

**République Algérienne Démocratique et Populaire**  
**Ministère de l'enseignement supérieur et de la recherche scientifique**



**UNIVERSITE MOULOU D MAMMERI DE TIZI-OUZOU**

**FACULTE DE GENIE ELECTRIQUE ET DE L'INFORMATIQUE**

**DEPARTEMENT D'ELECTRONIQUE**

## **Mémoire de fin d'Etudes**

Présenté pour l'obtention du **diplôme de Master Professionnel**

**En : Génie Electrique**

**Spécialité : Electronique industrielle**

**Par : Melle.DJETTOU Zahra**

### **Thème**

**Automatisation d'une installation d'alimentation en  
pression des machines d'insertion automatique des  
cartes électroniques au niveau de l'usine BOMARE  
COMPANY**

**Proposé par :**

Mr.LOUNES Ahmed

**Dirigé par :**

Mr.HAMICHE Hamid.

Promotion 2017-2018

*Je dédie ce travail*

*À mon compagnon de vie et mon cher mari Mohamed qui m'a  
toujours aidé, encouragé et orienté.*

*A mon trésor Rayan qui m'inspire la volonté.*

*A mes chers parents qui m'ont toujours soutenu.*

*A mes beaux-parents qui m'ont beaucoup encouragé.*

*À mes frères Chafaa, Tarek et Amine.*

*À Samir, Hakim et sa femme Amel, Nassim, Karim et Mehdi.*

*A ma sœur Wafia, son mari Farid et leurs petit prince Mayes.*

*A mon cousin Yacine.*

*A mes copines Lila, Lilia.*

*Zahra*

# Remerciements

À Dieu, le tout puissant, nous rendons grâce pour nous avoir donné santé, patience et la volonté pour réaliser ce modeste travail.

Je remercie, **Mr M.HASSENA** à qui j'exprime ma profonde gratitude pour ses encouragements, son aide, sa patience et ses conseils, ce travail n'existerait pas sous cette forme sans sa contribution.

Mes profonds remerciements s'adressent à mon promoteur **Mr H.HAMICHE** *enseignant au département électronique à L'UMMTO* qui, de par sa patience et ses conseils avisés, a fortement contribué à la réussite de ce travail

Je tiens aussi à remercier mon Co-promoteur, **Mr A.LOUNES**, *Responsable de Production a Bomare Company*. Je le remercie pour son accueil chaleureux au sein de l'entreprise, sa générosité, sa patience, sa disponibilité et aussi tous les conseils qu'il a pu me prodiguer.

Je tiens a remercié **Mr M.RAHALI** *Responsable de l'unité production des cartes électroniques chez Bomare Company*, pour son accueil au sein de l'unité production.

Je remercie aussi toute l'équipe de l'unité production des cartes électroniques de l'usine Bomare Company, pour leurs accueils chaleureux.

Par ailleurs, je remercie tous les enseignants du département électronique qui ont contribué à ma formation.

Je remercie également les membres du jury de me faire l'honneur d'évaluer ce modeste travail.

# Liste des figures

## **Chapitre I : Présentation de l'entreprise et problématique ..... 3**

Figure I.1 : Situation géographique de Bomare Company .....	5
Figure I.2 : Unité de fabrication de téléviseurs .....	7
Figure I.3: Unité de production des cartes électroniques .....	7
Figure I.4 : Unité de productions des smart phones et tablettes.....	8
Figure I.5 : Teste des cartes insérées.....	8
Figure I.6: Unité de réparation des écrans.....	8

## **Chapitre II :Les systèmes automatisés de production ..... 11**

Figure II.1: Structure d'un système de production .....	12
Figure II.2: Principe de l'automatisation .....	13
Figure II.3: Structure d'un système automatisé .....	15
Figure II.4:Structure de pré-actionneur électrique. ....	17
Figure II.5:Contacteur. ....	19
Figure II.6: Structure de pré-actionneur pneumatique. ....	19
Figure II.7: Fonctionnement de distributeur. ....	19
Figure II.8: Image d'un relais.....	20
Figure II.9: Structure d'un relais. ....	20
Figure II.10:Disjoncteur magnétothermique et leur symbole .....	20
Figure II.11: Disjoncteur .....	22
Figure II.12: Fusible.....	22
Figure II.13: Compresseur à vis .....	23

## **Chapitre III : Les automates programmables industriels ..... 24**

Figure III.1: Utilisation des automates programmable dans l'industrie d'automobile.....	25
Figure III.2: API de type compact.....	28
Figure III.3: API de type modulaire .....	28

Figure III.4: Structure générale d'un automate programmable .....	29
Figure III.5: Fonctionnement d'un API .....	31
Figure III.6: Priorité des tâche d'un API multi-tâches.....	32
Figure III.7: Ladder Diagram (LD) .....	33
Figure III.8: Instruction List (IL) .....	33
Figure III.9: Function Bloc Diagram (FBD) .....	33
Figure III.10: Structured Text (ST) .....	34
Figure III.11: Sequential Function Chart (SFC) (Grafcet).....	34

**Chapitre IV : Les modules programmables Zelio Logic et le logiciel de programmation Zelio Soft..... 40**

Figure IV.1: Module Zelio Logic compact .....	42
Figure IV.2: Association entre modules logiques modulaires et extensions.....	43
Figure IV.3:Afficheur de l'automate Zelio .....	44
Figure IV.4: Fonctions en symboles LADDER et symboles électriques .....	47
Figure IV.5: Différents outils de programmation en langage LADDER.....	47
Figure IV.6: Structure d'un réseau de contacts .....	48
Figure IV.7: Fenêtre de dialogue.....	49
Figure IV.8: Automate Zelio de Schneider .....	50
Figure IV.9: Étapes de programmation .....	51

**Chapitre V : Simulation de la gestion de pression avec l'automate Zelio ..... 57**

Figure V.1: Schéma synoptique de la distribution de pression actuelle.....	58
Figure V.2: Schéma synoptique de la distribution de pression automatisé.....	61
Figure V.3: la maquette réalisée.....	62
Figure V.4: Partie du programme Ladder .....	63
Figure V.5: Temporisation .....	64
Figure V.6: Organigramme de fonctionnement .....	65
Figure V.7:Étape 01 .....	66
Figure V.8: Étape 02 .....	66
Figure V.9: Étapes 3 & 4.....	67

# Liste des tableaux

<b>Chapitre III : Les automates programmables industriels .....</b>	<b>24</b>
Tableau III.1: Comparaison des langages .....	35
<b>Chapitre V : Simulation de la gestion de pression avec l'automate Zelio .....</b>	<b>57</b>
Tableau V.1: les éléments de la ligne 1 .....	59
Tableau V.2: les éléments de la ligne 2.....	59
Tableau V.3: élément de la ligne manuelle .....	60
Tableau V.4: Nomenclature du programme .....	61

# Sommaire

Dédicaces .....	i
Remerciements .....	ii
Liste des figures .....	iii
Liste des tableaux .....	v
Sommaire .....	vi
Résumé .....	x
Introduction générale .....	1
Chapitre I : Présentation de l'entreprise et problématique .....	3
I.1. Introduction .....	3
I.2. Présentation de l'entreprise Bomare Company .....	4
I.3. L'entreprise .....	4
I.4. La situation géographique .....	5
I.5. L'histoire de Bomare Company : .....	5
I.6. Organisation de Bomare Company .....	6
I.7. L'assurance qualité : .....	9
I.8. Problématique .....	9
I.9. Conclusion .....	10
Chapitre II : Les systèmes automatisés de production .....	11
II.1. Introduction .....	11
II.2. Définition d'un système de production .....	12
II.3. Définition des systèmes automatisés de production .....	12
II.3.1. L'automatisation .....	13
II.3.2. Objectifs de l'automatisation .....	13
II.3.3. Conduites et surveillance d'un système automatisé .....	14
II.4. Structure d'un système automatisé de production .....	14
II.5. Description des différentes parties .....	15
II.5.1. La partie opératives (PO) .....	15
II.5.2. La Partie commande (PC) .....	16
II.5.3. Le poste de contrôle. ....	16
II.6. Choix de la logique de réalisation des SAP .....	17
II.6.1. Critère 1: <i>Faisabilité</i> .....	17
II.6.2. Critère 2: <i>Optimisation</i> .....	17
II.7. Partie commande .....	17

II.7.1.	Automate Programmable Industriel .....	17
II.7.2.	Pré-actionneurs.....	17
II.7.3.	Pré-actionneurs électriques .....	17
II.8.	Partie opérative .....	21
II.8.1.	Les actionneurs.....	21
II.8.2.	Capteurs.....	22
II.9.	Compresseur a vis .....	23
II.10.	Conclusion .....	23
Chapitre III :	Les automates programmables industriels.....	24
III.1.	Introduction .....	24
III.2.	Historique .....	25
III.3.	Définition et description des automates programmables industriels.....	26
III.4.	Nature des informations traitées par les API.....	27
III.4.1.	Logiques « Tout ou rien (T.O.R) ».....	27
III.4.2.	Analogiques .....	27
III.4.3.	Numériques .....	27
III.5.	Aspect extérieur des automates .....	27
III.5.1.	De type compact.....	28
III.5.2.	De type modulaire .....	28
III.6.	Organisation et architecture générale d'un API .....	29
III.6.1.	L'alimentation .....	29
III.6.2.	Le processeur (CPU).....	29
III.6.3.	La mémoire .....	30
III.6.4.	Les modules d'entrées / sorties .....	30
III.7.	Fonctionnement et comportement des API .....	30
III.7.1.	Les tâches .....	30
III.7.2.	Cycles et phases .....	31
III.7.3.	Modes de programmation.....	32
III.8.	Langage de programmation pour API .....	32
III.9.	Comparaison des langages.....	35
III.10.	Critères de choix d'un automate .....	35
III.11.	Utiliser un PC au lieu d'un API .....	36
III.11.1.	Les points faibles du PC .....	36
III.11.2.	Les points forts des API.....	36
III.12.	Gestion de la sécurité .....	37
III.13.	Conclusion .....	39

Chapitre IV : Les modules programmables Zelio Logic et le logiciel de programmation Zelio Soft 2 .....	40
IV.1. Introduction .....	40
IV.2. Modules programmables Zelio Logic.....	41
IV.2.1. Présentation .....	41
IV.2.2. Programmation .....	41
IV.2.3. Modules logiques compacts/modulaires .....	41
IV.2.4. Communication .....	43
IV.3. Programmation sur écran.....	44
IV.4. Le logiciel de programmation Zelio Soft .....	45
IV.4.1. Présentation .....	45
IV.4.2. Langages utilisés .....	46
IV.4.3. Modes de saisie des schémas de commande .....	46
IV.4.4. Fonctionnalités .....	47
IV.4.5. Description d'un réseau de contacts.....	48
IV.5. Transfert d'une application .....	48
IV.5.1. Écrire d'un PC vers le Zelio Logic.....	48
IV.5.2. Transfert du programme Zelio Logic vers le PC.....	49
IV.6. Description des étapes de programmation réalisé avec le logiciel Zelio Soft 2 .....	50
IV.6.1. Présentations.....	50
IV.6.2. Description matériel de l'automate SR3B101FU .....	50
IV.6.3. Le logicielle Zelio soft 2 .....	51
IV.7. Conclusion .....	56
Chapitre V : Simulation de la gestion de pression avec l'automate Zelio .....	57
V.1. Introduction .....	57
V.2. Description de la distribution de pression actuelle dans l'unité de production des cartes électronique .....	58
V.3. Objectif de l'automatisation .....	60
V.4. Cahier des charges .....	62
V.5. Programmation .....	62
V.5.1. Création du programme de gestion de la pression .....	63
V.5.2. La nomenclature de programme .....	63
V.5.3. La temporisation .....	64
V.6. Organigramme de fonctionnement .....	65
V.7. Description de cycle de fonctionnement de la maquette .....	66
V.7.1. Étape 01 .....	66
V.7.2. Étape 02 .....	66
V.7.3. Étape 03 .....	67

V.7.4	Étape 04 .....	67
V.8.	Conclusion .....	67
	Conclusion générale et perspectives .....	68
	Références bibliographique.....	70

## Résumé

*Le travail présenté dans ce mémoire est basé essentiellement sur l'utilisation de l'**automate programmable industriel** de type **Schneider Electric**.*

*Nous avons réalisé une maquette qui nous permet de faire une simulation de la **gestion intelligente de la distribution de pression** d'air dans l'unité de production des cartes électroniques au sein de l'usine **BOMARE COMPANY**, par un **automate Zelio**.*

*Nous avons commencé par une introduction générale, une présentation de l'entreprise d'accueil et les systèmes de productions automatisés, ensuite on a décrit automates programmables industriels d'une manière générale, puis les modules programmable **Zelio logic** et le logiciel de programmation **Zelio Soft2**.*

*Ensuite, une réalisation d'un cahier des charges a eu lieu, programmé et transféré vers l'automate pour le mettre en marche dans une maquette.*

*Enfin, une conclusion générale donne une synthèse sur le travail réalisé, pour clôturé avec des suggestions.*

# Introduction générale

Aujourd'hui, la clé de la compétitivité pour les compagnies industrielles réside dans l'augmentation de la satisfaction client, en agissant sur trois axes principaux, les délais de livraison, la qualité, et le coût. Pour atteindre ces objectifs, certaines de ces entreprises ont mis en place des pratiques de fabrication et ont implémenté des démarches telles que l'automatisation de la production dont l'objectif consiste à produire le bon produit en quantités nécessaires, au bon moment. Le système sur lequel repose ce type de management a permis à l'entreprise de développer sa chaîne de valeur vis à vis du client final, par l'élimination de toutes sortes de défaillances.

Pour atteindre cet objectif, qui est l'optimisation et l'augmentation de la production dans les entreprises, l'Automate Programmable Industriels dit « API » [9] est devenu un produit incontournable dans les systèmes automatisés de production. Vue sa mobilité, sa flexibilité de son architecture et la facilité de sa programmation, de sa connexion et de son adaptation dans les milieux industriels.

L'air comprimé est l'une des principales sources d'énergie industrielle. Il possède de nombreux avantages, offre une grande souplesse d'application et se transporte aisément.

Le rôle d'une unité d'air comprimé est de fournir de l'air, à la pression convenable et en quantité suffisante, pour assurer un fonctionnement correct des machines de production.

Chez Bomare Company [24, 26], toute chute de pression entre le compresseur et le point d'utilisation de l'air comprimé constitue une perte irrécupérable. Afin d'augmenter le rendement des machines d'insertion automatique des cartes électroniques.

Notre travail consiste à élaborer un système de gestion intelligente des priorités de distribution de pression d'air dans ces machines en cas de défaut (chute de pression) à l'aide d'un automate Zelio de Schneider électroniques de type SR3B101FU.

Répondre à cette problématique une méthodologie de travail s'est imposée, elle est décrite par les cinq chapitres constituant ce mémoire.

- Le premier chapitre est consacré à la présentation de l'entreprise Bomare Company avec une description sommaire des différentes unités de production.

- Dans le deuxième chapitre, on présente les généralités sur les systèmes automatisés de production, avec la définition des parties opérative, commande et contrôle.
- Dans le troisième chapitre, on donne une idée complète sur les automates programmables industriels, leurs types et la nature des informations traitées, il est présenté, aussi le comportement de fonctionnement des automates programmables industriels et les différents langages de programmation associés.
- Dans le quatrième chapitre, on expose exclusivement l'automate programmable industriel de Schneider électrique de type SR3B101FU, sa programmation avec le logiciel Zelio Soft 2.
- Dans le cinquième chapitre, on décrit la maquette réalisée selon le cahier des charges défini, après la programmation de l'automate Zelio avec le langage Ladder à l'aide du logiciel Zelio soft 2, une mise en marche de la maquette sera observée.

En conclusion, une synthèse des résultats obtenus lors de cette thèse est présentée et des perspectives pour des travaux futurs sont proposées.

# Chapitre I

## Présentation de l'entreprise et problématique

### I.1. Introduction

Avant de rentrer dans le vif du sujet, il est important d'introduire et de présenter l'entreprise d'accueil qu'est Bomare Company, puis de s'intéresser davantage à la distribution de pression d'air dans l'unité de production électronique et l'impact de la pression sur le bon fonctionnement des machines d'insertion automatique des cartes électroniques afin de s'imprégner de ces derniers pour pouvoir en déterminer et cerner la problématique qui est l'objet de ce travail.

Dans ce chapitre, une présentation de l'entreprise Bomare Company est donnée dans un premier temps, la deuxième partie sera consacrée à la définition du cadre et de la problématique de notre étude.

## **I.2. Présentation de l'entreprise Bomare Company**

Leader dans la fabrication d'une gamme variée de produits de haute technologie tel que les : TV LED, OLED et Smart, récepteurs satellites numériques, Caméra de surveillance, affichage dynamique DID, Smartphones et Tablettes qui sont disponibles sur le marché sous la marque commerciale « Stream System ». Bomare company met ses produits au service des performances de ses clients et d'une meilleure qualité. Grâce à son offre unique, Bomare company possède des services après-vente exceptionnels et assure la qualité de ses produits sans après l'achat.

Après une série d'acquisitions pour compléter et étendre son offre vers l'international, en 2015 qu'elle a pu acquérir une stabilité dans les exportations en particulier vers la Grèce [23], En 2016, l'entreprise a signé un contrat de distribution vers l'Espagne et le Portugal d'une valeur de 50 millions de dollars pour une durée de 5ans [23]. En 2018, Bomare Company a conclu un accord de distribution de ses produits en Italie pour une valeur de 2,5 millions de dollars avec un objectif d'atteindre les 10 millions de dollars d'ici 2019[23]. Un protocole d'accord avec la côte d'ivoire vient de voir le jour [24]. Actuellement, Bomare Company négocie des partenariats avec la France, l'Allemagne et la Hongrie, outre, les projets en cours en Tunisie, Mauritanie, au Sénégal, en Afrique du Sud et en zone d'Afrique de l'ouest [24].

## **I.3. L'entreprise**

BOMARE COMPANY a un siège social à 26 Rue Mohamed Ouidir AMELLAL, El Biar, de forme SARL avec un capital de 4 Millions de dollars. Son Activité est la production de produits électroniques. BOMARE COMPANY emploie plus de 650 salariés algériens, Parmi eux, 40% de docteurs, d'ingénieurs et de techniciens. Son Chiffre d'affaire est passé de 10 million de dollars en 2011 à 65 million de dollars en 2017.

## I.4. La situation géographique

L'usine de BOMARE COMPANY est située à Birtouta (Alger), et occupe 15.000 m<sup>2</sup>, dont 5.500 m<sup>2</sup> sont des ateliers et des unités de production. Quant à son siège social il est situé El Biar. La Photo suivante montre la situation géographique de l'entreprise.

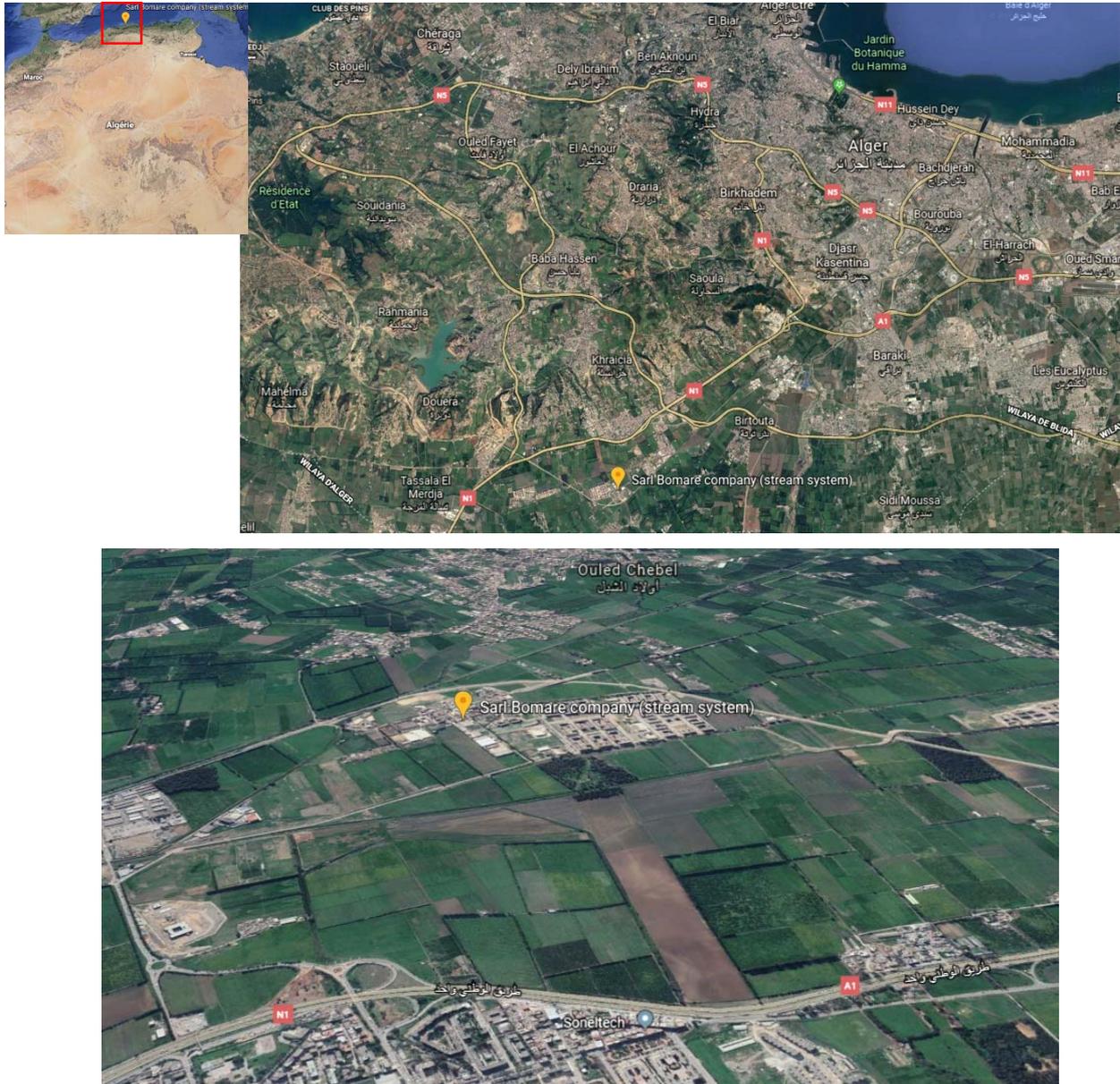


Figure I.1 : Situation géographique de Bomare Company

## I.5. L'histoire de Bomare Company :

Fondée en 2001 par le président A. BOUMEDIENE, dotée d'un capital de 850 million de dinars le début c'était une intégration manuelle de produits finis reçus de Corée, puis,

l'assemblage de kits [25]. L'entreprise Algérienne est devenue aussitôt un leader dans le marché de l'industrie électronique et un exemple de succès national et international.

En 2006, l'installation de la première ligne d'insertion des cartes électroniques et début de production. Depuis, ils utilisent la norme ROHS qui est relative à la restriction d'utilisation de certaines substances dangereuses dès les équipements électroniques et électriques.

En 2011, obtention d'une certification ISO9001 version 2008, et actuellement en cours de préparation pour l'intégration de la version 2015[26].

En 2013, inauguration de l'unité de réparation des écrans (LCD, LED, DLED, OLED et plasma), cette dernière dispose d'une salle blanche ISO5, la première en Afrique et en moyen orient. [24]

En 2013, Bomare Company a investi plus de 15 million de dollars consacré pour le développement de ses structures et l'acquisition de nouveaux équipements de chez les meilleurs équipementiers du monde tel que le japonais SAKI et l'américain UNIVERSAL INSTRUMENT [24].

Aussi, en 2013 un contrat de partenariat était signé avec le leader mondial de l'industrie LG, qui lui confie la production de certain de ses produits [24].

En octobre 2016, une conférence de presse s'est tenue conjointement animée par le DG de Bomare company, le CEO de Universal instrument et le recteur de l'université de Blida pour annoncer l'installation d'une ligne SMT d'un million et demi de dollars, au niveau de l'université de Blida afin d'accompagner les ingénieurs durant leurs cursus [24].

Avril 2016, Bomare Company a décroché l'empreinte « Basma Eldjazairia », grâce à ses téléviseurs, ses Smartphone et ses tablettes [26].

## **I.6. Organisation de Bomare Company**

L'entreprise Bomare Company est structurée en plusieurs départements pour assurer le bon déroulement de l'activité de production, on trouve le département production, le département technique, le département logistique, le département maintenance, le département assurance qualité et le service après-vente.

Le département production est dotée de trois unités de production qui sont aussi composées d'une ou plusieurs lignes, les unités de l'entreprise sont les suivantes :

- Unité de fabrication de téléviseurs (voir figure I.2).
- Unité de production des cartes électroniques (voir figure I.3).
- Unité de production du smart phone et tablettes (voir figure I.4).

L'unité de fabrication de téléviseurs est dotée de deux lignes d'assemblage des téléviseurs [24].



Figure I.2 : Unité de fabrication de téléviseurs



Figure I.3: Unité de production des cartes électroniques.

L'unité de production des cartes électroniques est dotée des lignes suivantes :

- La première ligne d'insertion automatique SMT, est installée en 2006 et remplacé par une nouvelle en 2017.
- La deuxième ligne de production est installée en 2013 [24].
- Une ligne d'insertion des composants manuelle et de teste des cartes insérées.



Figure I.4: Unité de productions des smart phones et tablettes

L'unité de production du smart phone et tablettes est inaugurée en novembre 2015 dispose d'une salle blanche ISO 06, un espace ISO 05 dédié à l'assemblage des écrans et deux lignes extensible pour l'assemblage et le teste des smart phones, tablettes [24].

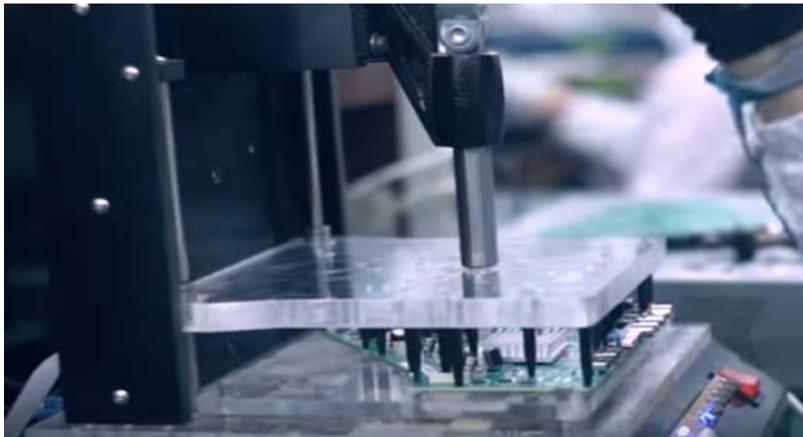


Figure I.5: Teste des cartes insérées.

L'unité de réparation de dalles, doté d'une salle blanche ISO 05 pour la réparation des parties optiques est liée au service après-vente.



Figure I.3: Unité de réparation des écrans

## **I.7. L'assurance qualité :**

Au niveau de BOMARE COMPANY, le contrôle qualité se fait en amont et en aval sur trois niveaux

IQC: incoming quality control

LQC: on-ligne quality control

OQC: outgoing quality control

A chaque étape de production un contrôle qualité se fait sur chaque produit. Bomare Company assure la conformité de ses produits dédiés au marché locale ou l'export selon les normes mondiales de sécurité et de santé d'où leur marquage CE [24].

## **I.8. Problématique**

Les machines d'insertion automatique Universal de Bomare Company fonctionnent toutes avec une pression d'environ 7 à 8 bars, mais elles sont très sensibles aux chutes de pression au-dessous de 7 bars.

En cas de chute de pression, tous les composants qui se trouvent sur chaque tête soient 60 composants par machine seront rejetés à n'importe quel endroit dans la machine et même sur la carte qui est en cours d'insertion. Ce qui fait un nombre élevé des composants perdus, car, même ceux déjà insérés avant la chute de pression seront jetés de fait qu'il y a un risque que les composants soient tombés sur quelques emplacements délicats comme les circuits Ball Grid Array (BGA). Ajouter à ça, les composants CMS ne sont pas disponibles sur le marché local, dont la nécessité d'attendre des jours pour avoir la liste demandée de chez les fabricants, en plus le rejet des composants affecte le fonctionnement de la machine surtout lorsque ces derniers tombent sur les encodeurs linéaires ou les caméras.

Comment faire une distribution intelligente de l'énergie pneumatique dans l'unité de production, qui consiste à couper la pression de toutes les machines et la garder sur les 5 machines d'insertion des composants CMS sur les cartes électroniques, suite à une coupure d'électricité qui implique l'extinction de compresseur d'air ou suite à un défaut sur ce dernier.

## **I.9. Conclusion**

Mon stage au niveau de Bomare Company, m'a donné l'occasion de comprendre le processus de production des cartes électroniques. Dans ce chapitre, une présentation générale de l'entreprise est donnée avec la description de son organisation des différentes unités de production.

Aussi une problématique a été mise en évidence. Mon travail consiste à contribuer à l'amélioration de la distribution de la pression par une gestion intelligente après un constat sur les différents processus de production allant de la sérigraphie des cartes électroniques jusqu'à la sortie du four. Le prochain chapitre est consacré à la description des systèmes automatisés de production.

# Chapitre II

## Les systèmes de production automatisés

### II.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. En plus Devant la compétitivité sans cesse croissante et de plus en plus dure demandée aux industries, qu'elles soient mécaniques, de transformation de produit ou de grande consommation, les industriels doivent garder leurs outils de production, performant et fiable.

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.

## II.2. Définition d'un système de production

Un système de production est un système à caractère industriel possédant les caractéristiques suivantes :

- L'obtention de la valeur ajoutée présente un caractère reproductible.
- La valeur ajoutée peut être exprimée et quantifiée en termes économiques.

Un système de production répond au besoin d'élaborer des produits, de l'énergie ou de l'information à un coût rentable pour l'utilisateur du système. L'élaboration de la valeur ajoutée sur les matières d'œuvre est obtenue :

- Au moyen d'un ensemble de dispositifs opératifs appelés **partie opérative (PO)**.
- Par l'action d'opérateurs humains et/ou de dispositifs de commande pour assurer la coordination des dispositifs opératifs [19].

Tout système de production possède une structure semblable au schéma suivant :

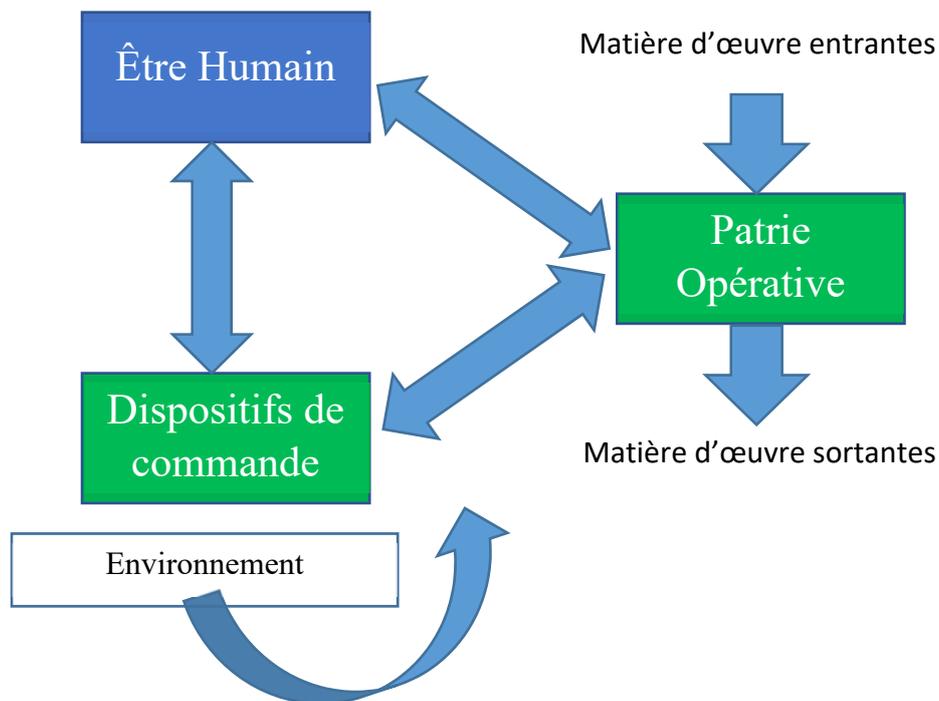


Figure II.1: Structure d'un système de production

## II.3. Définition des systèmes automatisés de production

L'automatisation tient une place très importante. Aujourd'hui, il serait difficile de concevoir un système de production sans avoir recours aux différentes technologies et composants qui forment les systèmes automatisés de production. Il faut cependant noter que les moyens de production malgré les avantages qu'ils présentent, peut engendrer certains inconvénients qu'il ne faut pas négliger.

### II.3.1. L'automatisation

L'automatisation de la production consiste à transférer tout ou partie des tâches de coordination, auparavant exécutées par des opérateurs humains, dans un ensemble d'objet technique appelé **Partie Commande (PC)**.

La partie commande mémorise le savoir-faire des opérateurs pour obtenir la suite des actions à effectuer sur les matières d'œuvre afin d'élaborer la valeur ajoutée. Elle exploite un ensemble d'informations prélevées sur la partie opérative (PO) pour élaborer la succession des ordres nécessaires pour obtenir les actions souhaitées [19].

Le schéma suivant détaille le principe de l'automatisation.

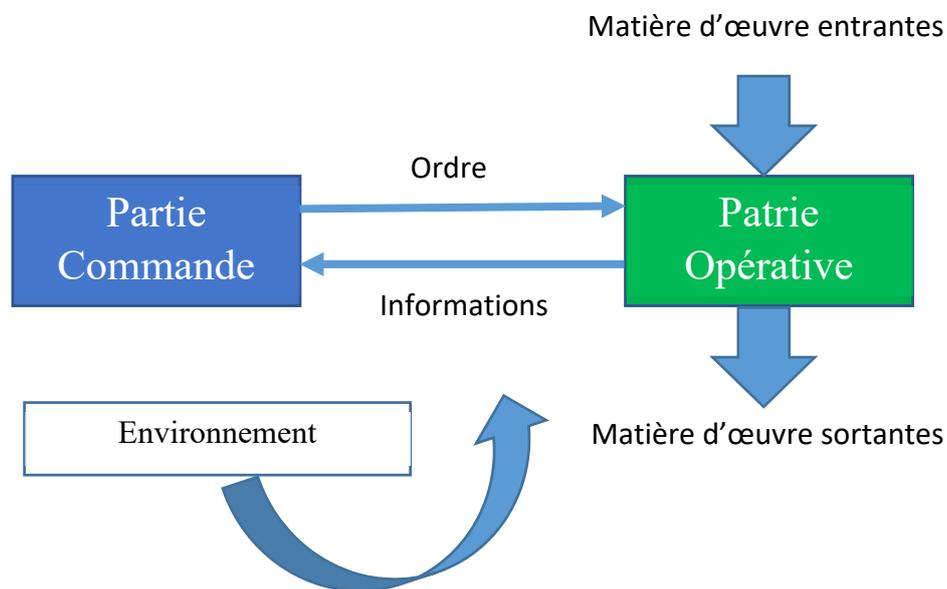


Figure II.2: Principe de l'automatisation

### II.3.2. Objectifs de l'automatisation

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

- La recherche des coûts plus bas pour le produit par la réduction des frais de main d'œuvre, d'économie d'énergie, d'économie de la matière, etc. ...
- La recherche d'une meilleure qualité du produit en limitant le facteur humain et multipliant les contrôles automatisés.
- L'amélioration de la flexibilité de la production.
- La suppression des travaux dangereux ou pénibles et l'amélioration des conditions de travail.

- La réalisation d'opérations impossibles à contrôler manuellement, par exemple des assemblages miniatures, des opérations très rapides, des coordinations complexes.
- La recherche de coûts plus bas par réduction des frais de main-d'œuvre [18].

### **II.3.3. Conduites et surveillance d'un système automatisé**

Il est difficile, dans la pratique, d'intégrer dans une partie de commande (PC) la totalité des savoir-faire humains. En effet, l'automatisation reste souvent partielle c'est-à-dire certaines tâches restent confiées à des intervenants humains. Ces tâches peuvent être classées en deux catégories: **Conduite** et **Surveillance** [19].

#### **II.3.3.1. Conduite :**

Cette catégorie regroupe les opérations de mise en marche du système, d'initialisation, de spécifications des consignes de fonctionnement, etc. [19].

#### **II.3.3.2. Surveillance :**

Le modèle de fonctionnement de la partie commande (PC) (choisi par le concepteur) correspond à un ensemble de situations prévues c'est-à-dire retenues par le concepteur parmi un ensemble de situations possibles. De ce fait, il est indispensable de pouvoir faire face à des situations non prévues (non retenues pour des raisons économiques compte tenu de leur faible probabilité d'apparition). À ce niveau, seul l'opérateur est appelé à intervenir et à prendre les décisions requises par cette situation. Il assure donc une fonction de surveillance [19].

Le concepteur devra, donc:

- Fournir à l'intervenant toutes les informations nécessaires à l'analyse de la situation
- Lui permettre d'agir sur le système.

Certaines tâches restent donc manuelles et l'automatisation devra donc prendre en compte la spécification du travail humain, en particulier :

- Assurer le dialogue entre l'être-humain et le système automatisé.
- Assurer la sécurité de ces intervenants lors de l'exécution de leurs tâches manuelles [19].

## **II.4. Structure d'un système automatisé de production**

Un système de production est dit automatisé, lorsqu'il peut gérer de manière autonome un cycle de travail préétabli qui se décompose en séquences ou étapes.

Les systèmes automatisés, utilisés dans le secteur industriel, possèdent une structure de base identique. Ils sont constitués de plusieurs parties plus ou moins complexes reliées entre elles que l'on nomme [BLUE01].

- Partie opérative (PO);
- Partie commande (PC);
- partie relation (pupitre de dialogue) (PR) [19].

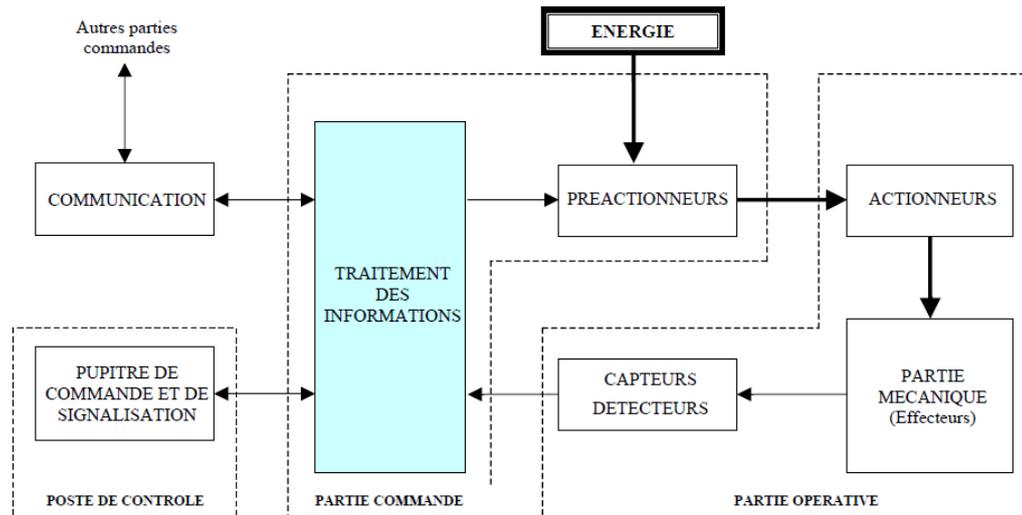


Figure II.3: Structure d'un système automatisé

## II.5. Description des différentes parties

### II.5.1. La partie opératives (PO)

C'est la partie visible du système. Elle comporte les éléments mécaniques du mécanisme qui agit sur la matière d'œuvre afin de lui donner sa valeur ajoutée, elle comporte :

- 1) Des *pré-actionneurs* (distributeurs, contacteurs), lesquels reçoivent des ordres de la partie commande;
- 2) Des *actionneurs* (vérins-moteurs) qui ont pour rôle d'exécuter ces ordres. Ils transforment l'énergie pneumatique (air comprimé), hydraulique (huile sous pression) ou électrique en énergie mécanique. Ils se présentent sous différentes formes comme :
  - Moteurs: hydraulique, pneumatique, électriques,
  - Vérins : linéaires (1 ou 2 tiges) rotatifs, sans tige;
- 3) Des *capteurs* qui informent la partie commande de l'exécution du travail. Ils existent sous différents types comme :

- Capteurs mécaniques, pneumatiques ou électriques;
- Capteurs magnétiques montés sur es vérins,
- Capteurs pneumatiques à chute de pression.

Dans un système automatisé de production, ce secteur de détection représente le service de surveillance et renseignement du mécanisme. Il contrôle, mesure, surveille et informe le PC sur l'évolution du système [19].

### **II.5.2. La Partie commande (PC)**

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 (réseau électrique, pneumatique ...) et les actionneurs Exemple : contacteur, distributeur ...

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 [18, 20].

### **II.5.3. 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 [18].

Sa complexité et sa taille dépendent de l'importance du système. L'outil de description s'appelle « GEMMA » (Guide d'Étude des Modes de Marche set Arrêts).

Ces outils graphiques (GRAFCEt et GEMMA) sont utilisés également par les techniciens de maintenance, pour la recherche des pannes sur les SAP (Système Automatisé de Production).

Pendant le fonctionnement, un dialogue continu s'établit entre les trois secteurs du système, permettent ainsi le déroulement correct du cycle défini dans le cahier de charges [19].

## II.6. Choix de la logique de réalisation des SAP

Pour choisir la meilleure technologie pour un automatisme donné, on utilise généralement deux critères: la **faisabilité** et l'**optimisation**.

### II.6.1. Critère 1: *Faisabilité* :

« La réalisation avec une technologie donnée est-elle possible ou non ? »

### II.6.2. Critère 2: *Optimisation* :

« La réalisation avec une technologie donnée conduit-elle au coût global le plus bas ? » [19]

## II.7. Partie commande

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

### II.7.1. 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. Cette partie sera développée dans le chapitre III.

### II.7.2. Pré-actionneurs

Le pré actionneurs peut être de type électrique ou pneumatique

### II.7.3. Pré-actionneurs électriques

Les pré-actionneurs électriques sont des contacteurs qui permettent le passage ou l'interruption de l'énergie électrique (voir figure 3). Ils sont appelés pré-actionneurs car ils se trouvent avant les actionneurs. Ces derniers peuvent être commandés à distance au moyen de

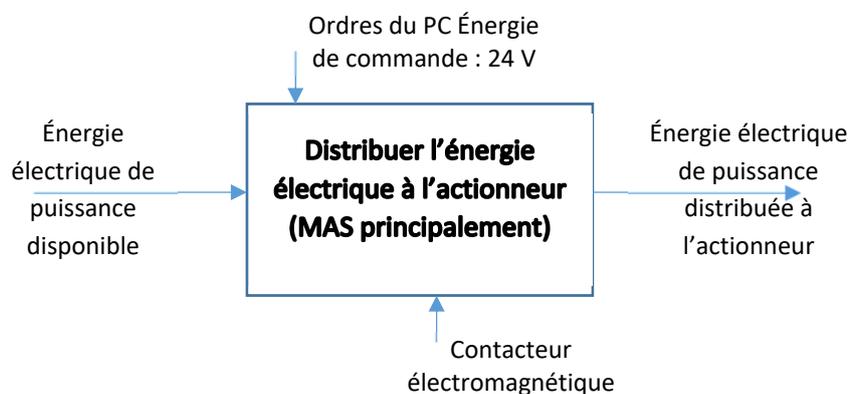


Figure II.4: Structure de pré-actionneur électrique.

contacts actionnés manuellement (bouton poussoir) ou automatiquement (asservi à une grandeur physique : pression, température, vitesse, etc.) [18].

Le contacteur est un appareil mécanique de connexion ayant une seule position de repos et une seule position de travail. Le contacteur est un appareil possédant un pouvoir de coupure.

Il est capable d'établir, de supporter et d'interrompre des courants dans les conditions normales du circuit, y compris les conditions de surcharges en service. L'intérêt du contacteur est de pouvoir être commandé à distance [18].

### ***II.7.3.1. 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 alternative de la bobine de contacteur [18].

### ***II.7.3.2. 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 [18].

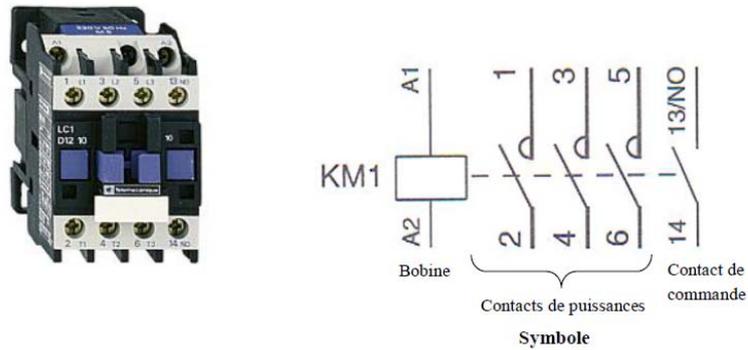


Figure II.5: Contacteur.

### II.7.3.3. Pré-actionneurs pneumatiques: les distributeurs

Ce sont des constitués chargés de distribuer l'énergie pneumatique vers les actionneurs pneumatiques sur ordre constituant de commande.

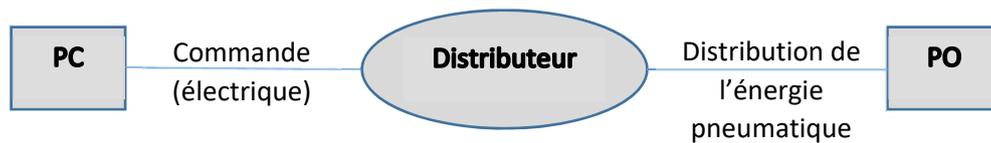


Figure II.6: Structure de pré-actionneur pneumatique.

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,
- Être des capteurs de position (pressostat, etc.).

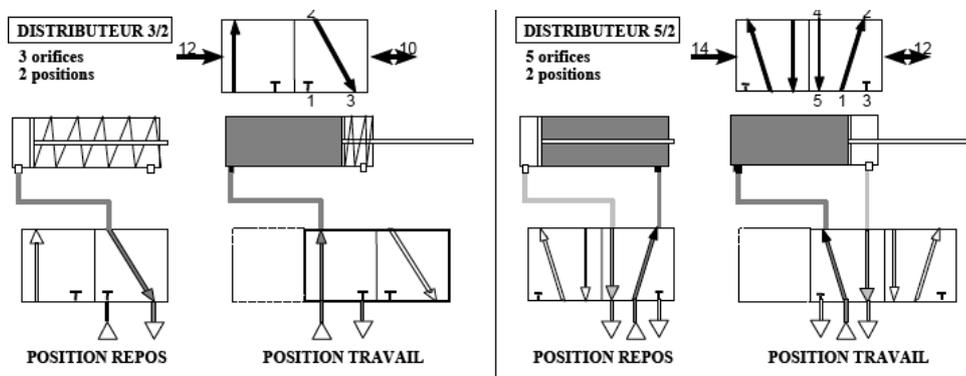


Figure II.7: Fonctionnement de distributeur.

### II.7.3.3.1 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 [20].

### II.7.3.4. 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 ...



Figure II.8: Image d'un relais.

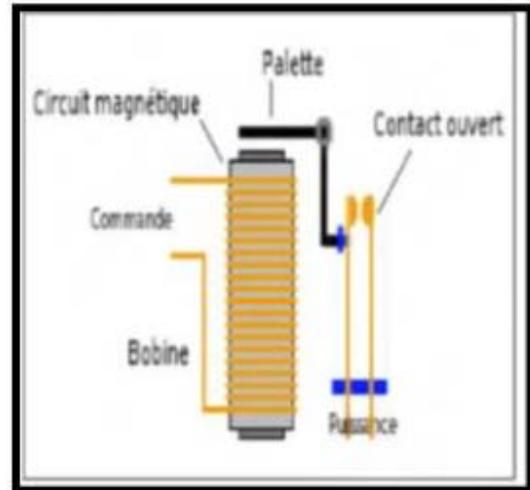


Figure II.9: Structure d'un relais.

### II.7.3.5. Protection

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

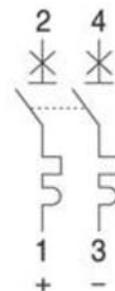


Figure II.8: Disjoncteur magnétothermique et leur symbole

**Disjoncteurs moteurs magnétiques** : pour la protection contre les courts circuits

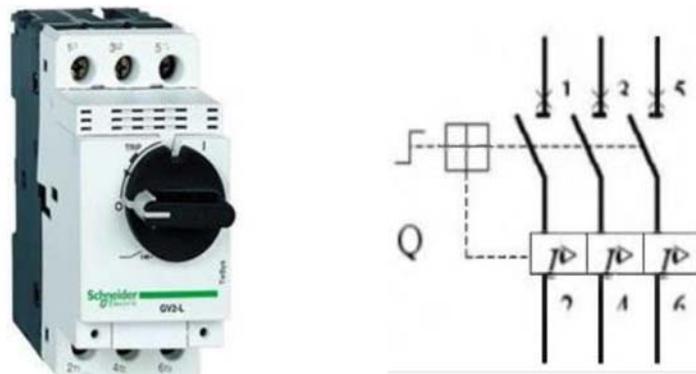


Figure 9: Disjoncteur moteurs magnétique.

**Fusible** : pour la protection contre les courts circuits.

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) [18].



Figure II.12: Fusible

## II.8. 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 [19].

### II.8.1. 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 agissent sur le système ou son environnement. Ces actionneurs s'appartiennent à trois technologies : électrique, pneumatique et hydraulique [18].

#### ***II.8.1.1. Les électrovannes :***

Une électrovanne est un dispositif commandé électriquement, permettant d'autoriser ou d'interrompre par une action mécanique, la circulation d'un fluide ou d'un gaz dans un circuit.

Il existe deux types d'électrovannes : "tout ou rien" et "proportionnelle".

Les "tout ou rien" possèdent deux états soit ouvertes, soit fermées.

Les électrovannes proportionnelles sont plus ou moins passantes en fonction de la tension qu'on leur applique.

Dans le secteur industriel, on les retrouve sur les groupes froids, les climatisations, les compresseurs [19].

#### ***II.8.1.2. Critères de choix des électrovannes :***

Le choix s'opère selon :

- Le coût et la compatibilité.
- La technologie et la fiabilité

### **II.8.2. Capteurs**

#### ***II.8.2.1. Définition***

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

#### ***II.8.2.2. Principe***

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

#### ***II.8.2.3. Pressostats basse pression :***

Ils sont des appareils utilisés essentiellement dans les installations frigorifiques et de conditionnement de l'air pour assurer une protection contre une pression d'aspiration trop faible ou une pression de refoulement trop élevée.

Ils peuvent également être utilisés comme appareils de régulation de pression. Ces appareils sont présents aussi dans les installations de conditionnement de l'air pour contrôler l'encrassement des filtres. En cas général, Le principe de régulation des pressostats est celui de

la boucle courte et fermée : Le paramètre mesuré sera contrôlé. C'est un appareil de régulation dit TOR (Tout Ou Rien) ou encore, Marche / Arrêt [18].

#### II.8.2.3.1 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.

### II.9. Compresseur a vis :

Le compresseur est une machine qui a pour fonction d'augmenter la pression du fluide compressible qui le traverse. Son nom traduit le fait que le fluide se comprime (son volume diminue) au fur et à mesure de l'augmentation de pression et cela est réalisé par rotation des deux rotors hélicoïdaux [17].



Figure II.13: Compresseur à vis

### II.10. Conclusion

Les systèmes automatisés de production deviennent indispensables 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 liés à ces systèmes pour la communication, la distribution d'énergie et la protection des machines. Nous avons présenté les différentes parties contrôle, puissance, communication et l'appareillage d'un système automatisé.

# Chapitre III

## Les automates programmables industriels

### III.1. Introduction

Tout système industriel automatisé est piloté par un ou plusieurs API, qui est utilisé pour la commande des machines (convoyage, emballage ...) ou des chaînes de production (automobile, agroalimentaire ...) où 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.

Dans ce chapitre, nous allons présenter dans un premier temps un bref historique suivi d'une description générale des automates programmables industriels pour pouvoir décrire leurs fonctionnalités, les différents langages de programmation et les critères de choix. Ce qui permet de Comprendre l'architecture matérielle des automates programmables industriels (API).

Ce chapitre a pour objectif de décrire les particularités des APIs ainsi que leurs langages de programmation.

## III.2. Historique

La deuxième partie du XXe siècle a passée à l'histoire comme étant l'ère de l'automatique [15]. Les automates programmables sont apparus aux USA vers les années 1969, à l'époque ils étaient destinés essentiellement à automatiser les chaînes de montages automobiles. Ils sont apparus en France en 1971, Ils sont couramment utilisés dans des systèmes critiques (avions, ascenseurs, trains, . . .), aussi leur validation est primordiale, ils sont de plus en plus employés dans l'industrie [2].

Les premiers processeurs d'automates furent donc construits à l'aide des circuits intégrés disponibles à l'époque [6].



Figure III.1: Utilisation des automates programmable dans l'industrie d'automobile

Leurs premières applications furent d'abord le remplacement des horloges de contrôle du temps des employés ; par la suite, leurs multiples utilisations industrielles, en particulier sur les lignes de production des usines, deviennent indispensables non seulement au point de vue contrôle, mais aussi du côté économique pour l'espace et l'entretien. C'est alors que de nombreux systèmes à relais durent céder leur place.

Les premiers automates programmables n'effectuaient que la commutation ON/OFF (et vice-versa) avec la possibilité de temporisation, comme les relais. Leurs applications étaient limitées seulement aux procédés répétitifs ainsi qu'à certaines machines.

De 1970 à 1974, la technologie des microprocesseurs (du moins les premiers) ajouta une plus grande flexibilité et une « intelligence » à l'automate programmable. Les capacités d'interface avec l'utilisateur s'améliorent. L'automate programmable peut maintenant exécuter

les opérations arithmétiques en plus des opérations logiques; il manipule les données et les adresses ; il effectue la communication avec d'autres automates ou ordinateurs, donnant ainsi une nouvelle dimension aux applications de l'automate programmable.

La console de programmation s'allie avec un moniteur permettant la programmation avec des symboles familiers de relais ce qui facilite beaucoup la compréhension et le dépannage car la logique peut être vue dans la même forme que les dessins à relais.

Les automates programmables utilisent une mémoire non-volatile (RAM + Pile, EEPROM ou EAPROM par exemple) pour emmagasiner les instructions. Ces derniers accompliront des fonctions logiques, arithmétiques, de temporisation, de comptage et de manipulation des données. En plus, les fonctions de contrôle PID et d'autres fonctions complexes comme le contrôle numérique de processus sont présentes. Puisque les automates programmables ont été conçus pour accomplir des opérations semblables à celles des relais, la programmation est basée généralement sur la nomenclature des diagrammes en échelle (ou schéma à relais). Des langages de haut niveau ont été aussi implantés sur certains automates afin de produire une plus grande flexibilité de programmation [15].

Ce qu'il est important de noter c'est que les automates furent au départ, et restent encore maintenant, des machines conçues par des automaticiens pour des automaticiens, indépendamment donc des constructeurs d'ordinateurs. Leur parfaite adéquation aux besoins industriels en est la conséquence la plus marquante [6].

### **III.3. Définition et description des automates programmables industriels**

Les automates Programmables Industriels (API) (PLC - Programmable Logic Controller) sont des appareils électroniques programmables (micro-ordinateurs simplifiés), adaptés à l'environnement industriel, qui réalise des fonctions d'automatisme pour assurer la commande de pré-actionneurs et d'actionneurs. Ils sont conçus pour traiter par le biais de programmes des problèmes de logique séquentielle, pour remplacer notamment les commandes d'automatismes en logique câblée réalisées avec des circuits logiques ou des relais [9].

Ces équipements offrent beaucoup moins de possibilités que les processeurs classiques, ils peuvent cependant être utilisés facilement par un personnel peu qualifié en informatique et ce grâce à des langages de programmations orientés application, de type langages relais, équations booléenne ou Grafset [9].

L'objectif de l'automatisation des systèmes est de produire, en ayant recours le moins possible à l'homme, des produits de qualité et ce pour un coût le plus faible possible.

Le déroulement des programmes d'automates est en principe de type synchrone, ce qui élimine toute sorte de complications liés à la multiprogrammation. Les automates programmables sont en général conçus dès l'origine pour fonctionner en milieu industriel. Ils sont dotés de toutes les protections contre l'environnement ainsi que des interfaces adaptées aux capteurs et aux actionneurs les plus courants [9].

### **III.4. Nature des informations traitées par les API**

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 :

#### **III.4.1. Logiques « Tout ou rien (T.O.R) »**

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

#### **III.4.2. Analogiques :**

C'est une information continue et peut prendre une valeur comprise dans une plage bien déterminée, c'est le type d'information délivrée par un capteur (pression, température).

#### **III.4.3. Numériques**

Elle 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.

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 [16].

### **III.5. Aspect extérieur des automates**

Les automates peuvent être de type compact ou modulaire. Les automates permettent de réaliser de nombreuses autres fonctions grâce à des modules intelligents que l'on dispose sur un ou plusieurs racks.

### III.5.1. De type compact

On distinguera les modules de programmation (LOGO de Siemens, ZELIO de Schneider, MILLENIUM de Crouzet ...) des micros 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.



Figure III.2: API de type compact

### III.5.2. De type modulaire

Le processeur, l'alimentation et les interfaces d'entrées / sorties résident dans des unités séparées (modules) et sont fixées sur un ou plusieurs racks contenant le "fond de panier" (bus plus connecteurs).

Ces automates sont intégrés dans les automatismes complexes où puissance, capacité de traitement et flexibilité sont nécessaires.



Figure III.3: API de type modulaire

### III.6. Organisation et architecture générale d'un API

Un automate programmable est un système construit autour d'un microprocesseur. Les éléments principaux que l'on rencontre habituellement dans un API sont : l'alimentation, le processeur (CPU), la mémoire et les modules d'entrées / sorties, comme indique dans le schéma ci-après [9] :

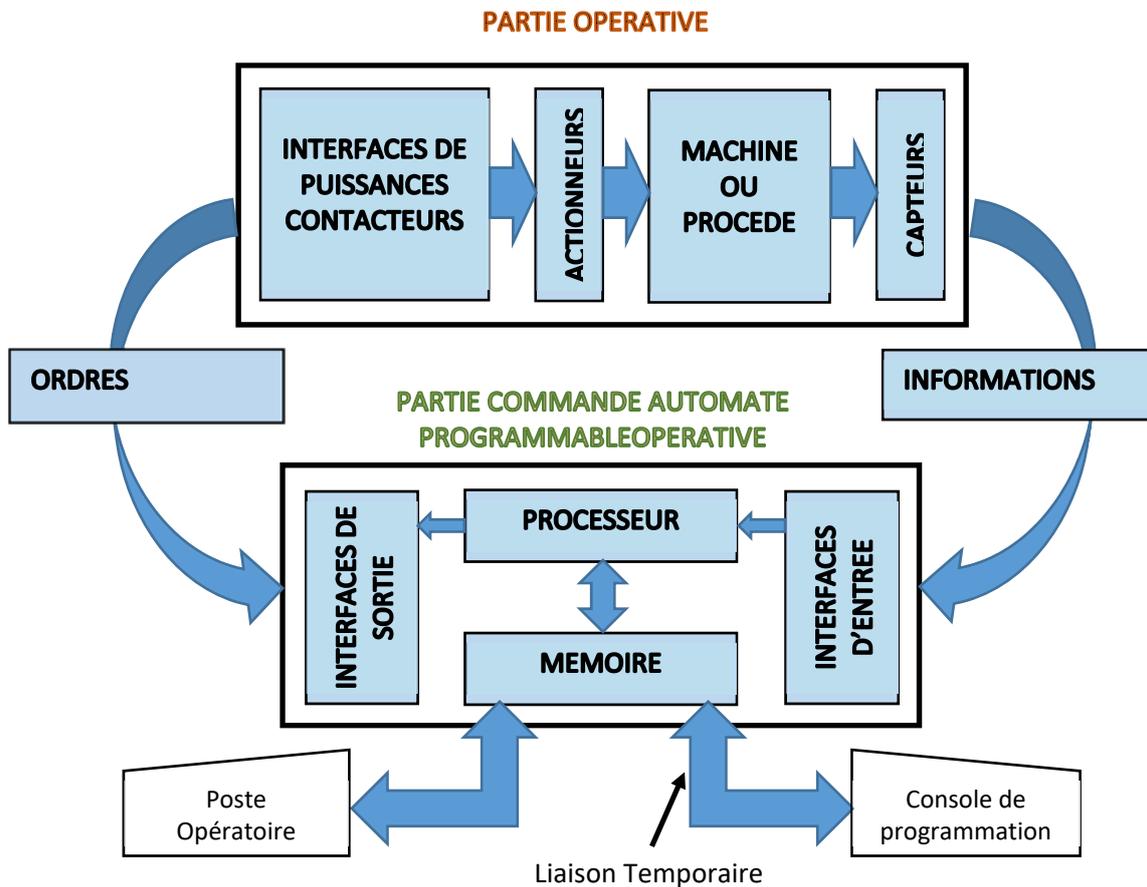


Figure III.4: Structure générale d'un automate programmable

#### III.6.1. L'alimentation

L'alimentation fournit une tension stable pour le fonctionnement du CPU, des modules d'entrées / sortie et de la mémoire. Cette alimentation ne fournit normalement pas de tension pour les signaux entrants et sortants des modules d'entrées / sorties. Il y a lieu de relever d'éventuelles exceptions : lignes de données sérielles, modules analogiques et modules intelligents [9].

#### III.6.2. Le processeur (CPU)

C'est la partie dite intelligente de l'automate. Il est géré par le programme système appelé FIREWARE. Le module CPU supporte des éléments internes tels que des indicateurs,

des temporisateurs, des compteurs et des registres de données. Les différents éléments sont accessibles au programme système par l'intermédiaire d'un bus interne.

Un compteur de programme pointe les instructions successives à effectuer au fur et à mesure du déroulement du programme. Le CPU à différents modes de fonctionnement : RUN ou STOP programmation. Pour la programmation et autres manipulations, une console de programmation doit être branché au CPU [9].

### **III.6.3. La mémoire**

La mémoire contient en premier lieu le programme utilisateur. Ce programme introduit par l'utilisateur décrit toutes les fonctions que l'automate est en mesure de réaliser. Selon les applications, la mémoire peut aussi contenir des textes et des données [9].

### **III.6.4. Les modules d'entrées / sorties**

Les modules d'entrées / sorties permettent les échanges d'informations vers l'environnement extérieur de l'automate. Ils adaptent les signaux entrants et sortants en tension et courants, comme ils filtrent les perturbations et protègent la partie interne de l'automate notamment des influences extérieures.

Les signaux ainsi adaptés et filtrés sont accessibles au programme système par l'intermédiaire d'un bus externe au CPU.

Un automate structure dispose de tous les éléments nécessaires pour réaliser les fonctions qu'on attend de lui. Il est évident que les modules d'entrées / sorties auront été connectés aux différents éléments périphériques [9].

## **III.7. Fonctionnement et comportement des API**

### **III.7.1. Les tâches**

Le moniteur d'exécution d'un API peut être composé de plusieurs sous-programmes appelés tâches. Une tâche est un ensemble d'opérations programmées pour s'exécuter successivement, puis s'arrêter jusqu'au prochain lancement. Dans un automate programmable industriel, une tâche est :

- ou bien cyclique : la tâche est immédiatement relancée après sa fin,
- ou bien périodique : la tâche est relancée toutes les T unités de temps,

– ou bien événementielle : la tâche est lancée à chaque fois qu'un événement prédéfini se produit [2].

### III.7.2. Cycles et phases

L'exécution d'une tâche est un cycle composé de trois phases, voir figure 5:

#### III.7.2.1 Phase 1 : l'acquisition des entrées

Les variables d'entrées sont accessibles en lecture seule. Pendant cette première phase, leurs valeurs sont lues et ensuite stockées dans la mémoire de l'API,

#### III.7.2.2 Phase 2 : Le traitement interne

C'est une phase d'exécution du programme et de calcul des valeurs de sorties à partir des valeurs stockées en mémoire dans la phase précédente, les résultats des calculs sont ensuite à leur tour stockés en mémoire.

#### III.7.2.3 Phase 3 : L'affectation des sortie

Les variables de sorties sont accessibles en écriture seule. Pendant cette phase, leurs valeurs sont mises à jour à partir des valeurs calculées dans la phase de traitement interne.

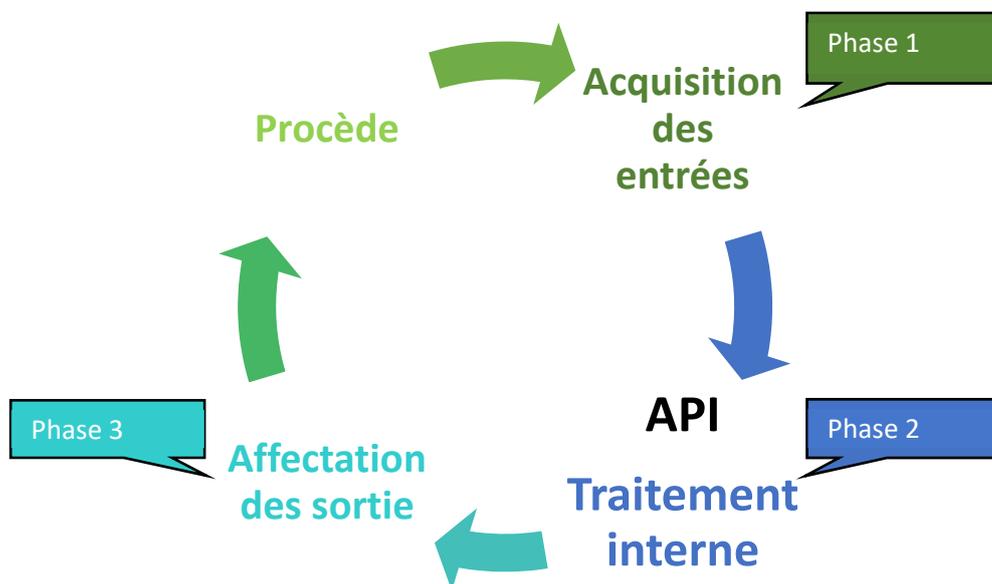


Figure III.5: Fonctionnement d'un API

### III.7.3. Modes de programmation

Les APIs peuvent être programmés selon deux modes différents :

- mode mono-tâche : le moniteur d'exécution comporte une unique tâche cyclique, appelée tâche maître.
- mode multi-tâches : le moniteur d'exécution comporte plusieurs tâches dont l'ordonnancement est réalisé en fonction de leurs priorités (figure 6). À tout moment, une seule tâche est active et chaque tâche possède son propre cycle d'acquisition des entrées, traitement interne et affectation des sorties. Les tâches possibles sont :
  - *la tâche maître* : elle est unique et cyclique.
  - *les tâches rapides* : elles sont optionnelles et périodiques. Elles peuvent lire un nombre limité d'entrées, mais modifier toutes les variables internes.
  - *les tâches événementielles* : elles sont optionnelles. Elles ont accès à un nombre limité d'entrées, de variables internes et de sorties.

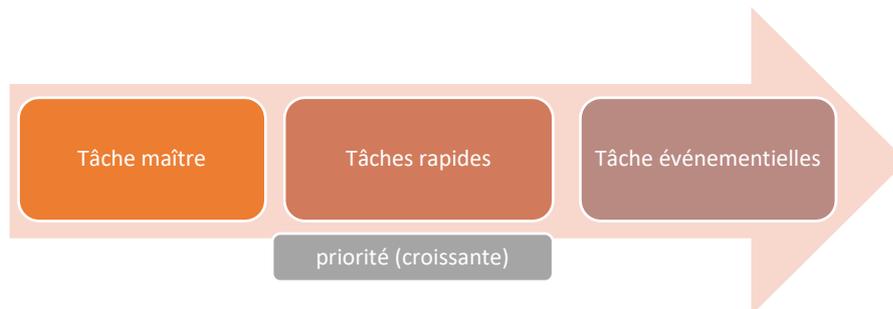


Figure III.6: Priorité des tâche d'un API multi-tâches

### III.8. Langage de programmation pour API

Les programmes utilisés avec les API peuvent être écrits dans différents formats. Pour que les ingénieurs ayant peu de connaissances en programmation puissent élaborer des programmes pour les API.

La CEI 61131-3 est une norme industrielle de la Commission électrotechnique internationale (CEI) définissant cinq langages de programmation qui peuvent être utilisés pour la programmation des automates programmables industriels.

Chaque automate se programme via une console de programmation propriétaire ou par un ordinateur équipé du logiciel constructeur spécifique [17].

### III.8.1.1 Langage à contact LD (Ladder Diagram)

C'est le plus utilisé, c'est une représentation graphique d'équations booléennes combinant des contacts (en entrée) et des relais (en sortie). Il permet la manipulation de données booléennes, à l'aide de symboles graphiques organisés dans un diagramme comme les éléments d'un schéma électrique [17].

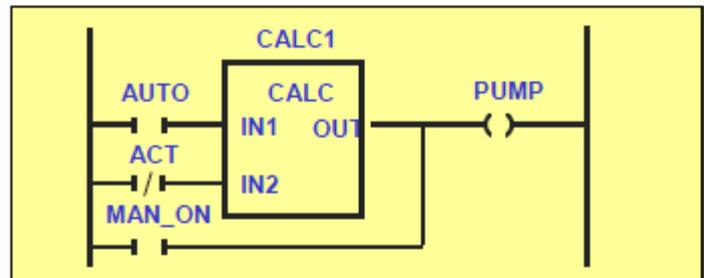


Figure III.7: Ladder Diagram (LD)

### III.8.1.2 Liste d'instruction IL (Instruction List)

Un langage textuel de bas niveau. Il est particulièrement adapté aux applications de petites tailles. L'opérateur indique le type d'opération à effectuer entre le résultat courant et l'opérande. Le résultat de l'opération est stocké à son tour dans le résultat courant. Très peu utilisé par les automaticiens [17].

```
A: LD  %IX1 (* PUSH BUTTON *)
   ANDN %MX5 (* NOT INHIBITED *)
   ST  %QX2 (* FAN ON *)
```

Figure III.8: Instruction List (IL)

### III.8.1.3 Blocs Fonctionnels FBD (Function Bloc Diagram)

C'est un langage graphique où des fonctions sont représentées par des rectangles avec les entrées à gauche et les sorties à droites. Il permet la construction d'équations complexes à partir des opérateurs standards, de fonctions ou de blocs fonctionnels. Les blocs sont programmés (bibliothèque) ou programmables. Utilisé par les automaticiens [17].

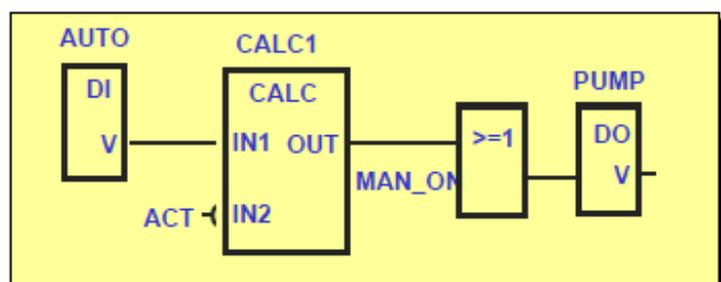


Figure III.9: Function Bloc Diagram (FBD)

### III.8.1.4 Langage littéral structuré ST (Structured Text)

Un langage textuel de haut niveau dédié aux applications d'automatisation. Ce langage est principalement utilisé pour décrire les procédures complexes, difficilement modélisables avec les langages graphiques. C'est le langage par défaut pour la programmation des actions dans les étapes et des conditions associées aux transitions du langage SFC [17].

```

VAR CONSTANT X : REAL := 53.8 ;
Z : REAL; END_VAR
VAR aFB, bFB : FB_type; END_VAR

bFB(A:=1, B:='OK');
Z := X - INT_TO_REAL (bFB.OUT1);
IF Z>57.0 THEN aFB(A:=0, B:="ERR");
ELSE aFB(A:=1, B:="Z is OK");
END_IF

```

Figure III.10: Structured Text (ST)

### III.8.1.5 Langage SFC (Sequential Function Chart) ou GRAFCET

C'est un langage graphique de spécification, le GRAFCET est utilisé par certains constructeurs d'automate (Schneider, Siemens) pour la programmation. Parfois associé à un langage de programmation, il permet une programmation aisée des systèmes séquentiels tout en facilitant la mise au point des programmes ainsi que le dépannage des systèmes. On peut également traduire un Grafcet en langage en contacts et l'implanter sur tout type d'automate [17].

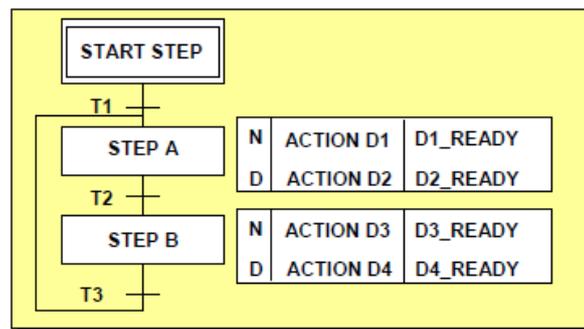


Figure III.11: Sequential Function Chart (SFC) (Grafcet)

### III.9. Comparaison des langages

Langage	Avantages	Inconvénients
<b>LD</b>	Facile à lire et à comprendre par la majorité des électriciens langage de base de tout PLC	Suppose une programmation bien structurée
<b>FBD</b>	Très visuel et facile à lire	Peut devenir très lourd lorsque les équations se compliquent
<b>ST</b>	Langage de haut niveau (langage pascal) Pour faire de l'algorithmique	Pas toujours disponible dans les ateliers logiciels
<b>IL</b>	Langage de base de tout PLC type assembleur	Très lourd et difficile à suivre si le programme est complexe Pas visuel.
<b>SFC</b>	Description du fonctionnement (Séquentiel) de l'automatisme. Gestion des modes de marches Pas toujours accepté dans l'industrie...	Peu flexible

Tableau III.1: Comparaison des langages

### III.10. Critères de choix d'un automate :

Le choix d'un API est en fonction de la partie commande à programmer. Il est impératif de tenir compte, en premier lieu, de plusieurs critères :

- Le choix d'une société ou d'un groupe (Schneider, Siemens, Allen Bradley, Rockwell, Mitsubishi, Delta, Omron ...)
- L'utilisation des langages de programmation de type Grafcet est préférable pour assurer les mises au point et dépannages dans les meilleures conditions.
- La possession d'un logiciel de programmation est une source d'économies (achat du logiciel et formation du personnel).
- La possession des outils de simulation des programmes.

Il faut ensuite quantifier les besoins :

- Le nombre d'entrées / sorties, selon les besoins, en prenant en considération les possibilités d'extension.
- Type de processeur : la taille du programme (mémoire), la vitesse de traitement et les fonctions spéciales offertes par le processeur.
- Les fonctions complémentaires: temporisateurs et compteurs.
- Le nombre de voies analogiques.
- La capacité de traitement arithmétique.
- La bibliothèque fonctionnelle.
- Fonctions ou modules spéciaux.
- 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 ...).

### **III.11. Utiliser un PC au lieu d'un API**

#### **III.11.1. Les points faibles du PC**

Le PC présente un ensemble de points faibles

- Les technologies du PC évoluent très rapidement elles deviennent obsolètes en peu de temps alors qu'une installation est en place pour 10 ou 20 ans.
- Les systèmes d'exploitations des PCs sont non robustes. Ils entraînent souvent un redémarrage du PC à la suite d'une exception.
- Le PC est sujet aux virus qui peuvent rendre les programmes de contrôle inopérants.
- La non tenue aux contraintes de l'environnement industriel (vibrations, chocs, températures, corrosion, rayonnement, etc.) conduit à un accroissement des prix du PC pour obtenir une meilleure résistance à l'environnement.

#### **III.11.2. Les points forts des API**

Par ailleurs, les API présentent un grand nombre de points forts :

- Matériel robuste et fiable même dans des conditions extrêmes (vibrations, chocs, températures, etc.)
- Une large gamme des modules d'entrée / sortie permet la communication avec des périphériques industriels variés.

- Les API peuvent être équipés d'un multi ou biprocesseur et d'un système d'exploitation temps réel (multitâches et traitement sur événement) pour garantir des performances en temps réel.
- Les API offrent le choix du mode de marche (manuel, semi-automatique, automatique), l'arrêt, remise en cycle.
- Le programme de contrôle est sauvegardé en permanence dans l'API et la mémoire peut être contrôlée constamment.
- Une ligne des programmes de contrôle peut être modifiée en cours d'exécution évitant ainsi la perturbation de l'exploitation de l'outil de production.
- Les modules de l'API peuvent être débrosés / embrosés sous tension ce qui évite la mise hors tension de l'API et par conséquent l'arrêt du système de contrôle.
- L'API n'est pas sensible aux perturbations et variations de la source électrique.
- L'état de chacune des entrées / sorties est lisible instantanément par des LEDs.
- Traitement en temps réel.

On voit ainsi qu'il est très difficile de remplacer l'API par un PC. Néanmoins, l'association API-PC fait que les forces de l'un annulent les faiblesses de l'autre.

L'API assure le contrôle de la partie opérative (ensemble d'outillages et d'actionneurs agissant sur le processus automatisé).

Le PC supporte le dialogue homme / machine et / ou le stockage de données et / ou le traitement d'informations permettant par exemple le suivi de la production et la gestion de la qualité.

### **III.12. Gestion de la sécurité**

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

Placé au cœur du système automatisé, l'automate se doit d'être un élément fiable car :

- Un dysfonctionnement de celui-ci pourrait avoir de graves répercussions sur la sécurité des personnes.
- Les coûts de réparation de l'outil de production sont généralement très élevés.
- Un arrêt de la production peut avoir de lourdes conséquences sur le plan financier.

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

**Contraintes extérieures** : l'automate est conçu pour supporter les différentes contraintes du monde industriel et a fait l'objet de nombreux tests normalisés (tenue aux vibrations, CEM ...)

**Coupures d'alimentation** : l'automate est conçu pour supporter les coupures d'alimentation et permet, par programme, d'assurer un fonctionnement correct lors de la réalimentation (reprises à froid ou à chaud)

**Mode RUN/STOP** : Seul un technicien peut mettre en marche ou arrêter un automate et la remise en marche se fait par une procédure d'initialisation (programmée)

Contrôles cycliques :

Procédures d'autocontrôle des mémoires, de l'horloges, de la batterie, de la tensions d'alimentation et des entrées / sorties

Vérification du temps de scrutation à chaque cycle appelée Watchdog (chien de garde), et enclenchement d'une procédure d'alarme en cas de dépassement de celui-ci (réglé par l'utilisateur)

**Visualisation** : Les automates offrent un écran de visualisation où l'on peut voir l'évolution des entrées / sorties.

La défaillance d'un automate programmable pouvant avoir de graves répercussions en matière de sécurité, les normes interdisent la gestion des arrêts d'urgence par l'automate ; celle-ci doit être réalisée en technologie câblée.

On peut également ajouter des modules de sécurité à l'automate (sécurité des machines).

Il existe enfin des automates dits de sécurité (APIs) qui intègrent des fonctions de surveillance et de redondance accrues et garantissent la sécurité des matériels.

### **III.13. Conclusion**

Dans ce chapitre, une description générale des automates programmables industriels est donnée. Cette description montre la richesse matérielle des API, notamment grâce à leurs structures qui permet d'adapter la constitution matérielle au besoin de l'application.

Pour construire une application autour d'un automate programmable, il est nécessaire d'explorer la capacité du logiciel de programmation de l'automate.

Donc, les automates programmables doivent être utilisés facilement par un personnel habitué aux techniques classiques d'automatisation et peu au courant de l'informatique. Ceci a conduit les constructeurs d'automates programmables à concevoir des langages d'application spécialement adaptés à la réalisation d'automatismes. Le prochain chapitre est consacré à la description des modules programmables Zelio Logic et le logiciel de programmation Zelio Soft 2.

# Chapitre IV

## Les modules programmables Zelio Logic et le logiciel de programmation Zelio Soft 2

### IV.1. Introduction

Les automates offerts par la filiale Télémécanique du Schneider Electric, exactement (Modules programmables Zelio Logic) sont programmables par le biais du logiciel Zelio Soft 2. Il permet la programmation et la simulation de procédures automatisées avec différents langages normalisés. Dans ce chapitre, nous allons présenter le Module programmable Zelio Logic et le logiciel de programmation Zelio Soft 2.

## **IV.2. Modules programmables Zelio Logic**

### **IV.2.1. Présentation**

Les modules logiques Zelio logic sont destinés à la réalisation de petits équipements d'automatismes. Ils sont utilisés dans les secteurs d'activité de l'industrie.

La simplicité de leur programmation, garantie par l'universalité des langages LADDER et blocs fonctions FBD, satisfait aux exigences de l'automaticien et répond aux attentes de l'électricien. Les modules logiques compacts répondent aux besoins d'automatismes simples, jusqu'à 20 entrées/sorties. Les modules logiques modulaires autorisent, si besoin, des extensions d'entrées/ sorties et une extension de communication sur réseau Ethernet, pour plus de performance et de flexibilité, de 10 à 40 entrées/sorties. [21].

### **IV.2.2. Programmation**

La programmation peut être effectuée :

- De façon autonome en utilisant le clavier du module logique (langage à contacts)
- Sur PC avec le logiciel « Zelio Soft 2 », la programmation peut être réalisée soit en langage à contacts (LADDER), soit en langage blocs fonctions (FBD) [21].

### **IV.2.3. Modules logiques compacts/modulaires**

#### *IV.2.3.1 Modules logiques compacts*

Les modules logiques compacts répondent aux besoins d'automatismes simples.

Les entrées/sorties sont au nombre de :

- 12 ou 20 E/S, alimentées en a 24 V ou c 12 V,
- 20 E/S, alimentées en a 48 V,

- 10, 12 ou 20 E/S, alimentées en a 100...240 V ou c 24 V.



Figure IV.1: Module Zelio Logic compact

#### IV.2.3.2 Modules logiques modulaires

Les entrées/sorties pour les modules logiques modulaires sont au nombre de :

- 26 E/S, alimentées en c 12 V,
- 10 ou 26 E/S, alimentées en a 24 V, a 100...240 V ou c 24 V.

Pour plus de performance et de flexibilité, les modules Zelio Logic modulaires peuvent recevoir des extensions afin d'obtenir un maximum de 40 E/S :

- Extensions de communication réseau Modbus ou Ethernet, alimentées en c 24 V par le module Zelio Logic de même tension.
- Extension d'entrées/sorties analogiques avec 4 E/S, alimentée en c 24 V par le module Zelio Logic de même tension,

- Extensions d'entrées/sorties TOR avec 6, 10, ou 14 E/S, alimentées par le module Zelio Logic de même tension.1



1 : Module Zelio Logic modulaire (10 ou 26 E/S)  
 2 : Extension entrées/sorties TOR (6,10 ou 14 E/S) ou analogiques (4 E/S)

1 : Module Zelio Logic modulaire (10 ou 26 E/S)  
 2 : Extensions de communication réseau modbus ou Ethernet  
 3 : Extension d'entrées/sorties TOR (6,10 ou 14 E/S) ou analogiques (4 E/S)

Figure IV.2: Association entre modules logiques modulaires et extensions

#### IV.2.4. Communication

Les outils de programmation permettent de connecter le module Zelio logic au PC équipé du logiciel “Zelio Soft 2” :

- Liaison par câbles : câble RS 232(SR2 CBL01) sur port série 9 contacts ou câble USB (SR2 USB01) sur port USB
- Liaison sans fil : Interface bluetooth SR2 BTC01

### IV.3. Programmation sur écran

Il est possible de programmer directement sur écran sans passer par le logiciel Zelio Soft 2, puisque le module Zelio possède un afficheur LCD sur la face avant ainsi que des boutons pour la manipulation, mais cela s’effectue uniquement en langage LADDER [21].

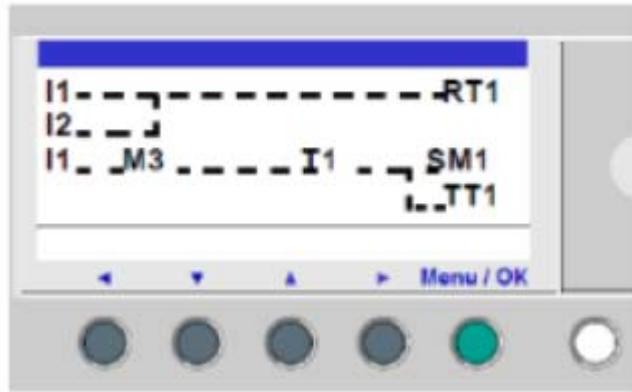
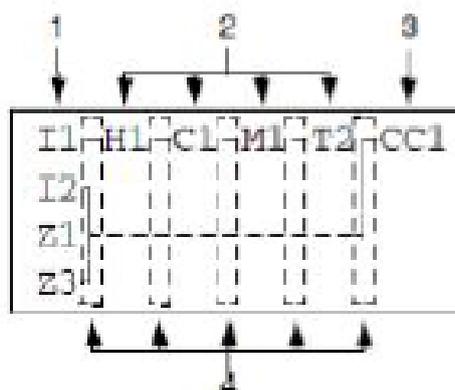


Figure IV.3:Afficheur de l’automate Zelio

Le module logique permet la saisie de 120 lignes de schémas de commande, tout comme sur le logiciel Zelio Soft 2. L’écran du module logique permet de visualiser ces lignes 4 par 4 et se présente de la façon Suivante :



Repère	Élément
1	Colonne réservée aux contacts (conditions).
2	Colonne réservée aux contacts (conditions ou aux liaisons).
3	Colonne réservée aux bobines (actions).
4	Colonne réservée aux liaisons.

Règles	Incorrect	Correct
Chaque bobine ne doit être saisie qu'une seule fois dans la colonne de droite		
Les éléments utilisés comme contacts peuvent être saisis autant de fois que nécessaire dans les 5 colonnes de gauche		
Les liaisons doivent toujours aller de gauche à droite		
Si dans un schéma, on utilise des bobine S (Set), utiliser aussi une bobine R (Reset).	Si on utilise pas de bobine R (Reset), la bobine correspondante sera toujours à 1.	Il est nécessaire d'utiliser une bobine R (Reset) pour la remettre à zéro.

## IV.4. Le logiciel de programmation Zelio Soft 2

### IV.4.1. Présentation

Le logiciel de programmation Zelio Soft 2 est conçu pour programmer les modules logiques de la gamme Zelio Logic. Zelio Soft 2 nous permet de choisir entre les langages de programmation, d'afficher les données du programme et des paramètres, de charger et télécharger des applications, ainsi que d'imprimer la documentation de l'application. [22]

#### Le logiciel "Zelio Soft 2" permet:

- La programmation en langage à contacts (LADDER) ou en langage à blocs fonctions (FBD)
- La simulation, le monitoring et la supervision
- Le chargement et le déchargement de programmes
- L'édition de dossiers personnalisés
- La compilation automatique de programmes l'aide en ligne. [21]

#### **IV.4.2. Langages utilisés**

Le module logique propose 2 modes de programmation :

- Mode LD : langage à contacts
- Mode FBD : langage en blocs fonction

Ces langages mettent en œuvre :

##### ***Des blocs fonction prédéfinis :***

- Temporisateurs
- Compteurs

##### ***Des fonctions spécifiques :***

- Gestion du temps
- Chaîne de caractères
- Communication...

Dans ce mémoire, seul le langage à contacts (LD) est utilisé pour la programmation de l'automate Zelio choisis, ce langage possède des symboles graphiques : contacts, bobines et blocs. L'exécution de calculs spécifiques est possible à l'intérieur des blocs opérations.

#### **IV.4.3. Modes de saisie des schémas de commande**

Le mode "saisie Zelio" permet à l'utilisateur ayant programmé directement sur le produit Zelio Logic de retrouver la même ergonomie, à la première prise en main du logiciel.

Le mode "saisie libre", plus intuitif, apporte à l'utilisateur un grand confort d'utilisation et de nombreuses fonctionnalités supplémentaires.

En langage de programmation LADDER, 2 types d'utilisation sont possibles :

- symboles LADDER,
- symboles électriques.

Le mode "saisie libre" permet aussi la création de mnémoniques et de commentaires associés à chaque ligne de programme. Le passage d'un mode de saisie à l'autre est possible à tout instant, par un simple clic de souris. Il est possible de programmer jusqu'à 120 lignes de schémas de commande, avec 5 contacts et 1 bobine par ligne de programmation [1, 12].

Fonction	Schéma électrique	Langage LADDER	Commentaire
Contact			i correspond à l'image réelle du contact câblé sur l'entrée du module. i correspond à l'image inverse du contact câblé sur l'entrée du module.
Bobine classique			La bobine est excitée lorsque les contacts auxquels elle est reliée sont passants.
Bobine à accrochage (Set)			La bobine est excitée lorsque les contacts auxquels elle est reliée sont passants. Elle reste enclenchée lorsque les contacts ne sont plus passants.
Bobine de décrochage (Reset)			La bobine est déexcitée lorsque les contacts auxquels elle est reliée sont passants. Elle reste inactivée lorsque les contacts ne sont plus passants.

Figure IV.4: Fonctions en symboles LADDER et symboles électriques

#### IV.4.4. Fonctionnalités

Le logiciel Zelio Soft 2 offre les fonctionnalités suivantes :

- 16 blocs fonctions Textes,
- 16 temporisateurs, chacun paramétrable parmi 11 types différents (1/10ème de secondes à 9999 heures),
- 16 compteurs/décompteurs de 0 à 32767,
- 1 compteur rapide (1 kHz),
- 16 comparateurs analogiques,
- 8 horloges, disposant chacune de 4 canaux,
- 28 relais auxiliaires,
- 8 comparateurs de compteurs,
- Écran LCD avec rétroéclairage programmable,
- Passage automatique heure d'été/heure d'hiver,
- Diversité des fonctions : bobine, à mémoire (Set/Reset), télérupteur, contacteur,
- 28 blocs messages [1, 12].

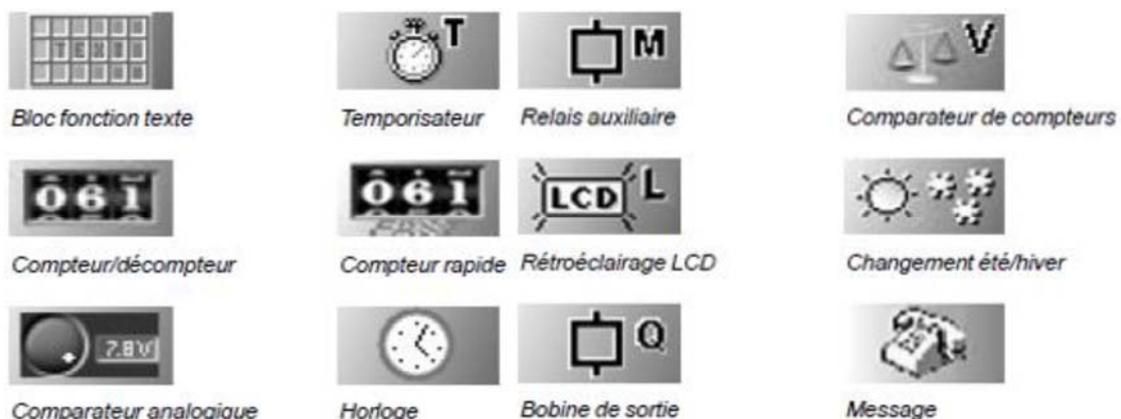


Figure IV.5: Différents outils de programmation en langage LADDER

### IV.4.5. Description d'un réseau de contacts

Un réseau de contacts est composé d'un ensemble d'éléments graphiques disposés sur une grille de :

- 120 lignes maximum de programme,
- Chaque ligne est composée de 5 contacts maximum et d'une bobine.

Il est réparti en deux zones :

- La zone test, dans laquelle figurent les conditions nécessaires au déclenchement d'une action (contacts),
- La zone action, qui applique le résultat consécutif à une combinaison logique de test (bobines) [1, 12].

La figure ci-après décrit la structure d'un réseau de contacts.

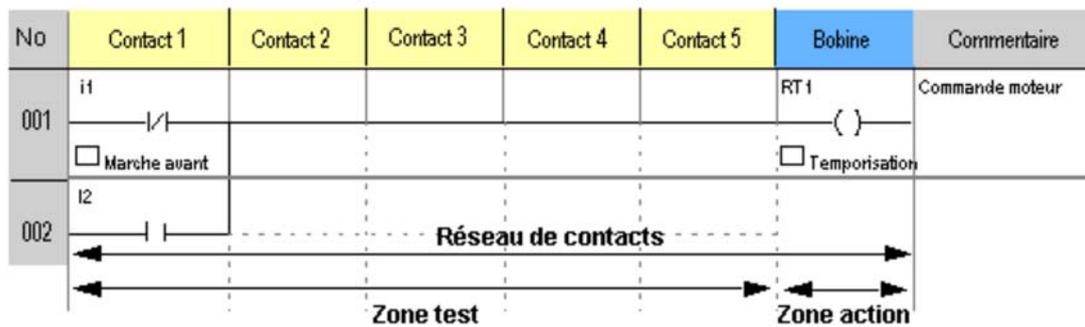


Figure IV.6: Structure d'un réseau de contacts

## IV.5. Transfert d'une application

### IV.5.1. Écrire d'un PC vers le Zelio Logic

Lorsque notre application est au point, on peut la transférer dans le Zelio Logic. Pour envoyer un programme vers le Zelio Logic, on doit aller dans le menu **Transfert, Transférer Programme** puis cliquez sur **PC->Module**.

Si le type de module sélectionné n'est pas compatible avec le type de module connecté, on peut changer le type de module dans **Module** puis **Choix du Module/Programmation**. Il est également possible d'effectuer un diagnostic du module connecté dans **Module** puis **Diagnostic du Module**.

Si le module connecté est en mode **RUN**, il est impossible de transférer le programme. On peut le mettre en mode **STOP** à partir du logiciel en sélectionnant **Transfert** puis **STOP Module** [1, 12].

Si le type de module sélectionné est le même que le type de module connecté, la fenêtre de dialogue suivante s'affiche :

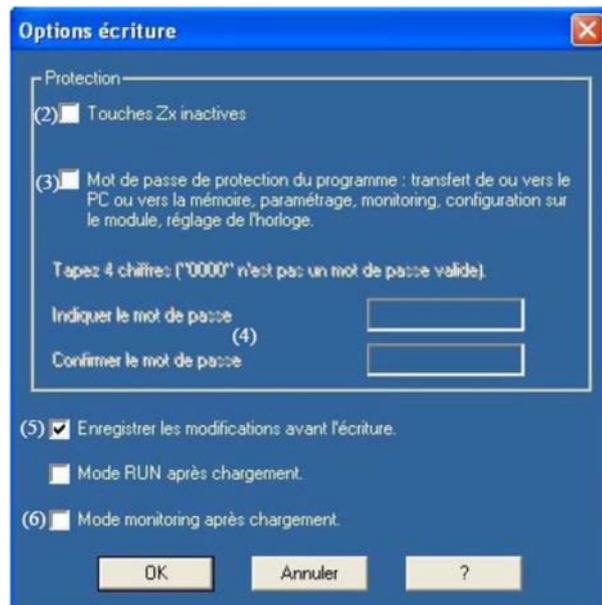


Figure IV.7: Fenêtre de dialogue

En (2) on choisit de rendre ou non les touches Zx (qui sont utilisées en entrée dans le programme) accessibles. Il est possible de protéger le programme présent dans le module par un mot de passe (3) qu'on saisit en (4). Pour enregistrer notre application sur notre ordinateur, on coche (5). Enfin pour lancer le mode monitoring, on coche (6). En cliquant ensuite sur **OK** le programme est transféré.

Le programme qui était présent dans le module avant le transfert est écrasé par le module.

Pour mettre en marche le module à partir du logiciel, on clique sur **Transfert** puis **RUN Module**.

Toutefois, lorsqu'on transfère un programme sur le module, celui-ci se met en mode **RUN** automatiquement [1, 12].

#### IV.5.2. Transfert du programme Zelio Logic vers le PC

Cette fonction de transfert permet de récupérer une application d'un module à l'aide du logiciel. À partir du logiciel, on va dans le menu **Transfert, Transférer Programme** puis on clique sur **Module-> PC**. Après une demande de confirmation, le transfert s'effectue. Le logiciel charge alors le programme présent dans le module.

Si le programme du module est verrouillé, le code vous sera demandé par le module avant le transfert [1, 12].

## IV.6. Description des étapes de programmation réalisé avec le logiciel Zelio Soft 2

### IV.6.1. Présentations

L'automate zelio de Schneider est l'automate utilisé pour la réalisation du travail présenté dans ce mémoire, il est choisi pour sa solution intelligente pour faciliter et réduire les câblages électriques. Avec une simplicité et la flexibilité de mise en œuvre de la solution.

### IV.6.2. Description matériel de l'automate SR3B101FU :

L'automate zelio utilisé dans ce travail comporte :

- 6 entrées appelées I1, i2, I3, I4, I5, I6 qui nous permet d'acquérir 6 informations en provenance de 6 capteurs différents.
- 4 sorties appelées Q1, Q2, Q3, Q4 qui nous permet de transmettre 4 ordres différents à exécuter.

Ci-dessous une description du boîtier automate Zelio.

- 1.Pattes de fixation rétractables.
- 2.Bornier à vis d'alimentation.
- 3.Bornier à vis des entrées TOR.
- 4.Afficheur LCD, 4 lignes, 18 caractères.
- 5.Touches de navigation ou après configuration boutons poussoir Z
- 6.Touche de sélection et validation (marche/arrêt).
- 7.Touche Shift.
- 8.Emplacement mémoire de sauvegarde ou câble de raccordement PC.
- 9.Bornier à vis sorties relais TOR.

Le logiciel Zelio soft 2 est utilisé avec le langage de contact Ladder (LD), la programmation nécessite une connexion PC via le port série à l'aide de cordon R2CBL01.

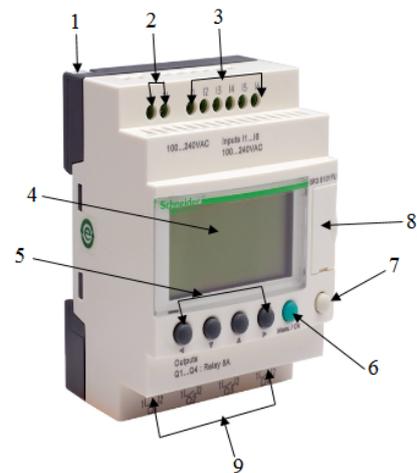


Figure IV.8: Automate zelio de Schneider.

### IV.6.3. Le logiciel Zelio soft 2 :

Est un logiciel 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 :

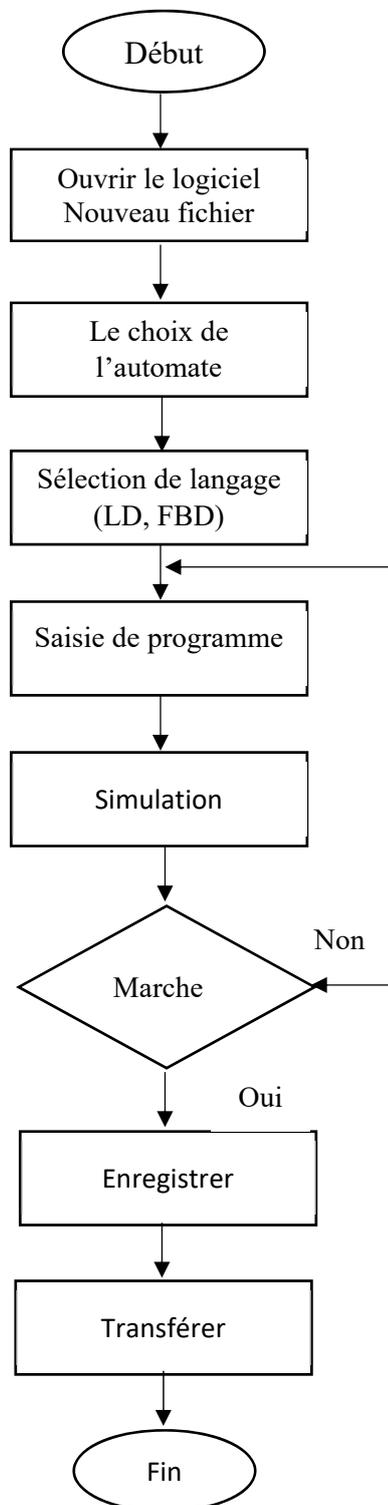
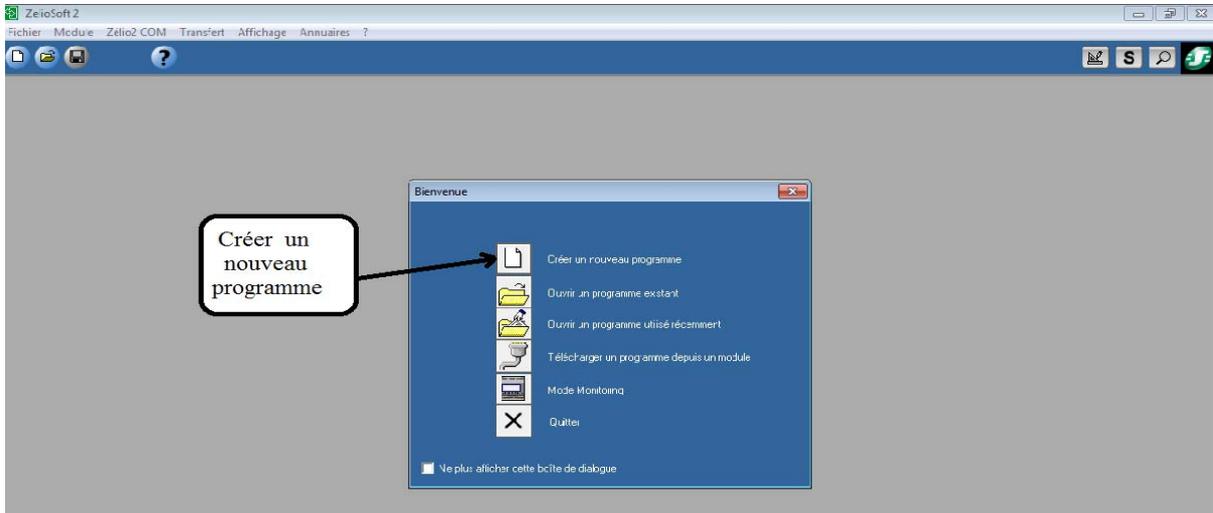


Figure IV.9: Étapes de programmation.

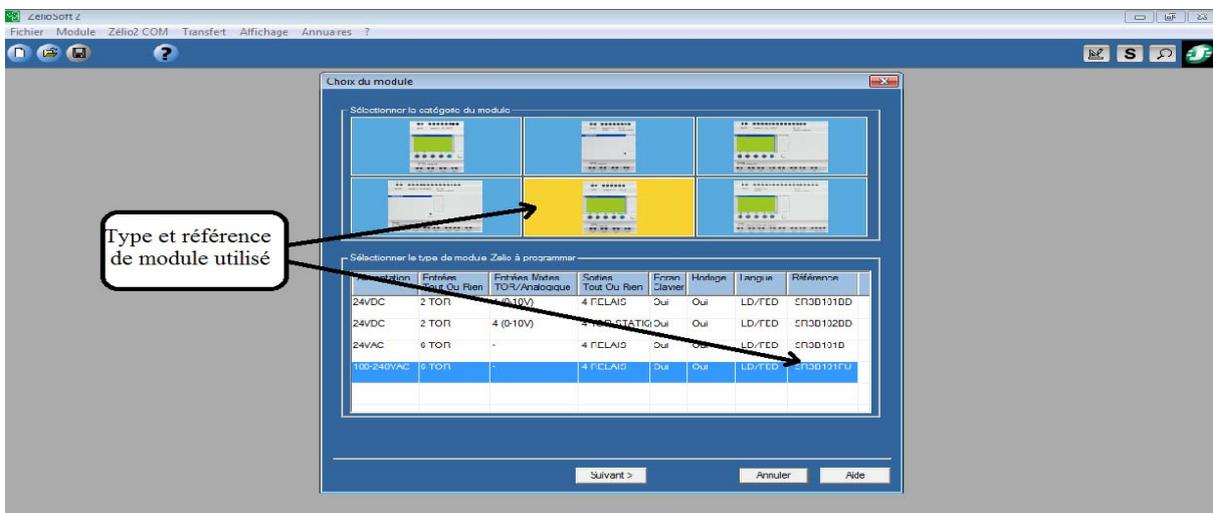
IV.6.3.1 Étape 1 :

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



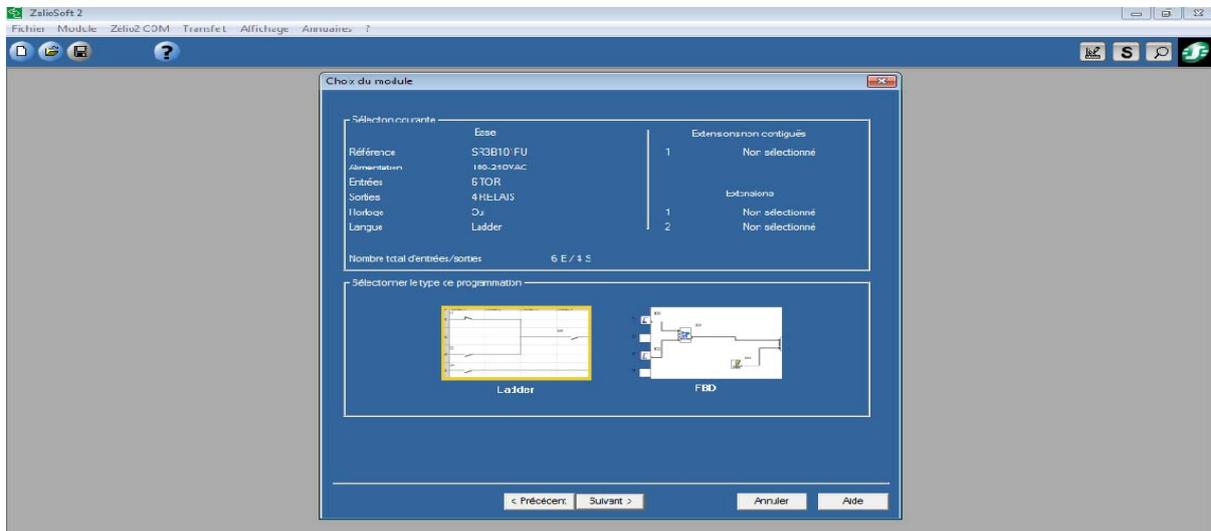
IV.6.3.2 Étape 2 :

Sélectionner le type et la référence de module à programmer, puis appuyer sur la touche Next



IV.6.3.3 Étape 3 :

Choisir le langage de programmation Ladder.

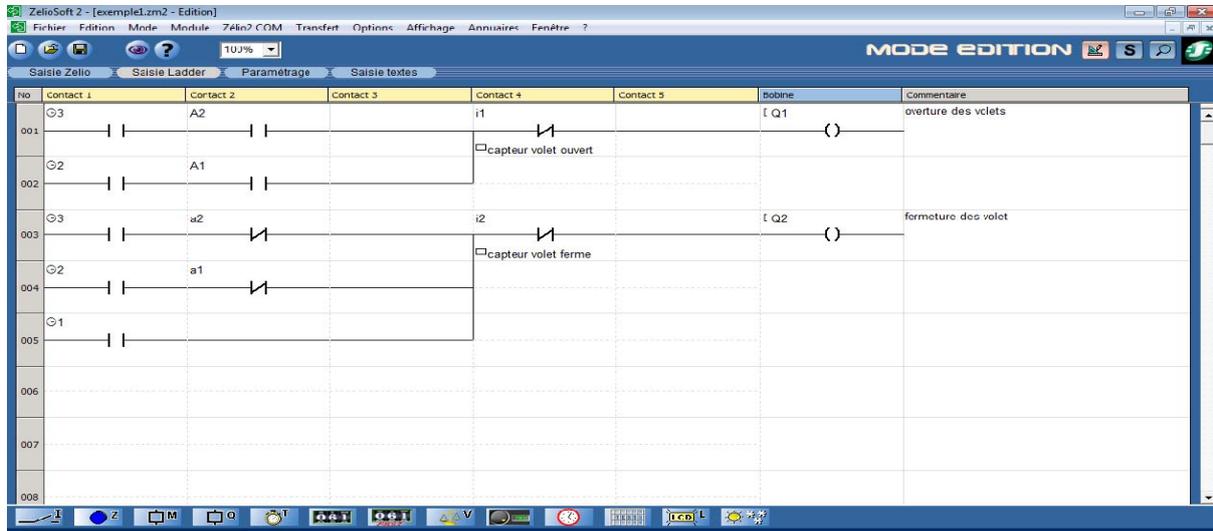


Une page d'édition de programme Ladder est affichée



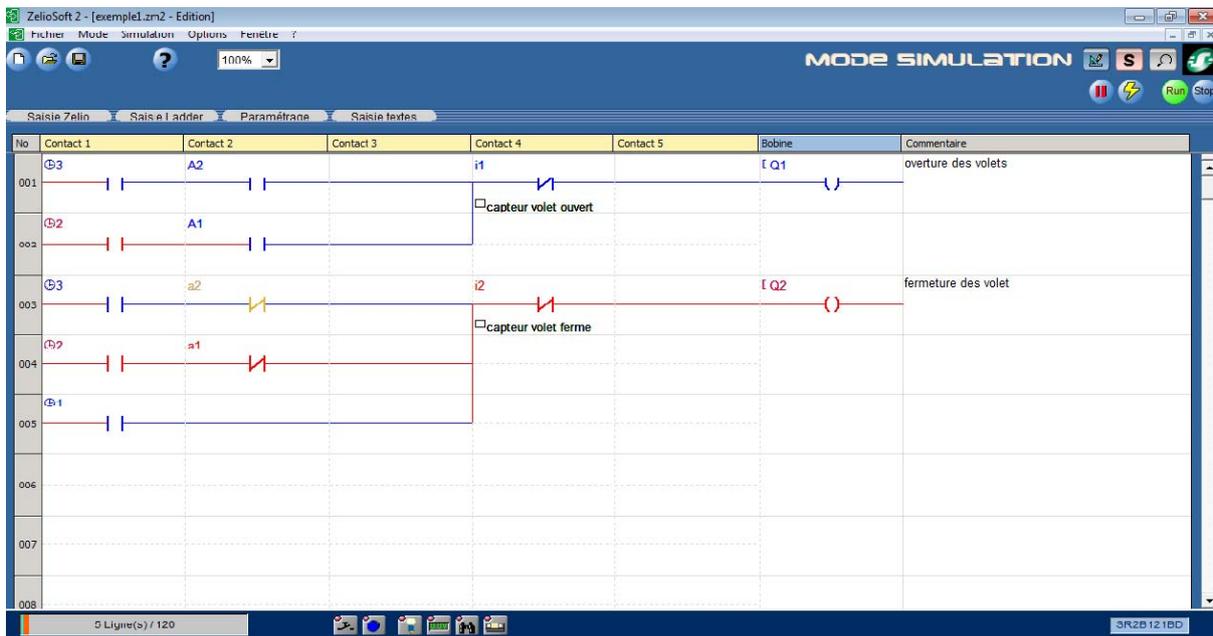
IV.6.3.4 Étape 4 :

Écrire le programme en langage Ladder.



IV.6.3.5 Étape 5 :

Simulation



La simulation permet d'exécuter le programme directement dans Zelio-Soft 2. Les procédures possibles sont la commande des entrées et l'affichage des sorties, la commande des touches Z, le pilotage des entrées analogiques, la visualisation/modification des paramètres des blocs fonctions et la simulation du bloc horloge

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



Pour passer en mode run, appuyer sur le bouton Run



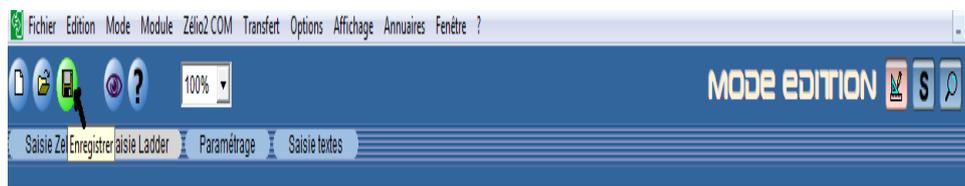
En cas de présence d'erreurs dans le programme on revient en mode édition pour apporter des modifications

Pour revenir mode édition, appuyer sur le bouton édition



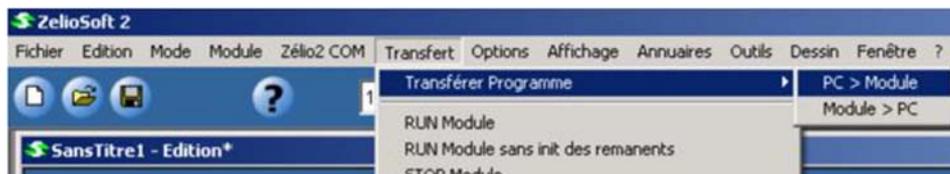
#### IV.6.3.6 Étape 6 :

Enregistrement du programme.



#### IV.6.3.7 Étape 7 :

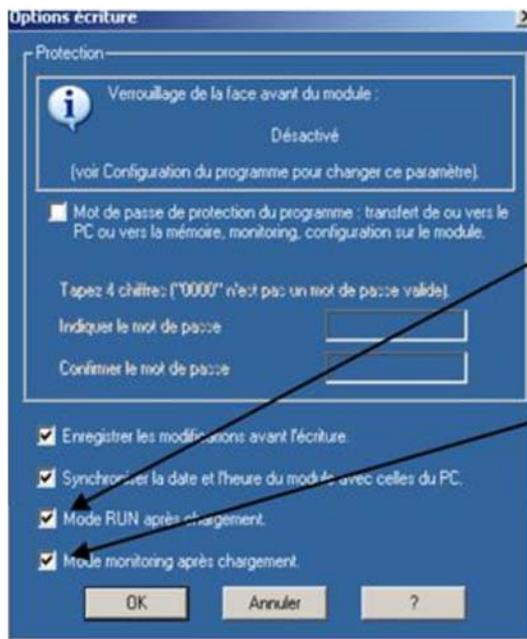
Transfert du programme dans le module.



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

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.

#### IV.7. Conclusion :

Dans ce chapitre nous avons parlés d'une description matérielle (boiter) et logiciel (le logiciel Zelio soft 2) de l'automate zelio SR3B101FU de Schneider utilisé dans notre manipulation.

Zelio Soft 2 est le logiciel nécessaire pour la programmation de l'automate Zelio, sachant que ce dernier est programmable directement sur son afficheur en langage Ladder , mais comme notre projet est une maquette sur laquelle une manipulation est effectués alors nous écrivons nos programmes avec le logiciel Zelio Soft 2, le programme sera donc sauvegardé. Pour effectuer une manipulation il suffira de transférer le programme correspondant vers l'automate.

# Chapitre V

## Simulation de la gestion de pression avec l'automate Zelio

### V.1. Introduction :

La coupure de l'alimentation en pression coute cher pour l'entreprise en terme de temps (arrêt de production pour diagnostique et résolution du problème) et en terme de bénéfice (rejet des composants) et en terme de risque sur les équipements (défaillance des encodeurs), pour cela une automatisation de la distribution de pression dans l'unité est nécessaire, c'est-à-dire concevoir un système qui favorisera l'alimentation des machine d'insertion automatique lors de la coupure de pression (chute de pression) au-dessous de 7bars qui est la valeur minimale acceptée par la machine.

Dans ce chapitre nous allons décrire la distribution actuelle dans l'unité de production des cartes électroniques, définir l'objectif et le cahier de charge de l'automatisation, puis une description de cycle de fonctionnement de la maquette associé à un organigramme et la partie programmation.

Ce chapitre a pour objectif de concrétiser la solution proposée pour répondre à la problématique.

**V.2. Description de la distribution de pression actuelle dans l'unité de production des cartes électronique :**

L'unité de cartes électronique est composée de de trois lignes d'insertion, dont la troisième est une ligne d'insertion manuelle et les deux autres sont des lignes d'insertion automatique.

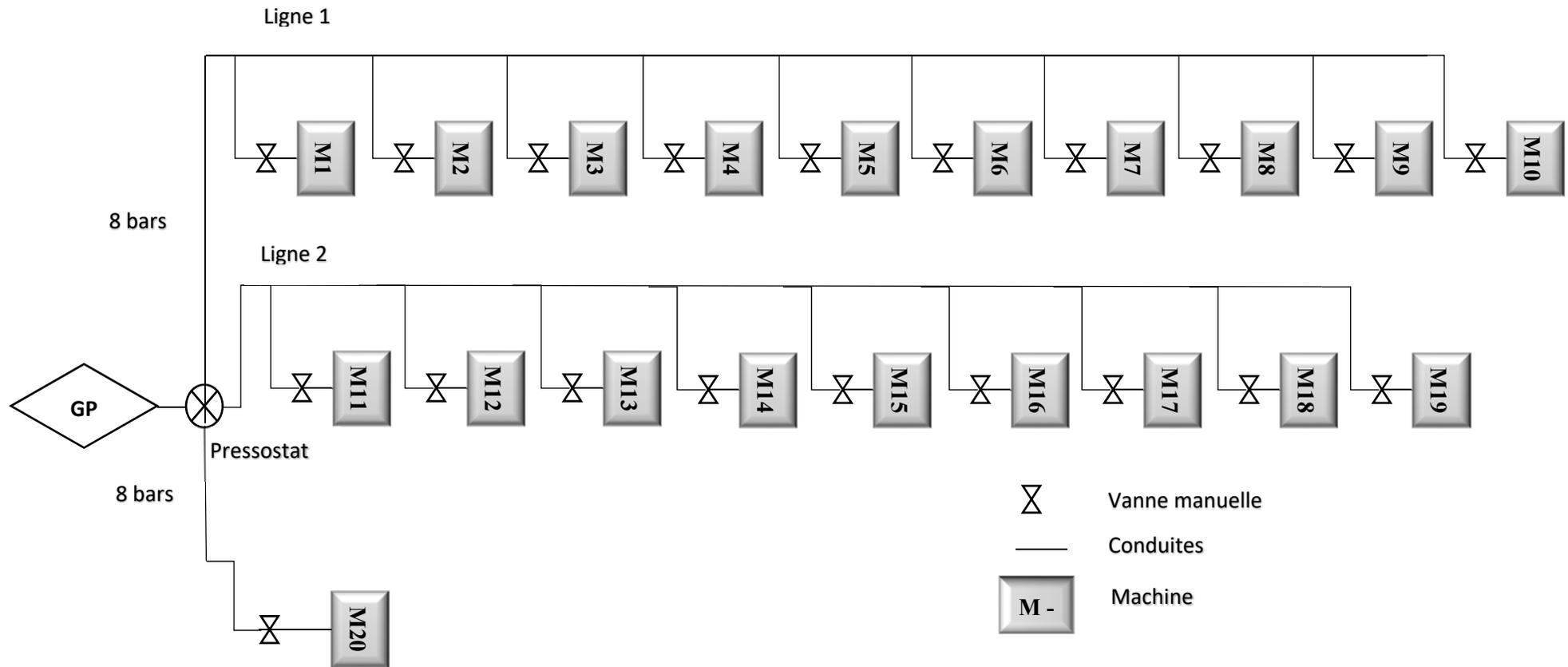


Figure V.1: Schéma synoptique de la distribution de pression actuelle.

La première ligne installée en 2006 est remplacée par une nouvelle acquise en 2017, elle est composée des organes suivants:

Abréviation	Nom et fonction de la machine
M1	Déchargeurs automatique de PCB «unloader »
M2	: Machine de sérigraphie pour le dépôt de crème à braser sur la carte. « screen printer »
M3	Machine d'inspection de la sérigraphie en 3D « SPI »
M4	Machine d'insertion automatique des petits composants CMS « chip shooter » à 4 têtes.
M5	Machine d'insertion automatique des petits composants CMS et des petits circuits intégrés «chip shooter » à 4 têtes.
M6	Machine d'insertion des Circuits intégrés « Multimount » à 2 têtes.
M7	: Machine d'inspection optique des composants insérés avant soudage au four « AOI »
M8	Machine d'inspection optique de la soudure des composants insérés après le four « AOI »
M9	chargeurs automatique de PCB «loader »
M10	Générateur d'azote 1

Tableau V.1: les éléments de la ligne 1

La deuxième ligne d'insertion automatique est installée en 2013, elle est composée des éléments suivants :

Abréviation	Nom et fonction de la machine
M11	Déchargeurs automatique de PCB «unloader »
M12	Machine de sérigraphie pour le dépôt de crème à braser sur la carte. « screen printer »
M13	Machine d'inspection de la sérigraphie en 3D « SPI »
M14	Machine d'insertion automatique des petits composants CMS « chip shooter » à 4 têtes.
M15	Machine d'insertion des Circuits intégrés « Multimount » à 2 têtes.
M16	Machine d'inspection optique des composants insérés avant soudage au four « AOI »
M17	Machine d'inspection optique des composants insérés après soudage au four « AOI »
M18	chargeurs automatique de PCB «loader »
M19	Générateur d'azote 2

Tableau V.2: les éléments de la ligne 2

Ligne d'insertion Manuelle est représentée par le bain de soudure (M20).

Abréviation	Nom et fonction de la machine
M20	Bain de soudure « Wave soledring »

Tableau V.3: élément de la ligne manuelle

### V.3. Objectif de l'automatisation :

Le principe est de couper la pression de toutes les machines, et de garder que les 5 machines M4 M5 M6 M14 et M15 alimentés, lors d'un défaut sur le compresseur d'air. Ces 5 machines assurent l'insertion des composants CMS sur le circuit imprimé, avec une grande vitesse nécessitant une pression minimale de 7 Bars. Au-dessous de cette valeur, elles s'arrêtent immédiatement et indiquent des défauts liés à l'insuffisance de la pression, tous les composants qui sont chargés sur chaque têtes seront automatiquement jetés à n'importe quel endroit à l'intérieur de la machine (risque de défaillance de la machine). Toutes les cartes en cours d'insertion seront éliminées du processus, de fait qu'elles sont incomplètes et exposées aux chutes des composants sur la crème à braser (qui est l'étain sous forme de crème), surtout sur les emplacements des circuits de précision.

Pour la réalisation, on propose de refaire l'installation actuelle par la séparation des 2 conduites, c'est-à-dire, placer une conduite qui alimente directement les 5 machines sensibles et une deuxième conduite alimente le reste des machines. Les deux conduites seront contrôlées par deux électrovannes.

Tout cela doit être accompagné d'un signal lumineux et sonore qui informe l'opérateur qu'une coupure a eu lieu afin d'intervenir et faire le nécessaire pour reprendre l'activité.

En cas ou le niveau de pression atteint à nouveau la valeur d'enclenchement prédéfinie, le système revient à l'état initial du fonctionnement normal.

Le schéma synoptique de la distribution de pression automatisé est donné ci-dessous :

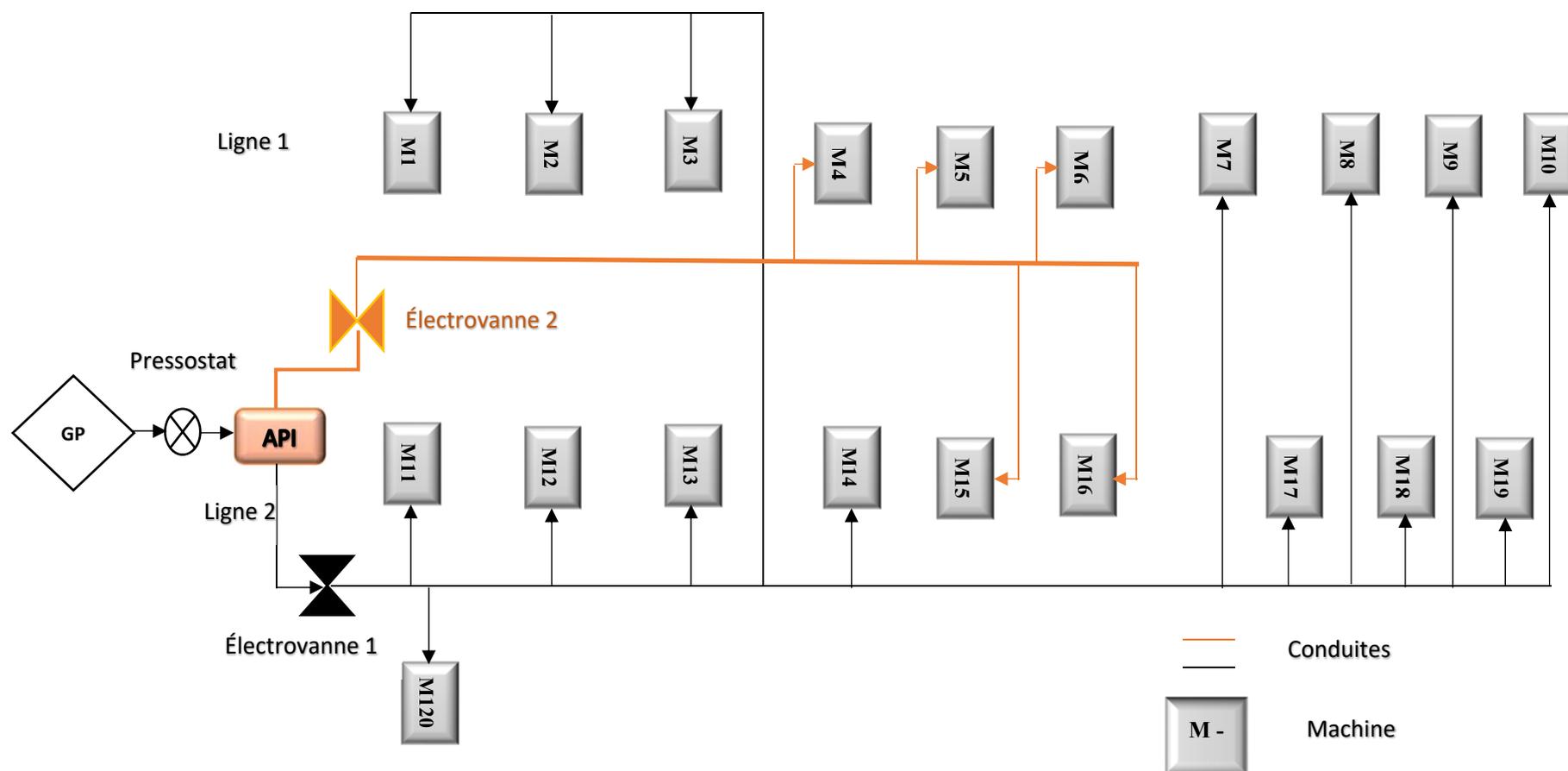


Figure V.2: Schéma synoptique de la distribution de pression automatisé.

#### V.4. Cahier des charges :

Les éléments cités ci-dessous sont les organes constituant la maquette réalisée :

- Un automate Zelio de Schneider de type SR3B101FU.
- Deux LED qui représentent les deux électrovannes utilisées dans l'automatisation.
- Une LED rouge qui représente le signal lumineux d'alarme.

Pour l'état de compresseur (marche /arrêt) et la valeur de pressostat (sup ou inf. à 7 bars) les touches de sélection de l'automate sont exploitées.

La fonction de l'automate consiste à allumer les deux LED jaune et verte, après la détection d'un défaut la LED rouge s'allume, ce qui donne ordre au deux LED de s'éteindre l'une après l'autre avec un temps de décalage.



Figure V.3: la maquette réalisée

#### V.5. Programmation :

La programmation des différentes actions du cycle de fonctionnement est effectuée en langage Ladder (LD) par le logiciel Zelio Soft 2. Avec une illustration de la temporisation utilisée dans le cycle de fonctionnement. Dans ce qui suit, nous présentons l'organigramme de fonctionnement ainsi qu'une description de cycle de fonctionnement de la maquette.

### V.5.1 Création du programme de gestion de la pression

Le module programmable utilisé avec le langage Ladder sous le logiciel Zelio Soft 2. Après l'installation de ce dernier sur un PC fonctionnant avec la plateforme Windows 7, l'installation du pilote du câble de communication est nécessaire.

Après l'édition du programme et la simulation, il est transféré vers l'automate Zelio SR3B101FU, ensuite on a procédé à son fonctionnement décrit ci-dessous. Le programme et la simulation avec l'automate Zelio ont répondu correctement aux cahiers des charges définis.

Ci-après une partie du programme Ladder de la gestion de pression dans l'unité de production de Bomare Company (figure 4)

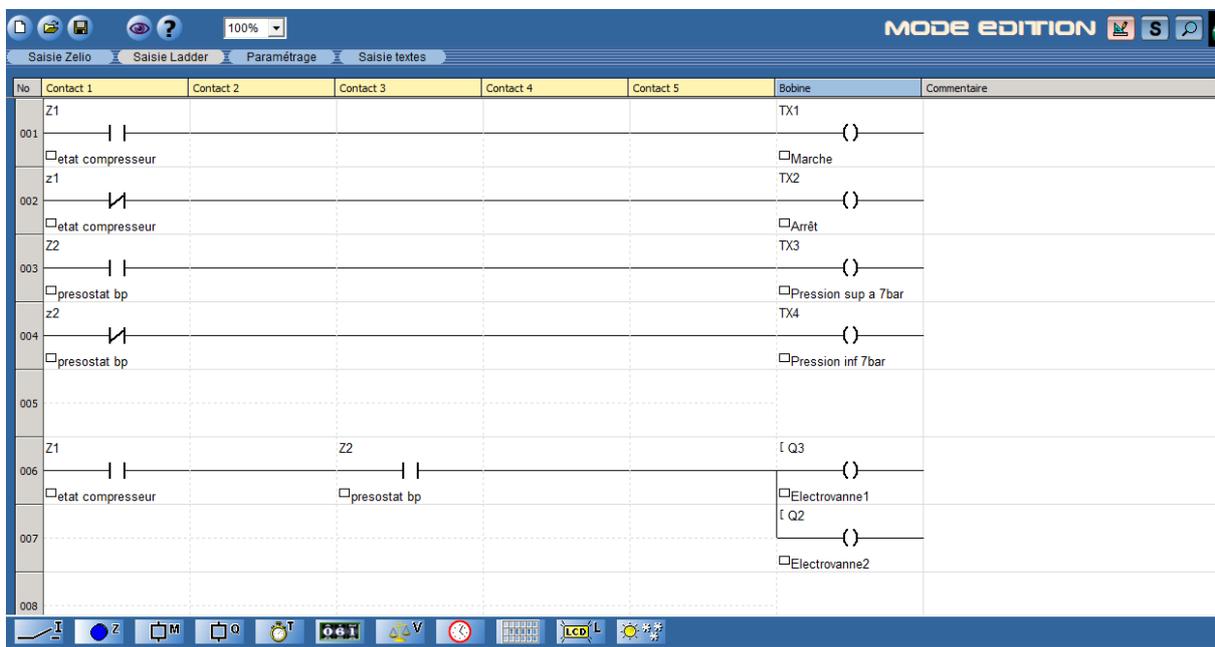


Figure V.4: Partie du programme Ladder

### V.5.2 La nomenclature de programme :

Le tableau suivant nous montre la nomenclature du programme de gestion de pression écrit avec en langage Ladder avec le logiciel Zelio soft 2.

La nomenclature	Le commentaire
Z1	Compresseur en marche
z 1	Compresseur en arrêt
Z2	Pressostat pression sup 7bars
z 2	Pressostat pression inf. à 7bars
Q1	Défaut signal lumineux
Q2	Electrovanne2
Q3	Electrovanne1
T1	La temporisation

Tableau V.4: Nomenclature du programme

### V.5.3 La temporisation :

Après une chute de pression, le programme signale un défaut pour donner l'information au personnel de l'unité de production afin de pouvoir intervenir, ce qui pousse à définir une temporisation (figure V.5) entre l'actionnement des deux électrovannes. Donner la priorité à l'électrovanne qui alimente les cinq (05) machines d'insertion des composants CMS.

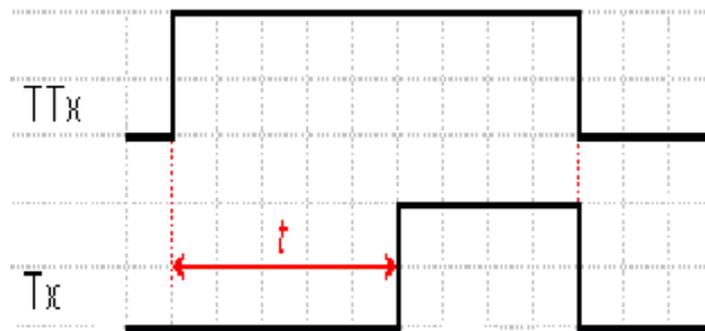


Figure V.5: Temporisation

### V.6. Organigramme de fonctionnement

Le fonctionnement de la maquette est donné dans l'organigramme suivant (figure V.6):

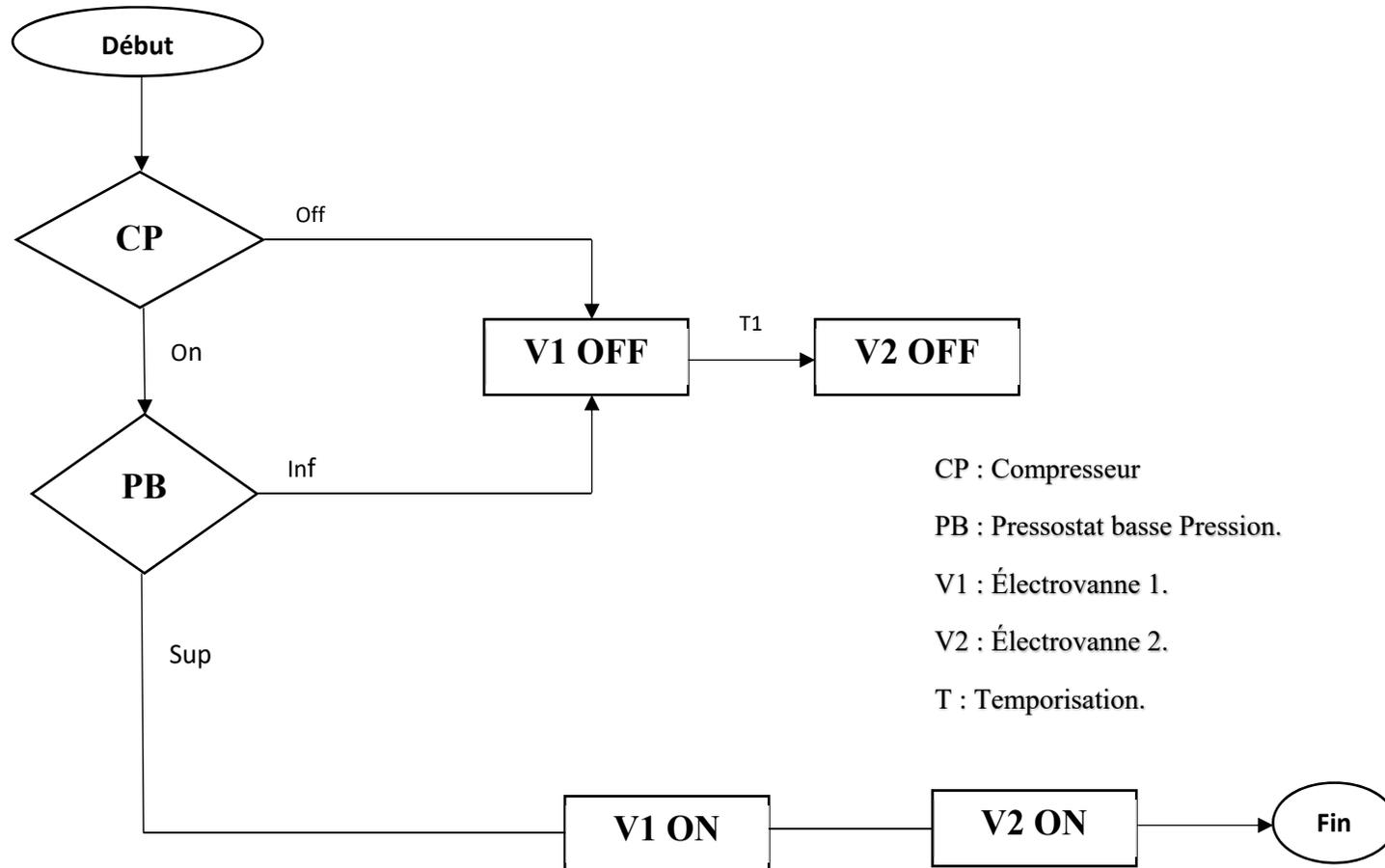


Figure V.6: Organigramme de fonctionnement

## V.7. Description de cycle de fonctionnement de la maquette :

Le début de cycle commence par un appui sur les boutons de sélections (1,2) sur le boîtier de l'automate Zelio.

Les actions du cycle se déroulent selon la chronologie suivante :

### V.7.1 Étape 01 :

En appuyant sur les deux boutons 1 et 2, les deux LED vert et jaune s'allument au même temps.

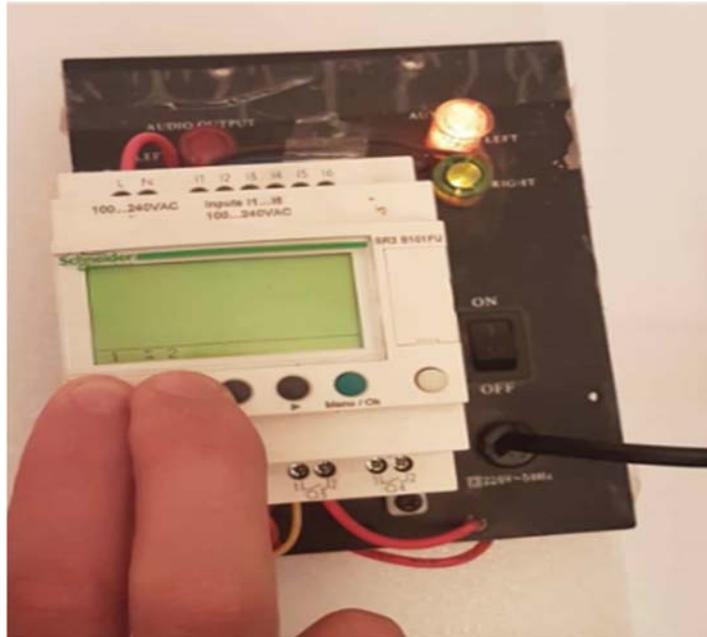


Figure V.7:Étape 01

### V.7.2 Étape 02 :

Une fois que l'un des boutons est relâché la LED rouge s'allume et la LED verte s'éteint.



Figure V.8: Étape 02

**V.7.3 Étape 03 :**

La LED jaune demeure allumée pendant 20s et puis elle s'éteint à son tour.

**V.7.4 Étape04 :**

La LED rouge demeure allumée jusqu'à ce que les deux boutons seront enfoncés à nouveau c'est-à-dire la valeur de pression sera rétablie.



Figure V.9: Étapes 3 & 4

**V.8. Conclusion :**

Dans ce chapitre nous avons présenté la distribution de pression existante dans l'unité de production de Bomare Company.

Après avoir décrit l'objectif de l'automatisation et le cahier des charges de notre projet, une maquette pour une simulation de gestion de pression est montée, ensuite la programmation avec le langage Ladder sous le logiciel Zelio Soft 2. Ensuite le transfert du programme à l'automate suivi d'un test de fonctionnement.

On peut conclure que la gestion de la pression de l'air dans l'unité de production de Bomare Company avec un automate programmable est une chose réalisable, simple et efficace.

# Conclusion générale et perspectives

Le programme avec le langage Ladder élaboré dans ce travail avec le logiciel Zelio Soft 2 injecté dans l'automate Zelio SR3B101FU, nous a permis de réaliser la simulation avec l'observation des résultats (allumage des LED) relatifs à la gestion de la pression d'air dans les machines d'insertion automatique dans l'unité de production de Bomare Company. Au cours de ce projet de fin d'études, nous avons pu concrétiser notre idée qui consiste à réaliser une maquette d'un système automatisé qui gère la distribution de la pression d'air dans l'unité de production des cartes électroniques au sein de de l'entreprise Bomare Company.

La réalisation de cahier des charges sur la maquette a passé par les étapes suivantes :

- Etablir l'organigramme de fonctionnement.
- Identifier les entrées et les sorties ainsi que les temporisateurs à utiliser.
- Créer un nouveau projet dans le logiciel Zelio Soft 2 en choisissant le module programmable dont on dispose.
- Elaborer les programmes sur le logiciel.
- Simuler les programmes avant le transfert vers l'automate.
- Transférer le programme et tester la maquette.
- Observation des résultats (allumage des LEDs qui représentent respectivement l'electrovanne1, l'electrovanne2 et le défaut).

Nous sommes arrivés à réaliser ce projet grâce aux connaissances théoriques que nous avons acquises au cours de notre formation académique d'une part, et d'autre part, les connaissances acquises pendant le stage pratique chez Bomare Company.

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 et spécialement l'automate Zelio de Schneider.

Afin d'élargir le spectre d'application de la méthode proposée, plusieurs perspectives peuvent être envisagées. Elles sont citées ci-dessous :

- Ce projet peut être réalisé, on se basant sur une autre solution qui consiste, à équiper chaque machines d'une électrovanne.
- Employer un automate qui possède plus de sorties afin de commander toutes les électrovannes.
- L'implémentation de la commande sur un automate programmable de plus haute gamme et l'utilisation d'un logiciel d'élaboration de l'interface Homme / machine plus exhaustif.

# Références Bibliographiques

- [1] : **Bouchra. FERHAOUI**, *gestion d'une maquette de station de pompage par un automate programmable Zelio*, mémoire de master en automatique, école nationale polytechnique, 2015.
- [2]: **Houda BEL MOKADEM**, *Vérification des propriétés temporelles des automates programmables industriels*, thèse de doctorat, l'école normale supérieure de Cachan, 2007.
- [3]: **Gérard GUIHENEUF**, *Automate programmable en Ladder et FBD, Domotique avec Zelio Logic – Etude – Programme, Réalisation – Mise en Service*, Catalogue, elector, 07/2015.
- [4]: **Anaïs GUIGNARD**, *Validation fonctionnelle de contrôleurs logiques: contribution au test de conformité et à l'analyse en boucle fermée*, thèse de doctorat, école normale supérieure de Cachan, 2015.
- [5]: **Gilles Michel**, *Architecture et applications des automates programmables industriels*, Dunod, 1987.
- [6]: **Ir. H. LECOCQ**, *Les automates programmables, Caractéristiques et méthodologie de programmation, Tome 1*, université de Liège, Faculté des sciences appliquées, 2005.
- [7]: **Naila ABDELATIF**, *Dimensionnement et optimisation du nouveau Centre de Distribution local, Cas Schneider Electric Algérie*, mémoire de fin d'études école nationale polytechnique, 2017.
- [8] : **Alla Eddine AOUADI, Lyes OUTIOUA**, *Contribution à l'amélioration de la chaîne logistique aval au niveau du centre de distribution Schneider Electric Algérie*, mémoire de fin d'études, école nationale polytechnique, 2011.
- [9]: **Sandra BENELMADJAT**, *Implémentation de la commande d'une machine de lyophilisation sur un automate programmable (avec élaboration d'une interface homme machine)*, mémoire de fin d'études, école nationale polytechnique, 2003.
- [10]: **Lyes Heythem BETTACHE**, *Commande et supervision d'une station de pompage didactique avec l'automate Zelio*, mémoire de fin d'études, école nationale polytechnique, 2017.
- [11]: **Salah BOULKHRACHEF**, *Analyse et commande d'un onduleur a cinq niveaux a structure a NPC. Application à la machine asynchrone commandée par logique floue*, mémoire de magister, école nationale polytechnique, 2001.
- [12]: **Amel HAMDY CHERIF, Imane MAYOUF**, *Commande et supervision d'une station de pompage à base d'automates programmables (Automates Scheider et Siemens)*, mémoire de fin d'études, école nationale polytechnique, 2018.
- [13]: **Anas MEDJAOURI, Mohamed ARIM**, *Commande et supervision du bâtiment intelligent Schneider Electric*, mémoire de fin d'études, école nationale polytechnique, 2010.

- [14] : **Abdelhafidh SAHRAOUI**, *Contribution à l'amélioration de la productivité par les outils VSM et MTM, Application : Ligne d'assemblage manuel SM6-36 Schneider Electric Algérie*, mémoire de fin d'études, école nationale polytechnique, 2014.
- [15] : **Anis SALLEMI**, Enseignant concepteur, Mohamed BEN AHMED, concepteur multimédia site web, lien : [http://www.uvt.rnu.tn/resources-uvt/cours/Automates/chap2/co/chap2\\_web.html](http://www.uvt.rnu.tn/resources-uvt/cours/Automates/chap2/co/chap2_web.html).
- [16] : **Gilles Michel, Claude Laurgeau et Bernard Espiau**, *Les automates programmables industriels*, Dunod, 1979.
- [17] : **William BOLTON**, *Automates programmables industriels*, Dunod, 2015.
- [18] : **Mohammed MAATOU, Abderrahman BELLAGH**, *Automatisation et réalisation à petite échelle (maquette) d'une chaîne transporteuse de briques*, mémoire de fin d'études, Université Hassiba Benbouali de chlef, 2016.
- [19] : **Simon Moreno et Edmond Peulot**, *LE GRAFCET : Conception-Implantation dans les Automates Programmables Industriels*, Edité par Casteilla, 1996.
- [20] : **Alain GONZAGA**, *Les automates programmables industriels*, cours.
- [21] : Schneiderelectric. Modules logiques Zelio Logic. Edition 1er semestre 2004
- [22] : Schneiderelectric. Guide de programmation du module Zelio Logic. Edition 10/2017.
- [23] : Bomare Company exporte ses produits « Stream System » vers l'Italie, <http://www.nticweb.com>
- [24] : Entretien avec le DG de Bomare Company Mr A.BOUMEDIENE, <http://www.nticweb.com>
- [25] : Présentation de Bomare Company, <https://www.youtube.com/watch?v=ahpmIP3UYz4>
- [26] : Site web Bomare Company, [www.bomarecompany.com/](http://www.bomarecompany.com/)