.....	12
1.5.1.1.Actionneur électrique	12
1.5.1.1.1. Les moteurs à courant continu	13
1.5.1.1.2.Moteurs asynchrones.....	14
1.5.1.1.3. Moteur synchrone.....	16
1.5.1.1.4. Les Moteurs pas à pas	19
1.5.1.2. Actionneurs pneumatiques	20
1.5.2. Capteurs.....	21
1.5.2.1. Définition	21
1.5.2.2. Principe.....	22
1.5.2.3. Structure et fonction principale d'un capteur.....	22
1.5.2.4. Principales caractéristiques des capteurs	23
1.5.2.5. Classification des capteurs	24
1.5.2.6. Type de capture	25

1.5.2.7. Critère de choix d'un capteur	26
1.6. Conclusion	27

CHAPITRE 2: Les Automates Programmable Industriel

2.1. Introduction.....	29
2.2. Structure d'un API	29
2.3. Différents types d'API.....	30
2.4. Architecteur d'un l'API	31
2.4.1. Microprocesseur	31
2.4.2. Mémoire	31
2.4.3. Interfaces d'entrées/sorties	32
2.4.4. Alimentation de l'automate programmable industriel	33
2.4.5. Modules périphériques	33
2.5. Câblage de l'automate	34
2.5.1. Alimentation de l'automate.....	34
2.5.2. Alimentations des entrées d'automate.....	35
2.5.3. Alimentations des sorties de automate	35
2.6. Protection de l'automate	36
2.7. Fonctionnement de l'automate.....	36
2.8. Programmation de l'API.....	37
2.9. Critères de choix d'un automate.....	38
2.10. Conclusion	38

CHAPITRE 3: Programmation de L'API

3.1. Introduction	41
3.2. GRAFCET	41
3.2.1. Description du GRAFCET	41
3.2.2. Les règles d'évolution du GRAFCET	44
3.2.3. GRAFCET de niveau 1 et 2	45
3.2.3.1. GRAFCET de niveau 1	45
3.2.3.2. GRAFCET de niveau 2	45
3.3. Langage Ladder.....	46

3.3.1. Principe.....	47
3.3.2. Les composants du langage.....	48
3.4. Logiciel de programmation	48
3.4.1. Zelio soft	48
3.4.2. Programmation sur logiciel Zelio Soft 2	49
3.5. Conclusion	55

CHAPITRE 4: Réalisation d'une Chaîne de Transport

4.1. Introduction	57
4.2. Cahier de charge.....	57
4.3 Liste de matériel utilisé	58
4.4. Description du cycle de fonctionnement	60
4.5. Schémas électriques.....	68
4.6. Grafset.....	70
4.7. Programmation	75
4.8. Conclusion	81
Conclusion Générale.....	83

INTRODUCTION GENERALE

Introduction générale

L'automatique est devenue indispensable dans l'industrie. Il a pour objectif de concevoir et d'étudier les divers automatismes en mettant en œuvre les actionneurs électriques et pneumatiques.

Chaque système automatisé possède une partie commande et une partie opérative. Dans la partie commande, l'automate programmable représente l'élément principal de la machine ou de l'installation, car c'est celui qui renferme le programme et doit procéder à son exécution en fonction de l'état des entrées et des sorties, mais la partie opérative représente en général le moteur ou bien les paramètres gérés.

Afin d'augmenter le rendement, mieux gérer et réaliser les tâches répétitives, on utilise des automates programmables industriels. Notre contribution intervient au niveau de l'automatisation partielle, d'une chaîne de transport dans une briqueterie, à l'aide d'un automate. Il nous a fallu collecter toutes les données et les liaisons entre les différents relais, acquérir les outils et méthodes afin de gérer de façon optimale un projet d'automatisme industriel.

Notre travail consiste à réaliser à échelle réduite une chaîne de transport de brique qui est une partie d'une unité de production de brique.

Ce mémoire se compose de quatre chapitres qui sont exposés de manière séquentielle :

- le chapitre 1 est consacré à la compréhension des systèmes automatisés de production et la présentation des différentes parties et appareillages d'un système automatisé,
- le chapitre 2 est destiné à la définition et la présentation des différents types d'automate. Ensuite, nous présentons l'architecture, la protection et les critères de choix d'automate,
- les langages de programmation et la programmation d'un automate font l'objet du chapitre 3 qui se termine avec un exemple de programmation sous LADDER,
- le chapitre 4 est dédié à la réalisation de notre maquette et de son câblage et à la programmation de l'automate choisi.

En fin, notre travail est terminé par une conclusion.

CHAPITRE I :

STRUCTURE DES SYSTEMES

AUTOMATISES DE PRODUCTION

1.1. Introduction

Les premiers systèmes conçus ont été des systèmes non mécanisés, c'est à dire des systèmes pour lesquels l'opérateur apporte non seulement son savoir-faire mais aussi l'énergie nécessaire à la modification de la matière d'œuvre. Les effecteurs sont les constituants qui permettent de modifier la matière d'œuvre. Une des premiers objectifs de l'automatisation est de supprimer les taches pénibles, diminuer l'effort que doit fournir.

La première évolution des systèmes est la mécanisation qui permet de limiter l'énergie apportée par l'opérateur. L'énergie est fournie par le milieu extérieur au système. L'opérateur autorise ou non le passage de l'énergie pour permettre la modification de la matière d'œuvre.

1.2. Objectif d'automatisation

Les objectifs poursuivis par l'automatisation peuvent être assez variés. On peut citer quelques-uns : la recherche de coûts plus bas par réduction des frais de main-d'œuvre et d'économie de matière et d'énergie, la suppression des travaux dangereux ou pénibles, l'amélioration des conditions de travail et la réalisation des opérations impossibles à contrôler manuellement. La compétitivité d'un produit final peut être définie comme sa capacité à être bien vendu. La compétitivité résulte essentiellement des résultats obtenus sur les facteurs suivants : coût, qualité, innovation, disponibilité.

1.3. Organisation d'un système automatisé

Tout système automatisé peut se décomposer selon le schéma ci-dessous en trois parties : post de contrôle, partie de commande et partie opérative (voir figure 1.1). [01]

Le poste de contrôle.

Le poste de contrôle est composé des pupitres de commande et de signalisation. Il permet à l'opérateur de commander le système (marche, arrêt, départ cycle ...). Il permet également de visualiser les différents états du système à l'aide de voyants, de terminal de dialogue ou d'interface homme machine (IHM) de type écran, clavier ou imprimante.

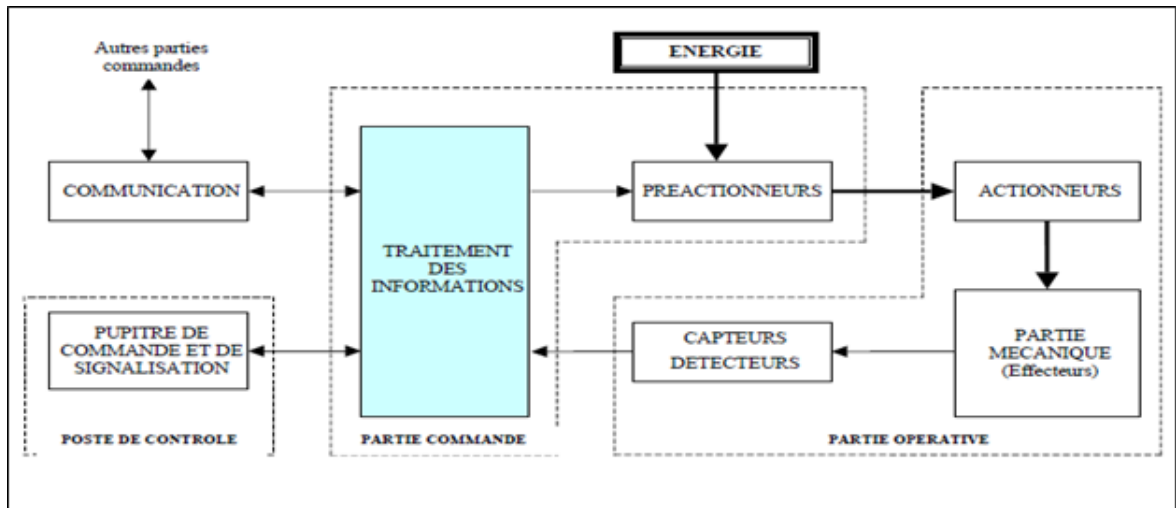


Figure 1.1 : Structure d'un système automatisé. [01]

La Partie commande

La partie de commande a pour tâche de donner les ordres de fonctionnement à la partie opérative. Elle est constituée de pré-actionneurs qui permettent de commander les actionneurs. Ils assurent le transfert d'énergie entre la source de puissance et les actionneurs. Ces pré-actionneurs sont commandés à leur tour par le bloc de traitement des informations. Celui-ci reçoit les consignes du pupitre de commande (opérateur) et les informations de la partie opérative transmises par les capteurs/détecteurs. En fonction de ces consignes et de son programme de gestion des tâches implanté dans un automate programmable (logique programmée) ou réalisé par des relais (logique câblée). Elle va commander les pré-actionneurs et renvoyer des informations au pupitre de signalisation ou à d'autres systèmes de commande et/ou de supervision en utilisant un réseau et un protocole de communication.

La Partie opérative

La partie opérative agit sur la matière d'œuvre afin de lui donner sa valeur ajoutée. Elle comporte:

- Les actionneurs : (moteurs, vérins) qui agissent sur la partie mécanique du système qui agit à son tour sur la matière d'œuvre,
- Les capteurs / détecteurs : permettent d'acquérir les divers états du système,

1.4. Partie commande

La partie commande est constitué de l'automate programmable industriel (API) et des pré-actionneurs

1.4.1. Automate Programmable Industriel

L'Automate Programmable Industriel est un appareil électronique programmable, adapté à l'environnement industriel, qui réalise des fonctions d'automatisme pour assurer la commande de pré-actionneurs et d'actionneurs à partir d'informations logiques, analogiques ou numériques.

1.4.2. Pré-actionneurs

Le pré actionneurs peut être de type électrique ou pneumatique

1.4.2.1. Pré-actionneurs électriques

Les pré-actionneurs électriques sont des contacteurs qui permettent le passage ou l'interruption de l'énergie électrique (voir figure 1.3). Ils sont appelés pré-actionneurs car ils se trouvent avant les actionneurs. Ces derniers peuvent être commandés à distance au moyen de contacts actionnés manuellement (bouton poussoir) ou automatiquement (asservi à une grandeur physique : pression, température, vitesse, etc.).

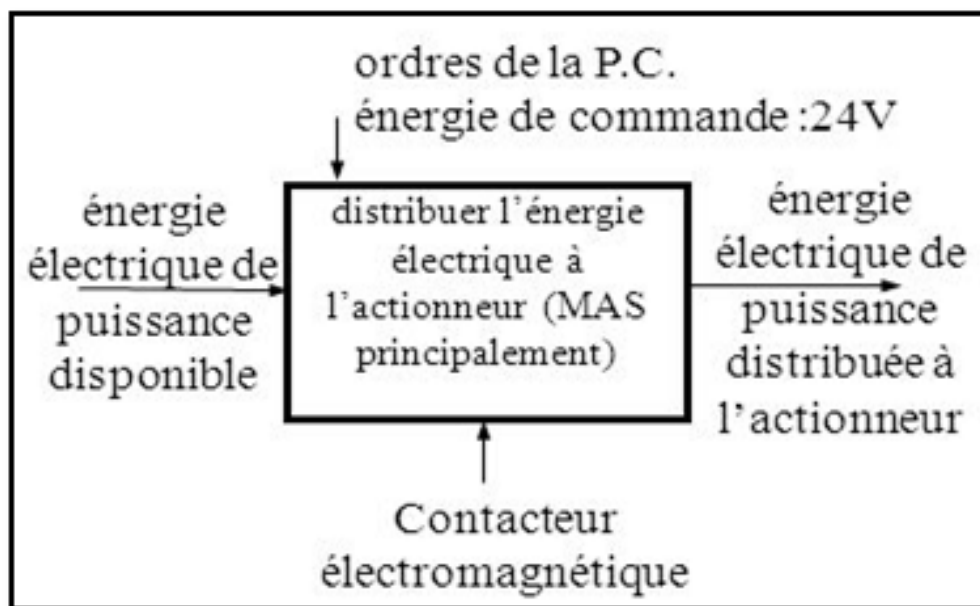


Figure. 1.2 : Structure de pré-actionneur électrique.

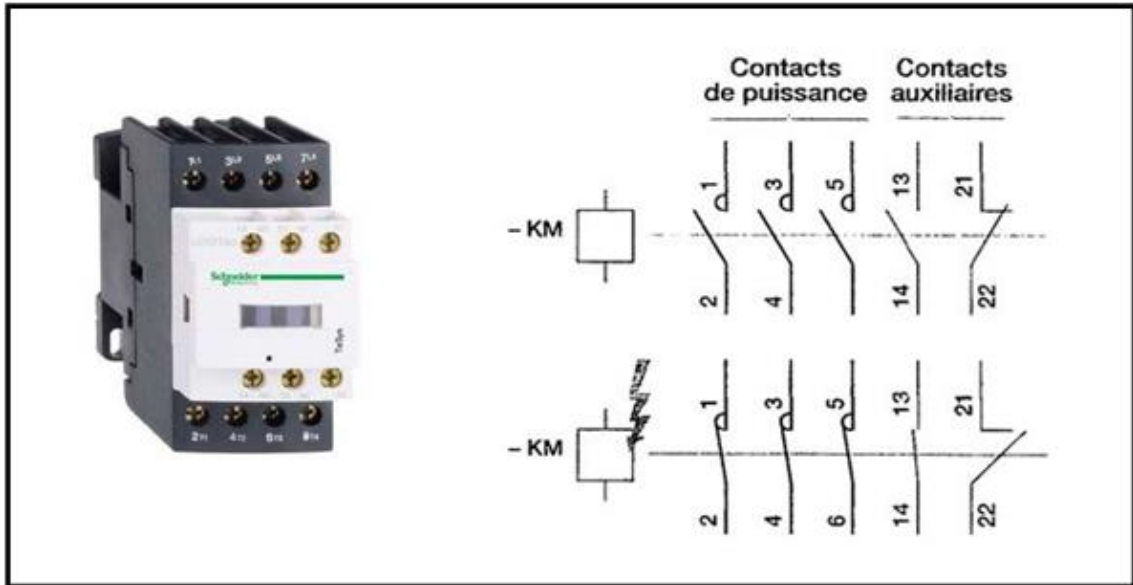


Figure 1.3 : Contacteur.

Nous venons de dire précédemment que le contacteur était un appareil possédant un pouvoir de coupure. Il est important de savoir que la séparation de deux contacts sous tension (c'est le cas des pôles principaux d'un contacteur) provoque généralement la formation d'un arc électrique qui doit être rapidement éteint puisque :

- le courant électrique continue de circuler tant que l'arc électrique n'est pas éteint d'où le risque de ne pouvoir mettre hors service l'installation,
- l'arc s'accompagne d'un dégagement de chaleur important qui provoque l'usure, voire la destruction de l'appareil de coupure (réduction de l'endurance électrique, risque de soudure des contacts),
- l'arc est dangereux de par sa mobilité. Il peut provoquer l'amorçage entre phase-terre ou phase-phase et le risque d'électrocution des personnels.

La capacité à "souffler" cet arc électrique (à le supprimer) sera donc donnée par le pouvoir de coupure. Un pouvoir de coupure de 10 KA (kilo ampère : 10000 A) permettra de couper un circuit où circule 10000 A max et de supprimer l'arc électrique qui résulte de l'ouverture de ce dernier.

Sans pouvoir de coupure, pas de suppression de l'arc électrique et donc « pas de possibilité d'ouverture du circuit » la fonction de commande n'est pas réalisée. Les constructeurs utilisent plusieurs procédés de suppression de l'arc. Parmi les plus répandus nous trouverons:

- allongement de l'arc électrique,

- utilisation de matériaux anti-arc (cuivre, bronze, zinc),
- guidage l'arc sur des contacts autres que les contacts utilisés dans l'installation,
- soufflage magnétique.

Constitution

Un contacteur (pré actionneur) est constitué:

- des pôles principaux de puissance,
- des contacts auxiliaires (possibilité d'ajouter au contacteur un bloc de contacts auxiliaires instantanés ou temporisés,
- une armature fixe et un autre mobile. un ressort de rappel,
- un circuit magnétique feuilleté de manière à réduire les pertes par courant de Foucault (dus à la présence d'un flux d'induction magnétique alternatif),
- une bobine (insérée dans le circuit de commande). Si la bobine est alimentée en courant alternatif le courant d'appel sur le circuit de commande lors de la fermeture du contacteur peut atteindre 6 à 10 fois le courant de maintien (utile pour le choix du transformateur de commande...). Une bobine peut être alimentée en courant continu (faire le bon choix lors de la commande du matériel) ce qui accroît la force d'attraction de l'électro-aimant constitué par la bobine et l'armature fixe,
- une "spire de franger" ou "bague de déphasage" qui évite les vibrations dues à l'alimentation en courant alternatif de la bobine de contacteur.

Caractéristique et choix d'un contacteur

Les caractéristiques d'un contacteur sont :

- la tension d'emploi assignée,
- courant d'emploi assigné,
- fréquence assignée,
- catégorie d'emploi,
- facteur de marche,
- fréquence de manœuvre, endurance électrique.

1.4.2.2 Pré-actionneurs pneumatiques: les distributeurs

Ce sont des constitués chargés de distribuer l'énergie pneumatique vers les actionneurs pneumatique sur ordre constituant de commande. Ces ordres supportés par un signal électrique en très base.

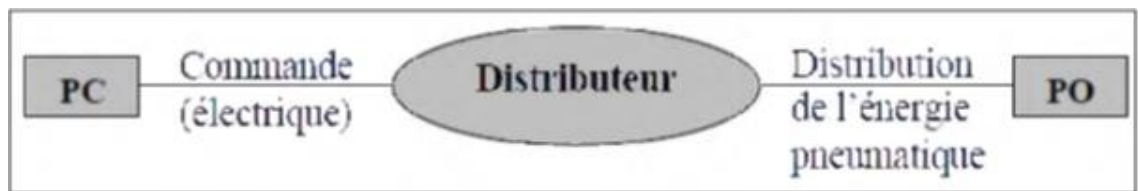


Figure 1.4 : Structure de pré-actionneur pneumatique.

Ils permettent de commuter et contrôler la circulation des fluides sous pression et assurent diverses fonctions :

- Contrôle de mouvement de la tige d'un vérin,
- Choisir le sens de circulation d'un fluide,
- Exécuter des fonctions logiques,
- Démarrer ou arrêter la circulation d'un fluide,
- Etre des capteurs de position (pressostat, vacuostat).

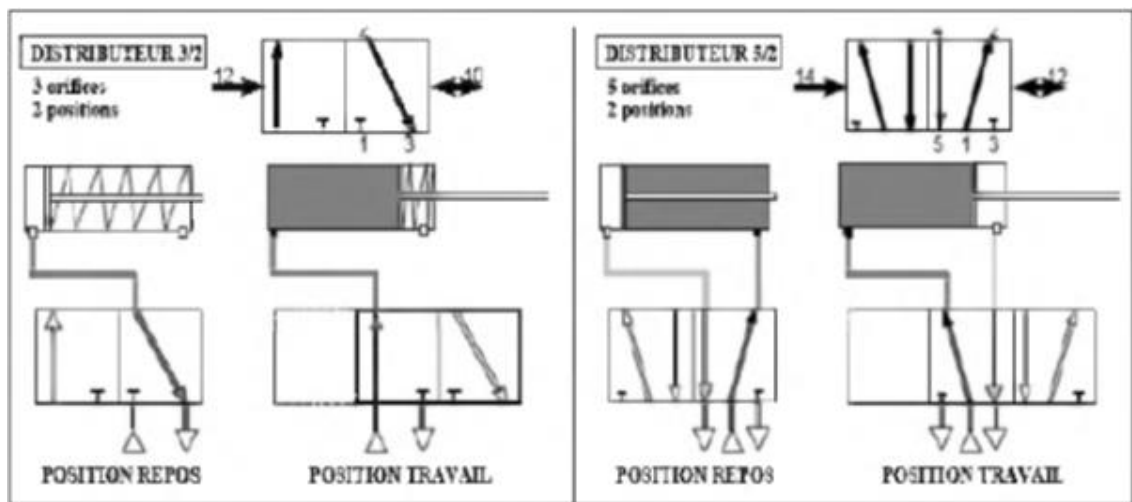


Figure 1.5: Fonctionnement de distributeur.

Caractéristique de choix des distributeurs

- Par les nombre des orifices 2, 3,4 ou 5,
- Par les nombre de distribution ou position : 2 ou 3,
- Par le type de commande de pilotage assurant le changement de position :
 - Simple pilotage avec rappel de ressort.
 - Ou double pilotage.
- Par la technologie de pilotage.

I.4.2.3. Relais

Comme son nom l'indique, il sert en tout premier lieu à « relayer », c'est-à-dire à faire une transition entre un courant faible et un courant fort. Mais il sert également à commander

plusieurs organes simultanément grâce à ses multiples contacts synchronisés. Il permet également la transition entre deux sources différentes en isolant ces dernières. Il autorise des temporisations, des verrouillages, des impulsions ... Bref, les fonctions d'un relais sont aussi nombreuses que différentes.



Figure 1.6 : Image d'un relais.

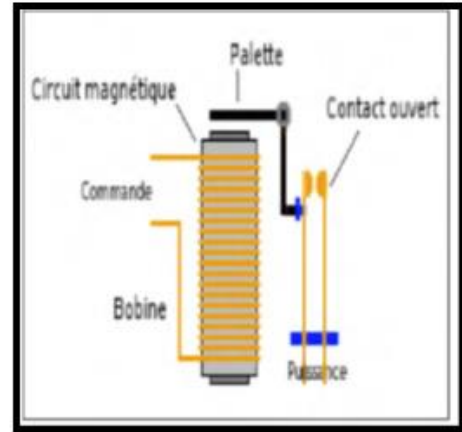


Figure 1.7: Structure d'un relais.

Constitution

Un Relais standard est constitué d'une bobine ou solénoïde qui lorsqu'elle est sous tension attire par un phénomène électromagnétique une structure ferromagnétique qui déplace des contacts (voir figure 1.7).

Caractéristiques

Un relais est caractérisé par :

- La tension de sa bobine de commande 5v à 220v,
- Le pouvoir de coupure de ses contacts qui est exprimé en ampère, 0.1A à 50A c'est le courant maximal qui pourra traverser les contacts,
- Le nombre de contacts souhaités.
- Son emplacement, circuit imprimé à visser, embrochage a soudé,
- Le type de courant de sa bobine, en générale du continu,
- La tension d'isolement entre la bobine et les contacts,
- La gamme de temps pour un relais temporisé,
- Son ambiance, vibration, humidité, poussière, température,

Différentes type de relais

Relais monostable

C'est le plus courant des relais, lorsque sa bobine est sous tension, l'armature mobile actionne les contacts qui changent d'état. Lorsque le courant cesse, l'armature revient à la position initiale ainsi que les contacts.

Relais bistable

Ce relais comporte généralement deux bobines montées en opposition. La mise sous tension d'une bobine déplace l'armature mobile et ses contacts qui restent en position par un système magnétique ou mécanique quand la bobine n'est plus alimentée. Pour changer la position il faut alimenter brièvement l'autre bobine.

1.4.2.4. Protection

Disjoncteur magnétothermique: destiné pour la protection contre les surcharges électriques et les courts circuits.



Figure 1.8 : Disjoncteur magnétothermique et leur symbole.

Disjoncteurs moteurs magnétiques : pour la protection contre les courts circuits

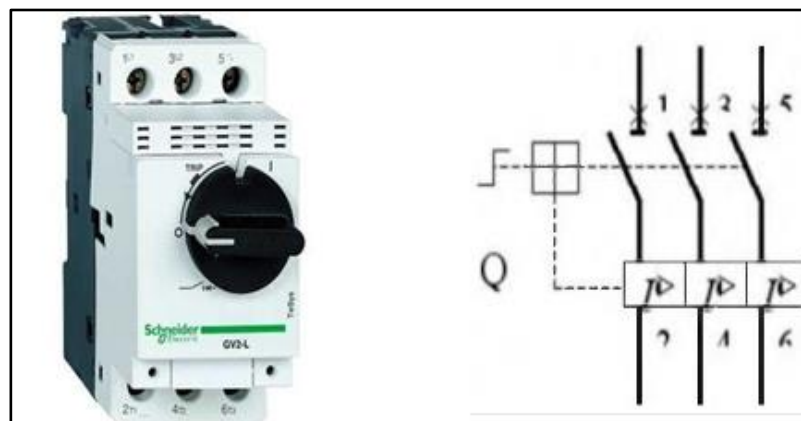


Figure 1.9 : Disjoncteur moteurs magnétique.

Fusible : pour la protection contre les courts circuits.



Figure 1.10: Fusible.

Les fusibles existent en deux versions : fusion rapide (gG) et fusion lente (aM). Les fusibles à fusion lente sont utilisés lorsque le circuit doit supporter une surintensité au démarrage (cas du moteur électrique ou du transformateur).

1.5. Partie opérative

La partie opérative est l'ensemble des moyens techniques qui effectuent directement le processus de transformation de la matière d'œuvre à partir des ordres fournis par la partie commande et l'opérateur. Les informations circulent d'une partie à l'autre par l'intermédiaire d'interfaces. Elle regroupe l'ensemble des opérateurs techniques qui assurent la production des effets utiles lesquels le système automatisé a été conçu. On retrouve dans la partie opérative les actionneurs et les capteurs.

1.5.1. Les actionneurs

Les actionneurs sont des éléments de la partie opérative qui reçoivent de l'énergie (électrique ou pneumatique) pour la transformer en énergie utilisable (mécanique) par le système. Ils exécutent les ordres reçus en agissant sur le système ou son environnement. Ces actionneurs s'appartiennent à trois technologies : électrique, pneumatique et hydraulique.

1.5. 1.1.Actionneur électrique

En fonction de la nature de l'énergie issue de la conversion effectuée par l'actionneur, on distingue différents types d'actionneurs électriques, selon la conversion de l'énergie électrique.

1.5.1.1.1. Les moteurs à courant continu

Il est surtout utilisé pour la traction de véhicules : chariots élévateurs, chariots filoguidés et pour obtenir les déplacements des organes de machines à commande numérique.

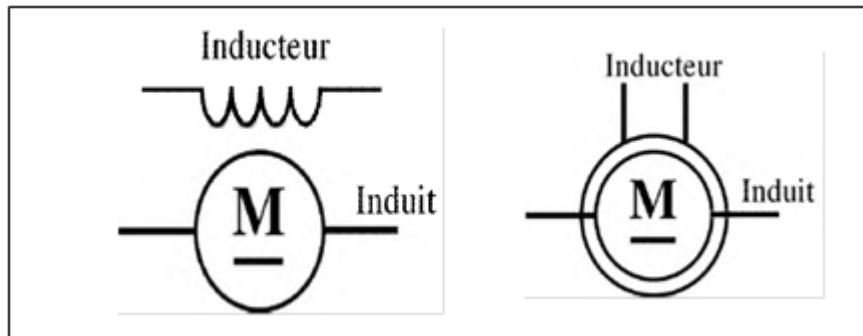


Figure 1.11 : Symbole d'un moteur courant continu.

Constitution

Le moteur comprend,

- un circuit magnétique comportant une partie fixe, le stator, une partie tournant, le rotor et l'entrefer l'espace entre les deux parties,
- une source de champ magnétique nommée l'inducteur (le stator) créée par un bobinage,
- un circuit électrique induit (le rotor) subit les effets de ce champ magnétique le collecteur et les balais permettent d'accéder au circuit électrique.

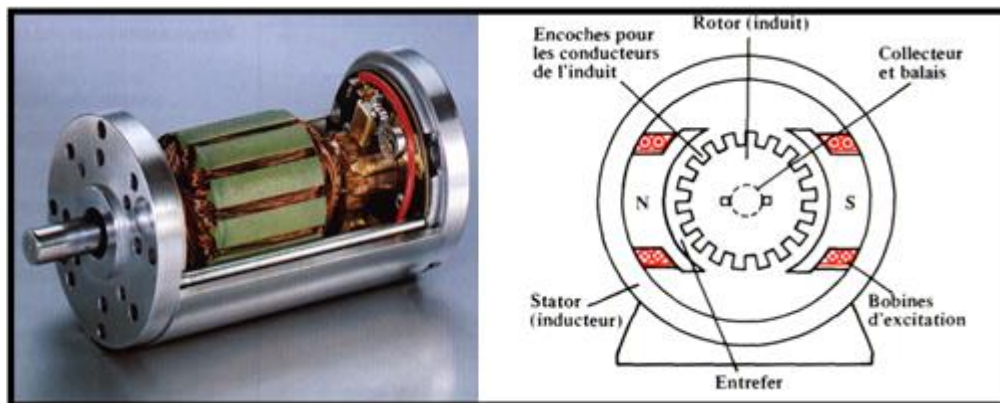


Figure 1.12 : Moteur à courant continu.

Le sens de rotation il est défini par les polarités de l'inducteur et de l'induit. Pour changer le sens de rotation de ce moteur, il suffit d'inverser la polarité de l'inducteur ou de l'induit.

Les types de moteur à courant continu

De construction, les moteurs à courant continu peuvent être :

- à excitation indépendante : inducteur et induit sont séparés,
- des moteurs « série » : inducteur et induit sont en série,

- des moteurs « dérivation » : inducteur et induit sont en parallèle.

Contrôler un moteur à courant continu

La boîte à bornes d'un moteur à courant continu comporte généralement quatre bornes : deux pour l'inducteur et deux pour l'induit. Elles sont généralement de dimension et/ou de couleurs différentes.

Si le moteur est à excitation indépendante ou en dérivation :

- la résistance entre les bornes de l'inducteur est de l'ordre de la centaine d'ohms,
- la résistance aux bornes de l'induit est d'environ 1Ω .

Si le moteur est un moteur série, l'inducteur et l'induit ont des résistances qui sont du même ordre de grandeur, environ 1Ω .

1.5. 1.1.2. Moteurs asynchrones

Transformer l'énergie électrique (courant alternative) en l'énergie mécanique.



Figure 1.13 : Représentation d'un moteur asynchrone.

Le moteur asynchrone, comme le moteur à courant continu, comporte deux parties (voir figure 1.14):

- Une partie fixe, le stator,
- Une partie mobile, le rotor,

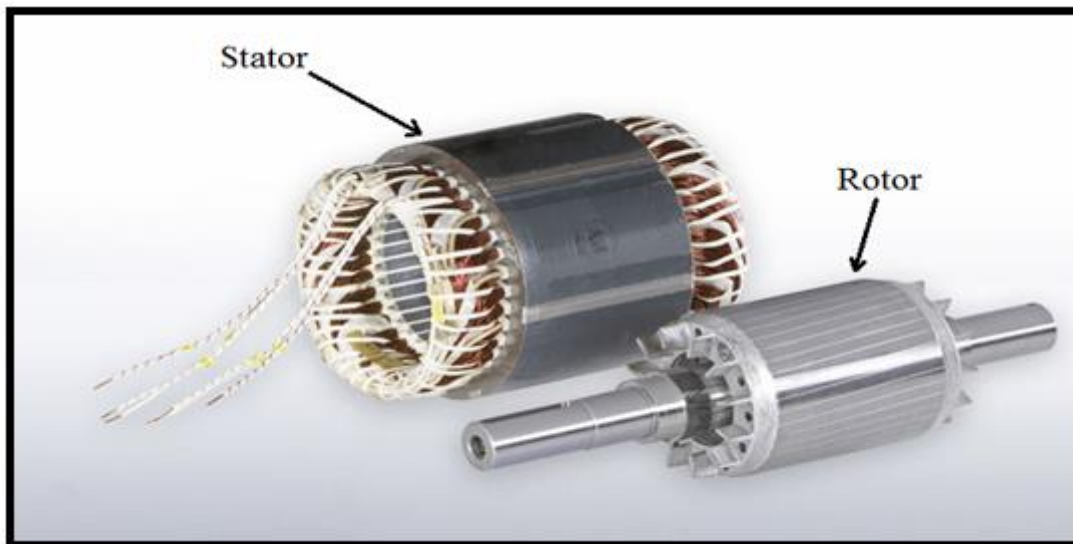


Figure 1.14: Stator et rotor d'un moteur asynchrone.

Le circuit électrique de stator est composé de trois enroulements qui sont reliés à la plaque à borne d'un moteur.

Le fonctionnement du moteur asynchrone est illustré par la figure 1.15.

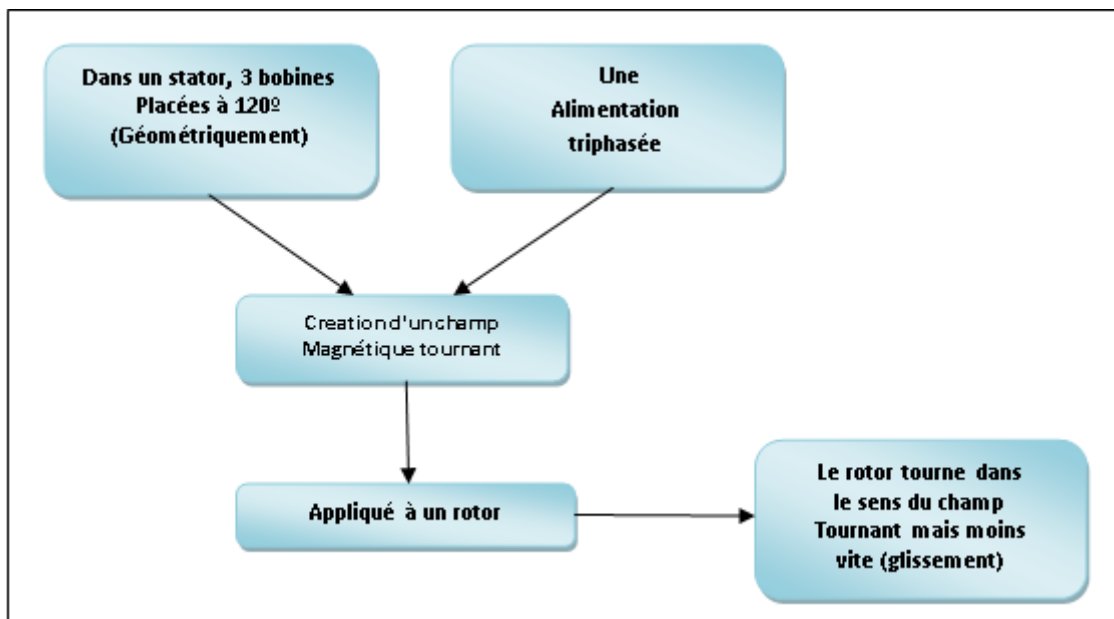


Figure 1.15 : Principe de fonctionnement d'un moteur.

Critères de choix

Pour choisir un moteur, on doit rechercher ses principales caractéristiques:

Caractéristiques :

- la tension d'alimentation
- la puissance nominale

exemple :

220/380 volts
17 kilowatts

- l'intensité nominale 60 ampère
- la vitesse nominale 1427 tour par minute

Mode de démarrage

Il y a plusieurs modes de démarrage des moteurs, les plus utilisés sont :

- **le démarrage direct**

Les enroulements du stator sont directement alimentés par la pleine tension du réseau. Cela engendre un très fort courant au démarrage qui peut atteindre jusqu'à dix fois le courant nominal. Ce mode de démarrage est réservé aux moteurs à rotor en court-circuit, et surtout à faible puissance 5 kilowatts, par exemple.

- **démarrage étoile/triangle**

Pour réduire le courant de démarrage, on démarre avec le couplage Y. Puis au bout de 3 à 5 secondes, on utilise un couplage triangle pour alimenter les enroulements avec la pleine tension du réseau. Ce mode de démarrage est réservé aux machines démarrant à vide ou à couple résistant faible. En général la puissance est 50 kilowatts.

Si une machine possède sur sa plaque signalétique deux tensions, pour utiliser le couplage et donc le démarrage Y, il faut que la plus petite des deux tensions soit égale à la tension du réseau. Sinon il est interdit d'utiliser le couplage triangle.

1.5.1.1.3. Moteur synchrone

Le moteur synchrone se compose, comme le moteur asynchrone, d'un stator et d'un rotor séparés par un entrefer. La seule différence se situe au niveau de la conception du rotor. La figure 1.16 montre un rotor à pôles saillants constitués d'aimants permanents ou d'électro-aimants alimentés en courant continu.

Après le démarrage, le moteur tourne en synchronisme avec le champ tournant. A vide les axes des pôles du champ tournant et du rotor sont confondus. En charge, les axes sont légèrement décalés. La vitesse du moteur synchrone est constante quelle que soit la charge.

On notera aussi que:

- la charge (le système d'ascenseur) ne doit pas dépasser l'effort de démarrage entre le rotor et le champ tournant,
- le couple moteur est proportionnel à la tension à ses bornes.

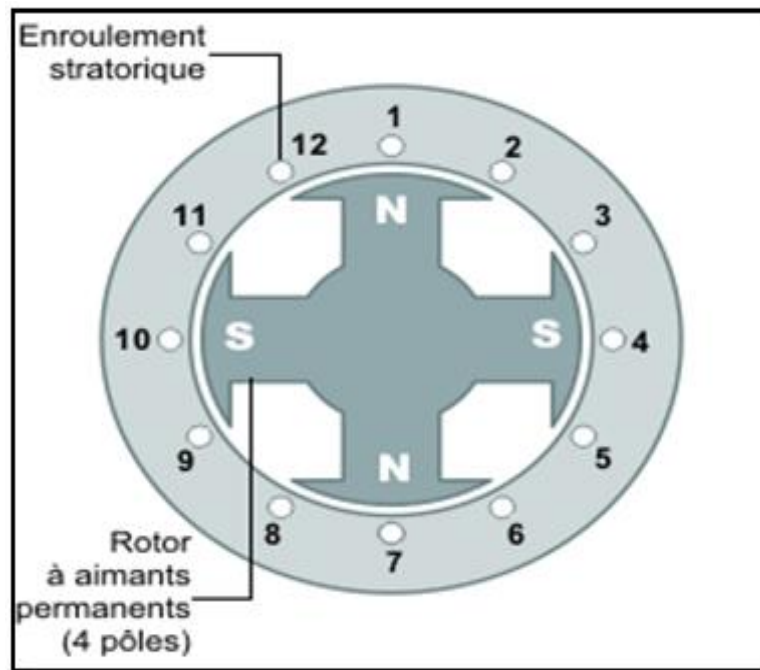


Figure 1.16 : Coupe d'un moteur à aimants permanents.

Les avantages

- le moteur synchrone peut travailler avec un facteur de puissance proche de l'unité ($\cos\phi \sim 1$). Il contribue donc à redresser le $\cos\phi$ global de l'installation électrique,
- la vitesse du moteur synchrone est constante quelle que soit la charge (intéressant dans le cas des ascenseurs),
- Il peut supporter des chutes de tension important sans décrocher.

Les inconvénients

S'il n'est pas associé à un variateur de vitesse, il a des difficultés à démarrer, il peut décrocher en cas de forte charge.

Le stator

Le stator est la partie statique du moteur synchrone, voir figure 1.17. Il s'apparente fort au stator des moteurs asynchrone. Il se compose principalement :

- de la carcasse,
- des paliers,
- des flasques des paliers,
- du ventilateur de refroidissement,
- le capot protégeant le ventilateur.

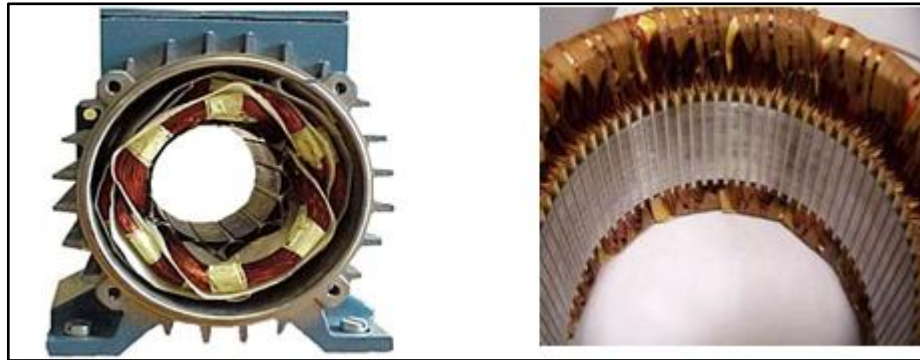


Figure 1.17: Stator.

L'intérieur du stator comprend essentiellement :

- un noyau en fer feuilleté de manière à canaliser le flux magnétique,
- les enroulements (ou bobinage en cuivre) des trois phases logés dans les encoches du noyau.

Dans un moteur triphasé, les enroulements sont au nombre minimum de trois décalés l'un de l'autre de 120° .

Lorsque les enroulements du stator sont parcourus par un courant triphasé, ceux-ci produisent un champ magnétique tournant à la vitesse de synchronisme. La vitesse de synchronisme est fonction de la fréquence du réseau d'alimentation et du nombre de paires de pôles. Vu que la fréquence est fixe, la vitesse du moteur peut varier en fonction du nombre de paires de pôles.

Tableau 1.1 : la vitesse du moteur peut varier en fonction du nombre de paires de pôles.

Paires de pôles	1	2	3	4	6
Nombre de pôles	2	4	6	8	12
n_0 [tr/min]	3000	1500	1000	750	500

Le rotor

Le rotor est la partie mobile du moteur synchrone. Couplé mécaniquement à un treuil d'ascenseur par exemple, il va créer un couple moteur capable de fournir un travail de montée et de descente de la cabine d'ascenseur. Il se compose essentiellement d'une succession de pôles Nord et Sud intercalés sous forme d'aimants permanents ou de bobines d'excitation parcourues par un courant continu. On distingue donc deux types de moteurs synchrones:

- à aimants permanents,
- à rotor bobines.

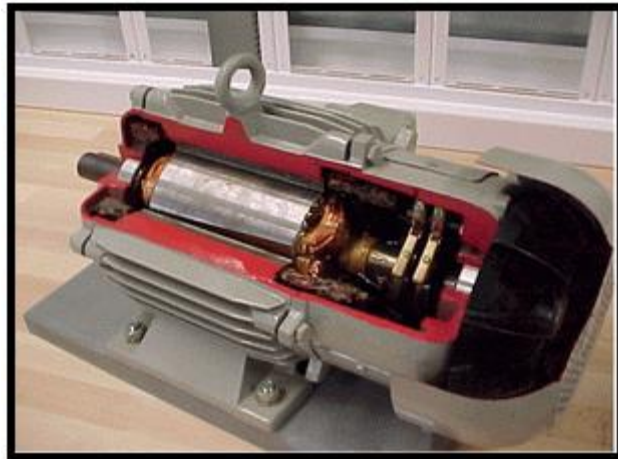


Figure 1.18: rotor.

Pilotage de la vitesse de rotation

Le pilotage de la vitesse de rotation du moteur synchrone est essentiel pour beaucoup d'applications. La relation suivante permet de cerner quels sont les paramètres qui peuvent influencer la vitesse de rotation.

On a:

$$n_0 = n$$

Avec,

- n_0 : vitesse du champ tournant [tr/min].
- n : la vitesse de rotation de l'arbre du moteur [tr/min].

Où:

$$n = 60 f / p$$

Avec,

- f : fréquence du réseau [Hz],
- p : le nombre de paires de pôles du stator.

On peut donc piloter la vitesse de rotation en intervenant sur :

- le nombre de paires de pôles (moteur à nombre de pôles variable),
- la fréquence du réseau.

1.5.1.1.4. Les Moteurs pas à pas

Le moteur pas à pas permet de transformer une impulsion électrique en un mouvement angulaire. Ce type de moteur est très courant dans tous les dispositifs où l'on souhaite faire du contrôle de vitesse ou de position, comme par exemple les imprimantes.

Le stator est constitué de bobines qui sont alimentées, à tour de rôle, en courant continu par une carte électronique. Le rotor est un croisillon, en métal ferreux ne conservant pas le magnétisme. Si on compte électroniquement les impulsions envoyées aux bobines on sait,

connaissant le pas, le nombre de rotations que le rotor a effectuées. Représentation du moteur pas à pas. La figure 1.19 représente les constituants du moteur pas à pas.

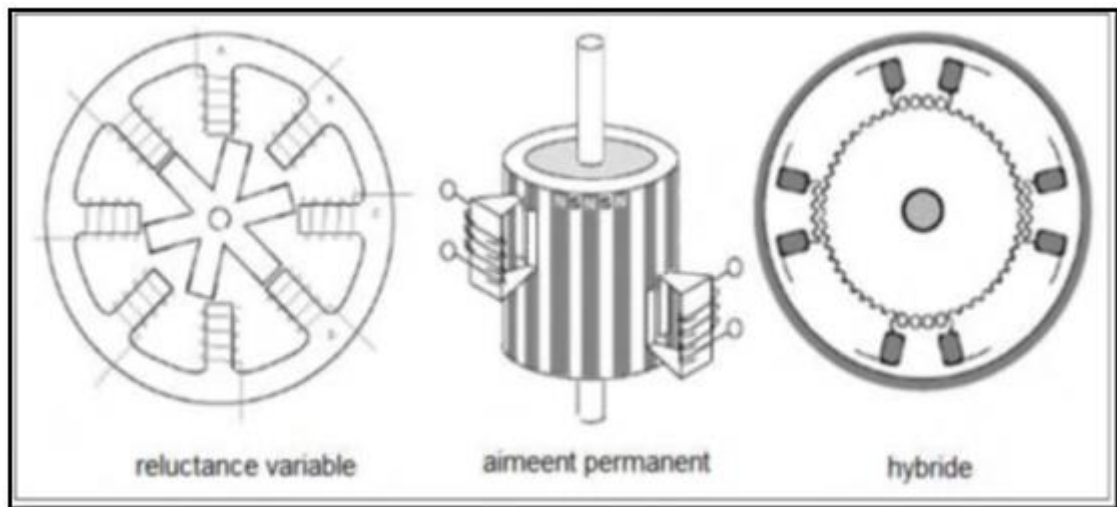


Figure 1.19 : Constituants d'un moteur pas à pas.

1.5.1.2. Actionneurs pneumatiques

Les actionneurs pneumatiques les plus répandus sont les vérins pneumatiques linéaires. Ils transforment l'énergie pneumatique (pression, débit) en énergie mécanique (effort, vitesse).



Figure 1.20 : Vérin pneumatique.

Il existe différents types de vérins. Les deux principaux sont :

Le vérin simple effet : c'est un composant monostable (Stable dans une seule position). Ce type de vérin ne peut produire un effort significatif que dans un seul sens, le rappel de tige est assuré par un ressort (voir figure 1.21).

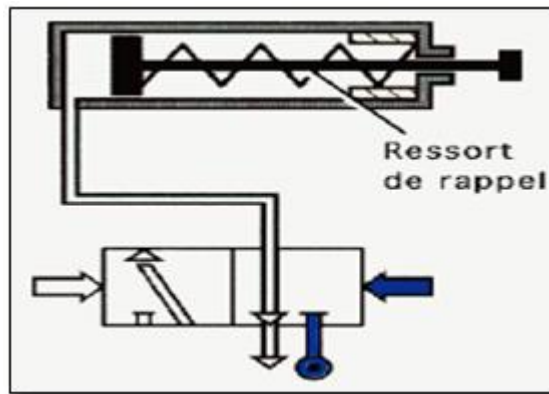


Figure 1.21: Vérin simple effet avec son distributeur

Le vérin double effet : le vérin double effet est un composant bistable (Stable dans deux positions). Ce type de vérin peut produire un effort significatif dans les deux sens, le rappel de tige est obtenu par inversion de l'alimentation des deux chambres (voir figure 1.22).

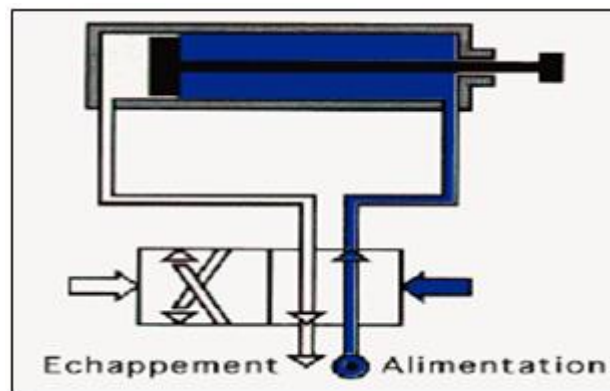


Figure 1.22 : Vérin double effet avec son distributeur

1.5.2. Capteurs

1.5.2.1. Définition

Capteur : Un capteur est un dispositif transformant l'état d'une grandeur physique observée en une grandeur utilisable.

Transmetteur : Elément qui transmet un signal. Il peut être un assemble d'éléments (capteur+ amplificateur).

Transducteur : Elément qui sert à transformer, suivant une loi, la grandeur mesurée en une autre grandeur.

Chaîne de mesure : Suite d'éléments allant du capteur au traitement de l'information.

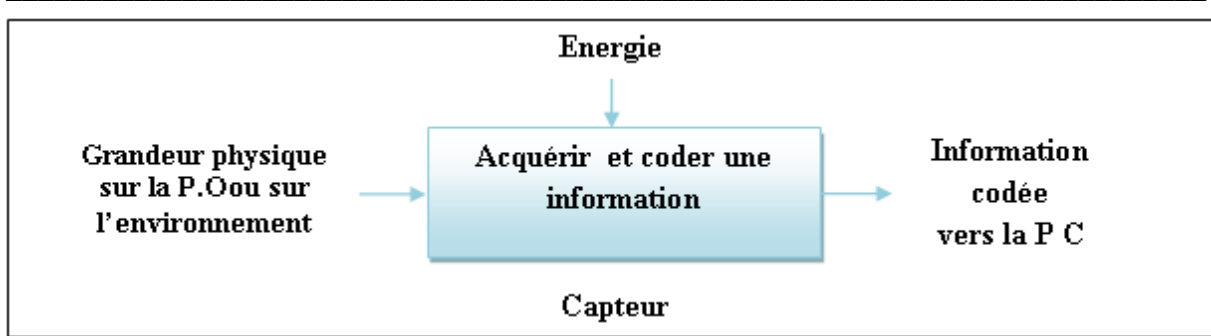


Figure 1.23 : Principe de fonctionnement d'un capteur.

1.5.2.2. Principe

Le capteur est chargé de prélever une grandeur physique à mesurer et de la transformer en une grandeur exploitable.

- la grandeur physique à mesurer, souvent appelée « mésurande », n'est en général pas directement utilisable. Elle constitue le signal d'entrée (ou stimulus) du capteur,
- la grandeur exploitable est souvent de nature électrique. Elle constitue le signal de mesure (ou signal de sortie, ou réponse) du capteur. Elle est une représentation de la grandeur à mesurer et doit être indépendante des autres grandeurs pouvant influencer sur le capteur. Ces grandeurs étrangères portent le nom de grandeurs d'influence,
- les grandeurs d'influence sont des grandeurs étrangères qui, selon leur nature et leur importance, peuvent provoquer des perturbations sur le capteur. C'est donc une cause d'erreurs agissant sur le signal de sortie. Citons en particulier :
 - la température
 - la pression environnement
 - les vibrations
 - l'humidité, la projection d'eau
 - les ambiances corrosives
 - les perturbations électromagnétiques
 - les accélérations et la pesanteur
 - l'alimentation électrique du capteur

1.5.2.3. Structure et fonction principale d'un capteur

De façon simple, un capteur peut être défini comme un transducteur convertissant une grandeur physique en un signal électrique. Cette transformation peut être directe dans quelques cas simples (exemple : thermocouple). Mais en réalité, la technologie des capteurs

fait souvent appel à plusieurs conversions de phénomène physique avant d'arriver au signal de sortie.

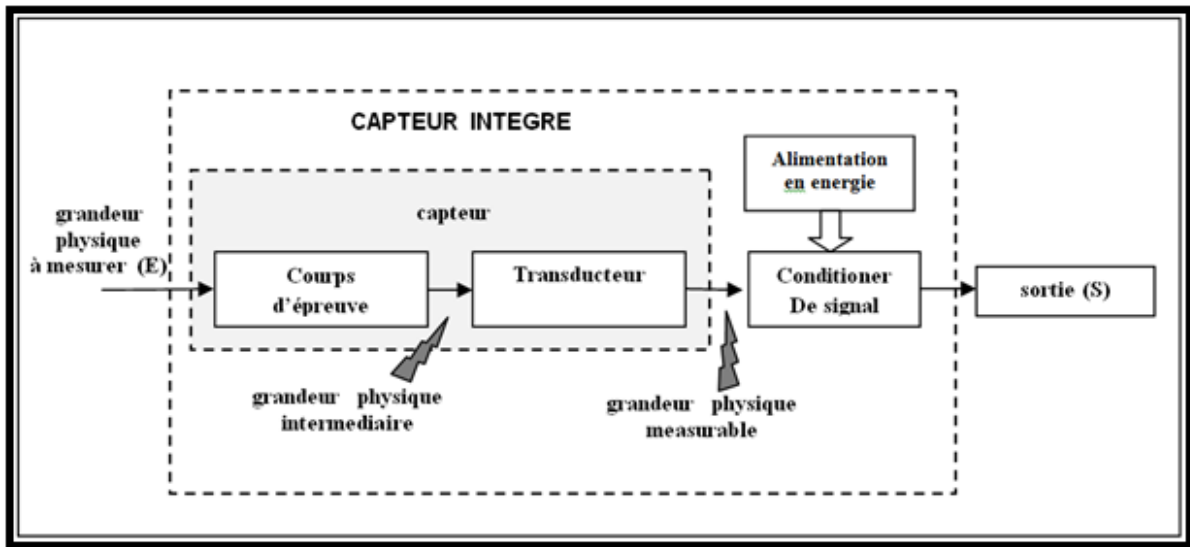


Figure 1.24: Schéma interne de capteur. [07]

Le capteur proprement dit est formé du corps d'épreuve et du transducteur :

Corps d'épreuve : C'est l'élément mécanique réagissant à la grandeur physique à mesurer.

Exemple : Support métallique pour une jauge de déformation.

Transducteur : C'est l'élément lié au corps d'épreuve traduisant la réaction reçue en un signal électrique, une variation de résistance, de capacité, d'inductance.

Exemple : Fil résistant disposé en zigzag pour une jauge.

Conditionneur : Circuit électronique traitant la grandeur mesurable pour délivrer un signal de sortie ayant des caractéristiques spécifiées (V, I, f, ...).

Exemple : Pont de Wheatstone suivi d'un amplificateur d'instrumentation

La loi obtenue $S=f(E)$ n'est pas toujours linéaire.

On intègre de plus en plus le conditionneur dans le même boîtier que le capteur : l'ensemble forme alors un capteur intégré. Les fonctions assurées par ce conditionneur vont de la simple mise en forme et adaptation du signal, jusqu'aux traitements élaborés de correction de grandeurs d'influence, formant ce qu'on appelle aujourd'hui les capteurs intelligents à microprocesseur.

1.5.2.4. Principales caractéristiques des capteurs

Etendue de mesure : c'est la différence entre le plus petit signal détecté et le plus grand perceptible sans risque de destruction pour le capteur.

Résolution : Plus petite variation de grandeur mesurable par le capteur.

Sensibilité : c'est la plus petite variation d'une grandeur physique que peut détecter un capteur.

Précision : c'est la culpabilité de répétabilité d'une information position, d'une vitesse,...

Rapidité : c'est le temps de réaction d'un capteur entre la variation de la grandeur physique qu'il mesure et l'instant où l'information prise en compte par la partie commande.

Linéarité : représente l'écart de sensibilité sur l'étendue de mesure sa reproductibilité.

La bande passante : est un intervalle de fréquences pour lesquelles l'amplitude de la réponse d'un système correspond à un niveau de référence.

1.5.2.5. Classification des capteurs

Les capteurs fonctionnent selon deux principes de base suivant l'origine du signal électrique de sortie. On distingue :

Les capteurs actifs : fonctionnement en générateur

Dans les capteurs actifs, une partie de l'énergie physique prélevée sur le mesurande est transformée directement en une énergie électrique qui constitue le signal de sortie.

Ce signal est un courant, une tension ou une quantité d'électricité. Les signaux de sortie délivrés par les capteurs actifs sont de très faible puissance, ils doivent être amplifiés pour pouvoir ensuite être transmis à distance.

Exemple : Thermocouple, Capteur piézoélectrique,

Les capteurs passifs : fonctionnant en modulateur,

Pour les capteurs passifs, c'est l'impédance du capteur qui est sensible aux variations de la mesure.

Ces variations d'impédance ne sont mesurables que par l'intermédiaire d'un circuit électronique de pré conditionnement.

Les capteurs passifs doivent être alimentés par une source d'énergie électrique extérieure

Exemple : Potentiomètre, Jauges extension métriques, ...

Le pré conditionnement peut se faire généralement de deux façons :

- le montage potentiométrique,
- le montage en pont.

1.5.2.6. Type de capture

On peut effectuer une première classification des capteurs par la nature des signaux transmis :

Capteur logique

Ils délivrent une sortie logique de type TOR (tout ou rien). Ils sont en général appelés détecteurs car ils servent surtout à prélever l'information "présence" ou "proximité" d'un objet.

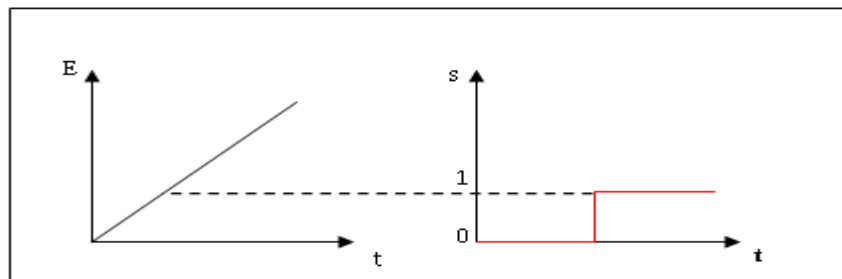


Figure 1.25 : Fonction logique d'un capteur TOR.

Capteur analogique

Ils délivrent un signal D_e de sortie sous la forme d'une tension ou d'un courant variant continuellement. Sur les capteurs industriels les plages de variation courantes sont +/- 50 mv, +/-1v, +/-10 v pour les tensions et 0-20mA, 4-20mA pour les courants. De tels signaux nécessitent un traitement particulier (conversion analogique-numérique) pour être exploitables par les API ou micro-ordinateur.

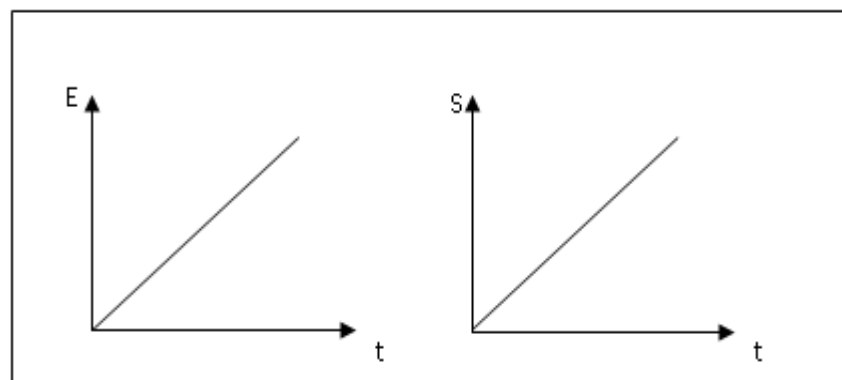


Figure 1.26 : Fonction d'un capteur analogique.

Capteur numérique

Ils délivrent un signal de sortie sous la forme, soit d'un train d'impulsions dont le nombre ou la fréquence est l'image de la grandeur d'entrée, soit d'un code numérique binaire. On trouve parmi les principaux capteurs numériques industriels, les capteurs de positions

angulaires incrémentaux, les codeurs absolus, les lecteurs de code à barres et les lecteurs de pistes magnétique.

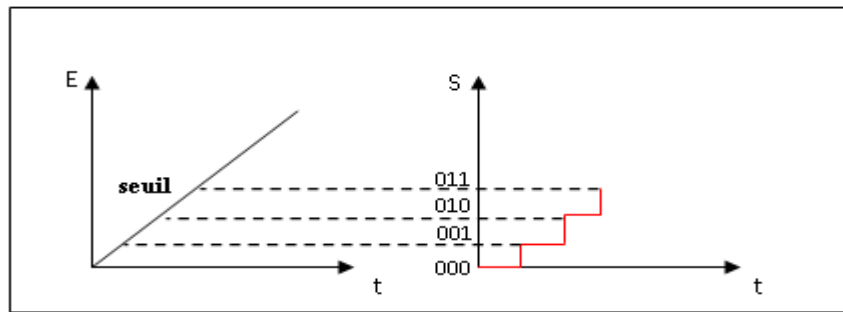


Figure 1.27 : Fonction d'un capteur numérique.

1.5.2.7. Critère de choix d'un capteur

Le choix s'opère suivant trois étapes

- le choix de technologie,
- le choix de la famille de capteur,
- la définition des caractéristiques, mécaniques, dimensionnelles, électriques, du capteur.

L'organigramme suivant illustre les premières étapes du choix d'un capteur.

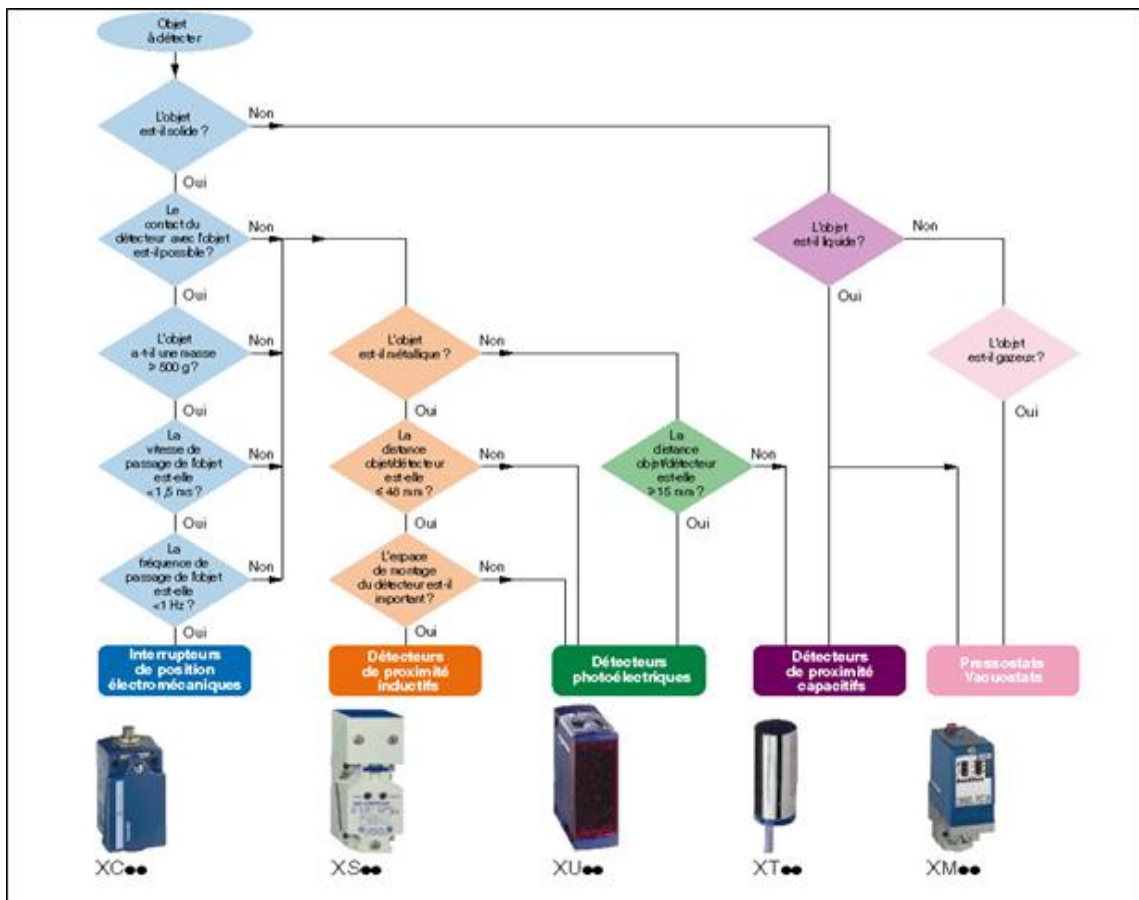


Figure 1.28 : Méthode de choix d'un capteur. [06]

1.6. Conclusion

Les systèmes automatisés de production devient indispensable pour obtenir une compétitivité des produits fabriqués de haute qualité. Dans ce chapitre on a vu en générale la structure des systèmes automatisés de production et les appareils essentiels à lies à ces systèmes pour la communication, la distribution d'énergie et la protection des machines. Nous avons présente les différents parties contrôle, puissance, communication et l'appareillage d'un système automatisé.

CHAPITRE II :
LES AUTOMATES
PROGRAMMABLE INDUSTRIEL

2.1. Introduction

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

L'information tout ou rien ne peut prendre que deux états (vrai/faux), (0 ou 1) C'est le type d'information délivrée par un détecteur, un bouton poussoir.

L'information Analogique 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).

L'information Numérique est contenue dans des mots codés sous forme binaire ou bien hexadécimale. C'est le type d'information délivrée par un ordinateur ou un module intelligent.

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

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

Le programme est écrit et compilé par des utilisateurs sur un matériel externe grâce à des langages adaptés. Il est téléchargé ensuite dans l'API. Les automates programmables industriels sont donc des systèmes embarqués destinée à piloter en ambiance industrielle et en temps réel des procédés automatiques.

2.2. Structure d'un API

L'API est un ensemble électronique qui gère et assure la commande d'un système automatisé. Il se compose de plusieurs parties, notamment d'une mémoire programmable dans laquelle l'opérateur écrit, dans un langage propre à l'automate, des directives concernant le déroulement du processus à automatiser. Son rôle consiste donc à fournir des ordres à la partie opérative en vue d'exécuter un travail précis comme par exemple la sortie ou la rentrée d'une tige de vérin, l'ouverture ou la fermeture d'une vanne. La partie opérative lui donnera en retour des informations relatives à l'exécution du dit travail. [02]

2.3. Différents types d'API

Les automates programmables peuvent être de type compact ou modulaire. Dans le type compact, on distingue les modules de programmation (LOGO de Siemens, ZELIO de Schneider, MILLENIUM de Crouzet ...) des micro-automates. 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) et recevoir des extensions en nombre limité. Ces automates, de fonctionnement simple, sont généralement destinés à la commande de petits automatismes.

Dans le 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.

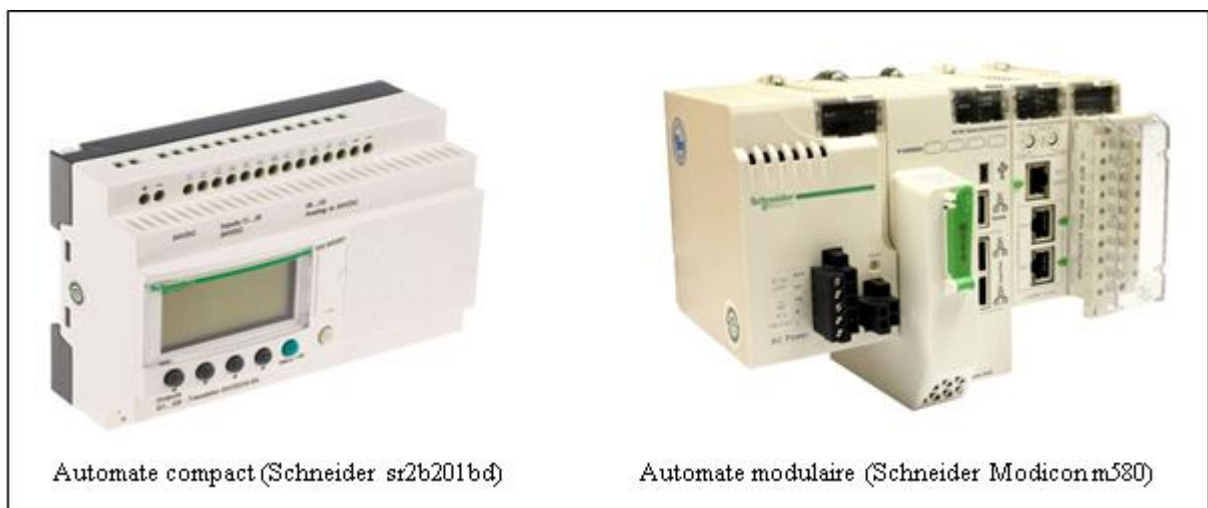


Figure. 2.1 : Automate programmable industriel. [02]

Les automates compacts permettent de commander des sorties en T.O.R et gèrent par fois des fonctions de comptage et de traitement analogique. Les automates modulaires permettent de réaliser de nombreuses autres fonctions grâce à des modules intelligents que l'on dispose sur un ou plusieurs racks. Ces modules ont l'avantage de ne pas surcharger le travail de la CPU car ils disposent bien souvent de leur propre processeur. [03]

2.4. Architecteur d'un l'API

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. Un API se compose donc de cinq parties : le processeur, la zone mémoire, les interfaces Entrées/Sorties, l'Alimentation et module périphériques (voir figure 2.2).

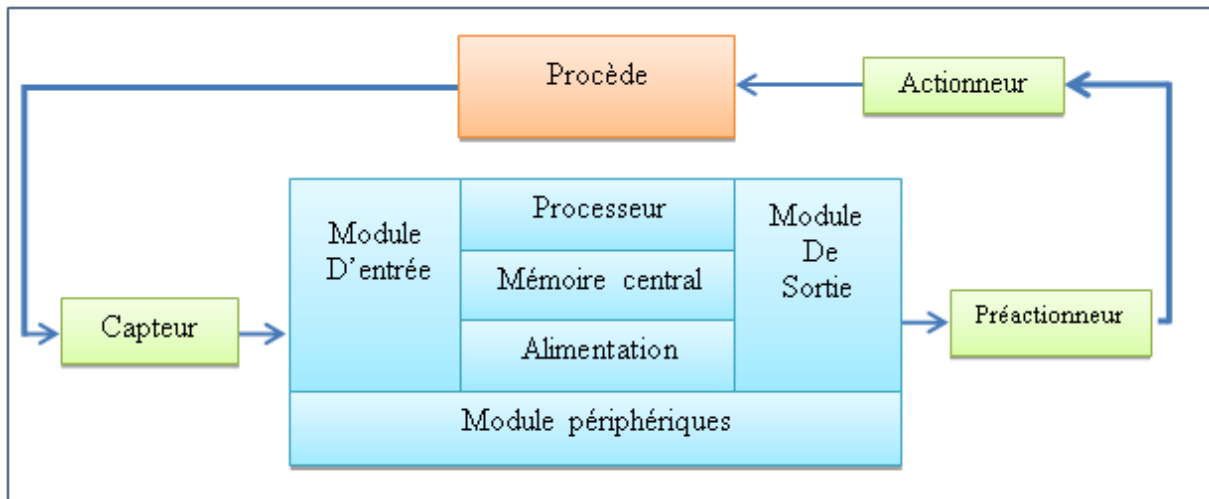


Figure .2.2: Architecteur d'un API.

2.4.1. Microprocesseur

Le microprocesseur réalise toutes les fonctions logiques (ET, OU, ...), les fonctions de temporisation, de comptage et de calcul à partir d'un programme contenu dans sa mémoire. Il est connecté aux autres éléments (mémoire et interface E/S) par des liaisons parallèles appelées 'BUS' qui véhiculent les informations sous forme binaire.

2.4.2. Mémoire

La zone mémoire permet de recevoir:

- les informations issues des capteurs d'entrées,
- les informations générées par le processeur et destinées à la commande des sorties (valeur des compteurs, des temporisations),
- et conserver le programme du système.

Plusieurs types de mémoire peuvent être distingués:

- RAM (Random Access Memory): mémoire vive dans laquelle on peut lire, écrire et effacer (contient le programme),
- ROM (Read Only Memory): mémoire morte dans laquelle on ne peut que lire,
- EPROM mémoires mortes reprogrammables effaçables aux rayons ultra-violetts,

- EEPROM mémoires mortes reprogrammables effaçables électriquement.

2.4.3. Interfaces d'entrées/sorties

Les entrées reçoivent des informations en provenance des éléments de détection et du pupitre opérateur. Les sorties transmettent des informations aux pré-actionneurs et aux éléments de signalisation du pupitre.

Les interfaces d'entrées sont destinées à :

- recevoir l'information en provenance des capteurs
- traiter le signal en le mettant en forme, en éliminant les parasites et en isolant électriquement l'unité de commande de la partie opérative (voir figure 2.3).

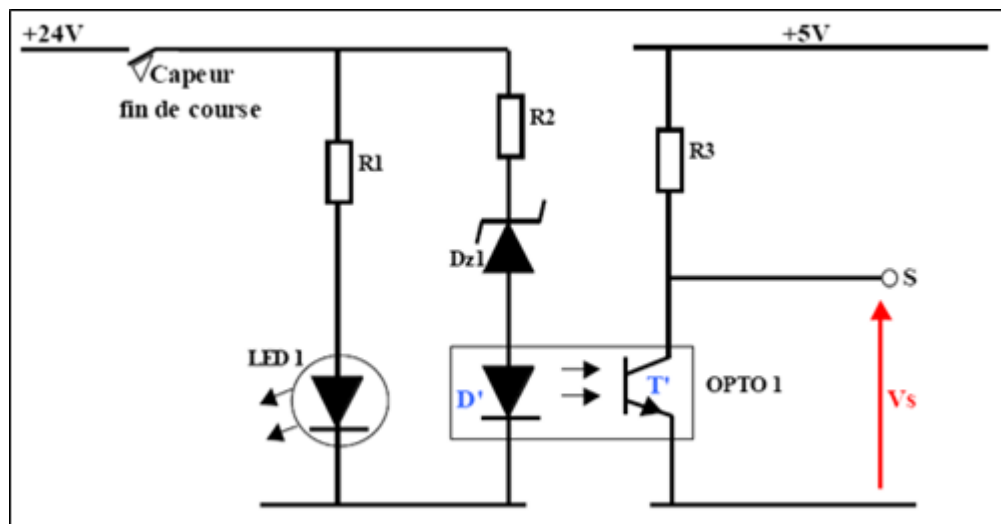


Figure 2.3: Exemple d'une carte d'entrées typique d'un API. [07]

Les interfaces de sorties sont destinées à :

- commander les pré-actionneurs et éléments de le signalisation du système.
- adapter les niveaux de tensions de l'unité de commande à celle de la partie opérative du système en garantissant une isolation galvanique entre ces dernières (voir figure.2.4).

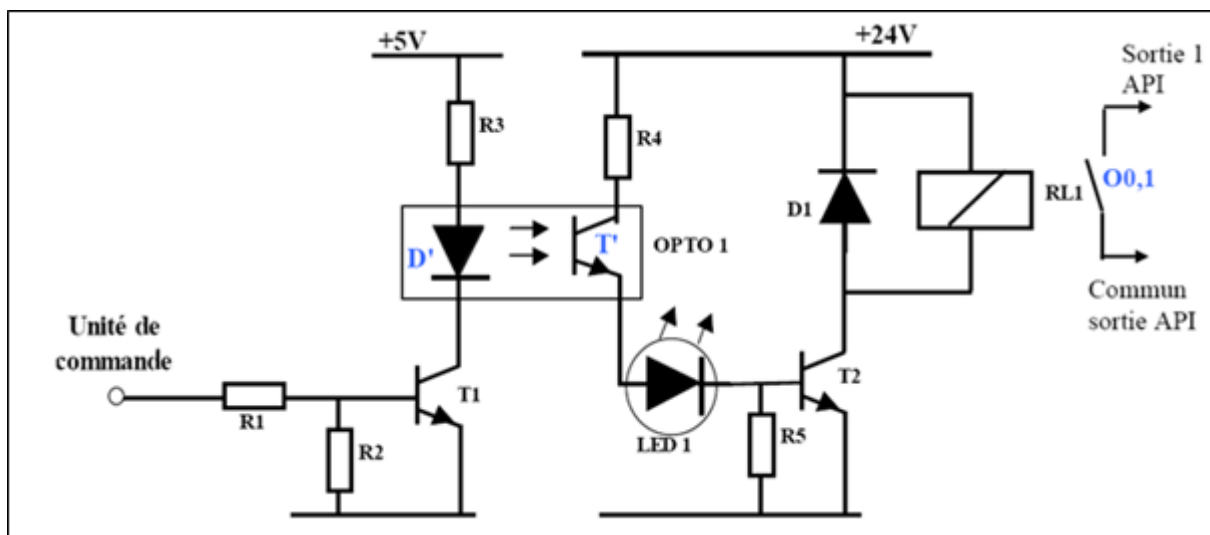


Figure 2.4: Exemple d’une carte de sortie typique d’un API. [07]

2.4.4. Alimentation de l'automate programmable industriel

L'alimentation, intégrée dans l'API, fournit à partir des tensions usuelles de réseau (230 V, 24 V) les tensions continues nécessaires au fonctionnement des circuits électroniques. Le module d'alimentation assure la distribution d'énergie aux différents modules.

2.4.5. Modules périphériques

Cartes de comptage rapide

Elles permettent d'acquérir des informations de fréquences élevées incompatibles avec le temps de traitement de l'automate.

Exemple : signal issu d'un codeur de position.

Cartes de commande d'axe

Elles permettent d'assurer le positionnement avec précision d'élément mécanique selon un ou plusieurs axes. La carte permet par exemple de piloter un servomoteur et de recevoir les informations de positionnement par un codeur. L'asservissement de position pouvant être réalisé en boucle fermée.

Cartes d'entrées /sorties analogiques

Elles permettent de réaliser l'acquisition d'un signal analogique et sa conversion numérique (CAN) indispensable pour assurer un traitement par le microprocesseur. La fonction inverse (CNA) est également réalisée. Les grandeurs analogiques sont normalisées : 0-10V ou 4-20mA.

Autre cartes

- Cartes de régulation PID,
- Cartes de pesage,
- Carte de communication (Ethernet ...),
- Carte d'entrée/sortie de portée.

2.5. Câblage de l'automate

2.5.1. Alimentation de l'automate

L'automate est alimenté généralement par le réseau monophasé 230V; 50 Hz, mais d'autres alimentations sont possibles (24 V etc...). La protection est de type magnétothermique (voir les caractéristiques de l'automate et les préconisations du constructeur). Il est souhaitable d'asservir l'alimentation de l'automate par un circuit de commande spécifique (contacteur KM1), voir figure 2.5. De même, les sorties seront asservies au circuit de commande et alimentées après validation du chien de garde. [04]

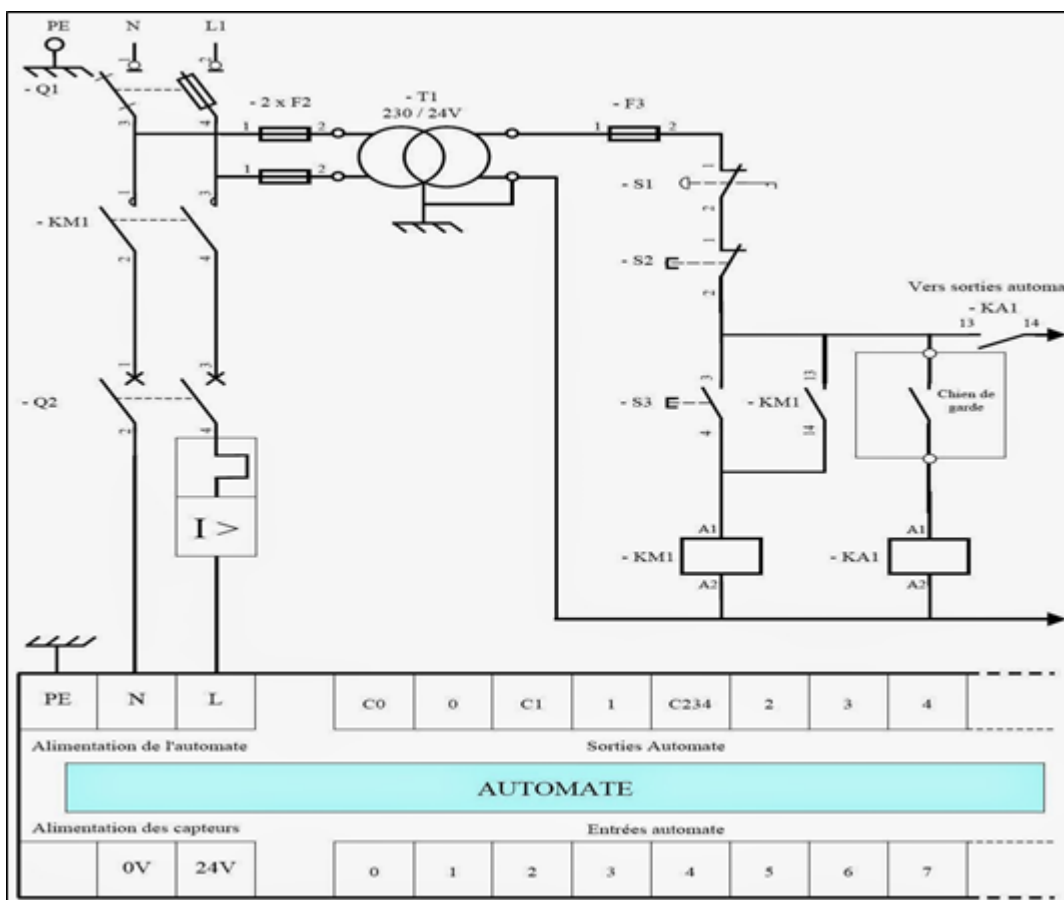


Figure. 2.5 : Alimentation de l'automate.

2.5.2. Alimentations des entrées d'automate

L'automate est pourvu généralement d'une alimentation pour les capteurs/détecteurs (attention au type de logique utilisée : logique positive ou négative). Les entrées sont connectées au OV (commun) de cette alimentation. Les informations des capteurs/détecteur sont traitées par les interfaces d'entrées.

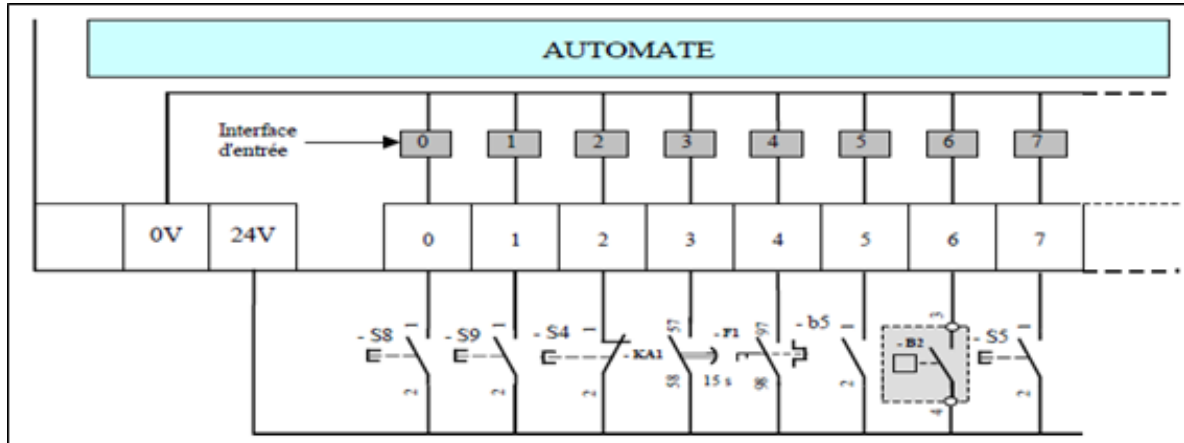


Figure. 2.6 : Alimentations des entrées d'automate. [07]

2.5.3. Alimentations des sorties de automate

Les interfaces de sorties permettent d'alimenter les divers pré-actionneurs. Il est souhaitable d'équiper chaque pré-actionneur à base de relais de circuits RC (non représentés).

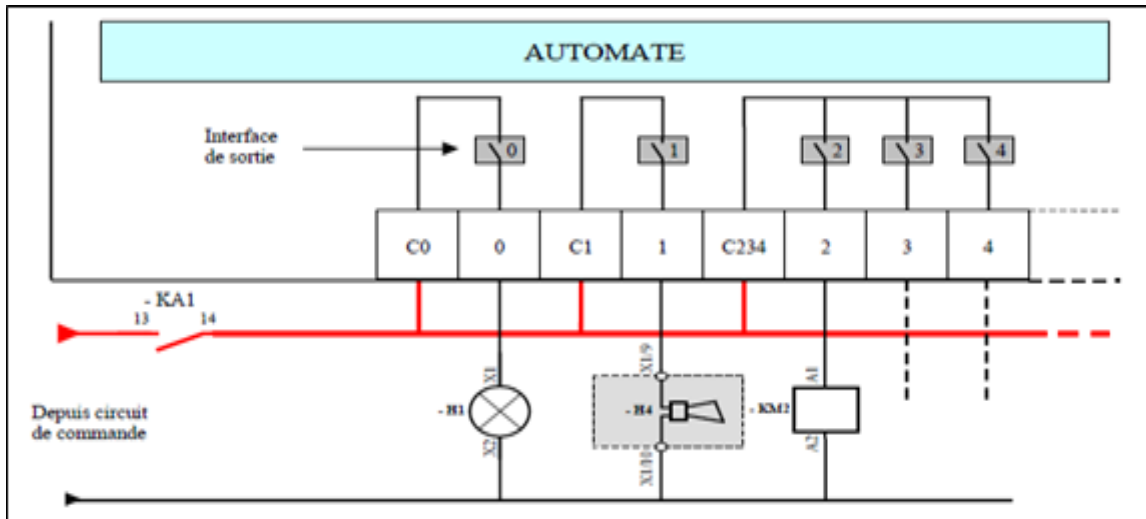


Figure. 2.7 : Alimentations des sorties de automate. [07]

2.6. Protection de l'automate

La protection des circuits d'entrée contre les parasites électriques est souvent résolue par découplage opto-électrique (voir figure 2.4). Le passage des signaux par un stade de faisceau lumineux assure en effet une séparation entre les circuits internes et externes. Du côté sorties, on doit assurer le même type de protection (voir figure 2.5), mais amplification de puissance, avec au final un courant continu ou alternatif selon les cas. Deux types de sortie sont utilisés :

Sorties statiques

Relais statiques intégrant des composants spécialisés : transistor bipolaires, thyristors. Ces composants n'ont aucune usure mécanique et leurs caractéristiques de commutation se maintiennent dans le temps.

Sorties relais électromagnétique

Où le découplage résulte de l'existence de deux circuits électriques (Bobines d'excitation, circuits de puissance), Ces relais électromagnétique ont l'avantage d'avoir une faible résistance de contact, une faible capacité de sortie et surtout un faible coût, mais une durée de vie et une vitesse de commutation inférieure aux sorties statiques.

2.7. Fonctionnement de l'automate

Le cycle de fonctionnement de l'automate est composé de trois phases: acquisition des entrées, traitement des données et émission des ordres.

Phase d'acquisition des entrées.

Durant cette phase, les signaux appliqués à l'interface d'entrée de l'API sont copiés en mémoire intermédiaire d'entrée (MIE) dans des emplacements accessibles au programme et qui correspondent aux variables d'entrée. Les variables d'entrée sont uniquement accessibles en lecture. Leurs valeurs resteront ainsi inchangées lors des deux phases suivantes du cycle, en d'autres termes, au moment d'acquisition des entrées.

Phase de traitement des données.

Lecture du programme et calcul des nouvelles valeurs de la variable de sortie.

Phase d'émission des ordres.

Lecture des variables de sortie dans la mémoire intermédiaire d'entrée (MIE) et les transférer vers le module de sortie pour pouvoir être appliquées aux actionneurs.

La MIE (Mémoire Intermédiaire des Entrées) est une mémoire temporaire qui reçoit les données d'entrées transmises par l'interface d'entrée.

La MIS (Mémoire Intermédiaire des Sorties) représente aussi une mémoire temporaire qui contient les données traitées (résultat) avant de les transmettre vers la sortie.

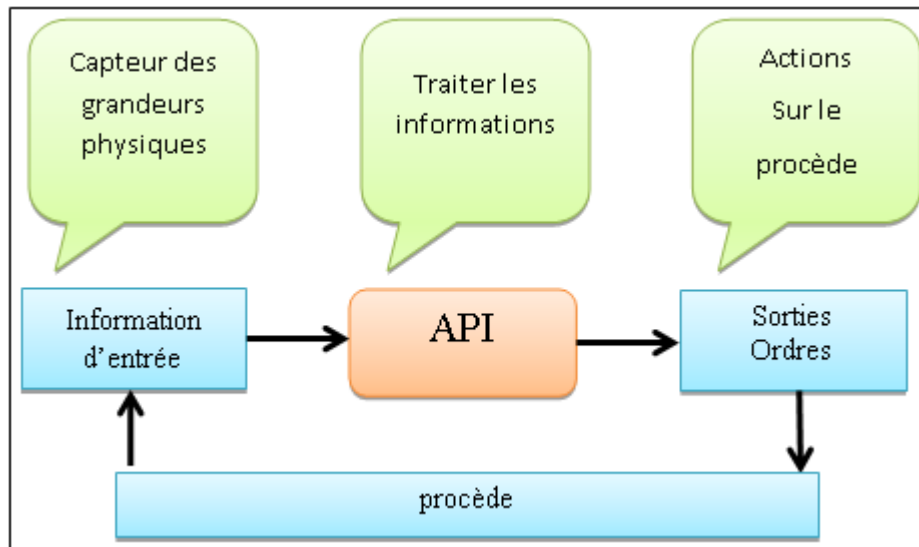


Figure. 2.8 : Fonctionnement de l'automate.

Ces trois opérations sont effectuées continuellement par l'automate (fonctionnement cyclique).

On appelle scrutation l'ensemble des quatre opérations réalisées par l'automate et le temps de scrutation est le temps mis par l'automate pour traiter la même partie de programme. Ce temps est de l'ordre de la dizaine de millisecondes pour les applications standards.

2.8. Programmation de l'API

Elle peut s'effectuer de trois manières différentes :

- sur l'API lui-même à l'aide de touches,
- avec une console de programmation reliée par un câble spécifique à l'API,
- avec un PC et un logiciel approprié avec lequel la programmation est plus conviviale, communiquant avec l'automate par le biais d'une liaison série RS232 ou RS485 ou d'un réseau de terrain.

2.9. Critères de choix d'un automate

Le choix d'un automate programmable est en premier lieu le choix d'une société ou d'un groupe. Les contacts commerciaux et les expériences vécues sont déjà un point de départ. Les grandes sociétés privilégieront deux fabricants pour faire jouer la concurrence et pouvoir se retourner en cas de perte de vitesse de l'une d'entre elles. Le personnel de maintenance doit toutefois être formé sur ces matériels et une trop grande diversité des matériels peut avoir de graves répercussions. [07]

Un automate utilisant des langages de programmation de type GRAFCET est également préférable pour assurer les mises au point et dépannages dans les meilleures conditions. La possession d'un logiciel de programmation est aussi source d'économies (achat du logiciel et formation du personnel), des outils permettant une simulation des programmes sont également souhaitables. Il faut ensuite quantifier les besoins :

- 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 ...).

2.10. Conclusion

L'API est un bon produit s'il est bien choisi et bien employé. Ce qui peut apparaître comme une lapalissade nous a amené à attirer l'attention sur des aspects parfois jugés triviaux, tels les types d'E/S, le dimensionnement des alimentations électriques, les modes d'exécution d'un programme, les limites des divers types de communication, car ce sont des points où sont parfois commises des erreurs qui entraînent des surcoûts d'installation ou limitent fâcheusement les performances obtenues.

CHAPITRE III :

PROGRAMMATION DE L'API

3.1. Introduction

L'écriture d'un programme consiste à créer une liste d'instructions permettant l'exécution des opérations nécessaires au fonctionnement du système. L'API traduit le langage de programmation en langage compréhensible directement par le microprocesseur. Ce langage est propre à chaque constructeur. Il est lié au matériel mis en œuvre.

Chaque automate possède son propre langage. Cependant, les constructeurs proposent tous une interface logicielle répondant à la norme CEI 61131-3 qui définit cinq langages de programmation utilisables : le grafcet et langage LADDER et langage ST (Structured Text) et langage IL (Instruction List) et langage FBD (Boîtes fonctionnelles).

3.2. GRAFCET

3.2.1. Description du GRAFCET

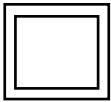
Le Grafcet (**GRA**phe **F**onctionnel de **C**ommande **E**tape **T**ransition) est créé en 1977 par un groupe de travail l'AF CET (Association Française pour la Cybernétique Economique et Technique). C'est un outil graphique de définition pour l'automatisme séquentiel en tout ou rien. C'est un langage universel qui peut se câbler par séquenceur et être programmé sur automate ou sur ordinateur.

Lorsque le mot GRAFCET en lettre capitale est utilisé, il fait référence à l'outil de modélisation. Lorsque le mot grafcet est écrit en minuscule, il fait alors référence à un modèle obtenu à l'aide des règles du GRAFCET présenté sous forme d'organigramme. Son but est la description du fonctionnement de l'automatisme contrôlant le procédé. C'est tout d'abord un outil graphique puissant directement exploitable. Il est aussi un langage pour la plupart des API existants sur le marché. Il comprend :

- des étapes associées à des actions,
- des transitions associées à des réceptivités,
- des liaisons orientées reliant étapes et transitions. [08]

Les étapes

L'étape symbolise un état ou une partie de l'état du système. Elle possède deux états possibles: active représentée par un jeton dans l'étape ou inactive. L'étape i , repérée numériquement, possède ainsi une variable d'état, appelée variable d'étape X_i . Cette variable est une variable booléenne valant 1 si l'étape est active, 0 sinon.



Etape initiale :

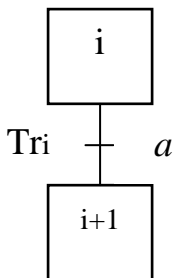
Elle représente le système à l'état de repos initial. Elle est activée au début du cycle.



Etape :

A chaque étape est associée une action ou plusieurs, c'est à dire un ordre vers la partie opérative ou vers d'autres grafquets.

Les transitions



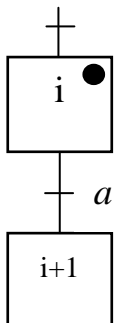
Une transition indique la possibilité d'évolution qui existe entre deux étapes et donc la succession de deux activités dans la partie opérative. Lors de son franchissement, elle va permettre l'évolution du système. A chaque transition est associée une réceptivité qui exprime la condition nécessaire pour passer d'une étape à une autre.

Cette condition est écrite sous forme d'une proposition logique, une fonction combinatoire calculée à partir :

- des variables d'entrées traduisant l'état des capteurs, des boutons poussoirs, etc,
- du temps,
- de l'état courant des étapes du grafcet (les X_i).

Si la réceptivité n'est pas précisée, alors cela signifie qu'elle est toujours vraie.

Règle : si l'étape i est inactive, $X_i = 0$, la transition Tri est sans effet. Cependant, attention, valider sans raison une transition peut avoir des conséquences graves, perturbant le cycle dans certains cas



Si l'étape i est active, $X_i = 1$, la transition a est validée, alors :

Si $a = 0$, alors attente

Si $a = 1$, alors l'étape i est validée $X_i = 0$ et l'étape suivante $i+1$ est activée, $X_{i+1} = 1$.

Les liaisons orientées

Une liaison orientée est le lien qui lie une étape à une transition ou l'inverse. Par convention, étapes et transitions sont placées suivant un axe vertical. Les liaisons orientées sont de simples traits verticaux lorsque la liaison est orientée de haut en bas, et sont munis d'une flèche vers le haut lorsque la liaison est orientée vers le haut.

Notation des entrées/sorties

Lors de l'établissement du grafcet de spécification de niveau 1, on utilise des noms explicites pour les entrées du système modélisé ainsi que pour les sorties. Lors du passage au grafcet de réalisation de niveau 2, on utilise plutôt des noms logiques : Ei pour les entrées et Si pour les sorties.

Les actions

L'action associée à l'étape peut être de 3 types : continue, conditionnelle ou mémorisée.

Action continue : La ou les sorties correspondant à l'ordre a sont mises à 1 tant que l'étape associée est active. Lorsque l'étape devient inactive, la ou les sorties sont mises à 0.

Action conditionnelle : Une action conditionnelle n'est exécutée que si l'étape associée est active et si la condition associée est vraie. Elles peuvent être décomposées en 3 cas particuliers:

- Action conditionnelle simple : Type C (Condition)
- Action retarder : Type D (delay)
- Action limitée dans le temps : Type L (limité)

Action mémorisée : On peut ainsi donner l'équation d'un ordre a en fonction des états des étapes, des conditions éventuelles et du temps.

Les réceptivités

Une réceptivité est associée à chaque transition. C'est une fonction booléenne calculée à partir des entrées du graphe, des états des étapes ou des temporisations. Une réceptivité est donc écrite en utilisant les opérateurs ET, OU, NON et front. Le front montant ou descendant d'une variable permet de situer dans le temps le changement de valeur de ce capteur. On les note respectivement E et \bar{E} . Les fronts ne sont à 1 que durant un délai, qui correspond au temps de prise en compte de l'évènement, c'est à dire le temps de franchissement d'une transition.

Les temporisations

Les temporisations sont des variables booléennes qui permettent une prise en compte du temps. Pour écrire ces temporisations, on fait appel à un opérateur normalisé "t1/Ei/t2" (CEI/IEC 617-12). Cet opérateur sert de base à la notation utilisée en GRAFCET.

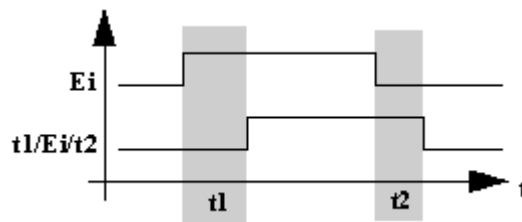
Les temporisations utilisées en GRAFCET font référence aux variables d'étapes et donc s'écrivent sous la forme "t1/Ei/t2" (alors t1 désigne le retard apporté au changement de l'état inactif à l'état actif de l'étape *i*, et t2 désigne le retard apporté au changement de l'état actif à l'état inactif de l'étape *i*).

Il est important de noter que:

L'étape *i* doit être active pendant un temps supérieur ou égal à t1 pour que "t1/Ei/t2" puisse passer à l'état vrai.

Si t1 =0, on note "Ei/t2", Si t2= 0 on note "t1/Ei".

Il faut préciser l'unité de temps à laquelle on fait référence.



3.2.2. Les règles d'évolution du GRAFCET

Règle N°1 – Condition initiale

A l'instant initial, seules les étapes initiales sont actives.

Règle N°2 – Franchissement d'une transition

Pour qu'une transition soit validée, il faut que toutes ses étapes amont (immédiatement précédentes reliées à cette transition) soient actives. Le franchissement d'une transition se produit lorsque la transition est validée, si et seulement si la réceptivité associée est vraie.

Règle N°3 – Evolution des étapes actives

Le franchissement d'une transition entraîne obligatoirement dans cet ordre la désactivation de toutes ces étapes amont et l'activation de ses étapes aval.

Règle N°4 – Franchissement simultané

Toutes les transitions simultanément franchissables à un instant donné sont simultanément franchies.

Règle N°5 – Conflit d'activation

Si une étape doit être simultanément désactivée par le franchissement d'une transition aval et activée par le franchissement d'une transition amont, alors elle reste active. On évite ainsi des commandes transitoires (néfastes au procédé) non désirées.

3.2.3. GRAFCET de niveau 1 et 2

3.2.3.1. GRAFCET de niveau 1

C'est en général la description de l'automatisme seul, c'est à dire l'enchaînement des actions et des transitions permettant de contrôler le procédé. Lorsque l'on aborde l'analyse et la description d'un système, on ne sait pas quelle technologie sera retenue pour les actionneurs, les capteurs et la commande. On décrira dans ce GRAFCET les actions et les évènements en termes généraux.

3.2.3.2. GRAFCET de niveau 2

C'est la description complète de l'automatisme qui tient compte de toutes les contraintes du procédé. Les points essentiels du GRAFCET de niveau 2 sont :

- la simplification du GRAFCET niveau 1, c'est à dire les parallélismes et les séquences répétées,
- les modes de fonctionnement de l'automatisme, c'est à dire la prise en compte de la marche automatique, de la marche par cycle et de la marche manuelle utile pour la maintenance et les réglages,
- les arrêts d'urgence entraînant la coupure d'alimentation, l'utilisation d'une variable AU (si arrêt d'urgence $AU = 1$) à rajouter dans les conditions logiques pour les différentes actions, une procédure de dégagement si $AU = 0$,
- les sécurités procédé, c'est à dire l'arrêt de l'automatisme si une condition anormale est détectée, par exemple la détection d'un objet incorrect ou mal positionné,
- les conditions initiales, c'est à dire les différentes positions au repos (au départ du cycle), l'état de l'alimentation et l'ensemble des tests destinés à vérifier les conditions initiales.

La complexité de cette tâche peut être grande. Le GRAFCET de niveau 2 doit être étudié avec le plus grand soin. Ceci peut éventuellement conduire à modifier le procédé, et à modifier si besoin les capteurs et les actionneurs. Dans certains cas, on peut être amené (après

étude) à rechercher d'autres solutions que l'automatisme séquentiel, par exemple la logique floue. Ces cas limites sont :

- une indétermination logique au niveau du procédé,
- une description logique inadaptée,
- une durée de cycle de scrutation ou d'acquisition trop longue,
- la difficulté à définir une séquence de dégagement pour l'arrêt d'urgence.

Remarque : En milieu industriel, les essais erreurs peuvent être dangereux ou catastrophiques : d'où l'intérêt de faire une étude du grafcet de niveau 2 très approfondie.

3.3. Langage Ladder

Ladder est le mot anglais qui signifie l'échelle. Ladder Diagramme (LD) ou le langage Ladder ou schéma à contacts est un langage graphique très populaire auprès des automaticiens pour programmer les automates programmables industriels. Il ressemble un peu aux schémas électriques voir figure 3.1. Il est facilement compréhensible.

L'idée initiale du Ladder est la représentation de fonction logique sous la forme de schémas électriques. Cette représentation est originalement matérielle : quand l'automate programmable industriel n'existait pas, les fonctions étaient réalisées par des câblages.

Le Ladder a été créé et normalisé dans la norme CEI 61131-3. Il est encore aujourd'hui souvent utilisé dans la programmation des automates programmables industriels, bien qu'ayant tendance à être délaissé en faveur de langages plus évolués, et plus adaptés aux techniques modernes de programmation, tels que le Structured Text (ST) par exemple, ou encore le Grafcet, plus adapté à la programmation de séquences.

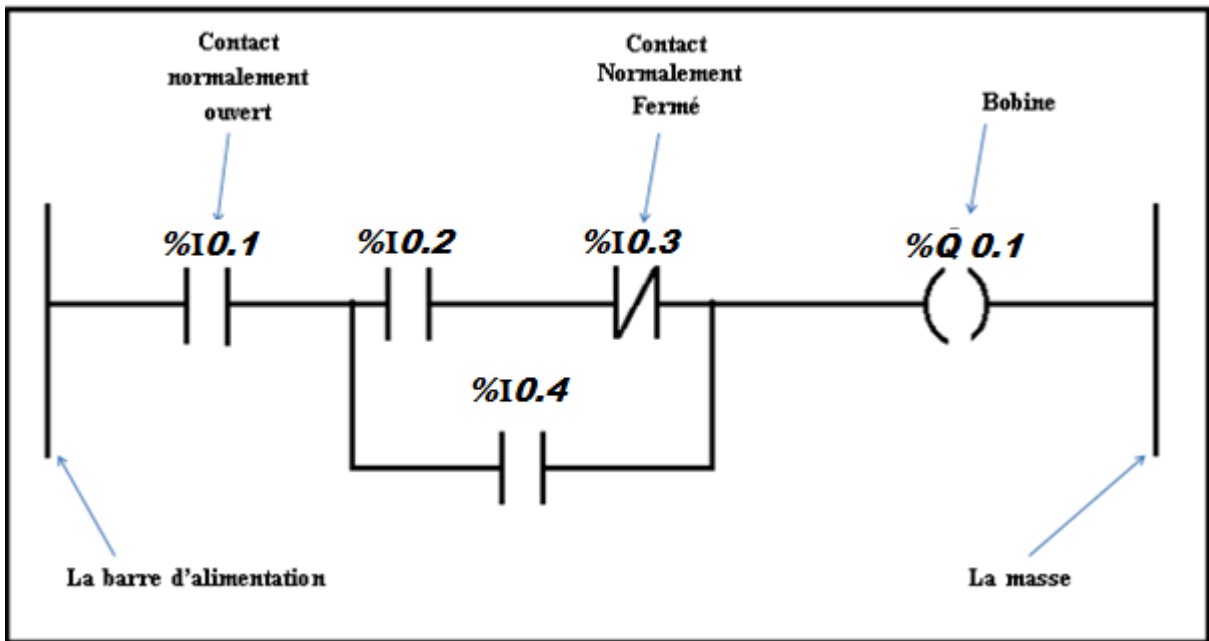


Figure. 3. 1: Exemple d'un programme en Ladder.

3.3.1. Principe

Un programme Ladder se lit de haut en bas et l'évaluation des valeurs se fait de gauche à droite. Les valeurs correspondent en fait, si on le compare à un schéma électrique, à la présence ou non d'un potentiel électrique à chaque nœud de connexion.

En effet, le Ladder est basé sur le principe d'une alimentation en tension représentée par deux traits verticaux reliés horizontalement par des bobines, des contacts et des blocs fonctionnels, d'où le mot 'Ladder' (échelle), (voir figure.3.1).

C'est un langage volontairement simple et graphique pour être compréhensible. Il a permis dans les années 1990 d'être utilisé sans lourde formation par un électricien. Il est aujourd'hui un peu dépassé. La puissance de calcul des CPU actuelles permettent de travailler directement en langage objets avec des notions de classe et d'héritage.

3.3.2. Les composants du langage

Il existe 3 types d'élément de langage :

- les entrées (ou contact), qui permettent de lire la valeur d'une variable booléenne,
- les sorties (ou bobines) qui permettent d'écrire la valeur d'une variable booléenne,
- les blocs fonctionnels qui permettent de réaliser des fonctions avancées.

Le tableau 3.1 donne les principaux éléments (contacts et bobines) d'un réseau LD.

Tableau. 3.1:les principaux éléments d'un réseau LD.

Objet graphique	nom
- -	Contact normalement ouvert
- / -	Contact normalement fermé
- P -	Contact fermé au front montant
- N -	Contact fermé au front descendant
-()-	Bobine normalement ouverte
-(/)-	Bobine normalement fermée
-(S)- ou-(L)-	Bobine Latch (maintenu à 1 une fois actionné)
-(R)-ou -(U)-	Bobine Reset (remise à 0 de la bobine latch)
-(P)-	Bobine active au front montant de son entrée
-(N)-	Bobine active au front descendant de son entrée
-<return>	Retour inconditionnel (vers le sous-programme appelant)
-<cond-<return>	Retour conditionnel
->>Label	Saut inconditionnel
-<cond->>Label	Saut conditionnel

3.4. Logiciel de programmation

Un automate programmable lit des entrées, commande des sorties et résout une logique basée sur un programme. La création d'un programme d'un automate zelio consiste à écrire une série d'instructions rédigées dans un des langages de programmation zelio.

3.4.1. Zelio soft

Le module Zelio Soft ou Logic trouve sa place partout où son intelligence pratique simplifie la vie. Dans les petits automatismes pour l'industrie et le résidentiel, l'électricien appréciera sa simplicité de mise en œuvre et sa facilité de programmation grâce à l'utilisation de symboles électriques, Ladder ou Zelio.

Zelio Logic combine dans un même boîtier des fonctions directement utilisables comme des temporisations, des compteurs, des horloges, des relais auxiliaires, combinables à volonté pour réaliser un petit automate simple. Grâce à son encombrement modulaire, il s'intègre parfaitement dans un coffret ou dans un tableau de distribution électrique et se programme directement à partir de son interface opérateur intégrée ou par un logiciel PC d'une extrême simplicité.

Le langage de programmation à contact permet de faire très simplement la relation entre le schéma électrique et l'automatisme à réaliser. Au-delà d'une application nécessitant deux ou trois temporisations, la solution Zelio Logic est performante en coût d'intégration et de mise en œuvre. Elle offre une solution économique, compacte, flexible et simple à installer, adaptée aux besoins. Il peut se trouver dans l'industrie (machines de contrôle de fin de chaîne, petites

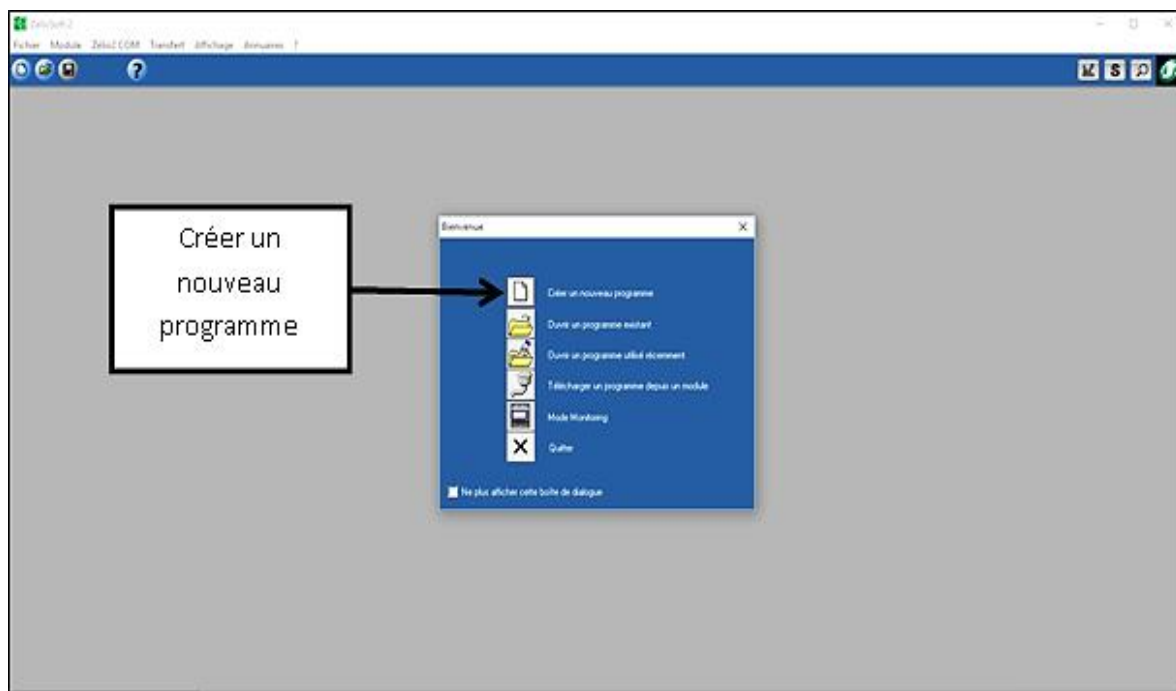
machines de conditionnement, d'emballage, machines automatiques) ou dans des installations du bâtiment (gestion de l'éclairage, ventilation, démarrage de moteurs en cascade, compresseurs, petits systèmes de manutention).

Cette solution apporte un confort supplémentaire dans le réglage des différents paramètres utilisés : réglages des temporisations par saisie numérique, verrouillages de certains réglages par l'installateur, affichage de messages de défauts sur l'interface du Zelio Logic en fonction de l'état de fonctionnement du système par exemple. [05]

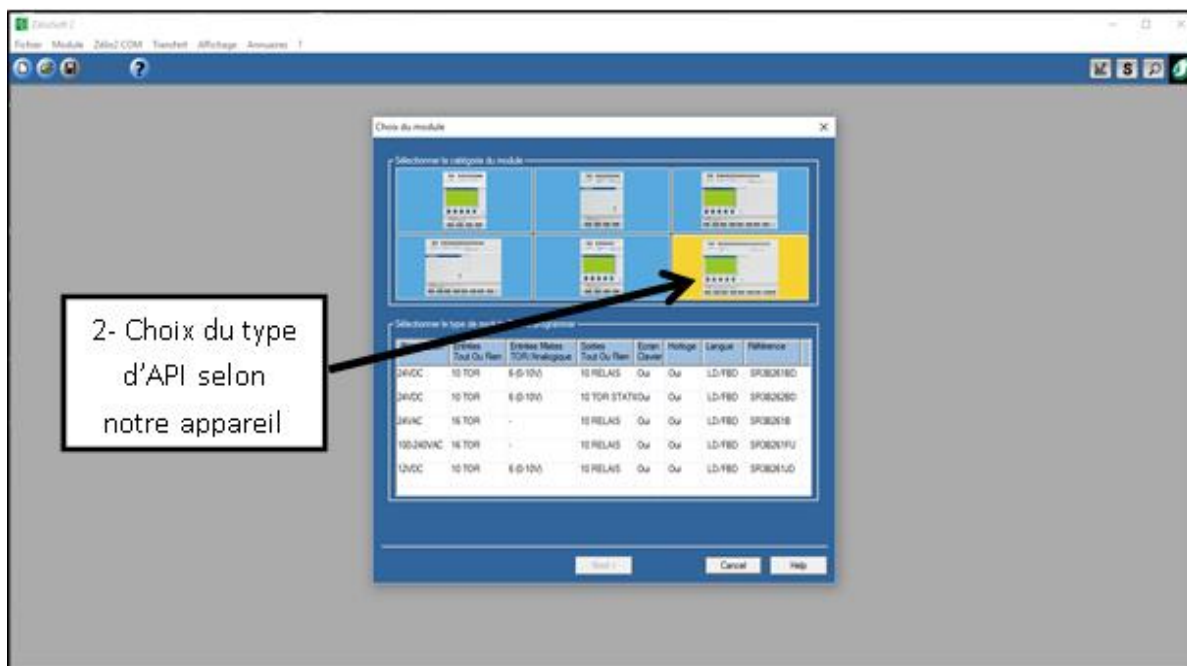
3.4.2. Programmation sur logiciel Zelio Soft 2

Zelio Soft 2 est un logiciel de programmation de type ladder destiné à la programmation des API de marque Schneider que nous allons utiliser dans la réalisation de notre projet. L'écriture d'un programme sur logiciel Zelio Soft 2 consiste à suivre les étapes suivantes.

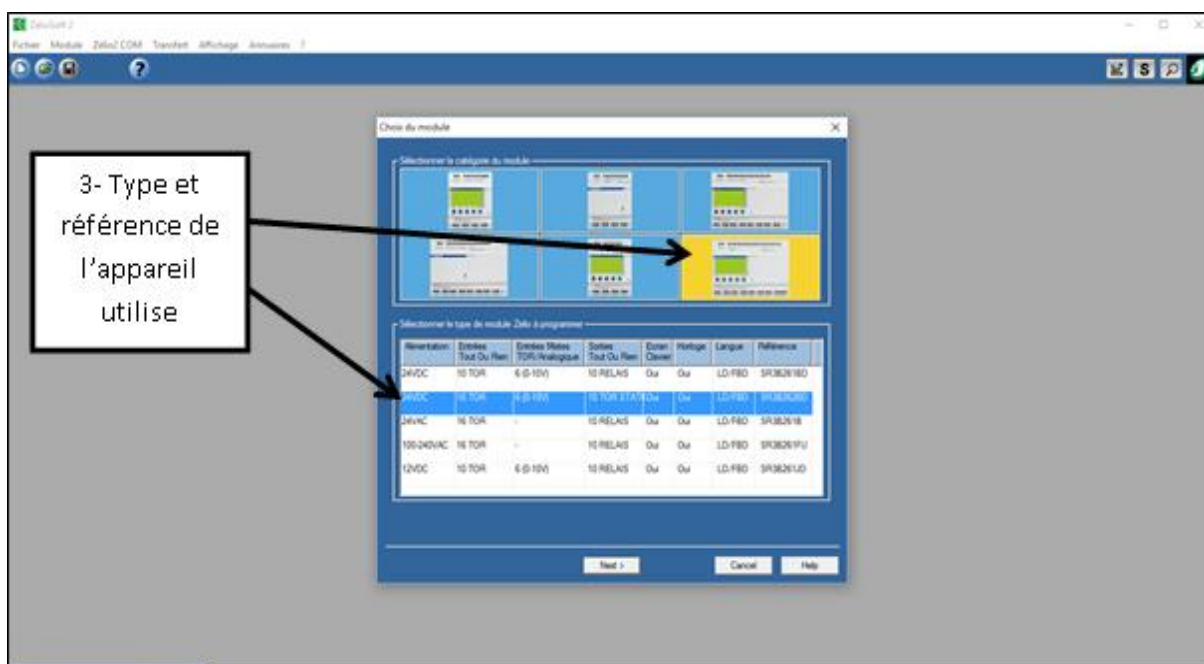
Etape 1 : Ouvrir le logiciel de programmation Zelio Soft 2 et créer un nouveau fichier de programmation.



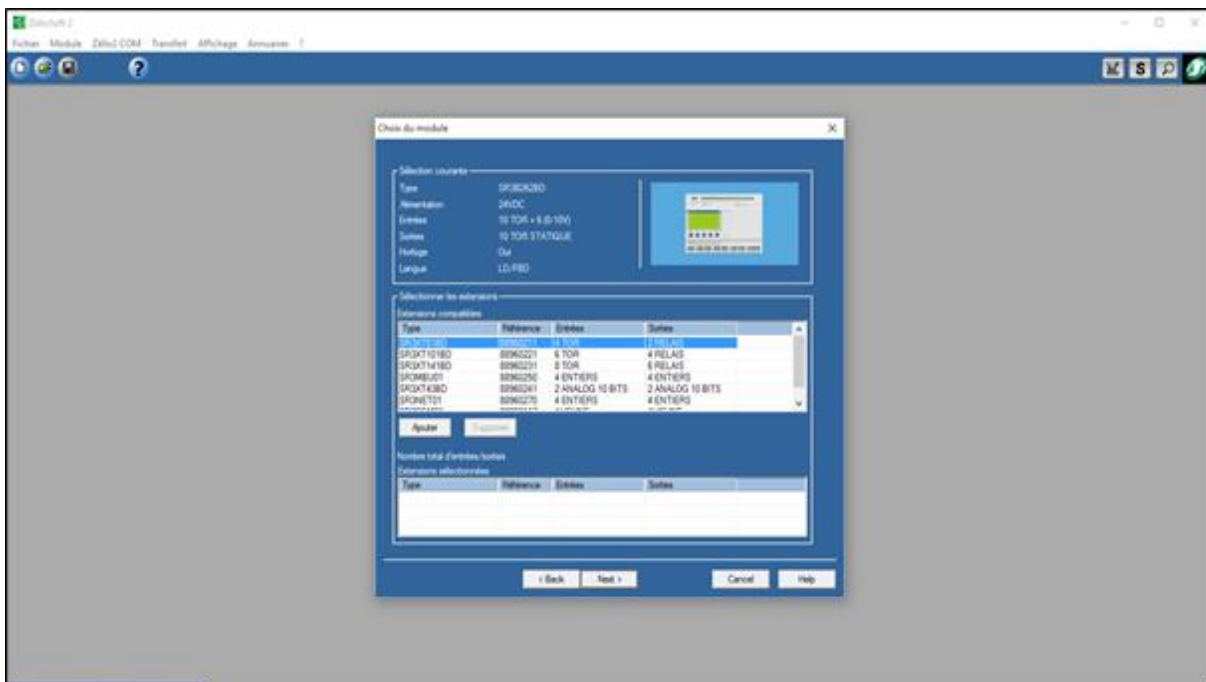
Etape 2 : choisir la catégorie d'automate parmi les catégories disponibles.



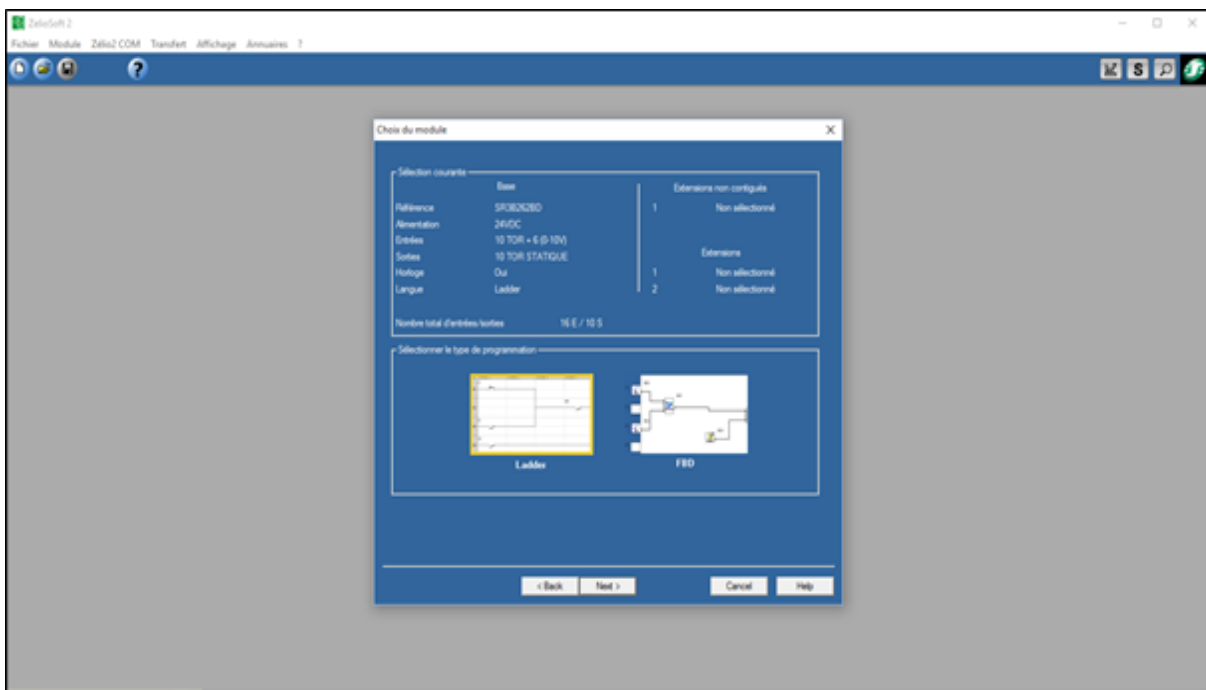
Etape 3 : sélectionner le type d'automate à programmer, puis appuyer sur la touche Next



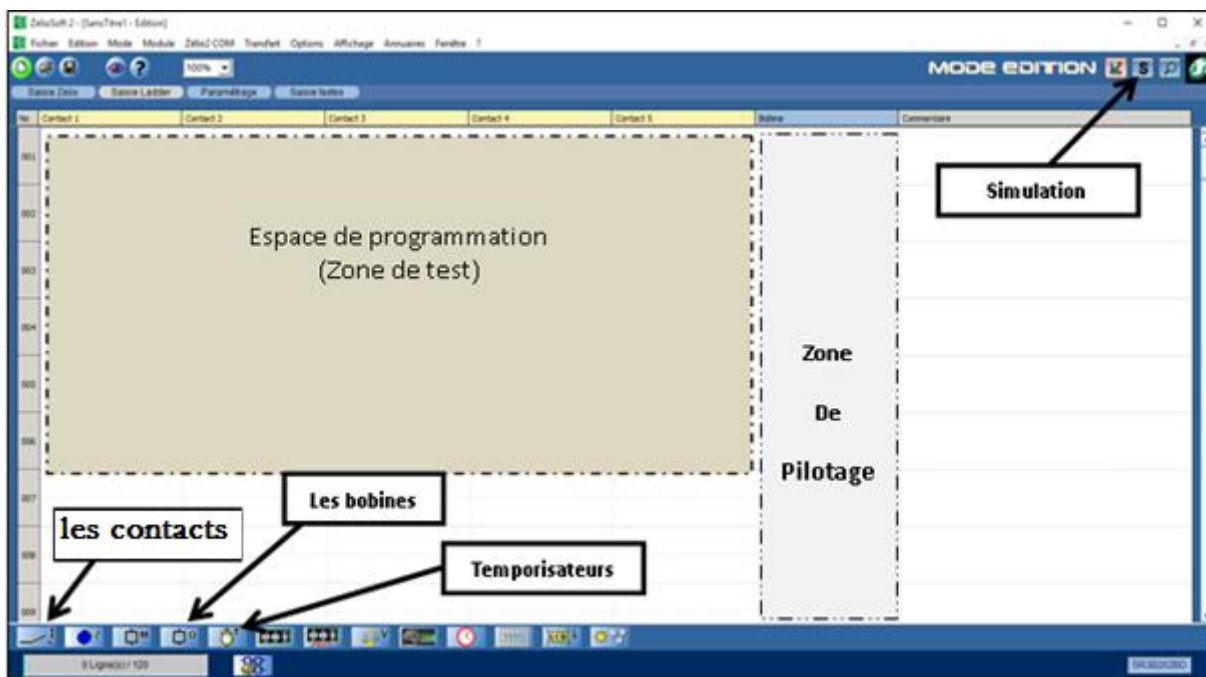
Etape 4 : facultative, pour ajouter un notre module si la capacité (nombre d'entrées sorties) de l'API choisi est insuffisante.



Etape 5 : choisir le langage de programmation Ladder ou FBD.



Une page d'édition de programme Ladder est affichée.

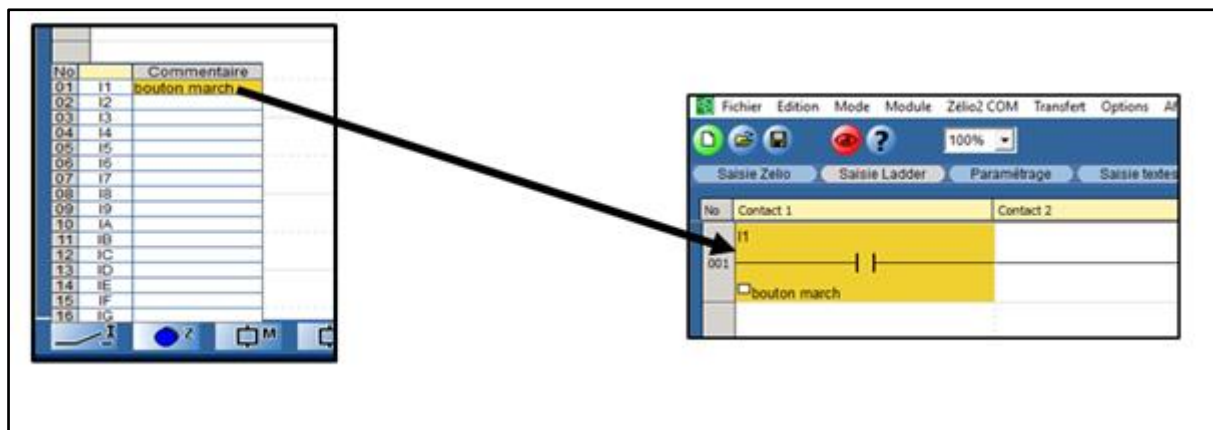


Etape 6: écrire le programme en langage Ladder.

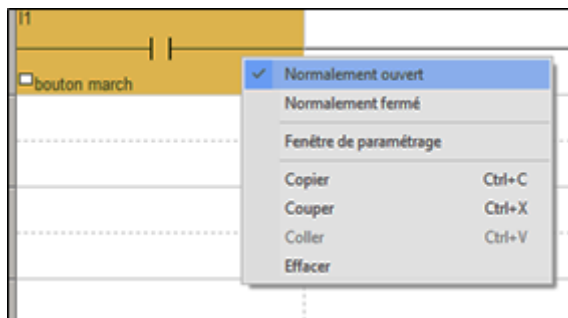
Exemple : marche et arrêt d'un moteur.

Pour ajouter un contact de marche (entrée):

- Cliquer sur le bouton I et choisir un contact.
- Faire glisser le contact choisi dans la zone de test à l'endroit désiré.



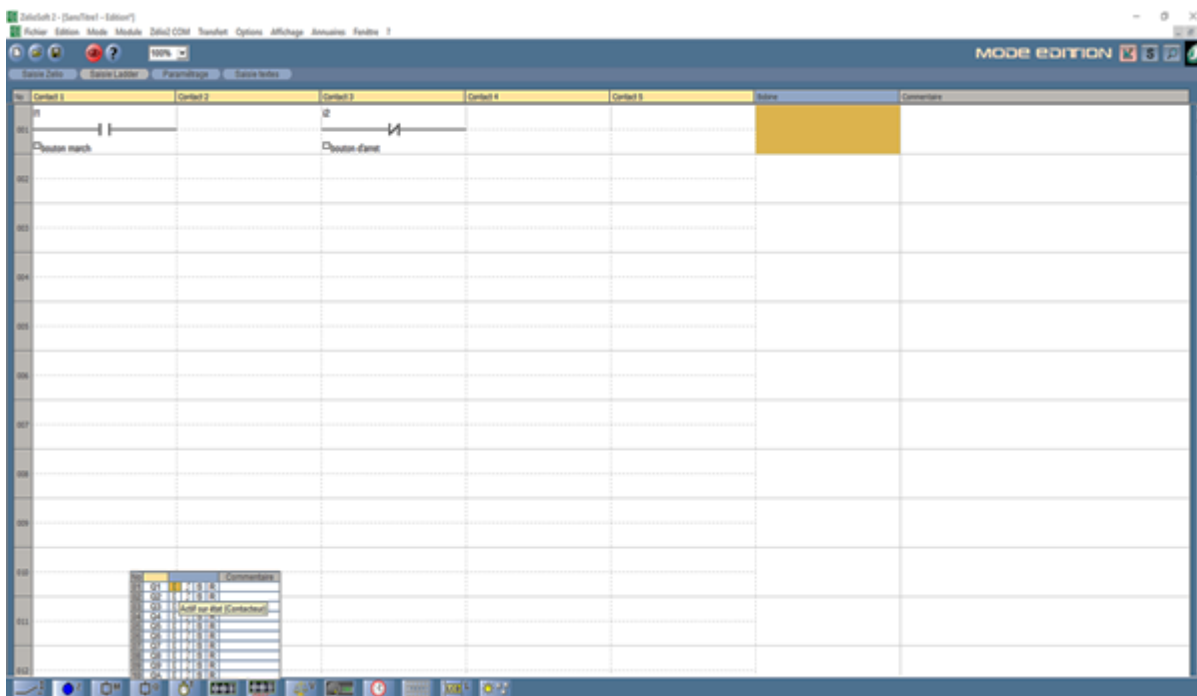
- Cliquer, par le bouton droit de la souris, sur le contact choisi, pour déterminer l'état fermé ou ouvert du contact au repos, l'état de contact par défaut est ouvert.



De la même façon, on ajoute un contact fermé pour l'arrêt.

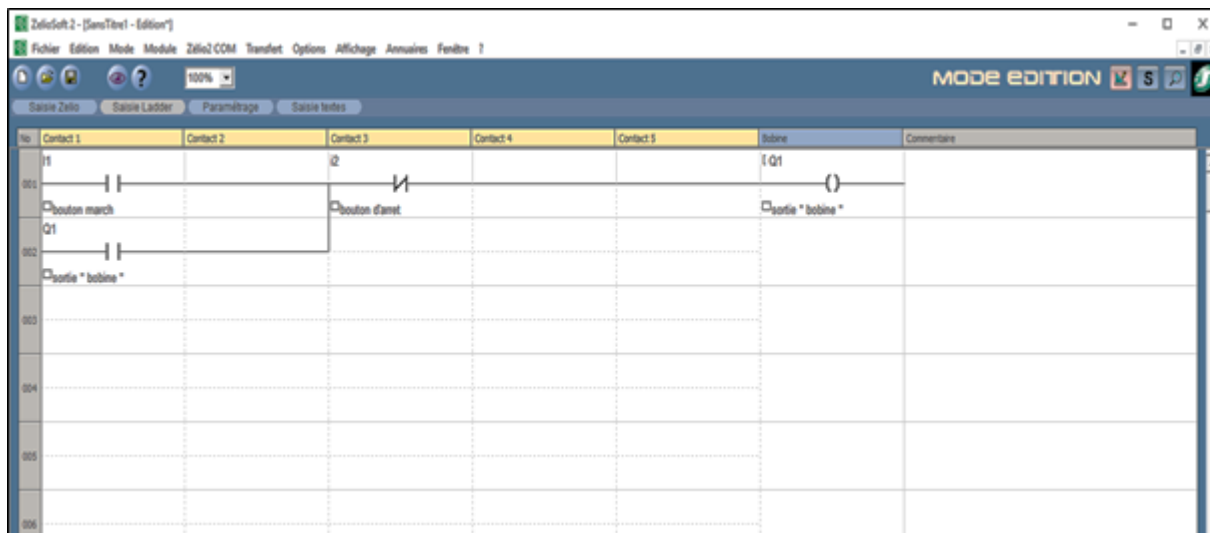
Ajouter une bobine (sortie) :

- Cliquer sur le bouton Q,
- Glisser la bobine choisie dans zone de pilotage.



Ajouter un contact de maintien, ouvert au repos, en parallèle avec le contact de marche.

Réaliser les liaisons électriques en cliquant sur les tronçons pour relier les différents éléments entre eux. On obtient le schéma Ladder de la figure suivante.



Etape 7 : Simulation

Pour passer en mode simulation, appuyer sur le bouton S.

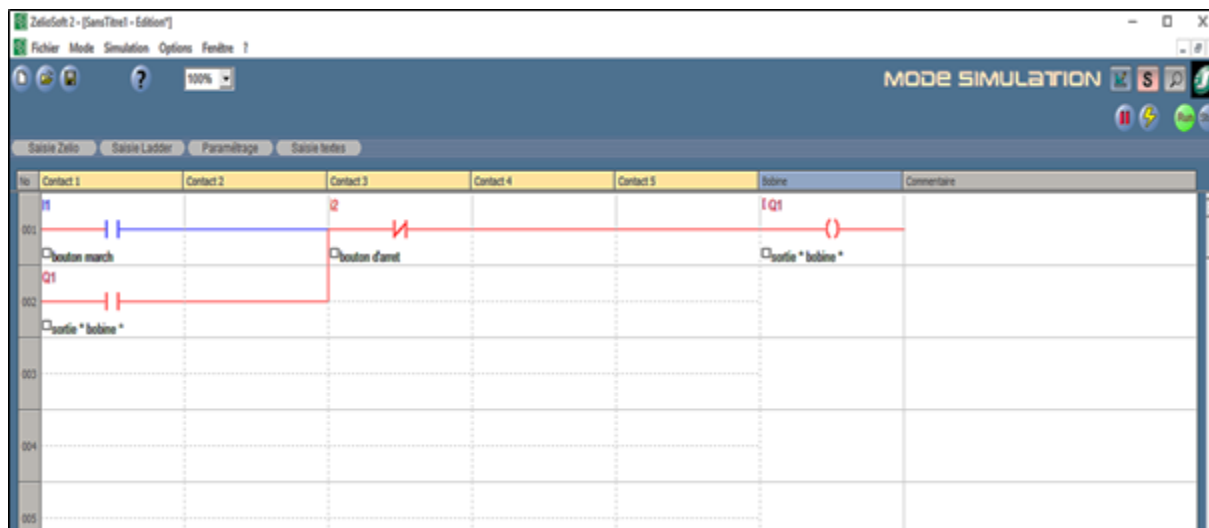


Pour passer en mode run, appuyer sur le bouton Run



Cliquer sur le contact de marche pour alimenter la bobine. Le contact de maintien se ferme et la bobine reste alimentée.

Pour arrêter le moteur, cliquer sur le contact d'arrêt.



Etape 8: Transfert du programme dans le module

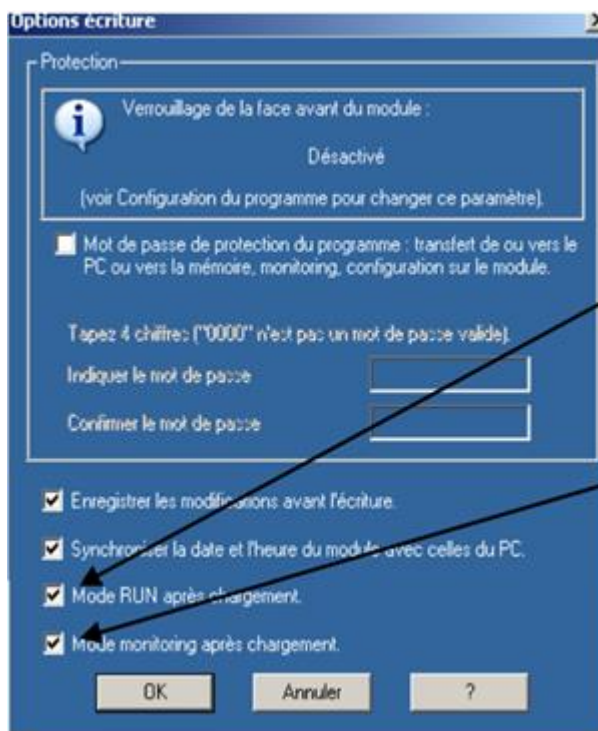
On ne peut transférer le programme que si l'on est dans le mode édition.

En mode édition, appuyer sur le bouton édition :



La page suivante est affichée. Appuyer sur le bouton transfert. Choisir l'option transférer programme. Puis choisir l'alternative PC/ Module.

Cliquer sur la touche OK.



Choisir **mode RUN après chargement** pour démarrer l'automate directement après le chargement du programme dans celui-ci.

Choisir également **Mode monitoring après le chargement** pour visualiser directement sur l'écran du PC l'évolution IN SITU du programme.

3.5. Conclusion

Dans ce chapitre nous avons présenté les langages de programmation de l'API à savoir le Grafcet et le Ladder, nous avons fondamentaux la dynamique de Grafcet. Ainsi que les principes du Ladder. Suive d'un exemple illustré de programmation des modes marche/arrêt d'un moteur sous le logiciel Zelio Sofet de Schneider.

CHAPITRE IV :
REALISATION D'UNE CHAINE
DE TRANSPORT

4.1. Introduction

L'objectif de notre projet, comme il a été mentionné en introduction, consiste à réaliser à échelle réduite (maquette) une partie d'une chaîne de transport de briques automatisée, voir figure 4.1.

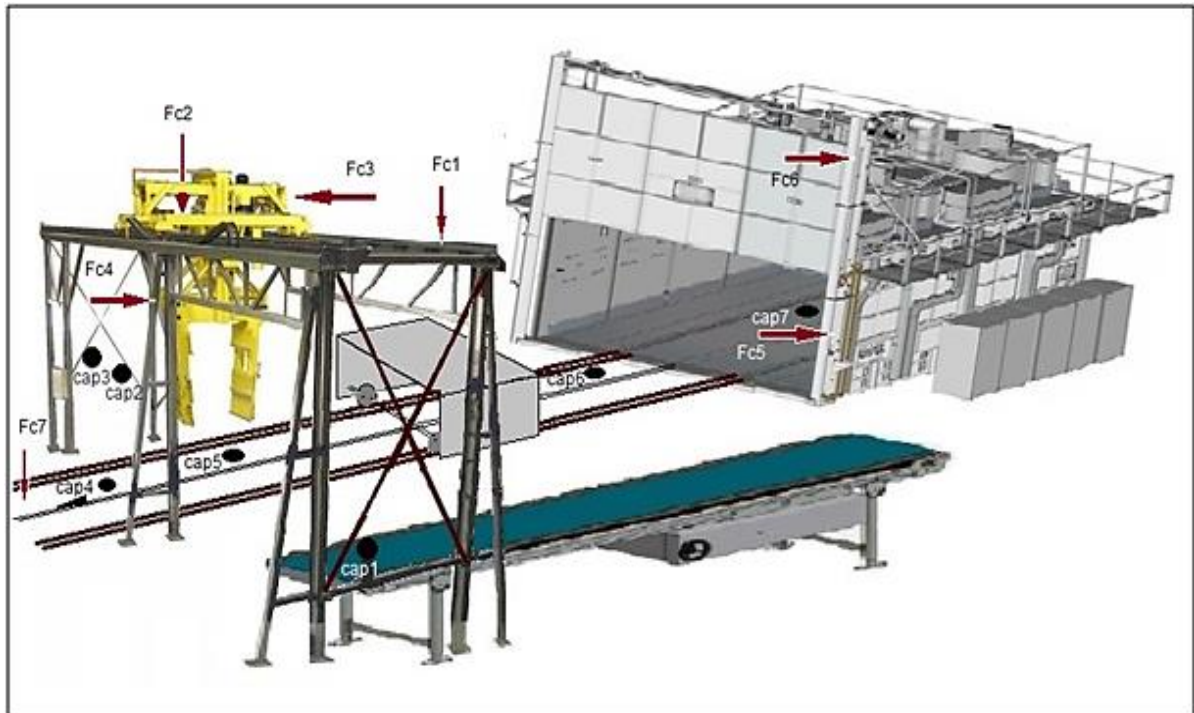


Figure 4.1. Schéma synoptique de la maquette

4.2. Cahier de charges

La maquette que nous allons réaliser est constituée de:

- un tapis roulant qu'amène la brique sèche,
- une pince qui déplace la brique sèche du tapis roulant vers un chariot,
- un pont roulant qui déplace la pince du tapis roulant vers le chariot et vis-versa,
- un chariot qui déplace la brique vers le four de cuisson,
- un dispositif d'avance chariot qui déplace, sur rails, le chariot vers le four de cuisson,
- un four de cuisson équipé d'une porte métallique qui s'ouvre pour faire entrer le chariot (plein de brique) puis se ferme,
- des actionneurs (moteurs à courant continu) pour la marche de la chaîne,
- des capteurs et des fins de course pour contrôler la marche et l'arrêt des actionneurs.

La fonction du tapis roulant consiste à amener la brique sèche à partir d'un hall de séchage. Il est entraîné par un moteur à courant continu (MCC). L'arrivée de la brique est

détectée par un capteur de proximité qui ordonne le MCC de s'arrêter pour décharger la brique du tapis roulant vers le chariot.

La pince, constituée de deux plaques métalliques, est entraînée par un MCC qui permet de déplacer cette dernière de bas en haut en présence de la brique. La pince descend jusqu'au niveau de la brique en position ouvert puis se ferme pour tenir la brique. La pince monte, ensuite, en plein charge puis descend au-dessus du chariot, après avoir été déplacée par le pont roulant. Enfin, la pince pose la brique sur le chariot. Tous ces mouvements de la pince sont contrôlés par des capteurs de proximité et des fins de course. La fermeture des plaques de la pince est effectuée par un vérin électrique entraîné par un MCC.

La fonction du pont roulant consiste à déplacer horizontalement la pince depuis le tapis roulant jusqu'au chariot et vis-versa. Le pont est entraîné par un MCC et contrôlé par deux fins de cours.

Le dispositif d'avance chariot, qui déplace le chariot, est entraîné par un MCC et contrôlé par quatre capteurs de proximité et un fin de course.

La porte du four s'ouvre et se ferme verticalement par un dispositif mécanique entraîné par un MCC et contrôlé par deux fins de course.

En fin, le fonctionnement global de la chaîne transporteuse de briques est contrôlée par un automate programmable industriel (API).

4.3 Liste du matériel utilisé

Le matériel utilisé (actionneurs, pré-actionneur, capteurs, fins de course et d'autres) pour réaliser cette maquette est donné dans le tableau 4.1.

Les actionneurs sont :

- un moteur à courant continu, démarrage directe, 1 sens de marche pour le tapis roulant,
- trois moteurs à courant continu, démarrage directe, 2 sens de marche (gauche/droite, descendre/monte et ferme/ouvrir) pour la pince,
- un moteur à courant continu, démarrage directe, 2 sens de marche pour le dispositif avance le chariot,
- un moteur à courant continu, démarrage directe, 2 sens de marche pour la porte du four.

Tableau 4.1 : Liste du matériel utilisé.

N	Type de matériel	Caractéristiques	Fonctions	Quantité
1	Automate programmable industriel	Schneider (SR3B261BD) 24 VDC, 16 entrées TOR ,10 sorties TOR	Automatisme de la chaîne	1
2	Pré-actionneurs	Relais (Schneider) électromagnétiques 24VDC	Relayage des actionneurs	10
3	Actionneurs	moteur à courant continu 12V, 2A	Entrainement des éléments de la chaîne	6
4	Capteurs	Proximité (Schneider)	Détection	7
5	Fins de course	NO / NC	Détection	7
6	Alimentation	220 VAC, 24 VDC	Alimentation électrique	2
7	Bouton poussoir	Bouton Marche Bouton arrêt	Marche et arrêt	2
8	Disjoncteur	disjoncteur magnéto thermique	Protection contre les surcharges et un court-circuit	1

La figure 4.2 représente la réalisation de la chaîne de la maquette

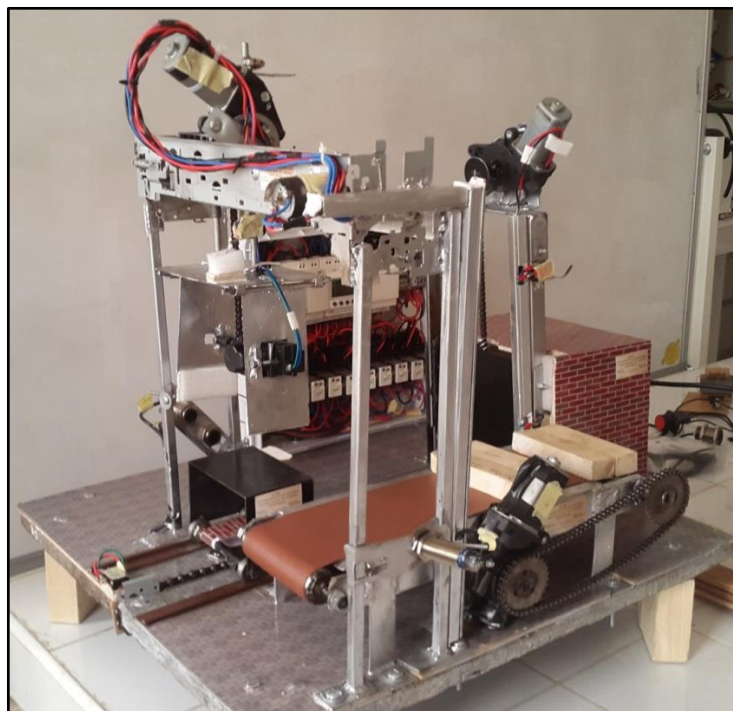


Figure 4.2. Projet de réalisation (maquette).

4.4. Description du cycle de fonctionnement

Le début de cycle commence par un appui sur le bouton poussoir de démarrage Bd, les actions du cycle se déroulent selon la chronologie suivante.

1. Le tapis roulant commence à tourner jusqu'à ce que le capteur CAP1 détecte la présence de la brique puis s'arrête, voir figure 4.4.
2. La pince descend, en position ouverte, juste au-dessus de la brique puis elle s'arrête par l'action de fin de course FC3, voir figure 4.5.
3. La pince se ferme pour tenir la brique, voir figure 4.6.
4. La pince monte, en pleine charge, vers le haut jusqu'au fin de course FC4 que arrête la pince à une certain hauteur, voir figure 4.7.
5. Le pont roulant déplace la pince jusqu'au-dessus du chariot puis s'arrête sur l'action du fin de course FC2, voir figure 4.8.
6. La pince descend puis s'arrête sur l'action du fin de course FC3, voir figure 4.9.
7. La pince s'ouvre pour déposer la brique sur le chariot, voir figure 4.10.
8. La pince remonte vers le haut et s'arrête par le fin de course FC4, voir figure 4.11.
9. Le pant roulant fait retourner la pince vers le tapis roulant puis il s'arrête sur l'action du fin de course FC1, en position suspendu.

Les actions 2 à 9 se répètent jusqu'au ce que le chariot soit remplie de briques. Le remplissage du chariot est achevé par le capteur de proximité CAP3, voir figure 4.12.

10. Une fois le chariot est rempli de briques, il commence à se déplacer vers le four de cuisson. Le capteur de proximité CAP6 fait arrêter le chariot juste devant la porte fermée de four, voir figure 4.13.
11. La porte du four s'ouvre vers le haut jusqu'au fin de course FC6, voir figure 4.14.
12. Le chariot entre dans le four. L'entrée totale du chariot est signalée par le capteur de proximité CAP7, voir figure 4.15
13. Une fois le chariot est à l'intérieur de four, la porte se ferme en descendant vers le bas jusqu'au fin de course FC5, voir figure 4.16.

Le dispositif avance chariot, qui avait déplacé le chariot vers le four, retourne à sa position initial pour avancer un autre chariot déjà remplie de brique et un autre cycle se répète.

Les figures 4.3 à 4.16 représentent les photos aux moments de déroulement des différentes actions du cycle de fonctionnement. La scène de déroulement des différentes actions du cycle de fonctionnement est filmée et partagée sur YouTube. Elle peut être vue et téléchargée en utilisant le lien suivant : <https://www.youtube.com/watch?v=v63335YODc8>

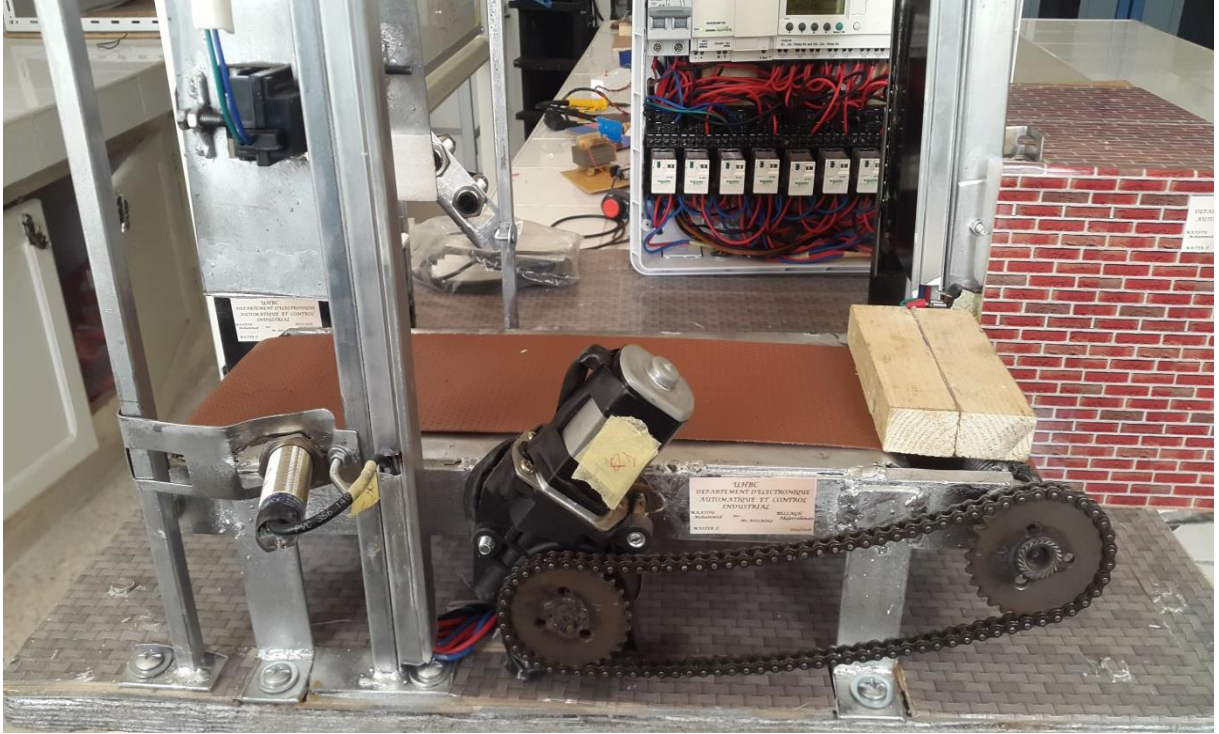


Figure 4.3 : Début du cycle de fonctionnement

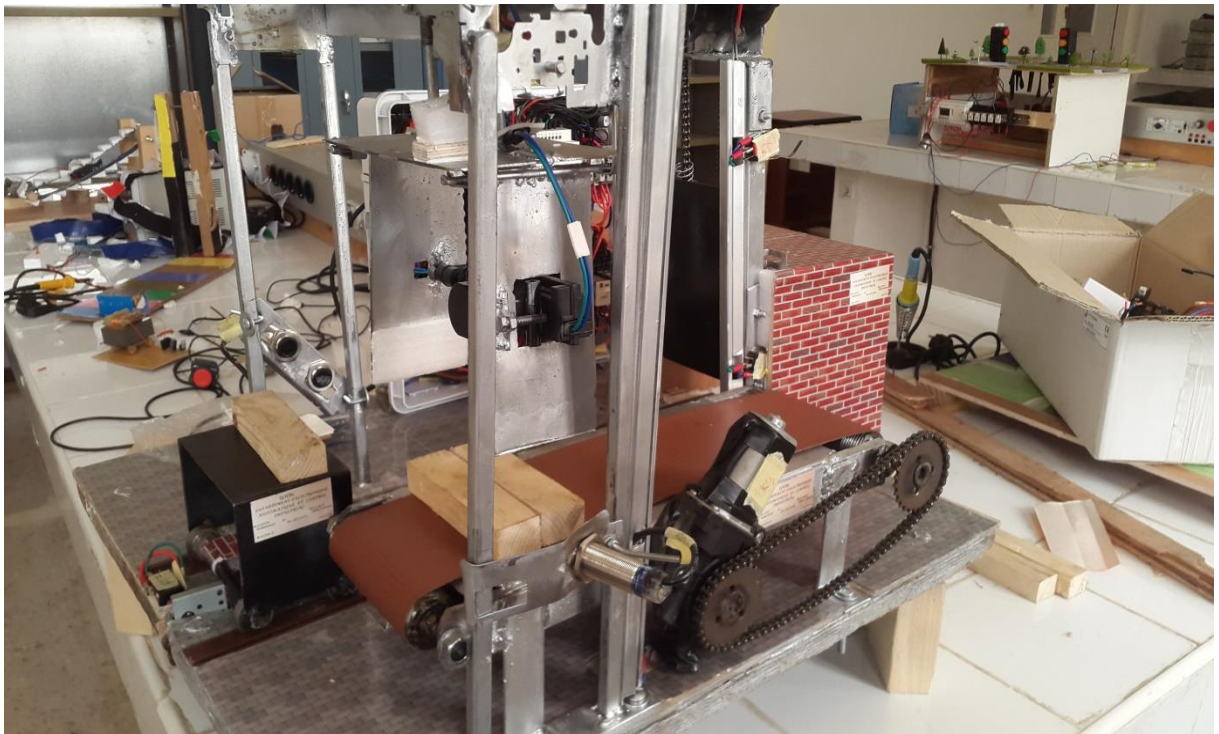


Figure 4.4 : Action 1 du cycle de fonctionnement

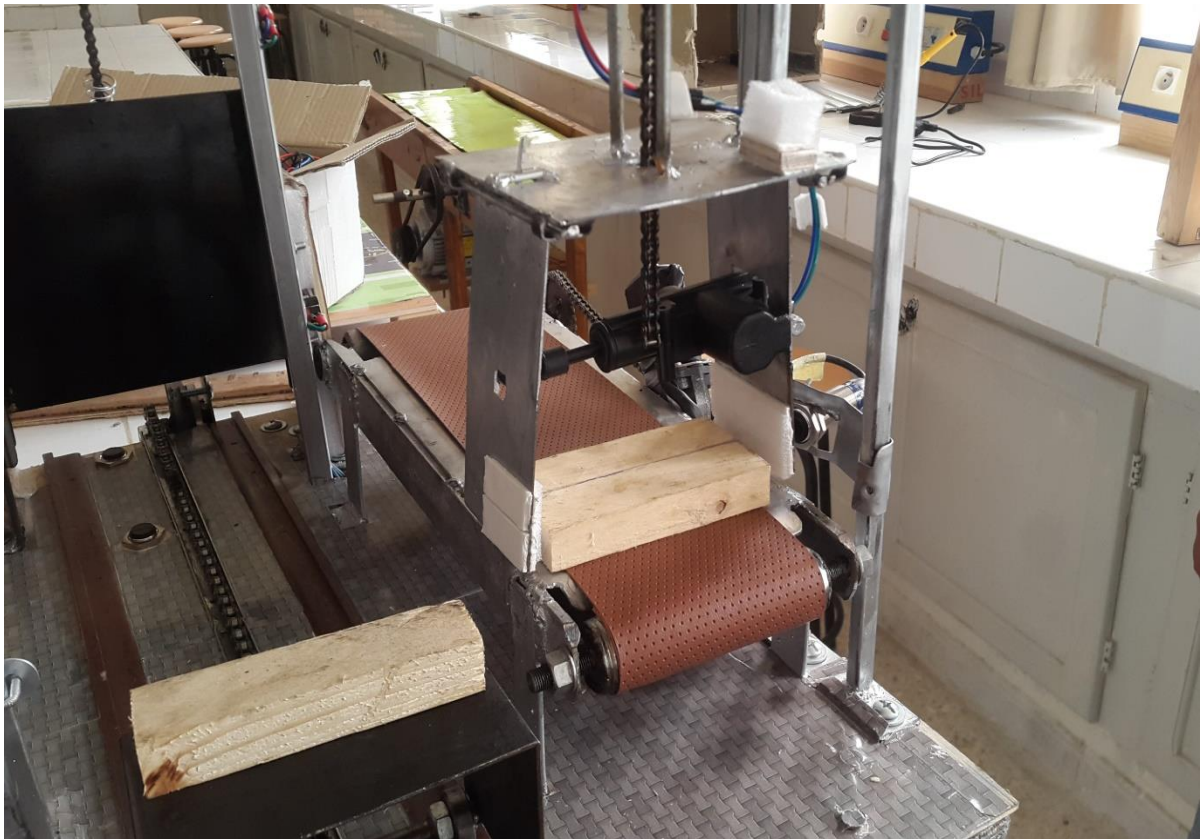


Figure 4.5 : Action 2 du cycle de fonctionnement



Figure 4.6 : Action 3 du cycle de fonctionnement

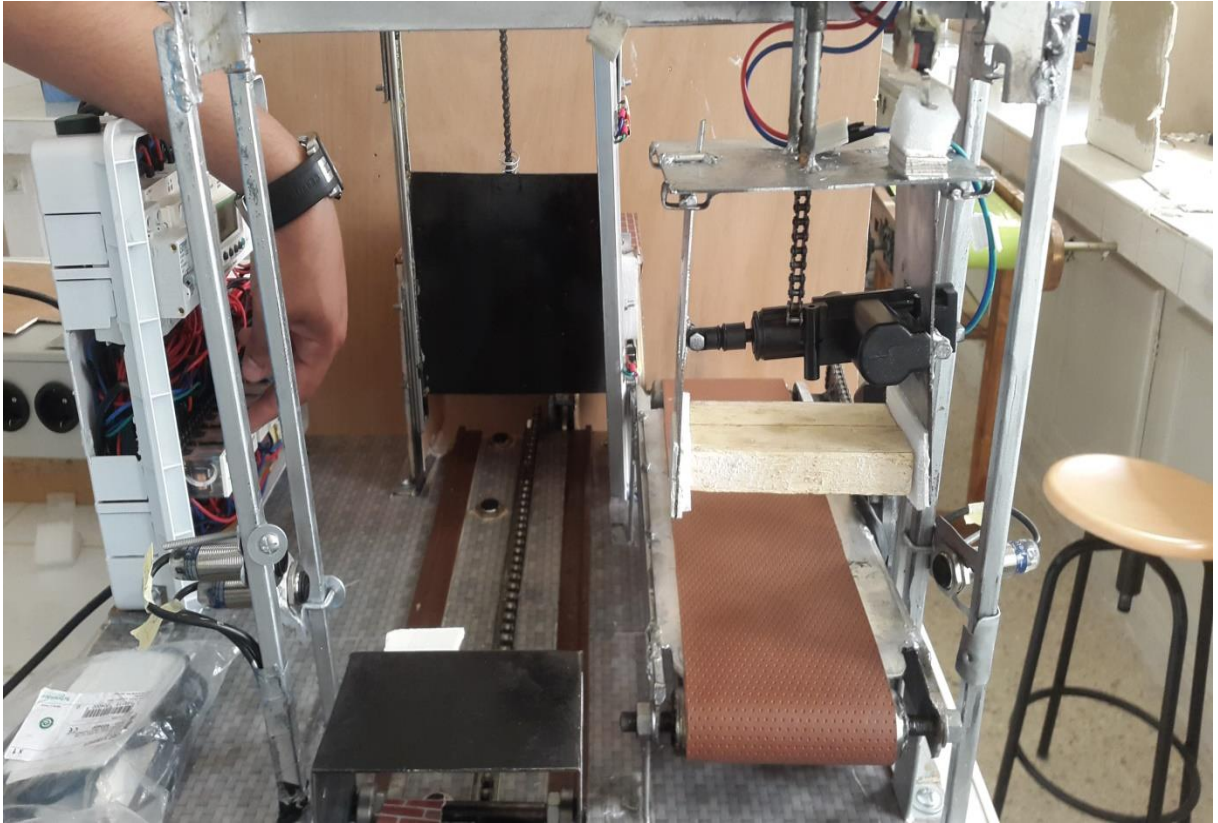


Figure 4.7 : Action 4 du cycle de fonctionnement

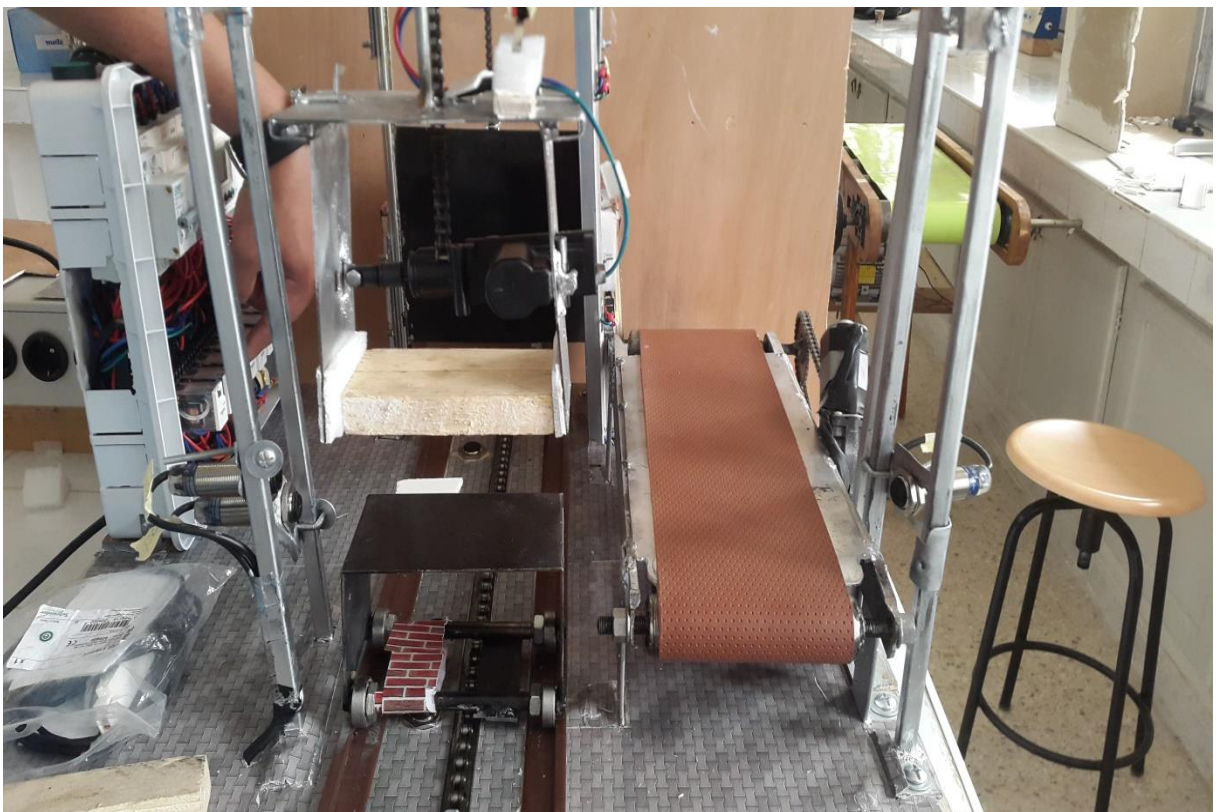


Figure 4.8 : Action 5 du cycle de fonctionnement

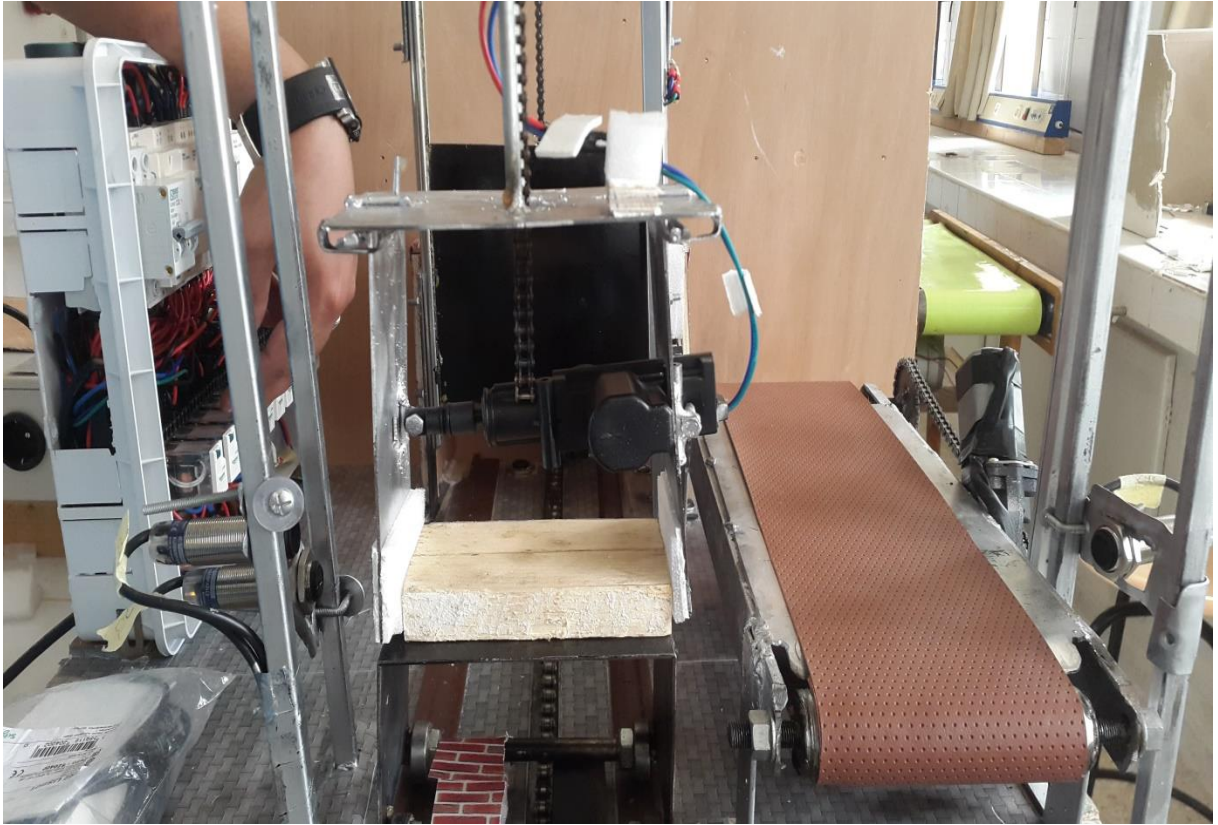


Figure 4.9 : Action 6 du cycle de fonctionnement

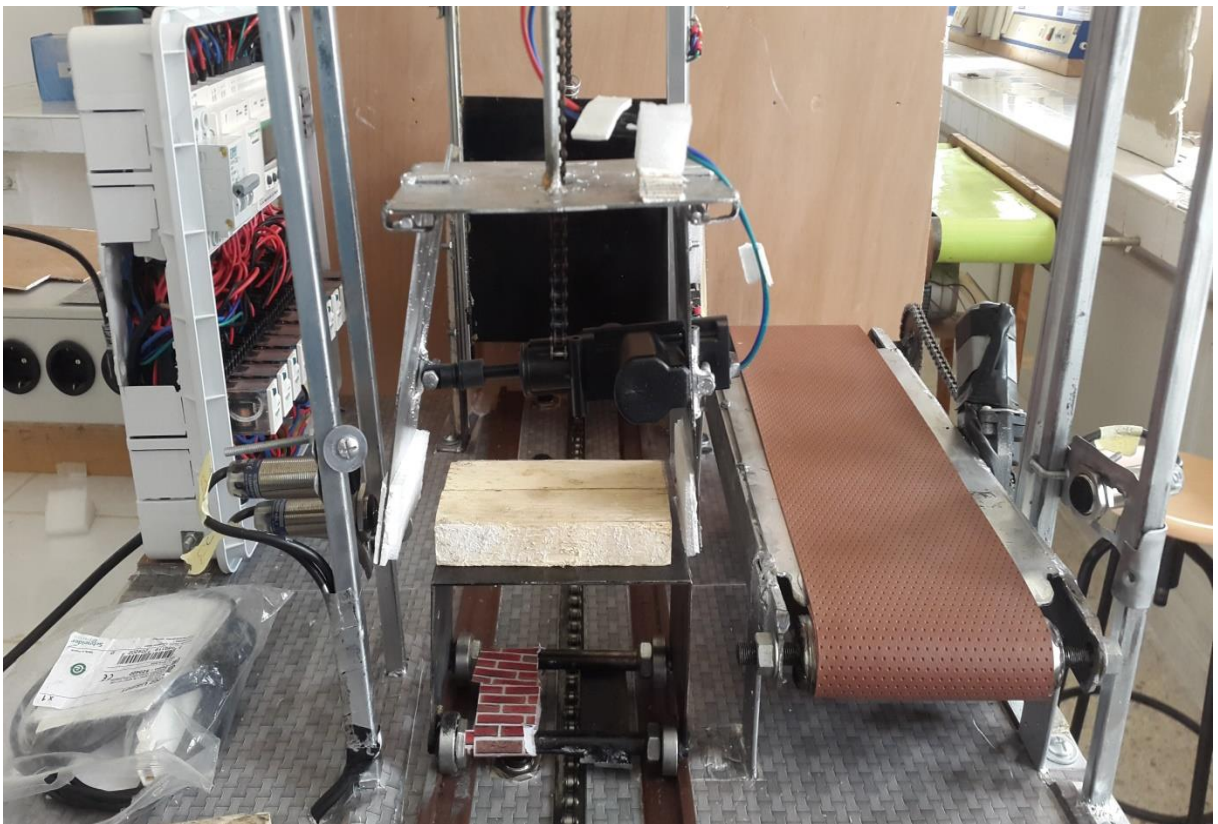


Figure 4.10 : Action 7 du cycle de fonctionnement

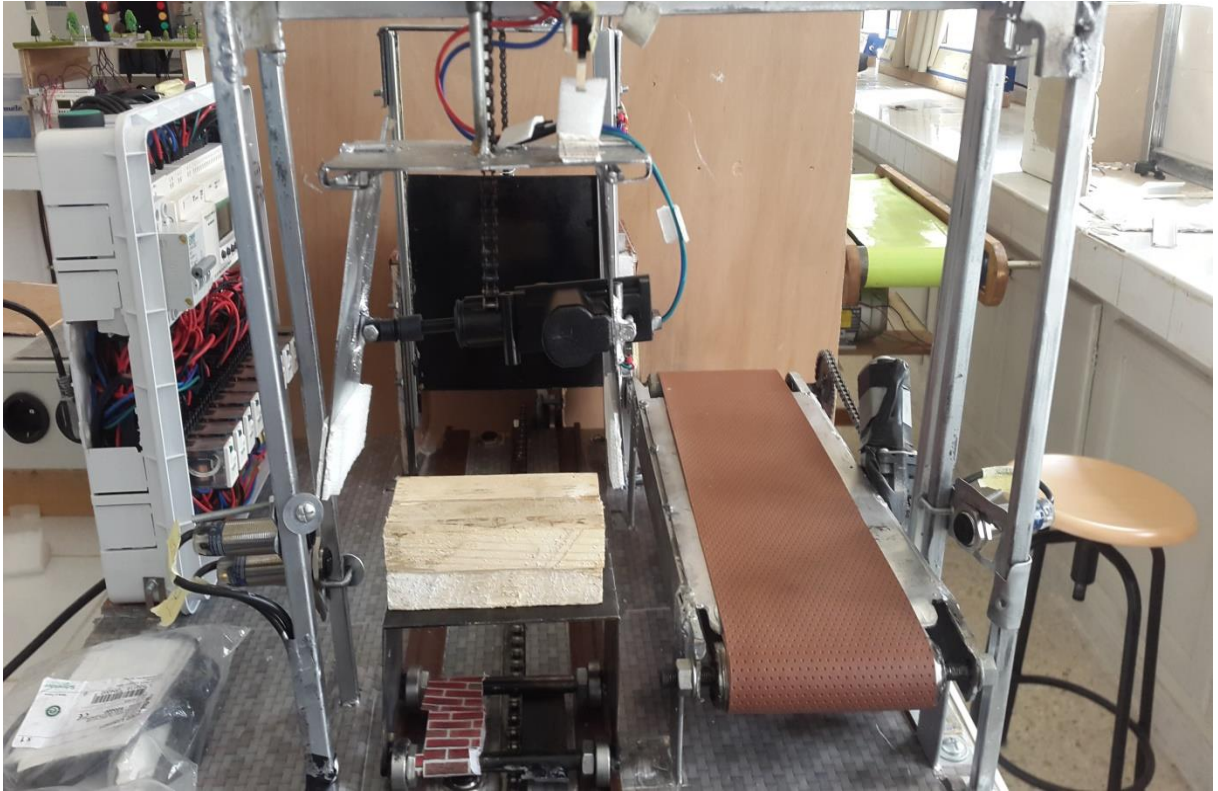


Figure 4.11 : Action 8 du cycle de fonctionnement



Figure 4.12 : Action 9 du cycle de fonctionnement



Figure 4.13 : Action 10 du cycle de fonctionnement



Figure 4.14 : Action 11 du cycle de fonctionnement



Figure 4.15 : Action 12 du cycle de fonctionnement



Figure 4.16 : Action 13 du cycle de fonctionnement

4.5. Schémas électriques

Les schémas électriques de commande et de puissance sont présents respectivement aux figures 4.17 et 4.18.

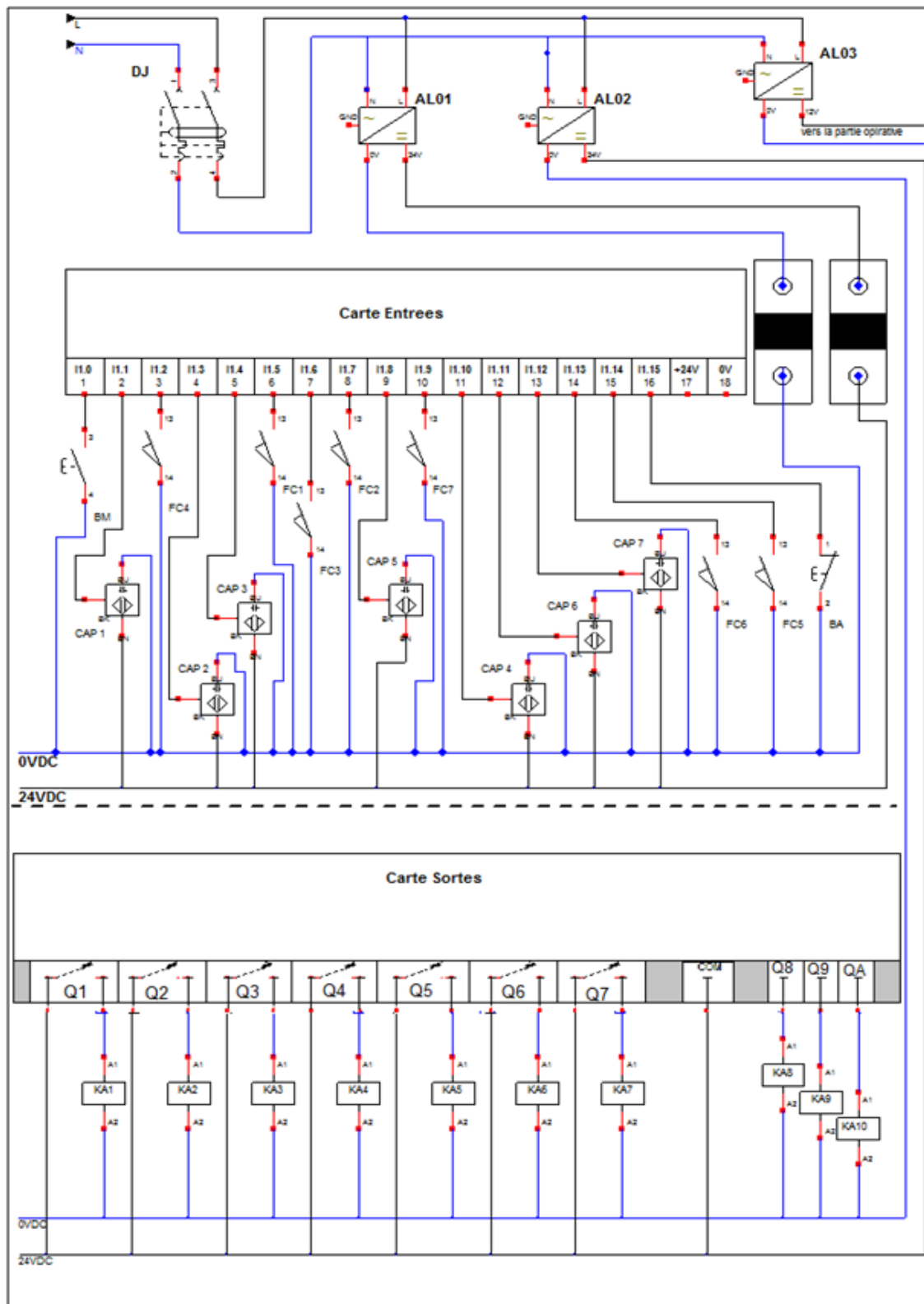


Figure 4.17 : Schéma de commande

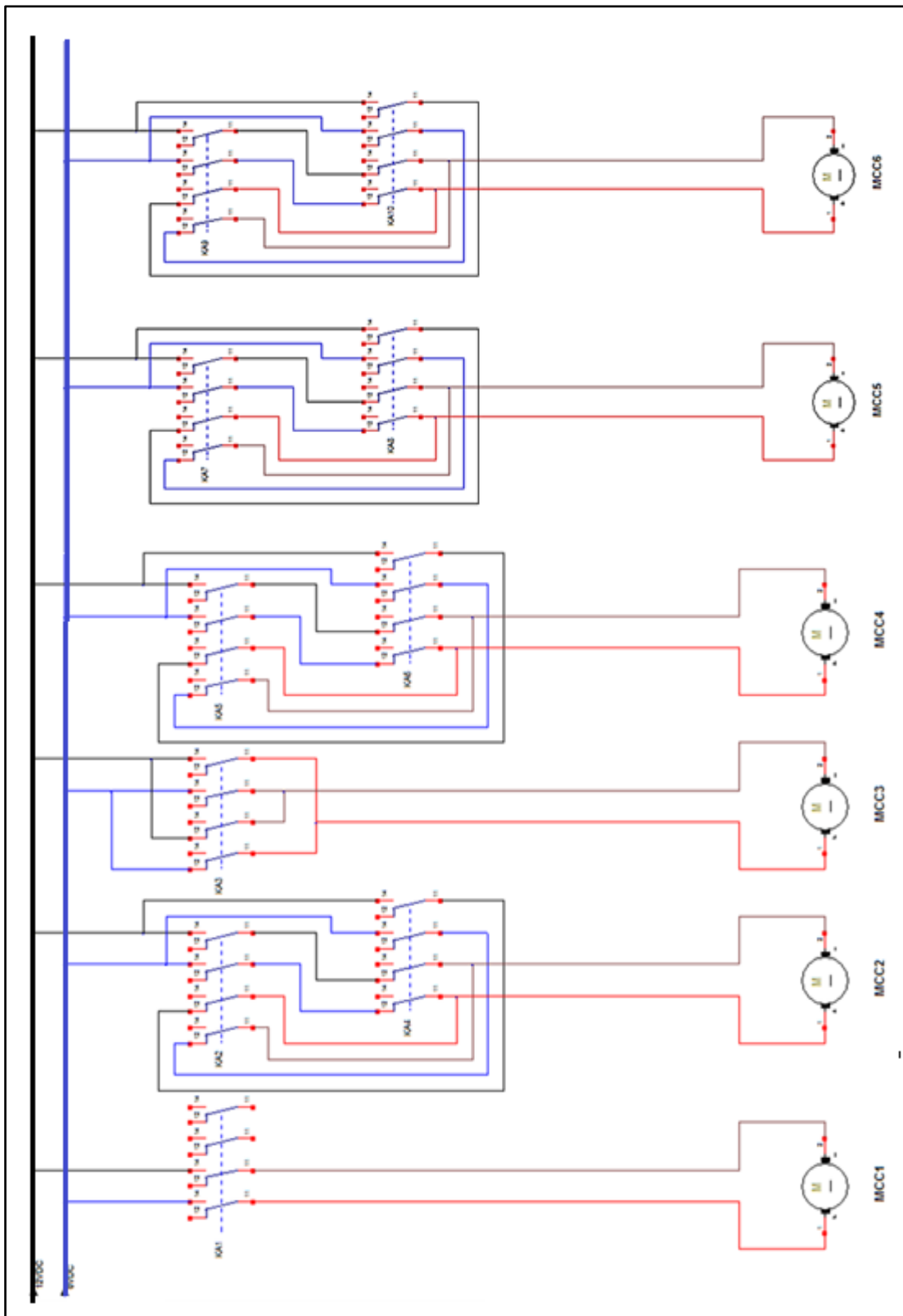


Figure 4.18 : Schéma de puissance

4.6. Grafcet

Les grafquets de niveaux 1 et 2 du cycle de fonctionnement sont représentés respectivement par les figures 4.19 à 4.22

Grafcet niveau 1

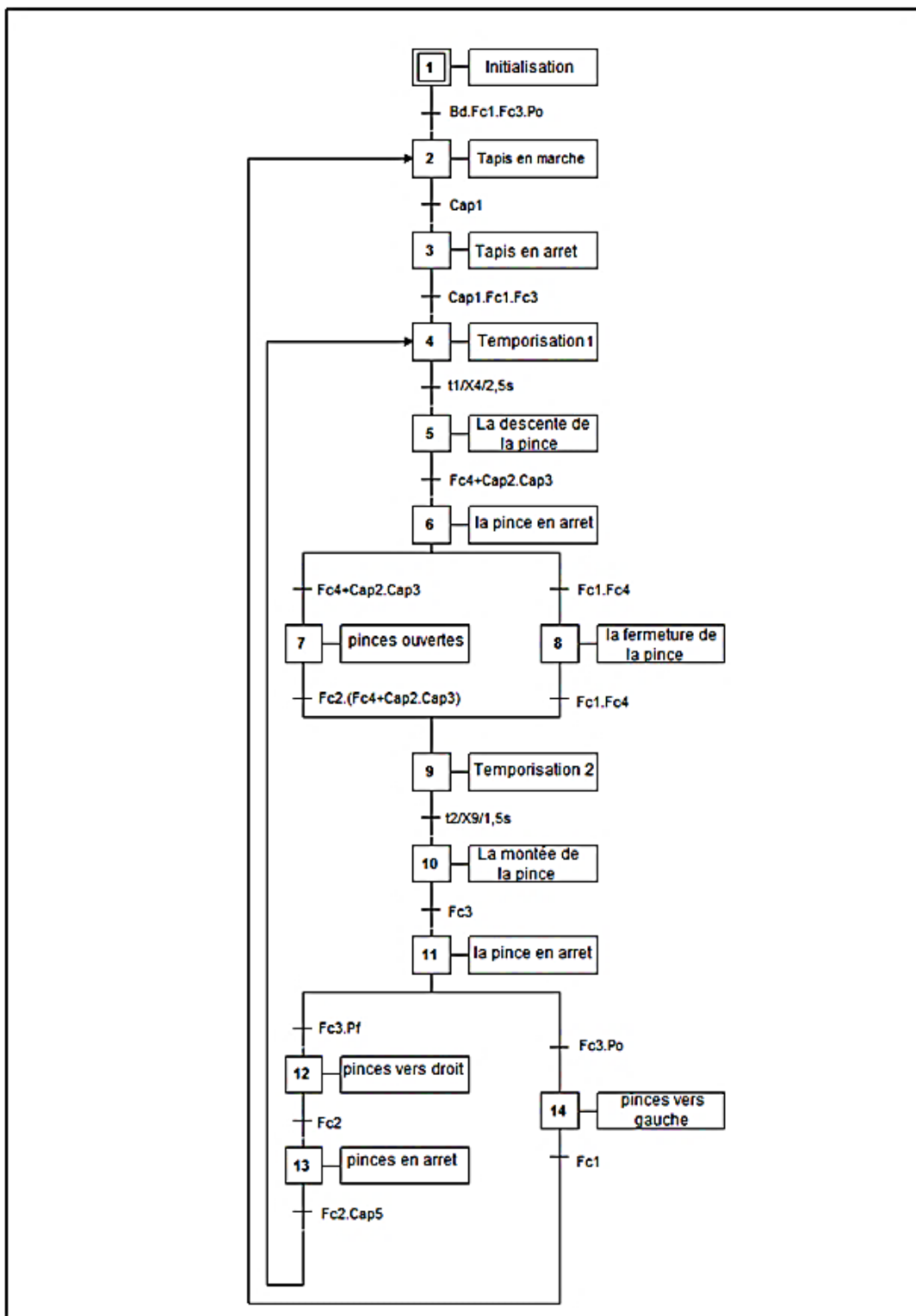


Figure 4.19 : Grafcet de niveaux 1

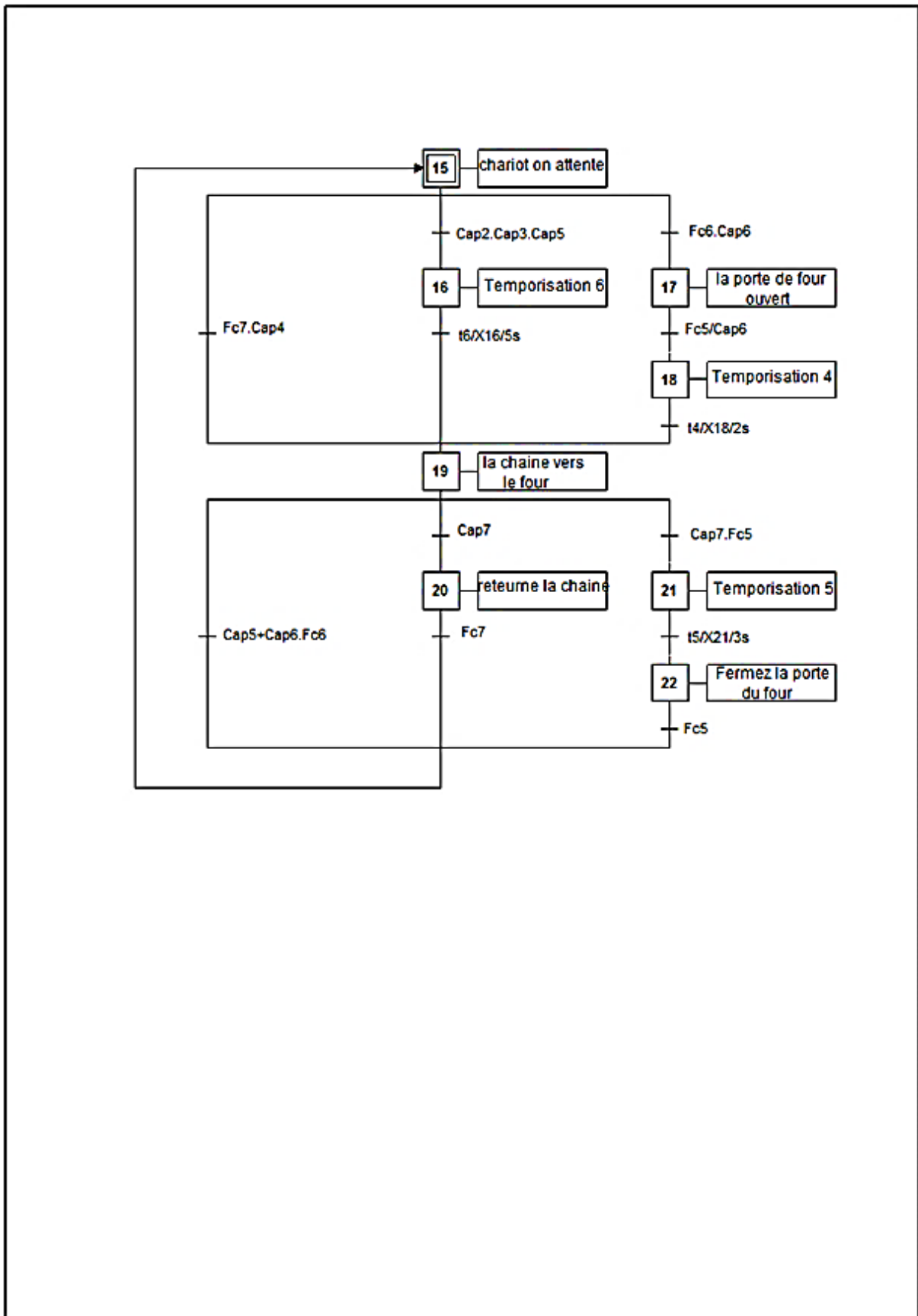


Figure 4.20 : Grafcet de niveaux 1 (suite)

Grafctet niveau 2

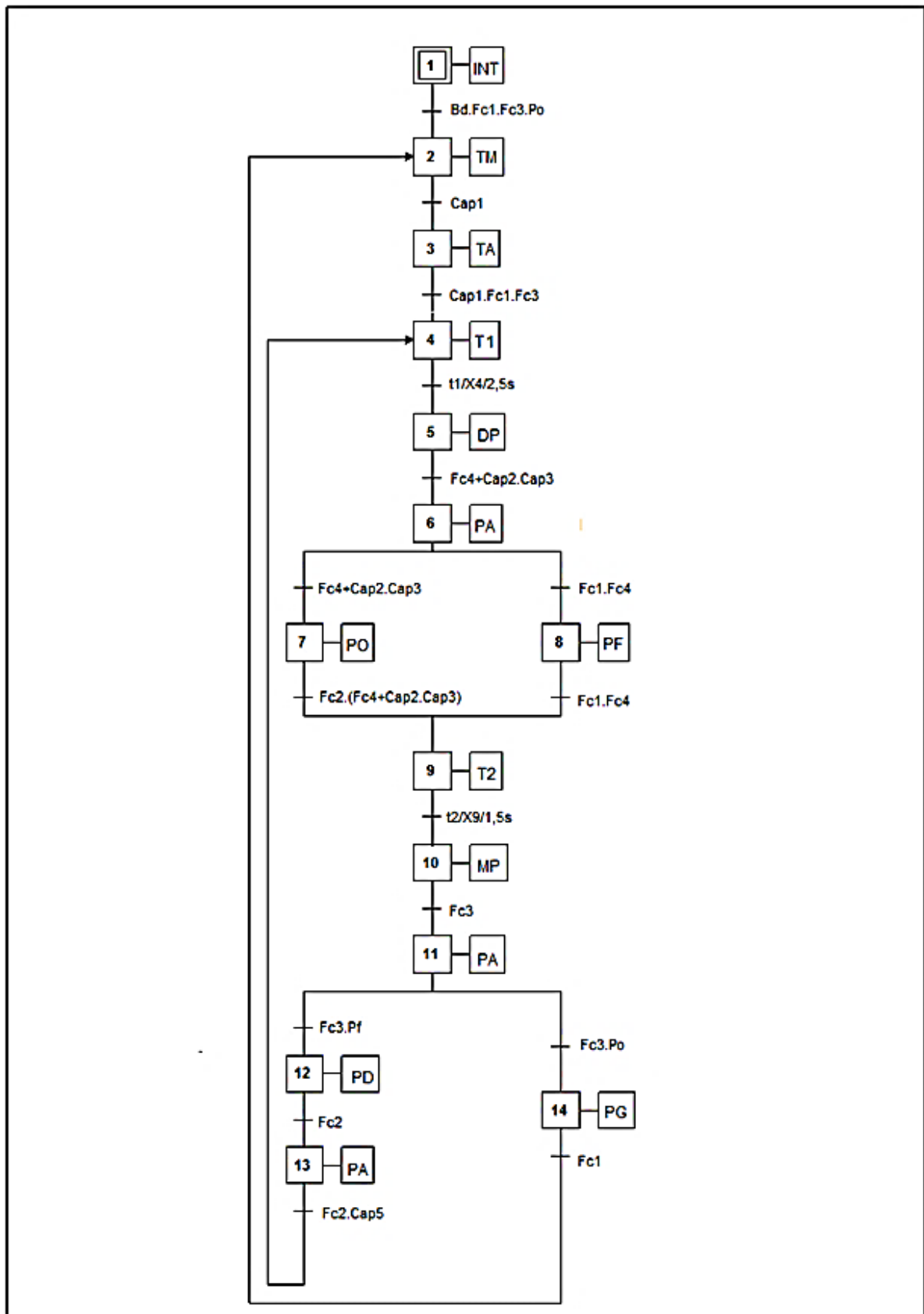


Figure 4.21 : Grafctet de niveaux 2

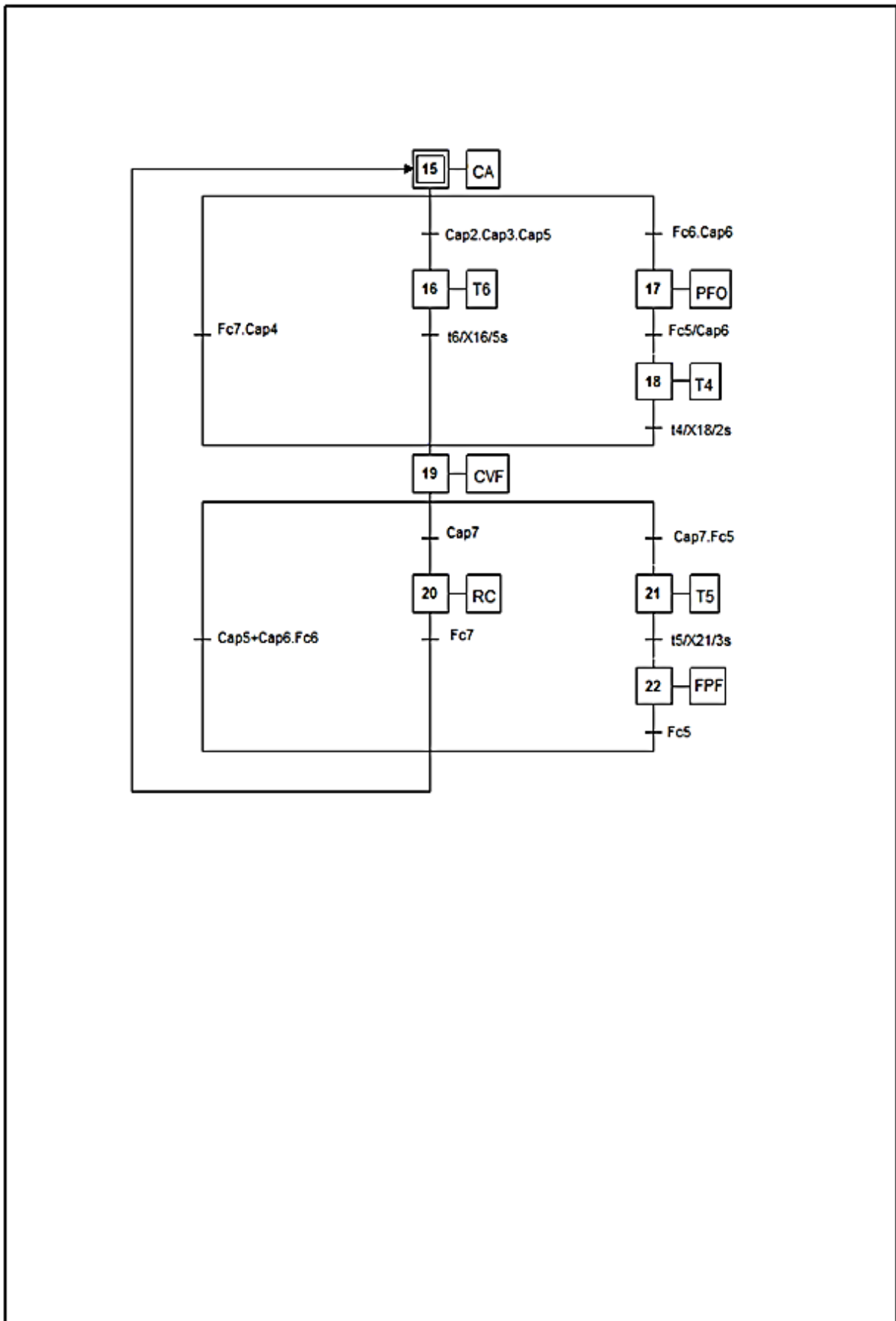


Figure 4.22 : Grafcet de niveaux 2 (suite)

Tableau.4.2: Les transitions.

Transitions	Commentaire
Bd	Bouton démarrage
Po	Pince ouvert
Fc1	fin de course 1
Fc2	fin de course 2
Fc3	fin de course 3
Fc4	fin de course 4
Fc5	fin de course 5
Fc6	fin de course 6
Fc7	fin de course 7
Cap1	capteur de proximité 1
Cap2	capteur de proximité 2
Cap3	capteur de proximité 3
Cap4	capteur de proximité 4
Cap5	capteur de proximité 5
Cap6	capteur de proximité 6
Cap7	capteur de proximité 7

Tableau.4.3: Les actions en niveau 1 et niveau 2

Action	Niveau 1	Niveau 2
1	Initialisation	INT
2	Tapis en marche	TM
3	Tapis en arrêt	TA
4	Temporisation 1	T1
5	La descente de la pince	DP
6	La pince en arrêt	PA
7	Pinces ouvertes	PO
8	La fermeture de la pince	PF
9	Temporisation 2	T2
10	La montée de la pince	MP
11	La pince en arrêt	PA
12	Pinces vers droit	PD
13	Pince en arrêt	PA

14	Pinces vers gauche	PG
15	Chariot en attente	CA
16	Temporisation 6	T6
17	La porte de four ouvert	PFO
18	Temporisation 4	T4
19	La chaîne vers le four	CVF
20	Retourne la chaîne	RC
21	Temporisation 5	T5
22	Fermez la porte de four	FPF

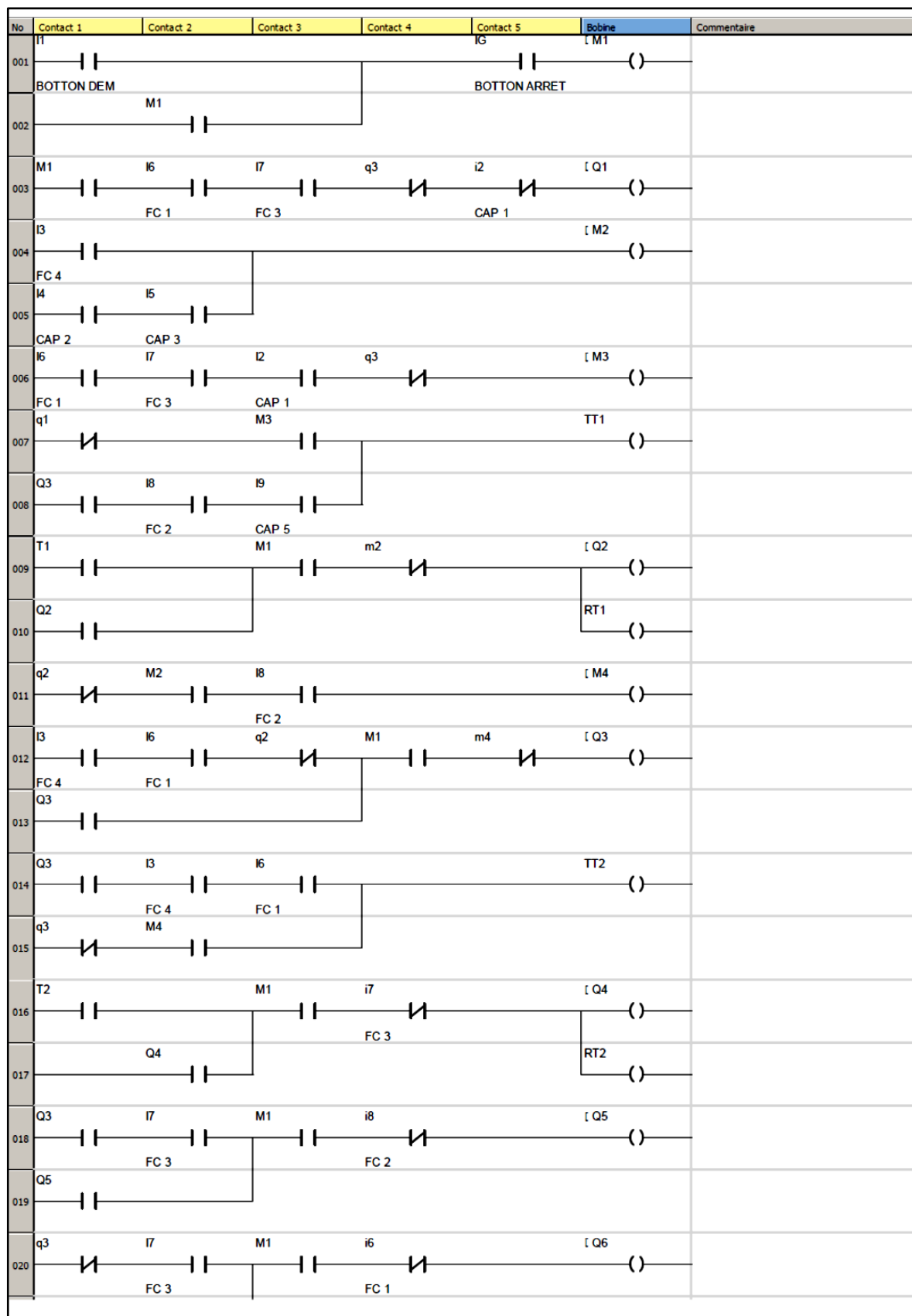
4.7. Programmation

La programmation des différentes actions du cycle de fonctionnement est effectuée en langage Ladder par le logiciel Zelio Soft 2. Dans ce qui suit, nous présentons le programme complet.

Les entrées et sorties physiques de l'API sont données aux tableaux 4.4 et 4.5.

Les temporisations des différentes actions du cycle de fonctionnement sont illustrées par les figures 4.23 à 4.28.

Schéma du programme en langage Ladder



No	Contact 1	Contact 2	Contact 3	Contact 4	Contact 5	Bobine	Commentaire
021	Q6						
022	IA FC 7 M5	IB CAP 4		i9 CAP 5		[M5	
023							
024	i5					TT6	
025	CAP 3					RT6	
026	i9 CAP 5	I4 CAP 2 M8	T6			[M8	
027					iC CAP 6	[M6	
028	M6						
029	IF FC 5	IC CAP 6 T4				TT4	
030				iD CAP 7		[M7	
031		M7				RT4	
032	M5					TT8	
033	M6					TT9	
034	M7					TTA	
035		T8					
036		T9			M1	[Q7	
037		TA					
038	i9 CAP 5					RT8	
039	IC CAP 6					RT9	
040	ID CAP 7					RTA	

No	Contact 1	Contact 2	Contact 3	Contact 4	Contact 5	Bobine	Commentaire
041	i9 CAP 5	IB CAP 4				I17 ()	
042	iC CAP 6						
043		T7				I M9 ()	
044			q7			RT7 ()	
045		ID CAP 7 M9		M1	iA FC 7	I Q8 ()	
046							
047							
048	IE FC 6	IC CAP 6	M1	iF FC 5		I Q9 ()	
049							
050	ID CAP 7	IF FC 5				TT5 ()	
051		T5	M1	iE FC 6		I QA ()	
052						RT5 ()	

Tableau 4.4 : Entrées physiques de l'API



























No	Symbole	Fonction	Verrou	Paramètres	Localisation (L/C)	Commentaire
I1		Entrées TOR	---	Pas de paramètres	(1/1)	BOTTON DEM
I2		Entrées TOR	---	Pas de paramètres	(3/5) (6/3)	CAP 1
I3		Entrées TOR	---	Pas de paramètres	(4/1) (12/1) (14/2)	FC 4
I4		Entrées TOR	---	Pas de paramètres	(5/1) (26/2)	CAP 2
I5		Entrées TOR	---	Pas de paramètres	(5/2) (24/1) (25/1)	CAP 3
I6		Entrées TOR	---	Pas de paramètres	(3/2) (6/1) (12/2) (14/3) (20/4)	FC 1
I7		Entrées TOR	---	Pas de paramètres	(3/3) (6/2) (16/4) (18/2) (20/2)	FC 3
I8		Entrées TOR	---	Pas de paramètres	(8/2) (11/3) (18/4)	FC 2
I9		Entrées TOR	---	Pas de paramètres	(8/3) (22/4) (26/1) (38/1) (41/1)	CAP 5
IA		Entrées TOR	---	Pas de paramètres	(22/1) (45/5)	FC 7
IB		Entrées TOR	---	Pas de paramètres	(22/2) (41/2)	CAP 4
IC		Entrées TOR	---	Pas de paramètres	(27/5) (29/2) (39/1) (42/1) (48/2)	CAP 6
ID		Entrées TOR	---	Pas de paramètres	(30/4) (40/1) (45/2) (50/1)	CAP 7
IE		Entrées TOR	---	Pas de paramètres	(48/1) (51/4)	FC 6
IF		Entrées TOR	---	Pas de paramètres	(29/1) (48/4) (50/2)	FC 5
IG		Entrées TOR	---	Pas de paramètres	(1/5)	BOTTON ARRET

Tableau 4.5 : Sorties physiques de l'API

No	Symbole	Fonction	Rémanence	Localisation (L/C)	Commentaire
Q1		Sorties TOR	Non	(3/6) (7/1)	
Q2		Sorties TOR	Non	(9/6) (10/1) (11/1) (12/3)	
Q3		Sorties TOR	Non	(3/4) (6/4) (8/1) (12/6) (13/1) (14/1) (15/1) (18/1) (20/1)	
Q4		Sorties TOR	Non	(16/6) (17/2)	
Q5		Sorties TOR	Non	(18/6) (19/1)	
Q6		Sorties TOR	Non	(20/6) (21/1)	
Q7		Sorties TOR	Non	(36/6) (44/3)	
Q8		Sorties TOR	Non	(45/6) (47/1)	
Q9		Sorties TOR	Non	(48/6) (49/1)	
QA		Sorties TOR	Non	(51/6) (52/1)	

Temporisateurs

Les temporisations qui fonctionnée dans le programme

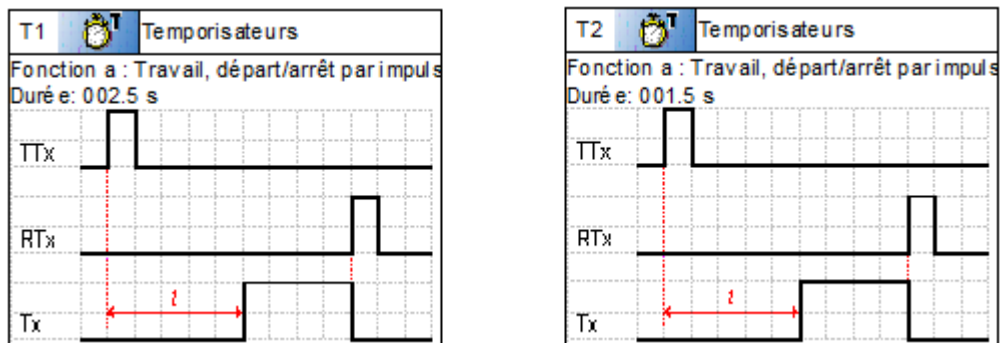


Figure 4.23 : Les temporisations des actions T1 et T2

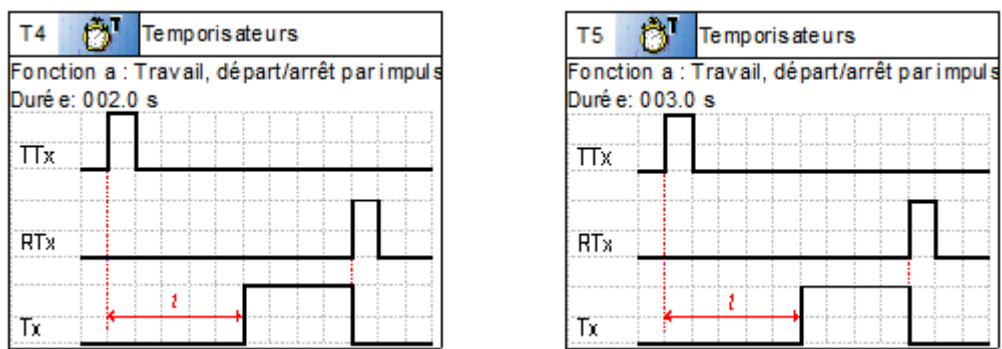


Figure 4.24 : Les temporisations des actions T4 et T5

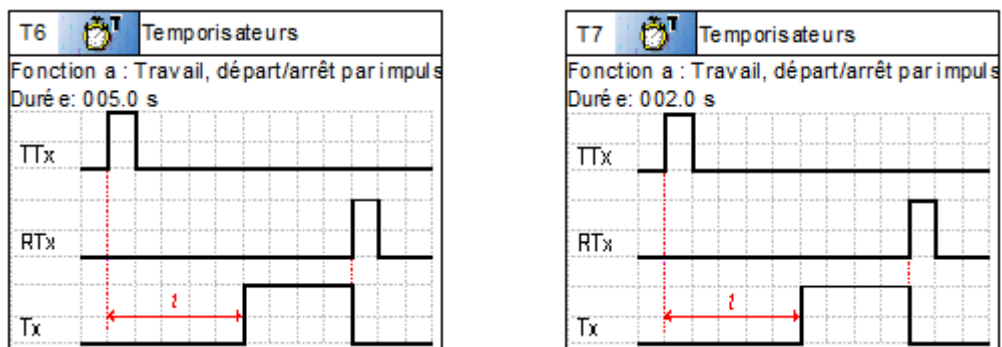


Figure 4.25 : Les temporisations des actions T6 et T7

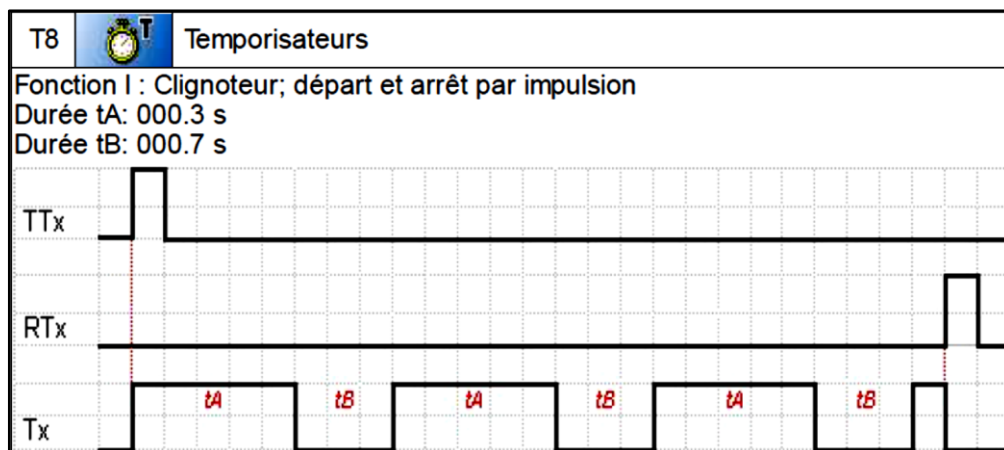


Figure 4.26 : Temporisation de l'action T8

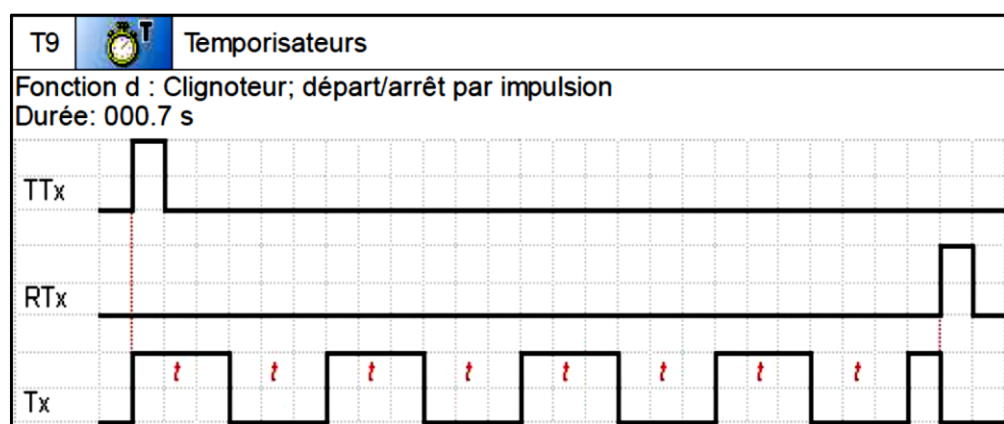


Figure 4.27 : Temporisation de l'action T9

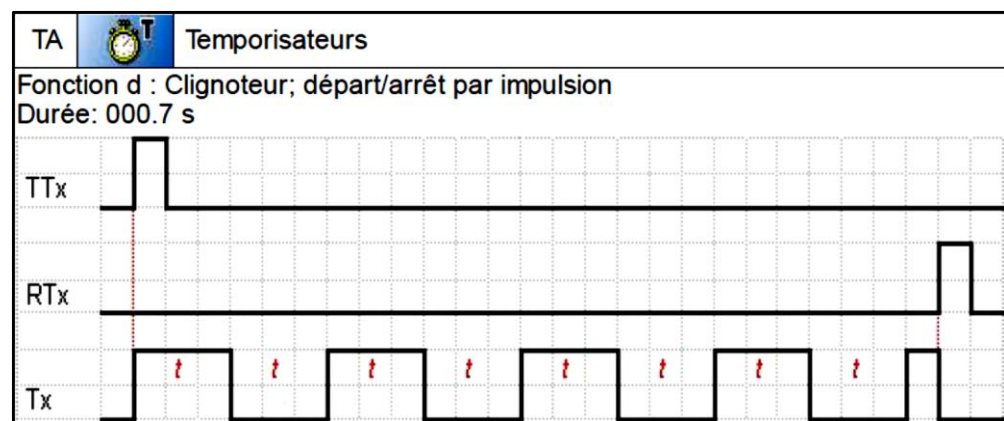


Figure 4.28 : Temporisation de l'action TA

4.8. Conclusion

Dans ce chapitre nous avons présenté le cahier de charge de notre projet et les outils utilisés dans la maquette. Présentation d'une suite illustrée des actions de fonctionnement, et ensuite le schéma électrique de câblage, le schéma de Grafcet et le programme de LADDER.

CONCLUSION GENERALE

Conclusion Générale

Au cours de ce projet de fin d'étude, nous avons pu concrétiser notre idée qui consiste à réaliser une maquette d'une chaîne de transport totalement automatisée. Nous sommes arrivés à réaliser ce projet grâce aux connaissances théoriques que nous avons acquies au cours de notre formation académique d'une part, et d'autre part, grâce aux informations que nous avons apprises durant notre stage de fin d'étude qui s'est déroulé dans une entreprise de fabrication de brique. Nous avons appris beaucoup de choses sur le fonctionnement des systèmes automatisés. Nous avons également appris beaucoup de choses sur les API et leur programmation.

Web Bibliographiques

[01] « Structure d'un système automatisé »

[http://foxi31.ovh.org/dl/2/ISI/04\)%20Structure%20d'un%20systeme%20automatise.pdf](http://foxi31.ovh.org/dl/2/ISI/04)%20Structure%20d'un%20systeme%20automatise.pdf)

[02] « Les automates programmables »

http://www.groupeisf.net/Automatismes/Automatesprogrammables/API_ATTOL/Bases_automatismes/an9_seq1_Place_et_role_de_l_API.ppt.

[03] <http://www.fichier-pdf.fr/2011/03/17/api/api.pdf>

[04] http://www.larmand.fr/fichiers/Ancien_site/enseigne/ressources/techno/bourse%20cours/COURS/automate%20programmable%20industriel%20introduction.pdf.

[05] « logiciel zelio soft 2 »

http://www.graczyk.fr/lycee/IMG/pdf/11-12_DR_Tutoriel_Zeliosoft.pdf

http://sitelec.org/download_page.php?filename=schneider/zelio.pdf

<http://www.schneider-electric.fr>

[06] <http://slideplayer.fr/slide/3249822/>

Autre référence :

[07] « Les automates programmables industriels »

Cours automatisation : de M.Benzouaoui Sidahmed enseignant de département d'électronique

[08] « Grafcet »

Livre : Du Grafcet aux réseaux de Petri ‘‘René David et Hassane Alla ‘‘