

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université 8 Mai 1945 – Guelma
Faculté des Sciences et de la Technologie
Département d'Electronique et Télécommunications



Mémoire de fin d'étude
Pour l'obtention du diplôme de Master Académique

Domaine : **Sciences et Technologie**

Filière : **Electronique**

Spécialité : **Instrumentation**

**Programmation et supervision d'un système didactique en utilisant
l'automate SIEMENS S7-300 et le système SCADA**

Présenté par :

Brahmia Abd Allah

Sous la direction de :

Dr.Nemissi Mohamed

Mai 2023

Résumé

L'objectif de cette étude consiste à étudier et programmer une station didactique faisant partie d'un ensemble de stations qui simulent une mini-usine d'assemblage de petits vérins. Cette station est située au département d'électronique et de télécommunications de l'université de Guelma. L'importance de cette étude réside essentiellement dans le fait que la station comporte différents aspects industriels tels que la pneumatique (vérins et distributeurs) et l'utilisation de divers capteurs. Les étapes suivies incluent l'examen du fonctionnement de la station, la création d'un Grafcet correspondant au fonctionnement souhaité, la configuration matérielle et la programmation du Grafcet en utilisant deux langages de programmation, à savoir le langage contact et le langage Graphes de fonctions séquentielles. De plus, une interface SCADA a été programmée pour superviser et contrôler le système étudié.

Summary

The objective of this study consists in examining and programming a didactic station that is part of a set of stations that simulate a mini-factory for assembling small cylinders. This station is located at the department of electronics and telecommunications of the University of Guelma. The importance of this study lies essentially in the fact that the station includes different industrial aspects such as pneumatics (cylinders and distributors) and the use of various sensors. The steps followed include the examination of the operation of the station, the creation of a Grafcet corresponding to the desired operation, the hardware configuration and the programming of the Grafcet using two programming languages, namely the Ladder language and the Sequential Function Graph language. Finally, a SCADA interface has been programmed to supervise and control the studied system

ملخص :

الهدف من هذه الدراسة هو فحص وبرمجة محطة تعليمية تشكل جزء من مجموعة من المحطات التي تحاكي مصنعا صغيرا لتجميع الأسطوانات الصغيرة. تقع هذه المحطة في قسم الإلكترونيات والاتصالات بجامعة قالمة. تكمن أهمية هذه الدراسة بشكل أساسي في أن المحطة تتضمن جوانب صناعية مختلفة مثل الهواء المضغوط (اسطوانات وموزعات) واستخدام أجهزة استشعار مختلفة. وتشمل الخطوات المتبعة دراسة تشغيل المحطة، وإنشاء "Grafcet" المطابق للعملية المطلوبة، وتكوين الأجهزة ل API وبرمجة " Grafcet " باستخدام لغتين البرمجة، وهي لغة الاتصال و الرسم البياني ولغة الوظائف التسلسلية وأخيرا، تمت برمجة واجهة SCADA للإشراف والتحكم في النظام المدروس

Dédicace

Je dédie ce modeste travail à :

Ma chère mère.

*Nos frères et nos sœurs, que j'aime
beaucoup.*

Nos cher ami (e) s.

A toute la promo 2022-2023.

*A tous ceux qui m'ont aidé de près ou de
loin pour réaliser ce travail.*

Que dieu leur accorde santé et prospérité

Remerciements

Avant tout, je tiens à remercier DIEU De m'avoir donné la force et le courage de mener à bien ce modeste travail.

*Je tiens à remercier mon encadreur **Mr. NEMISSI Mohamed** pour avoir accepté de diriger ce travail, je remercie pour sa confiance, ses conseils et son aide précieux.*

Aussi je remercie membres du jury pour leur présence, pour leur lecture attentive de ma mémoire ainsi que pour les remarques qu'ils m'adresseront lors de cette soutenance afin d'améliorer mon travail.

Je reconnaisances vont à tous ceux qui m'ont aidés de près ou de loin à réaliser mon mémoire

Merci à vous tous

SOMMAIRE

Introduction générale	1
------------------------------	----------

Chapitre I

Introduction au système automatisé

I.1 Introduction	3
I.2 Principe de l'automatisation	3
I.3 Les dispositifs d'entrée/sortie	5
I.3.1 Les dispositifs d'entrée	5
I.3.2 Les dispositifs de sortie	6
I.4 Pneumatique et hydraulique	6
I.4.1 Pneumatique	7
I.4.2 Hydraulique	9
I.4.3 Comparaison entre Pneumatique et hydraulique	9
I.5 Conclusion	15

Chapitre II

L'automate S7-300 de Siemens

II.1 Introduction aux Automates Programmables Industriels	12
II.2 Avantages des API et de la logique programmée	12
II.3 Structure des API	13
II.4 Langages de programmation des API (norme CEI 61131-3)	14
II.4.1 Le langage à contacts (LAD, Ladder Diagram)	15
II.4.2 Les listes d'instructions (IL, Instruction List)	16
II.4.3 Les diagrammes de schémas fonctionnels (FBD, Function Block Diagram).	16
II.4.4 Le texte structuré (ST, Structured Text)	17
II.4.5 Les graphes de fonction séquentielle (SLC, Sequential Function Charts)	17

II.5 Les API de Siemens	18
II.5.1 Aperçu général	18
II.5.2 CPU S7-300	18
II.5.3 Zones mémoire	20
II.5.4 Les Modules entrée/sortie (Signal modules)	21
II.5.5 Les Modules de communication	23
II.5.6 Interfaces de bus	24
II.5.7 Programmation structurelle	25
II.6 Conclusion	26

Chapitre III

Réseaux industriels et supervision

III.1 Introduction	27
III.2 Connexions distantes	27
III.2.1 Communications série et parallèle	29
III.2.2 Normes de communications séries	29
III.2.3 Normes de communications parallèles	31
III.2.4 Protocoles	31
III.3 Les réseaux industriels	31
III.3.1 Système distribués	32
III.3.2 Normes Réseaux	32
III.3.3 Exemples de réseaux industriels	33
III.4 Les réseaux de SIMATIC NET	34
III.4.1 MPI (Multi-Point Interface)	35
III.4.2 PROFIBUS	35
III.4.3 Ethernet	36
III.5 Supervision des systèmes industriels	37
III.5.1 Le système SCADA	37

III.5.2 Architecture de système SCADA	38
III.6 Conclusion	40

Chapitre IV

Application

IV.1 Présentation des systèmes MPS	41
IV.1.1 Introduction	41
IV.1.2 Présentation des systèmes MPS	42
IV.1.3 Présentation des stations	42
IV.2 La station de tri	44
IV.2.1 Présentation	44
IV.2.2 Fonction de la station	44
IV.2.3 Séquence	44
IV.2.4 Les composants de la station:	46
IV.2.5 Les entrées sorties de la station de tri	48
IV.3 Grafcet de la station	49
IV.3.1 Grafcet niveau 1	49
IV.3.2 Grafcet niveau 2	51
IV.4 Configuration et programmation de l'automate en utilisant grafcet	53
IV.4.1 Présentation du de step7	53
IV.4.2 Configuration matérielle	53
IV.4.3 Programmation	54
IV.5 Supervision avec SCADA	55
IV.6 Conclusion	55
Conclusion générale	56

Liste des Abréviations

API : Automate Programmable Industriel

SCADA: Supervisory Control And Data Acquisition

LIST: Langage Liste

LOG : Langage Logigramme

CONT : Langage Contact

OB : Organization Block

FB : Bloc Fonctionnel

DB : Data Bloc

FC : Bloc Fonctions

IHM : Interface Homme Machine

GRAFCET : Graphe de Commande Etape Transition

CPU : Unité Centrale de Traitement

PLC : Contrôleur Logique Programmable

ST: Structured Text

FBD: Fonction Bloc Diagram

LD : Ladder Diagram

IL : Instruction List

SFC : Organigramme des Fonctions séquentielles

DP : Decentralized Periphery

MPI : Multi Point Interface

Ptp : Point à Point

RTU : Remote Terminal Unit

WinCC: Windows Control Center

Liste des tableaux

Tableau II .1	Indication l'état de LED	20
Tableau II.2	Les diffèrent zones d'opérandes	21
Tableau III.1	Présentation et caractéristiques de MPI, PROFIBUS et sous-réseau Ethernet.	37
Tableau IV.1	Contenu et caractéristiques de la formation des stations MPS qui se trouve Au sein du département de l'électronique et télécommunication, université de Guelma	43
Tableau IV.2	Capteur et actionneur de la station de tri.	48

Liste figure

CHAPITRE I

Figure I.1	Les CAP, les CISP et les CAP effectuent le contrôle, la Communication et d'autres tâches.	4
Figure I.2	Système pneumatique.	8

CHAPITRE II

Figure II.1	Structure générale d'un API	13
Figure II.2	Exemple de programme en langage contact	15
Figure II.3	Exemple de programme en langage liste	16
Figure II.4	Exemple de programme en diagrammes de schémas fonctionnels	17
Figure II.5	Exemple de programme en langage séquentiel	17
Figure II.6	API SIEMENS S7-300	18
Figure II.7	Eléments d'affichage d'un CPU 317-2	19
Figure II.8	Exemples de modules d'entrée numérique	22
Figure II.8	Exemple de structure d'un programme d'automatisation selon la norme IEC61131-3.	26

CHAPITRE III

Figure III.1	Utilisation d'un module d'entrées-sorties distant.	28
Figure III.2	Utilisation d'API pour des entrées-sorties distantes.	28
Figure III.3	connecteur de type DB à 25 broches.	30
Figure III. 4	Niveaux des signaux envoyés via l'interface RS232.	30

Figure III.5	Topologies de réseaux : (a) en étoile, (b) en bus et (c) en anneau.	32
Figure III. 6	MPI typique (sous-réseau d'interface multi-points).	35
Figure III. 7	Sous-réseau PROFIBUS typique.	36
Figure III. 8	Sous-réseau Ethernet typique.	36
Figure III.9	Illustration de l'architecture d'un système SCADA simple.	39

CHAPITRE IV

Figure IV.1	Exemple de quatre stations du système MPS	41
Figure IV.2	La station de tri (Festo)	42
Figure IV.3	Formes et couleurs des pièces utilisées	45
Figure IV.4	Module de transport	46
Figure IV.5	Module glissière	47
Figure IV.6	Configuration matérielle du CPU utilisé pour l'automatisation de la station de tri	53
Figure IV.7	Interaction logiciel-matériel	54
Figure IV.8	Interface SCADA	55

Introduction générale

Introduction générale:

Actuellement, l'industrie est contrainte de produire des biens de haute qualité et en grande quantité pour répondre à la demande dans un environnement hautement concurrentiel. Pour y parvenir, les entreprises cherchent à améliorer leurs performances, notamment en optimisant leurs systèmes de production à l'aide de machines capables d'effectuer des tâches difficiles et pénibles et de fonctionner automatiquement. Les constructeurs de systèmes de commande et les ingénieurs en automatisation, qui sont bien informés sur les automates programmables, les considèrent comme un facteur important pour assurer une bonne compétitivité.

Les Automates Programmables Industriels (API) sont des machines électroniques programmables par un personnel automaticien, destinées à la commande et au contrôle industriel. Leur objectif principal est d'automatiser tous les mécanismes pour qu'ils puissent fonctionner de manière autonome. Ces appareils sont couramment utilisés dans des systèmes critiques tels que les avions, les ascenseurs et les trains.

De nos jours, la mise en place d'outils de supervision est indispensable pour implémenter des systèmes automatisés. La maîtrise d'un processus est indissociable de sa supervision pour aider les entreprises à améliorer leur productivité et leur qualité tout en réduisant les coûts. Cette supervision permet également aux entreprises de garantir la sécurité de leurs équipements et de leur personnel (les opérateurs), ainsi que de protéger l'environnement.

L'objectif de ce projet est d'étudier et de programmer une station didactique faisant partie d'un ensemble de stations simulant une mini-usine. Cette station se trouve dans le département d'électronique et de télécommunication de l'université 8 Mai 1945 de Guelma. L'importance de cette étude réside principalement dans le fait que la station comprend différents aspects industriels tels que la pneumatique (vérins et distributeurs) et une variété de capteurs photoélectriques.

Les étapes suivies pour la réalisation de ce projet sont les suivantes :

- Étude du fonctionnement de la station, en particulier des vérins et des distributeurs pneumatiques ainsi que des capteurs photoélectriques.
- Réalisation du Grafcet correspondant au fonctionnement souhaité.
- La configuration matérielle de l'automate avec le logiciel Step7.

- La programmation du Grafset en utilisant deux langages de programmation : le langage contact et le langage Graphe de fonctions séquentielles.
- Le test du programme avec le logiciel PLCSIM et son transfert à la station.
- Réalisation d'un système de supervision basé sur SCADA

Le présent mémoire est reparti en quatre chapitres décrivant les volets principaux confiés :

Dans le premier chapitre, nous présenterons l'automatisation industrielle et nous touchons les dispositifs d'entrée/sortie, nous donnons également un aperçu sur la pneumatique et l'hydraulique.

Le second chapitre sera dédié en sa globalité aux automates programmables industriels et aux leurs différents langages de programmation.

Le troisième chapitre donne une présentation générale des réseaux industriels et du système de supervision industriel (SCADA).

Le dernier chapitre décrit la station étudiée et son automatisation. Nous y présentons toutes les étapes à suivre pour l'automatisation et la supervision du système étudié.

On termine ce travail par une conclusion générale.

CHAPITRE I

Introduction au système automatisé

CHAPITRE I

Introduction au système automatisé

I.1 Introduction

Les êtres humains fabriquent des outils et des machines depuis des milliers d'années. À l'origine, la plupart des produits étaient fabriqués individuellement au besoin ; si un outil était nécessaire, il était fabriqué à la main et utilisé pour fabriquer plus d'outils. Au fil du temps, des techniques plus complexes ont été développées pour aider les gens à accomplir des tâches de fabrication et de production. La technologie du travail des métaux, les métiers à tisser, les moulins à meuler à eau et le développement des moteurs à vapeur et à essence ont tous contribué à une plus grande capacité de fabriquer divers produits, mais les choses étaient encore généralement faites une à la fois par des artisans qualifiés dans diverses techniques. Ce n'est qu'après la révolution industrielle et l'utilisation commune de l'énergie électrique et des mécanismes que la fabrication de produits à grande échelle est devenue monnaie courante.

L'automatisation est l'utilisation de commandes de programmation logique et d'équipement mécanisé pour remplacer la prise de décision et les activités manuelles de commande-réponse des êtres humains. Historiquement, la mécanisation, comme l'utilisation d'un mécanisme de synchronisation pour actionner un levier ou un cliquet et une patte, aidait les humains à exécuter les exigences physiques d'une tâche. L'automatisation, cependant, nécessite une mécanisation. Il s'agit de réduire considérablement les besoins sensoriels et mentaux tout en optimisant la productivité [1].

I.2 Principe de l'automatisation

La tâche du contrôle des processus industriels a considérablement changé au cours des dernières années à partir d'un fonctionnement manuel complet, passant par le contrôle analogique et l'automatisation de bas niveau, puis récemment, jusqu'aux méthodes de contrôle et d'automatisation entièrement informatisées.

Aujourd'hui, un système de contrôle et d'automatisation industriel est un terme général qui couvre de nombreux types d'appareils numériques tels que les ordinateurs personnels industriels (I-PC), les automates programmables industriels (API), les contrôleurs d'automatisation programmables (PAC), API intégrés et d'autres contrôleurs numériques spécifiques. De plus, les configurations de systèmes d'automatisation comprennent des plates-formes logicielles et matérielles telles que des sous-systèmes de contrôle et d'acquisition de

données (SCADA), des sous-systèmes de contrôle distribués (DCS), et des sous-systèmes de communication industrielle.

Après l'introduction des premiers ordinateurs personnels et automates puissants, les ingénieurs en automatisation se sont divisés en deux groupes. Le premier groupe préfère les PC avec un matériel d'entrée/sortie (E/S) approprié pour des fonctions d'automatisation industrielle appropriées ; tandis que le second groupe considère les PC comme des dispositifs informatiques inadaptés à l'industrie, tout en favorisant les CAP. Cependant, les deux catégories ont leurs propres caractéristiques, avantages et inconvénients.

La concurrence naturelle entre les fournisseurs d'automates programmables et les groupes d'ingénierie industrielle, ainsi que les similitudes générales sur le marché des contrôleurs industriels, expliquent pourquoi divers fournisseurs trouvent des moyens de repousser les limites de ces deux technologies matérielles et d'ajouter des fonctionnalités avancées, l'une est le "PC industriel". Les PC industriels se sont considérablement développés et améliorés ces dernières années, comblant le fossé entre les PC et les automates, mais cette catégorie n'a toujours pas remplacé les automates et n'est pas largement acceptée et installée.

Un PAC est généralement un contrôleur industriel multifonction qui peut simultanément surveiller et contrôler les signaux d'E/S numériques, analogiques et série provenant de plusieurs sources sur une seule plate-forme tout en prenant en charge plusieurs protocoles de communication.

La figure I.1 présente un aperçu des composants informatiques fondamentaux disponibles pour la mise en œuvre de systèmes d'automatisation et de contrôle industriels.

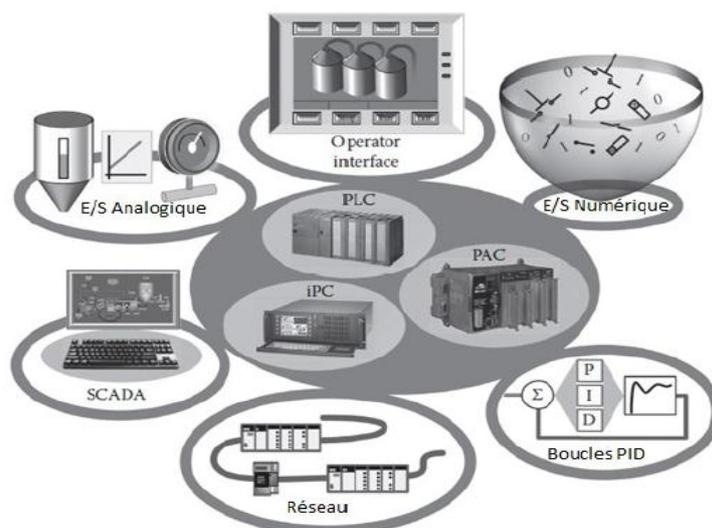


Figure I.1: Les CAP, les CISP et les CAP effectuent le contrôle, la communication et d'autres tâches.

Le concept SCADA a été introduit en raison du besoin de collecter des données et de surveiller les sous-systèmes de contrôle en temps réel dans les grandes usines de traitement industriel. Quelle que soit son objectif initial, le terme SCADA représente aujourd'hui un système matériel et logiciel combiné, comprenant des dispositifs de terrain distants, des réseaux, des équipements de stations centrales et des plates-formes logicielles.

Étant donné que les systèmes SCADA sont des systèmes de contrôle distribués (DCS), ces systèmes se composent d'un réseau puissant et de contrôleurs de processus avancés, souvent avec des processeurs très puissants, permettant simultanément plusieurs contrôles en boucle fermée.

Un système SCADA est un outil qui aide à garder un œil sur ce qui se passe à différents endroits en même temps. Il utilise différentes façons de palier à tous ces endroits pour obtenir des informations [2].

I.3 Les dispositifs d'entrée/sortie

I.3.1 Les dispositifs d'entrée

Le terme capteur fait référence à un périphérique d'entrée qui fournit une sortie utilisable en réponse à une entrée physique donnée. Par exemple, un thermocouple est un capteur qui convertit une différence de température en une sortie électrique. Le terme transducteur fait généralement référence à un dispositif qui convertit un signal d'une forme en une autre forme physique.

Les capteurs numériques produisant des sorties discrètes (c'est-à-dire à commutation) peuvent être facilement connectés aux ports d'entrée de l'API. Les capteurs analogiques produisent une sortie proportionnelle à la grandeur mesurée. Ces signaux analogiques doivent être convertis en signaux numériques avant de pouvoir être envoyés aux ports d'entrée de l'automate.

Voici quelques termes les plus utilisés pour définir les performances d'un capteur :

- La précision correspond à l'erreur maximale de la valeur indiquée du système ou de l'élément de mesure
- La plage de la variable système correspond à la limite de variation de l'entrée
- Le temps de réponse est le temps écoulé après une augmentation soudaine de l'entrée d'un système ou d'un composant de zéro à une valeur constante, jusqu'à ce que le système ou le composant produise un point de sortie correspondant à un pourcentage donné, par exemple 95 %

- La sensibilité fait référence au changement de la sortie du système ou du composant lorsque la quantité mesurée change d'une quantité donnée
- La stabilité d'un système est liée à sa capacité à produire la même sortie lorsqu'il est utilisé pour mesurer une entrée constante sur une période de temps.
- La fiabilité d'un système de mesure ou d'un élément est définie comme la probabilité qu'il fonctionnera à un niveau de performance convenu pendant une période de temps spécifiée dans des conditions environnementales spécifiées

Des exemple de quelques dispositifs d'entrée les plus utilises avec les API, sont les suivants:

- Interrupteurs mécaniques
- Interrupteurs de proximité
- Capteurs et interrupteurs photoélectriques
- Encodeurs
- Capteurs de température
- Capteurs de position/déplacement
- Capteurs de pression
- Détecteurs de niveau de liquide
- Mesure du débit d'un fluide [3]

I.3.1 Les dispositifs de sortie

Les ports de sortie de l'automate sont équipés de relais ou de photo-coupleurs avec transistor ou triac, selon l'appareil à commander. Généralement, les signaux numériques des canaux de sortie de l'API sont utilisés pour contrôler les actionneurs et donc le processus. Le terme actionneur fait référence à un organe qui convertit un signal électrique en une action plus puissante, qui conduit ensuite au contrôle du processus. Des exemple de capteurs sont :

- Relais
- Vannes de commande directionnelle
- Moteurs
- Vérins [3]

I.4 Pneumatique et hydraulique

Conventionnellement, l'utilisation de liquides ou de gaz sous pression pour actionner les actionneurs est connue sous le nom de puissance des fluides.

L'hydraulique définit l'énergie produite par les liquides, comme l'huile minérale ou l'eau, et la pneumatique définissent l'énergie produite à l'aide de gaz, comme l'air ou l'azote. La puissance des fluides peut actionner les vérins de façon linéaire, les moteurs en mouvement de rotation ou les actionneurs en mouvement de rotation de moins de 360°.

Les avantages de l'hydraulique et pneumatique varient. Alors que Les systèmes pneumatiques sont généralement moins coûteux à mettre en œuvre et à utiliser, ils sont moins précis et sont généralement limités à des services publics en raison de la rapidité de dilatation élevée une fois le gaz décompressé. D'autre part, si l'hydraulique peut être plus précise, elle est plus coûteuse à construire et à entretenir en raison de la nécessité de disposer d'un moyen de drainage et de récupération des liquides et d'un équipement plus grand [1].

I.4.1 Pneumatique

L'air comprimé ou pneumatique est utilisé pour déplacer les vérins ou les actionneurs et est même utilisé dans une logique de commutation. L'air est généralement filtré, séché et régulé à une pression utilisable et distribué d'un compresseur aux divers dispositifs et actionneurs où il est nécessaire. Divers dispositifs et raccords à déconnexion rapide ont été conçus en dimensions métriques et standard et sont largement utilisés dans l'industrie. Les générateur d'air comprimé sont utilisés dans la plupart des installations industrielles et sont facilement disponibles pour un large gamme d'utilisations.

La pression d'air est habituellement appliquée à un filtre régulateur supplémentaire avec un manomètre pour régler la pression sur une machine ou un système. Un lubrifiant supplémentaire est parfois utilisé pour appliquer une petite quantité d'huile pour lubrifier l'intérieur des bouteilles d'air. Les régulateurs de filtre et les lubrifiants sont relativement peu coûteux et utilisés régulièrement au point d'entrée d'un système de commande pneumatique. Des régulateurs de pression, des régulateurs de débit et des vannes supplémentaires sont utilisés pour contrôler les actionneurs pour l'effet désiré. Les vannes de commande sont courantes dans les variétés 120VAC et 24VDC. La figure I.2 montre comment ces composants sont combinés dans un circuit pneumatique typique.

La pression d'air dans les installations industrielles peut atteindre plusieurs centaines de psi, mais les vérins pneumatiques fonctionnent habituellement à 60 à 80 psi. Les régulateurs de pression peuvent être utilisés pour réduire la pression dans le système, au besoin.

Les commandes de débit peuvent être placées à l'une ou l'autre des extrémités d'un vérin pneumatique pour régler la vitesse de mouvement. Le terme « mater in » fait référence à la restriction du débit d'entrée d'air vers un vérin, tandis que le terme « mater out » fait référence à la restriction du débit sortant de l'autre extrémité. Une fois qu'un vérin a été mis en place sans débit d'air de l'un ou l'autre des orifices, la pression a tendance à s'évacuer lentement. Cela peut provoquer une « fente » soudaine du vérin lorsque la pression est réintroduite. Pour contrôler le mouvement dans une direction sélectionnée, il est généralement préférable « mater out » plutôt que « mater in » pour rendre le mouvement plus cohérent. Dans la plupart des vérins pneumatiques, des contrôles de débit sont utilisés sur les deux orifices et ajustés pour le mouvement souhaité dans les deux sens.

La plupart des vannes sont de type marche/arrêt ou discrètes, fournissant un flux d'air (et donc un mouvement) à un vérin jusqu'à ce qu'il soit éteint [1].

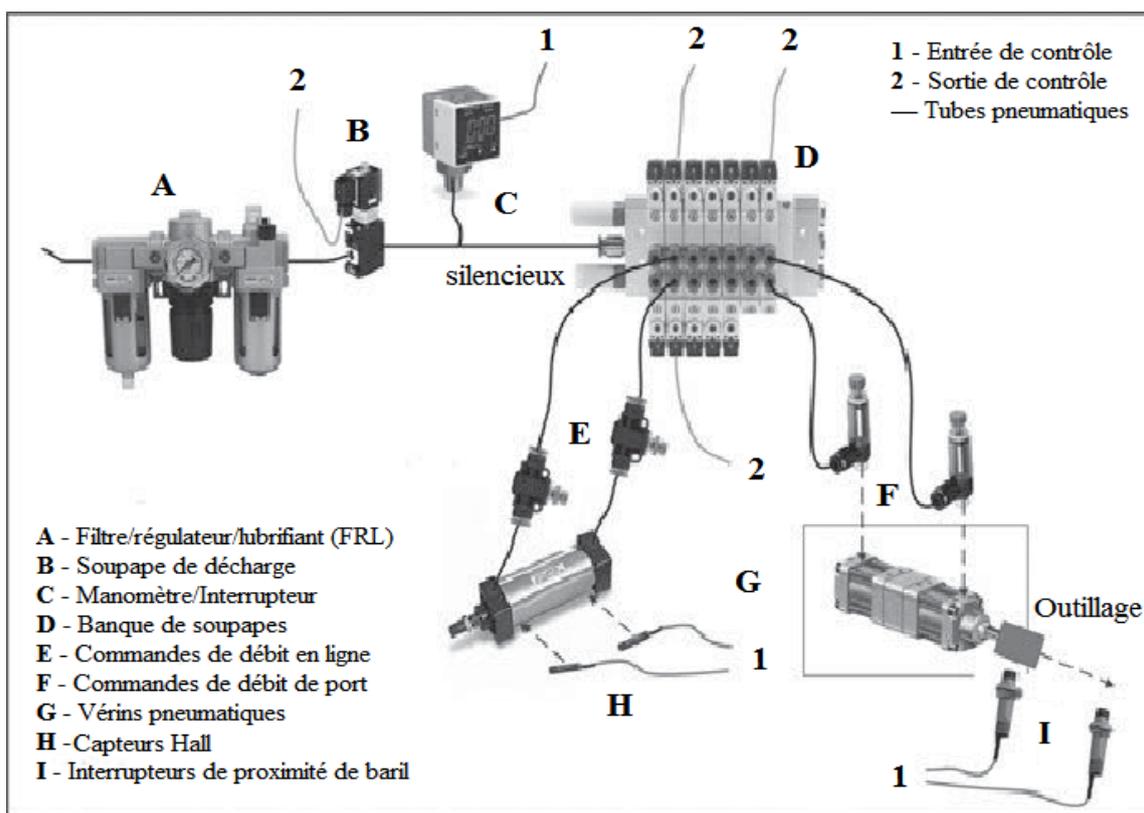


Figure I.2: Système pneumatique.

Des vannes d'air proportionnelles sont également utilisées pour contrôler rapidement le mouvement Air pulsé ou pression ou débit variable selon la position de l'actionneur. Cela permet une forme simple de contrôle du mouvement pour la position et la vitesse.

La force qu'un vérin peut produire est basée sur la taille du piston du vérin et la pression d'air disponible pour le vérin. L'équation simple de la force qu'un cylindre peut produire est

$$\text{Force} = \text{pression} \times \text{surface} \quad (\text{I.1})$$

La pression dans cette équation est la pression d'air disponible (en force/surface unités) et Surface est la taille du piston disponible pour déplacer le piston dans le cylindre — typiquement $\text{Surface} = \text{PI} \times \text{rayon du piston}^2 - \text{diamètre de la tige}^2$ (si la tige réduit la surface disponible dans la direction dans laquelle la force est appliquée). Ces données sont disponibles dans les fiches techniques ou catalogues du fabricant.

I.4 2. Hydraulique

Les systèmes hydrauliques, contrairement aux systèmes pneumatiques ou des systèmes d'air, ont généralement une pompe située à chaque machine et sont donc plus autonomes. La pression hydraulique est utilisée lorsque plus de force est requise qu'une application pneumatique. Comme les systèmes pneumatiques, il existe une grande variété de vérins hydrauliques et actionneurs disponibles auprès des fabricants pour chaque application. Les systèmes hydrauliques sont beaucoup plus coûteux à mettre en œuvre que les systèmes pneumatique puisque, ils fonctionnent habituellement à une pression plus élevée et plus de précautions doivent être prises pour s'assurer que les fluides ne fuient pas du système. Divers autres composants, tels que les refroidisseurs d'huile et les intensificateurs, peuvent également ajouter au coût des systèmes hydrauliques.

Les systèmes hydrauliques ne fonctionnent pas aussi rapidement que les systèmes pneumatiques, mais, en raison de la plus faible compressibilité de l'huile, ils peuvent être contrôlés avec plus de précision. Les pressions et la force obtenues à partir des systèmes hydrauliques sont généralement beaucoup plus élevées que celles des systèmes pneumatiques et sont donc souvent utilisés dans les applications de formation des métaux [1].

I.4 3. Comparaison entre Pneumatique et hydraulique

Avantages de la pneumatique :

- Propreté
- Simplicité de conception et de contrôle
 - Les machines sont facilement conçues à l'aide de vérins standard et d'autres composants. Le contrôle est facile car il consiste à commande simple de type marche/arrêt.

- Fiabilité
 - Les systèmes pneumatiques ont généralement une longue durée de vie et nécessitent très peu d'entretien.
 - Parce que le gaz est compressible, l'équipement est moins susceptible d'être endommagé par un choc. Le gaz en pneumatique absorbe une force excessive.
- Stockage
 - Le gaz comprimé peut être stocké, ce qui permet l'utilisation de machines lorsque l'électricité est coupée.
- Sécurité
 - Très peu de risques d'incendie (par rapport à l'huile hydraulique).
 - Les machines peuvent être conçues pour être protégées contre les surcharges.
- Coût généralement moins élevé
 - L'air comprimé est facilement disponible dans la plupart des de fabrication.

Avantages de l'hydraulique :

- Le fluide n'absorbe pas l'énergie fournie.
- Les systèmes hydrauliques fonctionnent à des pressions beaucoup plus élevées (généralement 3000 psi) que les systèmes pneumatiques, ce qui permet d'obtenir une force plus importante pour une taille d'alésage donnée.
- Le fluide hydraulique est fondamentalement incompressible, ce qui réduit au minimum l'effet de ressort. Lorsque l'écoulement du fluide hydraulique est interrompu, le moindre mouvement de la charge relâche la pression sur la charge ; il n'est pas nécessaire de "purger" l'air sous pression pour relâcher la pression sur la charge.
- Peut arrêter le vérin hydraulique à mi-course, ce qu'un vérin pneumatique ne peut généralement pas faire [1].

I.5 Conclusion

L'utilisation de l'automatisation présente plusieurs avantages, tels que le remplacement des opérateurs humains qui effectuent des tâches physiquement exigeantes ou répétitives et ceux qui travaillent dans des environnements dangereux. L'automatisation peut également faciliter des tâches qui dépassent les capacités humaines, telles que la manipulation d'objets lourds ou petits ou la fabrication de produits rapidement ou lentement. La production automatisée est généralement plus rapide et moins coûteuse par produit que les opérations manuelles. Les systèmes d'automatisation peuvent également intégrer des contrôles et des vérifications de la

qualité pour réduire le nombre de pièces hors tolérance et permettre un contrôle statistique des processus. L'automatisation a joué un rôle important dans l'amélioration des économies des entreprises et des sociétés. Par exemple, l'automatisation de la production d'armes, d'automobiles, de textiles et d'autres biens destinés à l'exportation a considérablement amélioré le produit national brut et le niveau de vie de l'Allemagne et du Japon au cours du XXe siècle.

Bien que l'automatisation offre plusieurs avantages, elle présente également certains inconvénients. L'un des principaux inconvénients est que la technologie actuelle ne peut pas automatiser toutes les tâches souhaitées, telles que la production ou l'assemblage de produits avec des tailles de composants non uniformes ou des tâches nécessitant des compétences manuelles. De plus, certaines tâches peuvent coûter plus cher à automatiser qu'à effectuer manuellement. L'automatisation est généralement mieux adaptée aux processus reproductibles, cohérents et à volume élevé. Le coût de la recherche et du développement d'un processus automatisé est également difficile à prévoir à l'avance, et il peut être découvert après l'automatisation d'un processus qu'il n'y a aucun avantage économique à le faire. Les coûts initiaux de l'automatisation sont également relativement élevés, nécessitant un investissement important par rapport au coût unitaire du produit. Même les machines dont le coût de développement a déjà été récupéré sont chères en termes de matériel et de main-d'œuvre. Les lignes de production personnalisées peuvent avoir des coûts exagérés, et un service de maintenance qualifié est souvent nécessaire pour maintenir le système d'automatisation en bon état de fonctionnement, car le non-respect de cette consigne entraînera finalement une perte de production et/ou la production de pièces défectueuses. Malgré ces inconvénients, les avantages de l'automatisation l'emportent sur les inconvénients, comme en témoigne le niveau de vie plus élevé dont bénéficient les pays qui ont adopté l'automatisation. Bien que des préoccupations concernant l'impact de l'automatisation sur les emplois soient souvent soulevées, il est clair que la productivité augmente avec l'application appropriée des techniques d'automatisation [1].

CHAPITRE II

L'automate S7-300 de Siemens

CHAPITRE II

L'automate S7-300 de Siemens

II.1 Introduction aux Automates Programmables Industriels

Un automate programmable industriel (API) est une forme spéciale de contrôleur à microprocesseur qui utilise une mémoire programmable pour stocker des instructions et exécuter diverses fonctions, qu'il s'agisse de logique, de séquençage, de synchronisation, de comptage ou d'arithmétique, pour contrôler les machines et les processus. Il est destiné à être utilisé par des ingénieurs qui peuvent avoir une connaissance limitée des ordinateurs et des langages de programmation. La création et la modification de programmes API ne se limitent pas aux informaticiens. Les concepteurs de API l'ont préprogrammé afin que le programme de contrôle puisse être saisi dans un langage simple et intuitif [3].

II.2 Avantages des API et de la logique programmée :

Un avantage majeur des Automates Programmables Industriels (API) est qu'un même automate de base peut être utilisé avec une grande variété de systèmes de commande. Pour modifier le système de commande et les règles appliquées, il suffit à l'opérateur de saisir une suite d'instructions différente. Cela permet d'obtenir un système flexible et économique, qui peut être utilisé avec une grande variété de systèmes de commande, quelles que soient leur nature et leurs complexités. En comparaison avec les systèmes à relais, les avantages des API sont :

- Les modifications sont facilitées car elles sont implémentées de manière logicielle plutôt que par des solutions matérielles plus complexes.
- L'ajout de nouveaux modules peut être facilement effectué sur les API, alors que des changements matériels sont sinon requis.
- Ils sont plus robustes et fiables en raison de la réduction du nombre de composants mécaniques.
- Les API sont plus compactes.
- Nécessitent moins d'entretien.
- Sont plus rapides, et donc plus performants.

Les API sont comme des ordinateurs. Cependant, les ordinateurs sont optimisés pour les tâches de calcul et d'affichage, tandis que les API sont optimisées pour les tâches de contrôle et les environnements industriels. Voici en quoi une API diffère d'un ordinateur :

- Les API sont conçues pour être robustes et capables de supporter les vibrations, les températures extrêmes, l'humidité et le bruit dans les environnements industriels. En revanche, les ordinateurs personnels ne sont pas conçus pour fonctionner dans des environnements hostiles.
- Les interfaces d'entrée et de sortie sont intégrées à l'API et peuvent être facilement étendues pour prendre en charge un plus grand nombre d'entrées-sorties avec les modules d'extension.
- La programmation des API est simple et leur langage de programmation est facile à apprendre, car il est axé sur les opérations logiques et de commutation. Cela les rend plus conviviaux pour les ingénieurs et les opérateurs.
- Les capacités de stockage à long terme et d'analyse de données des API sont moins développées que celles des ordinateurs personnels
- Les API sont plus fiables et moins sujets aux dysfonctionnements que les ordinateurs personnels dans les environnements industriels [3]

II.3 Structure des API

En général, la structure d'un API repose sur plusieurs éléments de base, comme l'unité de traitement, la mémoire, l'unité d'alimentation, les interfaces d'entrées-sorties, l'interface de communication et le périphérique de programmation (Voir Figure II.1) :

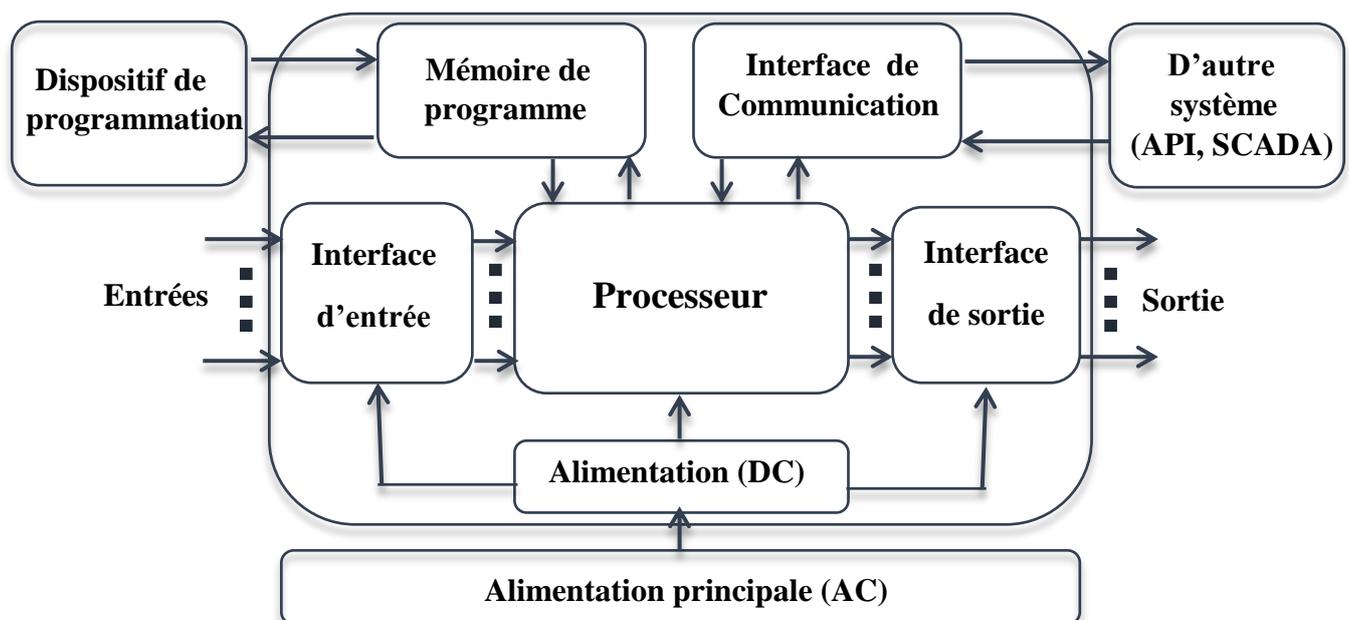


Figure II.1 : Structure générale d'un API

Le processeur, également appelé unité centrale de traitement (CPU, Central Processing Unit), est équipé d'un microprocesseur. Ce dernier interprète les signaux d'entrée et exécute les actions de commande conformément au programme stocké en mémoire, communiquant les décisions sous forme de signaux d'action aux sorties correspondantes.

- L'unité d'alimentation est un composant essentiel car elle convertit une tension alternative en une basse tension continue de 5 V nécessaire au fonctionnement du processeur et des modules d'entrées-sorties.
- Le périphérique de programmation est utilisé pour entrer le programme dans la mémoire du processeur. Le programme est développé sur le périphérique, puis transféré dans la mémoire de l'API.
- La mémoire contient le programme qui définit les actions de commande effectuées par le microprocesseur. Elle contient également les données provenant des entrées pour leur traitement ainsi que les données des sorties.
- Les interfaces d'entrée-sortie permettent au processeur de recevoir et d'envoyer des informations aux dispositifs externes.
- L'interface de communication est utilisée pour recevoir et transmettre des données sur des réseaux de communication qui relient l'API à d'autres API distantes. Elle est impliquée dans des opérations telles que la vérification d'un périphérique, l'acquisition de données, la synchronisation entre les applications et la gestion de la connexion [3].

II.4 Langages de programmation des API (norme CEI 61131-3)

Au fil des ans, plusieurs normes de programmation pour les contrôleurs programmables ont été proposées. Des suggestions ont été émises par différents comités nationaux et internationaux pour développer une interface commune. En 1979, un groupe de travail international d'experts en PLC a été formé pour proposer une première ébauche d'un standard PLC complet.

La première norme sur les langages de programmation a été publiée en mars 1993 sous la désignation IEC 61131-3. D'autres ajouts ont été publiés en 2002 et un troisième, et provisoirement dernier, est paru en 2013. La plupart des fabricants de PLC suivent actuellement cette norme à divers degrés. Elle comprend plusieurs langages de programmation :

- ST: Structured Text (Texte structuré)

- FBD : Function Block Diagram (Schéma fonctionnel des fonctions)
- LD : Ladder Diagram (Diagramme d'échelle)
- IL : Instruction List (Liste des instructions)
- SFC : Sequential Function Chart (Organigramme des fonctions séquentielles).

II.4.1 Le langage à contacts (LD, Ladder Diagram)

Malgré le fait que les langages ST et SFC soient plus efficaces dans la plupart des contextes, le diagramme d'échelle (LD) est encore largement utilisé par de nombreux programmeurs PLC. Ceci est dû au fait que LD est simple à comprendre et est basé sur les schémas électriques traditionnels, tels que les schémas de relais. La popularité de ce langage bien établi persiste car les nouveaux ingénieurs et techniciens doivent encore apprendre à le lire, à le comprendre et à l'appliquer. De plus, certaines personnes appliquent le concept de diagramme de relais au code de programme écrit en LD

Les LD consistent essentiellement en un ensemble d'instructions qui exécutent les types de fonctions de contrôle les plus élémentaires : logique, contrôle du temps et comptage, ainsi que des opérations mathématiques simples. La plupart des fabricants d'automates permettent aujourd'hui d'implémenter des fonctionnalités supplémentaires de haut niveau en LD, souvent Intégrées à d'autres langages tels que FBD et ST [4].

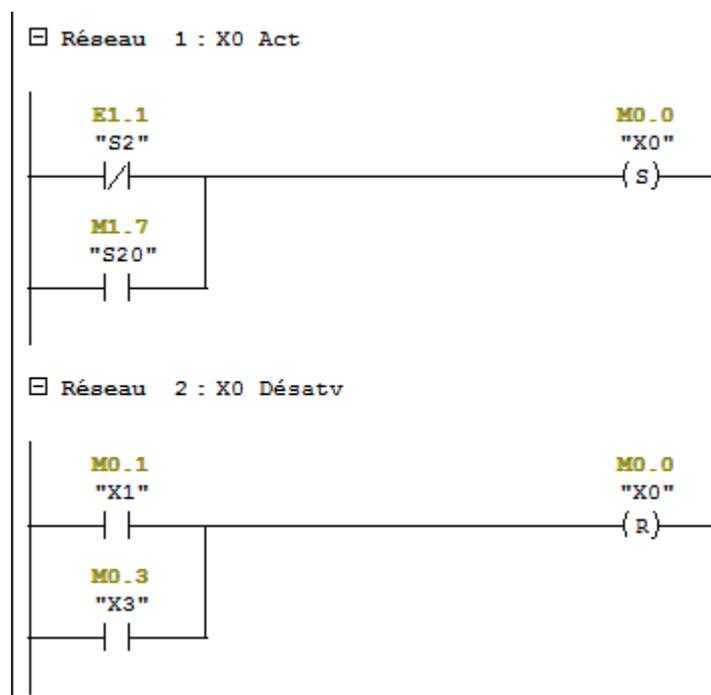


Figure II.2 : Exemple de programme en langage contact

II.4.2 Les listes d'instructions (IL, Instruction List)

IL est un langage d'assemblage de bas niveau. Bien qu'il y ait des inconvénients associés à l'utilisation d'un langage de bas niveau comme IL, l'avantage est qu'il ne nécessite pas beaucoup de puissance informatique. La raison pour laquelle ce langage continue d'être utilisé est qu'il est l'un des plus anciens langages de programmation de la norme, en plus de LD. De nombreux anciens PLC ne peuvent être programmés qu'avec IL/LD. Ainsi, il peut y avoir des situations où IL est utilisé, par exemple, lorsqu'un code de programme écrit en IL est extrait d'un ancien PLC pour être modifié ou analysé.

```
▣ Réseau 9 : X4 Actv
  U      "X3"          M0.3
  U(
  O      "S1"          E1.0
  O      "S10"         M1.6
  O      "S3"          E1.2
  O      "S30"         M2.0
  )
  S      "X4"          M0.4

▣ Réseau 10 : X4 Désactiv
  O      "X5"          M0.5
  ON     "S2"          E1.1
  O      "S20"         M1.7
  R      "X4"          M0.4
```

Figure II.3 : Exemple de programme en langage liste

II.4.3 Les diagrammes de schémas fonctionnels (FBD, Function Block Diagram).

FBD est un langage graphique, décrit de manière très simplifiée, basé sur l'enchaînement de fonctions et de blocs fonctionnels. Le langage comprend, entre autres, l'utilisation de fonction logique standard telle que (AND, OR, NOT, etc.) et de fonctions de blocs fonctionnels (FB) telles que des temporisateurs et des compteurs. Il peut être commode d'écrire des algorithmes logiques (fonctions booléennes) et des fonctions de contrôle (telles que des structures de régulation, etc.) dans ce langage [4].

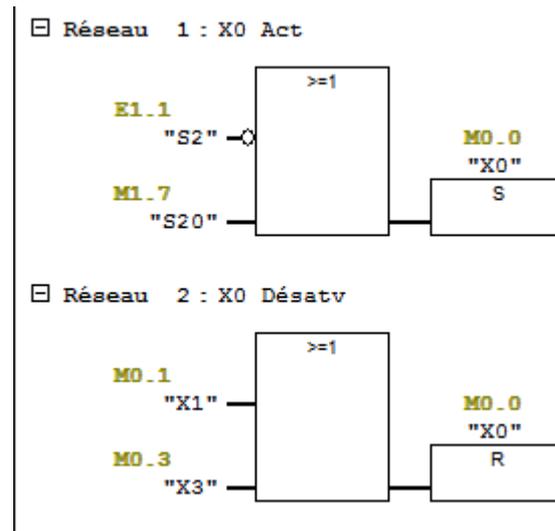


Figure II.4 : Exemple de programme en diagrammes de schémas fonctionnels

II.4.4 Le texte structuré (ST, Structured Text)

ST tire son nom du fait qu'il est un langage basé sur le texte. C'est un langage de haut niveau qui permet d'effectuer de nombreuses opérations et instructions avec une seule ligne de commande. ST est similaire à Pascal ou C et a été spécialement conçu pour programmer des fonctions arithmétiques complexes et manipuler des tables.

II.4.5 Les graphes de fonction séquentielle (SLC, Sequential Function Charts)

SFC est l'outil graphique idéal pour programmer des commandes séquentielles et mettre en œuvre des algorithmes de commande basés sur l'état. En fait, SFC n'est pas un langage de programmation au sens traditionnel, mais une manière graphique de structurer le code de programmation. Il s'adapte également avec brio à la norme selon laquelle tous les autres langages peuvent être utilisés en combinaison avec SFC, et il est nécessaire qu'au moins un de ces langages implémente toutes les conversions et opérations nécessaires [4].

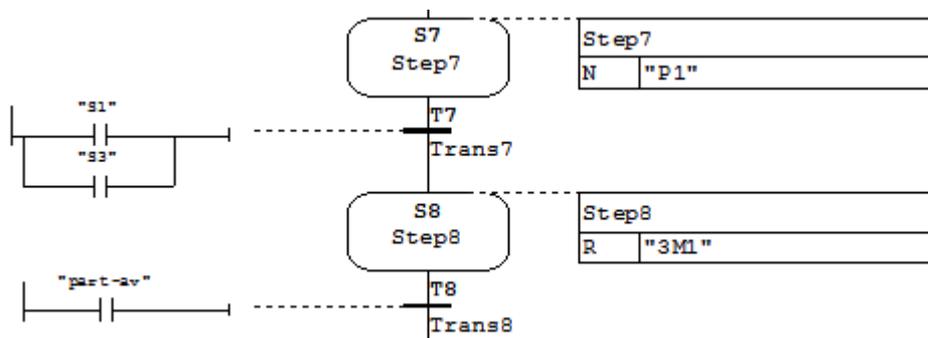


Figure II.5 : Exemple de programme en langage séquentiel

II. 5 Les API de Siemens

II.5.1 Aperçu général

Le système SIMATIC S7-300 est un mini-automate modulaire pour les plages de performances faibles et moyennes. Différentes versions du contrôleur peuvent adapter les performances à l'application correspondante. Sur demande, l'automate programmable peut être étendu avec des modules d'entrée/sortie pour les signaux numériques et analogiques jusqu'à quatre racks avec huit modules chacun.

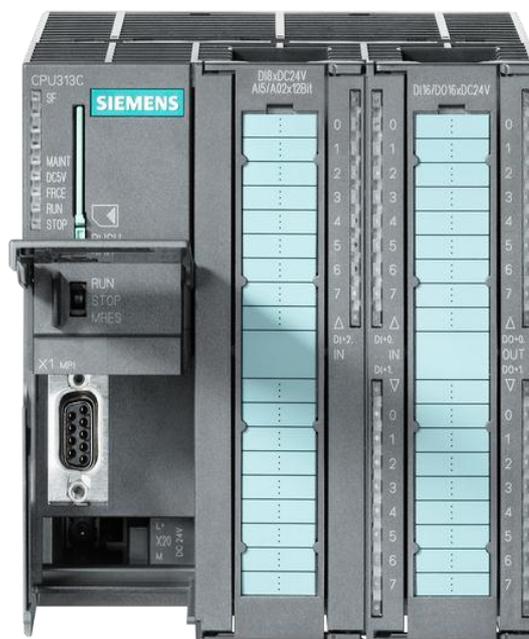


Figure II.6 : API siemens S7-300

II.5.2 CPU S7-300

a) Versions CPU

Il existe plusieurs versions de CPU pour le S7-300, adaptées à différentes applications. La portée des fonctions de contrôle (opérandes, types d'étiquette, types de données, opérations logiques binaires, arithmétique à virgule fixe et à virgule flottante, etc.) varie selon la version de CPU choisie. Les CPU diffèrent également par leur taille de mémoire, leur gamme d'opérandes et leur vitesse de traitement. L'utilisation d'une carte mémoire micro (MMC) est requise pour le fonctionnement [5].

b) Commutateur et leds d'affichage

Les LED d'état (Tableau II.1) et le commutateur de mode se trouvent à l'avant de la CPU (figure II.7). Le commutateur de mode est un commutateur à bascule avec les positions RUN, STOP et MRES. En position RUN, le programme utilisateur s'exécute normalement et l'appareil de programmation a un accès illimité à la CPU.

En position STOP, le programme utilisateur n'est pas exécuté, mais le CPU reste apte à communiquer. Par exemple, un nouveau programme utilisateur peut être téléchargé à l'aide de l'appareil de programmation ou le tampon de diagnostic peut être lu avec le CPU en position STOP.

En mode MRES (réinitialisation principale), les paramètres du processeur sont remis à zéro. Il s'agit d'un bouton de réinitialisation. Pour réinitialiser la mémoire du CPU, il est possible d'utiliser une séquence d'entrée spéciale ou de le réinitialiser à son état d'origine [5].

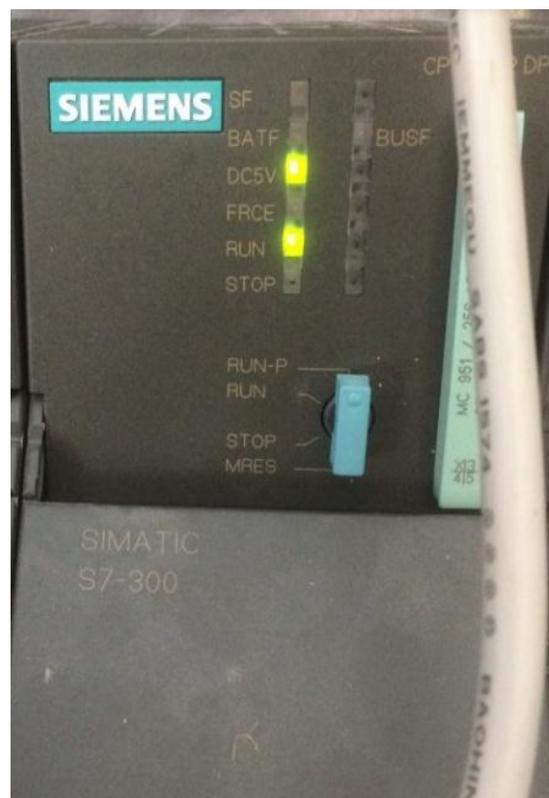


Figure II.7: Eléments d'affichage d'un CPU

LED	COULEUR	L'ETAT DE LED
SF	ROUGE	Cette LED s'active en cas d'erreur logicielle ou matérielle.
BF	ROUGE	Cette LED s'illumine lorsque le bus est en panne. Chaque interface de bus dispose d'une LED de défaut, qui est numérotée consécutivement si plusieurs bus d'interface sont présents (BF1, BF2, etc.).
MAINT	JAUNE	Cette LED est activée en cas de demande de maintenance.
DC5V	VERT	Cette LED s'active si l'alimentation est présente au CPU.
FRACE	JAUNE	Cette LED s'illumine lorsque le forçage est actif et clignote à une fréquence de 2 Hz pendant le test de clignotement du nœud.
RUN	VERT	Cette LED s'active en mode RUN, clignote à 0,5 Hz en mode HALT et clignote à 2 Hz en mode STARTUP.

Tableau II.1 : Indication l'état de LED

II.5.3 Zones mémoire

La mémoire système contient les opérandes que vous adressez depuis votre programme. La zone de composition des opérandes (opérandes), qui contient un certain nombre d'opérandes par CPU (Tableau II.2). Par exemple, les opérandes peuvent être des entrées pour interroger l'état du signal des boutons poussoirs et des interrupteurs de fin de course, et des sorties pour contrôler les contacteurs et les lumières [5].

Les zones d'opérandes	Définition
Entrées (I)	C'est une image (image de traitement) représentant les modules d'entrée numérique.
Sorties (Q.A)	C'est est une image (image de traitement) représentant les modules de sortie numérique.
Mémoires de bits (M)	Ce sont des mémoires d'informations et peuvent être traitées dans le programme utilisateur à partir de n'importe quelle position.
Fonctions de la minuterie SIMATIC (T)	Ce sont des minuteurs utilisés pour la mise en œuvre des temps d'attente et de contrôle.
Fonctions de comptage SIMATIC (C)	Il s'agit des compteurs logiciels qui ont la capacité de compter dans les deux sens, à la fois en augmentant et en diminuant.
Données locales temporaires (L)	La mémoire système inclut des tampons pour les tâches de communication ainsi que pour les alarmes système (tampon de diagnostic).

Tableau II.2 : Les diffèrent zones d'opérandes

II.5.4 Les Modules entrée/sortie (Signal modules)

a) Module d'entrée numérique

Les modules d'entrée numérique sont employés par l'unité centrale pour enregistrer les états de fonctionnement de la machine ou des dispositifs commandés. Ces modules servent de conditionneurs de signaux pour les signaux d'entrée de processus binaires. Les signaux de processus présentant un niveau de tension continue allant de 24 V à 120 V ou une tension alternative allant de 120 V à 230 V sont transformés en signaux possédant un niveau interne[5].



Figure II.8 : Exemples de modules d'entrée numérique

b) Module de sortie numérique

Les modules de sortie numérique sont employés par l'unité centrale pour réguler la machine ou l'installation connectée. Ces modules servent de conditionneurs de signaux pour les signaux de sortie de processus binaires. Les signaux internes sont renforcés et délivrés en plages de courant et de tension suivantes (valeurs nominales) :

- Des amplificateurs électroniques pour une tension alternative de 24 V à 120 V AC et un courant de 0,3 A ou 2 A.
- Des amplificateurs électroniques pour une tension alternative de 120 V à 230 V AC et un courant de 1 A ou 2 A.
- Des contacts de relais pour une tension continue de 24 V ou une tension alternative de 230 V et un courant pouvant atteindre 5 A.

c) Module d'entrée/sortie numériques

Les modules d'entrée/sortie numériques sont employés par l'unité centrale pour enregistrer les états de fonctionnement de la machine ou de l'installation connectée, ainsi que pour commander la machine ou l'installation. Ils servent de conditionneurs de signaux pour les signaux d'entrée et de sortie de processus binaires [5].

d) Modules analogiques

L'unité centrale peut utiliser des modules d'entrée analogiques pour traiter les variables analogiques mesurées après leur conversion en valeurs numériques par les modules. Ces

modules servent de conditionneurs de signaux pour les signaux d'entrée des processus analogiques. Tous les modules d'entrée analogiques signalent l'absence de tension de charge ou un défaut du module grâce à la LED SF.

L'unité centrale peut également utiliser des modules de sortie analogiques pour fournir en permanence aux actionneurs des points de consigne analogiques. Ces modules sont des conditionneurs de signaux pour les signaux de sortie analogiques du processus. Tous les modules de sortie analogiques indiquent une tension de charge manquante ou un défaut sur le module par le biais de la LED SF.

De plus, l'unité centrale peut utiliser des modules d'entrée/sortie analogiques pour traiter des variables analogiques et fournir en permanence aux actionneurs des points de consigne analogiques. Ces modules sont des conditionneurs de signaux pour les signaux de processus analogiques.

Chaque canal analogique correspond à une valeur de 16 bits, soit deux octets. Les modules de sorties et d'entrées analogiques sont disponibles avec 2, 4 ou 8 canaux, correspondant à une plage d'adresses de 4, 8 ou 16 octets [5].

II.5.5 Les Modules de communication

Les modules de communication (également appelés processeurs de communication, CP) permettent à l'unité centrale de déléguer les tâches de communication. Ils assurent la connexion physique avec un partenaire de communication, s'occupent de l'établissement de la connexion et du transport des données, et fournissent les services de communication nécessaires à l'unité centrale et au programme utilisateur.

Les modules de communication suivants sont proposés :

- CP 340 : module de communication pour couplage point à point
- CP 341 : module de communication pour couplage point à point
- CP 343-2 : AS-Interface master, CP 343-2P : AS-Interface master avec prise en charge de la configuration
- CP 342-5 : connexion à PROFIBUS DP avec interface électrique, CP 342-5FO : connexion à PROFIBUS DP avec interface optique
- CP 343-5 : connexion au FMS PROFIBUS
- CP 343-1 : connexion Lean à Ethernet industriel

- CP 343-1 : connexion à Ethernet industriel
- CP 343-1 : connexion avancée à Ethernet industriel [5].

II.5.6 Interfaces de bus

Selon la conception de l'unité centrale, les interfaces de bus suivantes peuvent être présentes :

a) DP (Decentralized Peripherals)

L'interface DP permet de connecter l'unité centrale au système de bus PROFIBUS DP. L'unité centrale peut servir de maître DP ou d'esclave DP. Si l'unité centrale dispose de deux interfaces DP, il n'est pas possible que les deux interfaces soient simultanément en mode DP esclave. La station de communication de base S7 peut être utilisée pour configurer le transfert de données via une interface DP. En outre, un appareil de programmation ou une unité de commande et d'affichage de l'opérateur peuvent être connectés à une interface DP.

b) MPI (Multi Point Interface)

Chaque CPU 300 est équipée d'une connexion MPI (interface multipoint) à travers laquelle l'appareil de programmation peut être connecté. La MPI peut également être employée pour le transfert de données (communication de base S7 interne à la station) et pour connecter la commande de l'opérateur et les unités d'affichage. La connexion MPI est toujours nommée X1[5].

a) MPI/DP (Multi Point Interface/ Decentralized Peripherals)

Certaines unités centrales sont équipées d'une interface MPI/DP combinée en tant que première connexion. À la livraison, l'interface est configurée en tant qu'interface MPI : si vous voulez l'utiliser comme une interface DP, vous devrez modifier les paramètres.

b) PN (PROFINET)

L'interface PN relie l'unité centrale au système de bus Ethernet industriel en modes PROFINET IO et PROFINET CBA. En mode PROFINET IO, l'unité centrale peut être utilisée comme contrôleur ou dispositif dans le système PROFINET IO. L'interface PN dispose de deux ports qui sont interconnectés par un commutateur, permettant une configuration facile d'une structure quasi-ligne sur le bus Ethernet. De plus, un appareil de programmation ou une unité de commande et d'affichage de l'opérateur peut être connecté à une interface PN. Le transfert de données vers d'autres appareils est possible grâce à une communication ouverte entre les utilisateurs via un Ethernet industriel [5].

c) PtP (Point à Point)

L'interface PtP (point à point) permet de connecter un CPU compact adéquatement équipé à un autre appareil via un couplage point à point. Les pilotes suivants sont disponibles : Pilote ASCII, procédures 3964 (R) et RK 512 [5].

II.5.7 Programmation structurelle :

En plus des langages de programmation présentés, la norme CEI 61131-3 définit les concepts et les composants de la programmation structurelle, qui remplace la liste inutilisable et simple des instructions dans un seul programme unifié. Le terme programmation structurelle signifie que le programme d'automatisation est composé de différents composants avec des tâches distinctes et sont différents, définies comme des modules de programme. La norme CEI 61131-3 introduit donc un modèle logiciel complet pour les PLC. Ces composantes structurelles sont appelées des unités d'organisation de programme (POUs). Cette norme définit trois types de POU:

- Fonctions
- Blocs de fonctions
- Programmes

a) Fonction

Une fonction est un bloc de programmes qui comporte une ou plusieurs entrées et une seule sortie. Pour qu'une fonction soit exécutée, toutes les entrées doivent avoir une valeur numérique, et elle ne peut stocker aucune information sur son état actuel ou antérieur. Ainsi, l'exécution d'une fonction avec les mêmes valeurs d'entrée, chaque fois qu'elle est appelée, fournira toujours la même valeur en sortie. La norme CEI 61131-3 définit la fonction standard fournie par le fabricant du PLC, ainsi que les fonctions définies par l'utilisateur.

b) Blocs de fonction (FB)

Un bloc de fonction est un bloc de programme qui comporte une ou plusieurs entrées et une ou plusieurs sorties. Contrairement aux fonctions, un bloc de fonction peut stocker des informations sur son état. Cela signifie que les valeurs en sortie dépendent non seulement des valeurs en entrée, mais également des informations stockées. Pour y parvenir, chaque bloc de fonction est directement lié à un bloc de données (DB) où les valeurs sont stockées. Lorsqu'un bloc de fonction est exécuté, toutes les entrées doivent avoir une valeur numérique.

c) Programmes

Les programmes sont des blocs de programme qui sont construits à partir de fonctions et de blocs de fonction, ainsi que d'instructions simples. Les programmes n'ont pas d'entrées ni de sorties, toutes les variables utilisées doivent être universelles, et ils peuvent être programmés dans tous les langages standard. Avec l'introduction et l'utilisation des PDR mentionnés ci-dessus dans un grand système industriel, le programme d'automatisation peut adopter une structure adaptée à la nature distribuée des processus industriels. En général, les processus industriels ne sont pas concentrés en un seul endroit ; ils sont plutôt constitués de sections individuelles ayant des tâches différentes et des emplacements géographiques distincts. La figure II.8 présente la structure d'un programme d'automatisation composé de plusieurs PDR, tels que les OB, FB et FC, qui peuvent correspondre parfaitement au fonctionnement des différentes parties d'un processus industriel [5].

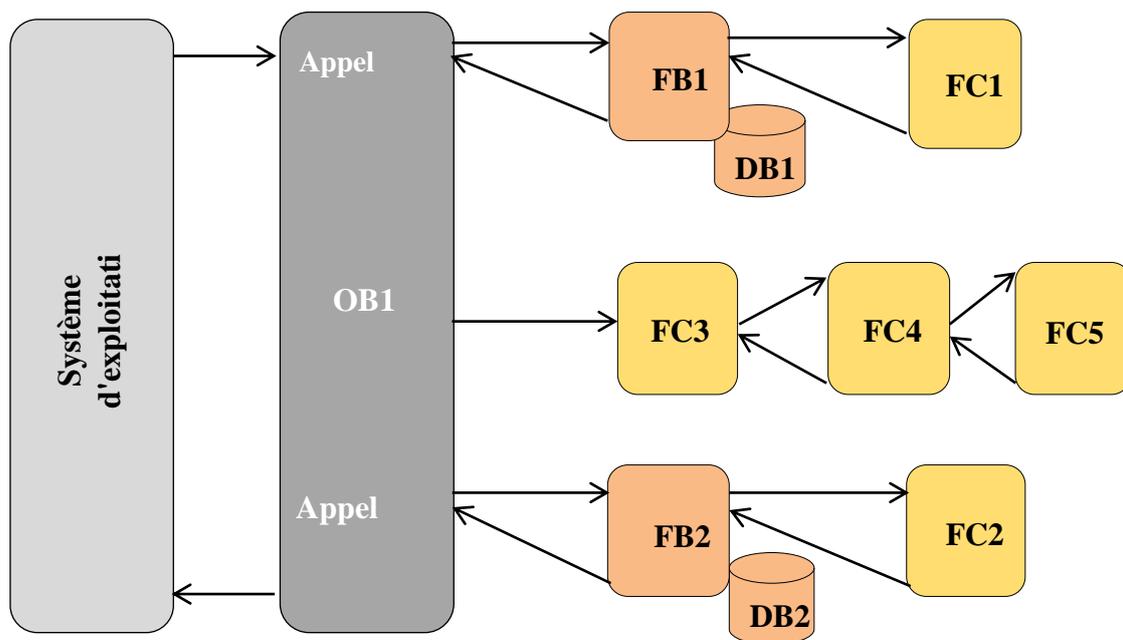


Figure II.8: Exemple de structure d'un programme d'automatisation selon la norme IEC61131-3.

II.6 Conclusion

L'étude du chapitre nous a mis en contact avec la technologie de l'automate S7-300, leurs structures, fonctionnement et langages de programmation.

Dans le chapitre suivant, nous verrons Réseaux industriels et Supervision des systèmes industriels avec SCADA.

CHAPITRE III

Réseaux industriels et supervision

CHAPITRE III

Réseaux industriels et supervision

III.1. Introduction

L'histoire des réseaux industriels locaux remonte à la fin des années 1970 avec l'avènement des équipements industriels numériques intelligents et des réseaux informatiques bureautiques. Leur création a pour objectif pour répondre aux :

Premièrement, la demande croissante de productivité dans le secteur industriel et l'automatisation de la communication entre divers appareils industriels (contrôle et mesure) pour éliminer les pertes de temps et les risques d'erreurs dues à l'intervention humaine

Deuxièmement, l'interconnexion de dispositifs industriels informatisés hétérogènes, introduits anarchiquement dans l'environnement industriel, c'est-à-dire en résolvant chaque problème individuellement sans égard à l'intégrité de l'ensemble du système industriel.

Ainsi, les réseaux locaux industriels sont progressivement introduits dans les systèmes d'automatisation, en différentes étapes selon le domaine d'application. Ils sont nés avec le développement de l'électronique et des matériels numériques programmables. L'avènement des régulateurs numériques et des contrôleurs logiques programmables a incité les fournisseurs à les connecter aux réseaux industriels et à ramener les informations nécessaires aux salles de contrôle à des coûts de câblage réduits [6].

Quant à la supervision, c'est une technique industrielle utilisée pour la surveillance informatique et le contrôle des processus de fabrication automatisés. Il s'agit d'acquisition de données (mesures, alarmes, retour d'état de fonctionnement) et de paramètres de contrôle de processus normalement confiés à des automates programmables. En informatique, la supervision consiste à surveiller le bon fonctionnement d'un système ou d'une activité.

III.2 Connexions distantes

Lorsque de nombreuses entrées ou sorties sont situées loin de l'API, il est possible de faire passer des câbles entre chaque appareil et l'API, mais une solution plus économique consiste à placer les modules d'E/S à côté des entrées et des sorties et de les connecter sur de longues distances en utilisant des câbles mono conducteurs vers API au lieu des câbles multiconducteurs nécessaires sans ces modules d'E/S déportées (Figure III.1) [3].

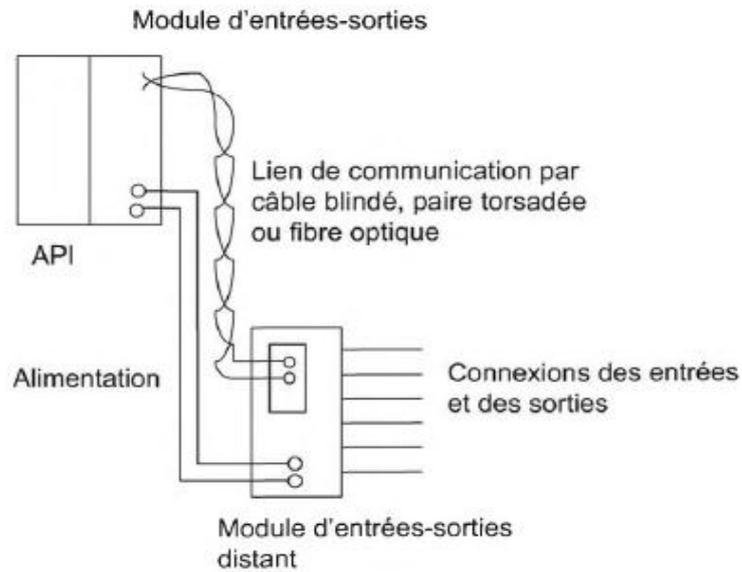


Figure III.1: Utilisation d'un module d'entrées-sorties distant.

Dans certains cas, plusieurs automates peuvent être connectés à un automate maître, qui envoie et reçoit des données d'entrée et de sortie d'autres unités (Figure III.2). Les automates distants n'ont pas de programme de contrôle car tout le traitement est effectué sur l'automate principal [3].

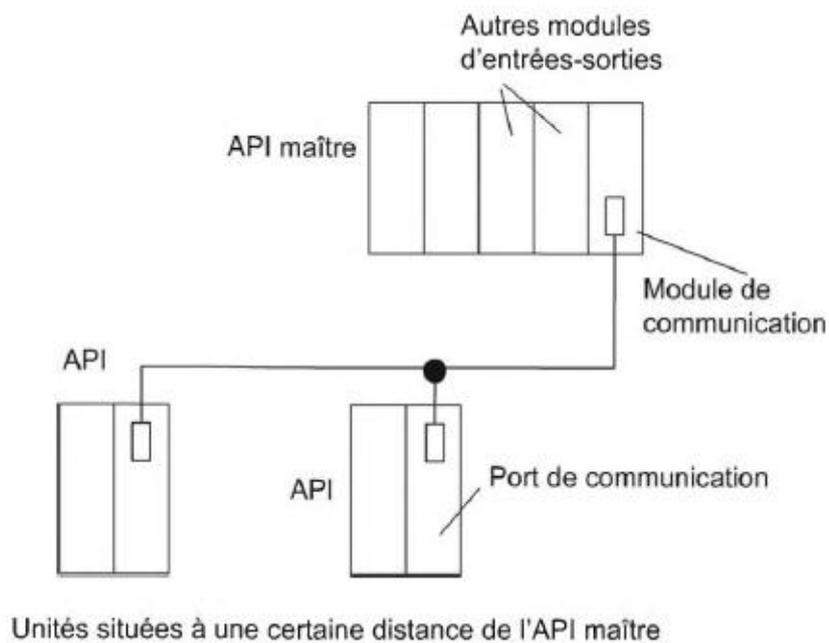


Figure III.2 : Utilisation d'API pour des entrées-sorties distantes.

Les câbles utilisés pour l'échange de données entre les modules d'E/S distants et l'automate central, et entre les automates distants et l'automate maître sont généralement des câbles à paires torsadées placés dans des conduits métalliques mis à la terre pour réduire les effets du bruit électrique. Les câbles coaxiaux peuvent atteindre des débits de données plus élevés et ne nécessitent pas de tube métallique de protection. Les fibres optiques offrent les avantages d'une plus grande immunité au bruit, d'une taille plus petite et d'une plus grande flexibilité. C'est pourquoi ils sont de plus en plus utilisés [3].

III.2.1 Communications série et parallèle

Dans une communication parallèle, tous les bits qui composent un mot sont transmis en parallèle le long du câble en même temps. Cela permet aux données d'être envoyées à des vitesses très élevées sur de courtes distances.

La communication série est utilisée pour transmettre des données sur de longues distances. Il est moins coûteux de faire passer des câbles monoconducteurs sur de longues distances pour les communications série que d'utiliser les câbles multiconducteurs requis pour les communications parallèles. Dans les systèmes basés sur l'API, la communication série peut être utilisée pour connecter un ordinateur utilisé comme terminal de programmation à API.

III.2.2 Normes de communications séries

Pour que les communications série puissent se faire correctement, il est nécessaire de préciser les points suivants :

- La signification de la séquence de bits transmise et comment construire le message. Différents mots sont envoyés sur le même câble, il doit donc y avoir un moyen de déterminer où un mot commence et se termine, et où commence le mot suivant.
- La vitesse à laquelle la séquence de bits est envoyée.
- La synchronisation des horloges à chaque extrémité.
- Protocole ou contrôle de flux utilisé pour échanger des informations telles que « capable de recevoir des données » ou « non prêt à recevoir des données ». Pour cela, deux fils de signal supplémentaires sont utilisés, appelés fils de conversation, l'un pour indiquer au récepteur que l'émetteur est prêt à envoyer des données et l'autre pour indiquer à l'émetteur que le récepteur est prêt à recevoir des données.
- Vérification des erreurs afin que la séquence de bits puisse être vérifiée pour une éventuelle corruption des données pendant la transmission.

L'interface de communication série la plus couramment utilisée est appelée RS232. La connexion se fait via un connecteur de type DB à 25 broches (Figure III. 3), mais pas toujours, avec une prise mâle sur le câble et une prise femelle sur le matériel. Toutes les broches ne sont pas utilisées dans toutes les applications [3].

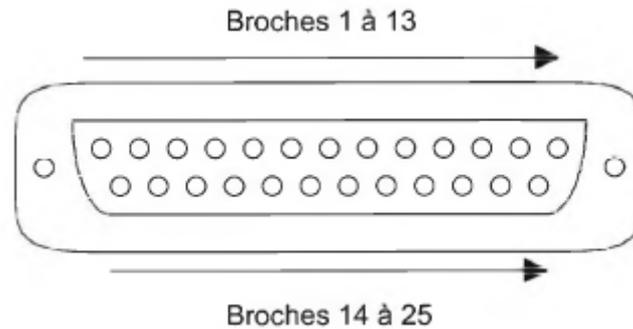


Figure III.3 : Connecteur de type DB à 25 broches.

Tous les bits transmis ne sont pas utilisés pour les données. Certains représentent le début et la fin des éléments de données de la série, qui sont généralement appelés fanions. D'autres sont utilisés pour vérifier si les données sont endommagées pendant le processus de transmission. La figure III. 4 montre le type de signal qui peut être envoyé via l'interface RS232. Ajout d'un test de couplage pour détecter les dommages des données. S'il s'agit d'une paire de couples étranges, ajoutez un point à 1 pour obtenir 1 pair comme 1.

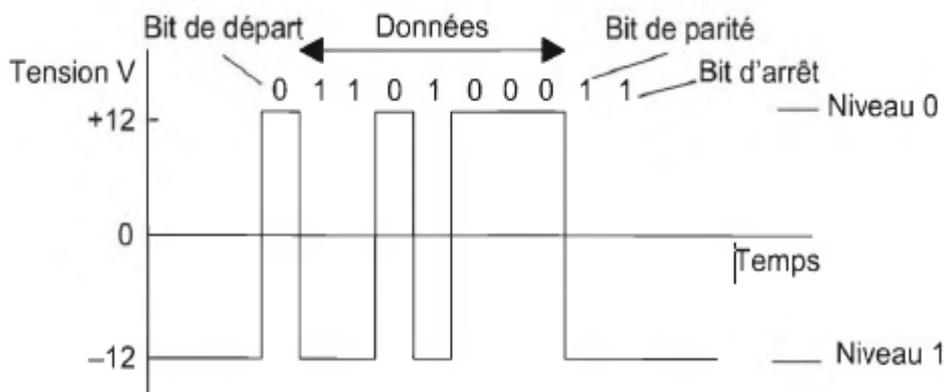


Figure III. 4 : Niveaux des signaux envoyés via l'interface RS232.

D'autres normes, comme RS422 et RS423, sont comparables à la norme RS232. RS232 présente des limites sur la longueur des câbles. RS422 peut être employée pour des distances

plus importantes. Cette technologie utilise deux lignes pour la transmission, le signal transmis étant une différence de tension entre les deux lignes [3].

III.2.3 Normes de communications parallèles

L'interface standard la plus employée pour les communications parallèles est L'interface IEEE-488. Un bus comprend vingt-quatre lignes, dont huit lignes bidirectionnelles servent au transport des données et des commandes entre les différents appareils connectés, cinq lignes sont utilisées pour les signaux de contrôle et d'état, trois participent au dialogue entre les appareils et huit sont des lignes de masse. Le dialogue (handshaking) désigne le transfert des informations de contrôle, comme les signaux DATA READY et INPUT ACKNOWLEDGED, entre deux appareils. Chaque appareil connecté au bus possède sa propre adresse. Les adresses sont envoyées en parallèle sur les lignes de données sous forme d'un mot de sept bits, les cinq bits de poids faible précisant l'adresse de l'appareil et les deux autres apportant des informations de contrôle.

III.2.4 Protocoles

Le flux de données entre les deux appareils doit être contrôlé pour définir la composition et comment démarrer et arrêter la communication. Tous ces aspects forment un protocole. L'appareil doit être en mesure de pointer un autre appareil qu'il devrait démarrer ou arrêter d'envoyer des données. Cet échange peut se faire à L'aide des fils de dialogue qui connectent l'émetteur et le récepteur. Le signal indique au récepteur que l'émetteur se prépare à envoyer (RTS, Ready To Send) et que le signal sur l'autre fil indique que l'émetteur est prêt à recevoir (CTS, Clear-To- Send). Les lignes RTS et CTS existent sur l'interface de communication de la série RS232.

III.3. Les réseaux industriels

L'utilisation croissante de l'automatisation dans l'industrie a conduit à la nécessité d'une communication et d'un contrôle au niveau de l'entreprise, ainsi que d'API, d'ordinateurs, de machines numériques et de robots interconnectés. Les réseaux existent dans trois topologies de base. Dans un réseau en étoile (Figure III. 5 a), les terminaux appelés esclaves sont tous directement connectés à un ordinateur central appelé hôte ou maître. L'hôte comprend la mémoire, le traitement et le matériel de commutation qui permettent aux terminaux de communiquer. Dans un réseau en bus (Figure III. 5b), tous les terminaux sont connectés au même câble, de sorte que chaque émetteur/récepteur a un chemin direct vers tous les autres

émetteurs/récepteurs du réseau. Dans un réseau en anneau (Figure III. 5c), des câbles relient tous les terminaux de l'anneau. Les réseaux en bus et en anneau sont également appelés réseau pair à pair car chaque terminal a le même statut. Ces systèmes permettent à plusieurs stations d'utiliser le même réseau [3].

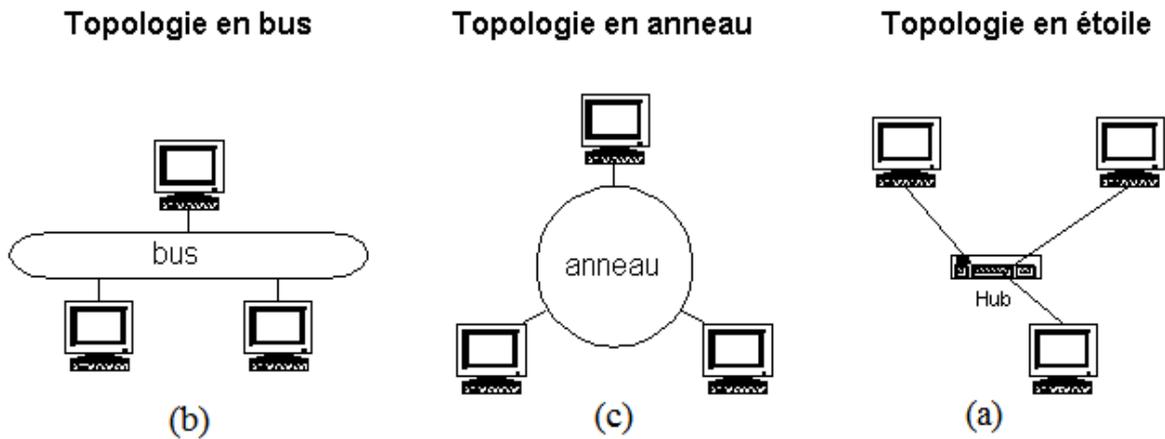


Figure III. 5 : Topologies de réseaux : (a) en étoile, (b) en bus et (c) en anneau.

III.3.1 Système distribués

Les API résident généralement dans une hiérarchie de communication. Le niveau plus bas est constitué de périphériques d'entrée et de sortie, tels que des capteurs et des moteurs, qui sont connectés à la couche suivante via des interfaces d'entrée et de sortie. Le niveau suivant implique des automates, tels que de petits automates ou de petits ordinateurs connectés via un réseau, et à des niveaux supérieurs, des automates et des ordinateurs plus grands pour le contrôle local. À leur tour, ils peuvent faire partie d'un réseau où il y a un ordinateur central contrôlant l'ensemble. Les systèmes de contrôle et de surveillance des processus industriels sont de plus en plus utilisés. Cela implique un contrôle et la collecte de données. Le terme SCADA (Supervisory Control and Data Acquisition) est largement utilisé pour décrire de tels systèmes de surveillance et d'acquisition de données à distance [3].

III.3.2 Normes réseaux

L'interconnexion de plusieurs appareils peut entraîner des problèmes de compatibilité. Par exemple, ils peuvent fonctionner à des débits de données différents ou utiliser des protocoles différents. Pour faciliter la communication entre les appareils, l'ISO (Organisation internationale de normalisation) a défini en 1979 un modèle pour normaliser l'interconnexion

des systèmes ouverts (OSI, Open Systèmes Interconnections), appelé modèle ISO OSI. Les liaisons de communication entre les appareils numériques sont définies par des normes physiques, électriques, de protocole et d'utilisateur. La partie 5 de la norme CEI 61131 pour les automates programmables industriels concerne les communications. Il établit la norme pour les capacités de communication des API, qu'elles agissent en tant que serveurs (fournissant des informations et répondant aux demandes de service) ou clients (demandant des informations et faisant des demandes de service). Afin que les automates conformes à la norme IEC puissent échanger des informations et des signaux de contrôle, plusieurs blocs de communication standard ont été définis [3].

III.3.3 Exemples de réseaux industriels

a) MAP (Manufacturing Automation Protocol).

En 1990, General Motors avait un problème pour automatiser ses opérations de fabrication. Tous les systèmes de l'entreprise devaient pouvoir communiquer entre eux. Par conséquent, ils ont développé un système de communication standard d'automatisation appelé MAP (Manufacturing Automation Protocol). Le système a été appliqué à toutes les machines de l'atelier, telles que les robots, les automates et les postes de soudage.

b) Ethernet

Ethernet n'implique aucune station maître. Toutes les stations connectées ont le même état, ils communiquent donc de pair à pair. Une station souhaitant envoyer un message sur le bus détermine si le bus est disponible, et si c'est le cas, place une trame de message sur le bus. Chaque message consiste en une séquence de bits indiquant l'adresse de destination, l'adresse source, les données à transférer et la séquence de contrôle du message.

Chaque station réceptrice vérifie l'adresse de destination de la trame pour voir si elle lui était destinée. Si oui, elle accepte le message. Ethernet est largement utilisé dans les configurations où un automate doit communiquer avec un ordinateur.

c) ContrôlNet

Allen-Bradley utilise ce réseau. Les données sont placées sur le réseau sans aucune indication, telle que celle du destinataire. Par conséquent, toutes les stations utilisant les données peuvent les accepter en même temps.

d) PROFIBUS :

PROFIBUS (Process Field Bus) est un système développé en Allemagne utilisé par Siemens. PROFIBUS DP (Decentralized Periphery) est un bus au niveau de l'appareil qui fonctionne généralement avec un seul maître DP et plusieurs esclaves. Plusieurs de ces systèmes DP peuvent être installés sur des réseaux PROFIBUS. Les transmissions se font par RS485 ou par fibre optique [3].

III.4. Les réseaux de SIMATIC NET

SIMATIC NET représente une famille de composants et de services de communication. Conçu pour couvrir une gamme complète d'exigences de réseau. Avec des composants conformes aux normes internationales, SIMATIC NET fournit des communications ouvertes qui prennent en charge un environnement multi-fournisseurs pour les APIs, HMIs, ordinateurs, les modules d'E/S, et une variété de systèmes et d'appareils d'automatisation de production. SIMATIC NET englobe les composants tels que les processeurs de communication (PC) pour les API et les PC, logiciels de configuration et de diagnostic, et pilotes logiciels.

Comme indiqué dans le Tableau III.1. La famille SIMATIC NET comprend les réseaux MPI, Ethernet industriel [IEEE 802.3/802.3u) et PROFIBUS (IEC 61 158/EN SO 170).

➤ Réseaux et sous-réseaux

Dans une usine de fabrication. Un réseau connecte des dispositifs tels que les API, ordinateurs, interfaces homme-machine (HMIs). Et d'autres dispositifs à des fins de communication. Souvent, la portée d'un réseau d'usine couvre un ou plusieurs bâtiments. Et comprend un ou plusieurs sous-réseaux. Dans un sous-réseau, toutes les stations sont connectées à un support commun, des mêmes caractéristiques physiques et opérationnelles, et communiquent via le même protocole. Un sous-réseau peut, cependant, impliquer deux ou plusieurs segments de câbles identiques reliés par des répéteurs. Dans un projet STEP7, nous pouvons travailler avec MPI, PROFIBUS et Ethernet industriel ou sous-réseaux point à point (PTP). Les grands projets peuvent impliquer plusieurs sous-réseaux pour compléter le réseau[7].

III.4.1 MPI (Multi-Point Interface)

Dans l'environnement S7, l'interface multipoint (MPI) (Figure III. 6) sert de réseau à faible performance, soutenant de petites quantités d'échange de données entre les API, les dispositifs de programmation, les dispositifs d'interface homme-machine (par exemple, panneaux de commande), ainsi que d'autres systèmes Simatic. Chaque CPU S7 dispose d'une interface MPI intégrée. Qui lui permet de se connecter en tant que nœud MPI sans modules réseau supplémentaires. En tant que réseau, MPI prend en charge jusqu'à 32 nœuds. Utilise le même RS-485 et les mêmes supports de transmission de fibre, et fonctionne généralement à 187.5 Kbs [7].

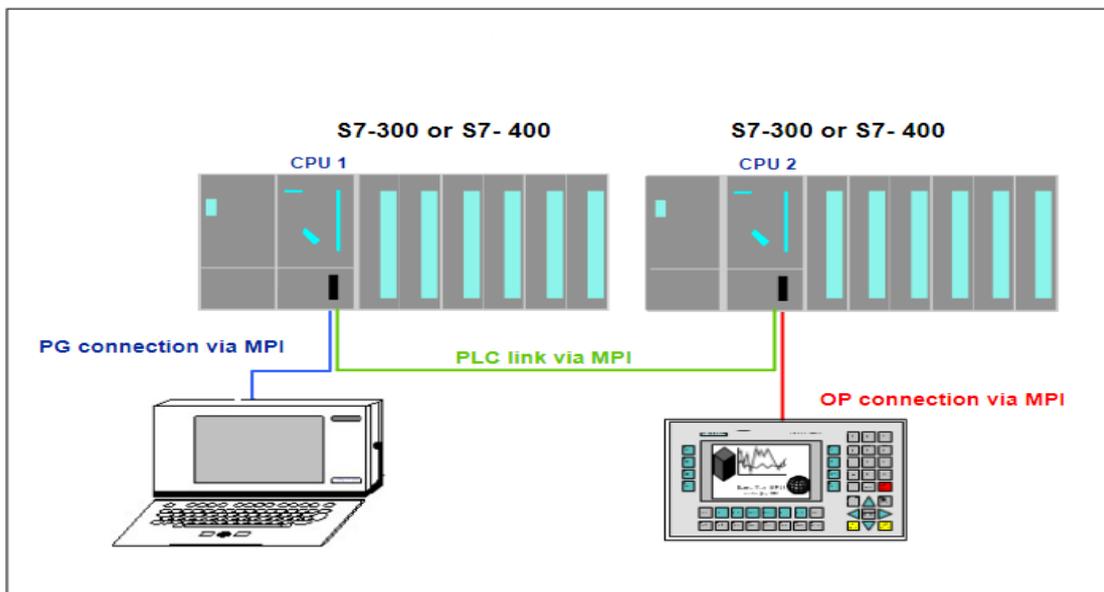


Figure III. 6: MPI typique (sous-réseau d'interface multi-points).

III.4.2 PROFIBUS

PROFIBUS, acronyme pour (PROcess Field BUS), est une norme selon les normes européennes CEI 61 158 et EN 50170 Vol.2.

Le réseau PROFIBUS (Figure III. 7) utilise une paire torsadée blindée, en fibre de verre ou en plastique, prend en charge 126 nœuds, et des taux de transmission jusqu'à 12 Mbaud. Les stations SIMATIC S7-300, S7-400, et les stations PC nécessitent un processeur de communication pour se connecter à un sous-réseau PROFIBUS. Les CPU disposant d'une interface PROFIBUS-DP intégrée (par exemple, CPU 315-2 DP. CPU 416-2 DP), ne nécessitent pas de modules supplémentaires [7].

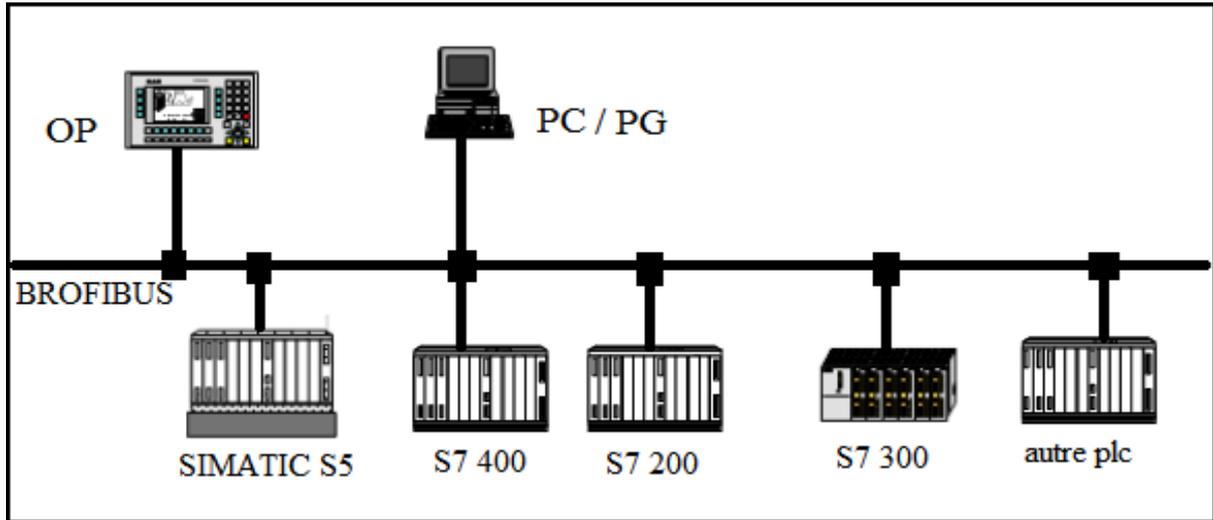


Figure III. 7: Sous-réseau PROFIBUS typique.

III.4.3 Ethernet

Ethernet industriel (Figure III. 8) répond aux besoins des zones d'usine où les systèmes de fabrication sont supervisés et coordonnés. Ethernet industriel prend en charge des protocoles de transport ISO et TCP/IP, et par conséquent la transmission de grandes quantités de données sur des réseaux locaux standardisés avec accès aux réseaux mondiaux.

Le réseau Ethernet industriel utilise un câble coaxial à double blindage, paire torsadée industrielle, verre ou fibre de plastique. Ethernet supporte plus de 1000 nœuds à des vitesses de transmission de 10 Mbit/s ou 100 Mbit/s [7].

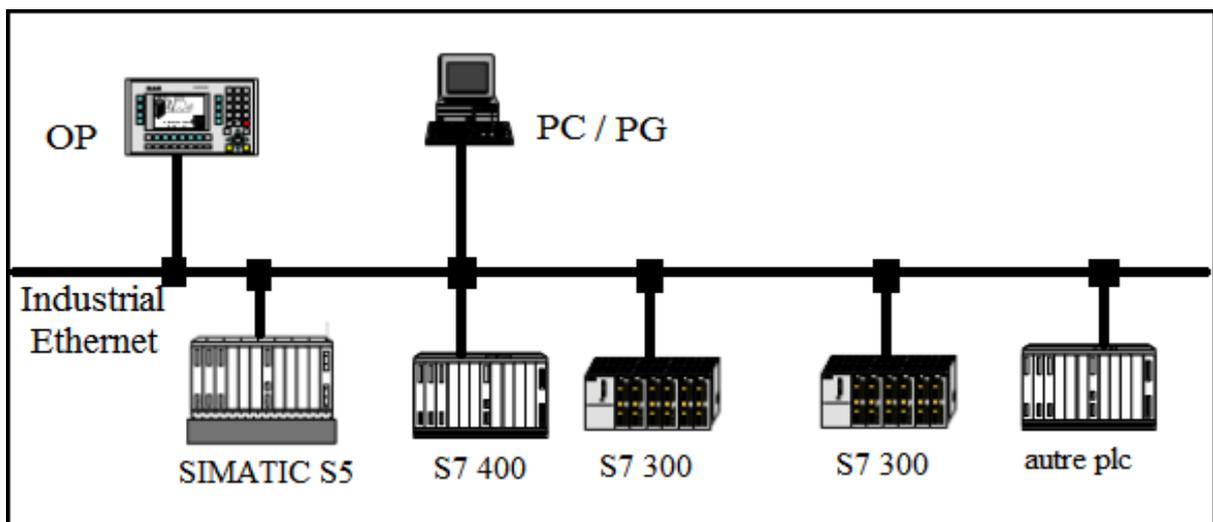


Figure III. 8: Sous-réseau Ethernet typique.

➤ **Caractéristiques de MPI, PROFIBUS et sous-réseau Ethernet.**

Caractéristique		MPI	PROFIBUS	ETHERNET
Standard		Procédure Siemens	EN 50170 Vol.2	IEEE 802.3
Supports de transmission	électrique	Blindage à 2 cœurs	Blindage à 2 cœurs	Paire torsadée Coaxial blindé
	Fibre optique	Fibre de verre Fibre plastique	Fibre de verre Fibre plastique	Fibre de verre Fibre plastique
Débits de transmission		19.2Kbits/187.5 Kbits 12.0Mbits	9.6 Kbits/s a 12.0 Mbits/s	10 Mbits/s 100 Mbits/s
Distances LAN	Électrique	100 m (max)	9.6 km (max)	1.5 Km (max)
	Optiques	-	90 km (max)	200 km (max)
WAN Distances		-	-	Worldwide vio TCP/IP
Nombre de nœuds				
- Typique		2-10	2-16	2-100
- Maximum		32	126	> 1000
Topologie		Line	Line, Tree, Ring. Star. Redundant	Line, Tree, Ring. Star. Redundant
Nivea Automation		Cell/ champ	Cell/ champ	Cell / Gestion
Systems attachable		SIMATIC S7/M7/C7 SIMATIC PG/PC SIMATIC HMI	SIMATIC S7/M7/C7 SIMATIC PG/PC SIMATIC HMI SIMATIC S5 ordinateur hôte	SIMATIC S7 /M7/C7 SIMATIC PG/PC SIMATIC HMI SIMATIC S5 station de travail

Tableau III.1 : Présentation et caractéristiques de MPI, PROFIBUS et sous-réseau Ethernet.**III.5. Supervision des systèmes industriels****III.5.1 Le système SCADA**

SCADA, en Anglais, **S**upervisory **C**ontrol **A**nd **D**ata **A**cquisition, est un système d'acquisition et de contrôle de données. Comme son nom l'indique, il ne s'agit pas d'un système de contrôle complet, mais qui se concentre plutôt sur le niveau de supervision. En tant que tel, c'est un paquet purement logiciel qui est positionné sur le matériel auquel il est interfacé, en général via des API ou d'autres systèmes de contrôle.

En fait, l'industrie SCADA est essentiellement née du besoin d'un utilisateur en front d'un système de contrôle contenant des PLC. Tandis qu'un PLC fournit un contrôle automatisé et préprogrammé sur un processus, ils sont généralement distribués à travers une usine, ce qui rend difficile de recueillir des données manuellement. En outre, les informations PLC sont généralement dans un format brut non compréhensible par l'utilisateur. Le SCADA recueille

dès les données auprès des PLC par une méthode de communication, et combine et formate les informations. En Europe, SCADA est appelé aussi interface homme-machine (MMI, Man Machine Interface).

Les systèmes SCADA sont utilisés dans les procédés industriels, comme la fabrication de l'acier et la production et distribution d'électricité. La taille de ces centrales varie de quelques 1000 à plusieurs milliers de canaux d'entrée/sortie (E/S). Les systèmes SCADA évoluent rapidement sur le marché des installations industrielles, électriques et de communication avec un certain nombre de canaux d'E/S de plusieurs 100K [8].

III.5.2. Architecture du système SCADA

Les systèmes SCADA ont fait des progrès substantiels ces dernières années en termes de fonctionnalité, d'évolutivité, de performance et d'ouverture, de sorte qu'ils constituent une alternative au développement interne, même pour les systèmes de contrôle très exigeants et complexes comme ceux des expériences de physique. Les principaux composants d'un système SCADA sont :

- Plusieurs APIs ou unités terminales distantes (RTU, Remote Terminal Unit).
- Poste principal et ordinateur(s) HMI.
- Infrastructure de communication

a) Architecture matérielle

La figure III.9 montre l'architecture matérielle typique d'un système SCADA simple. Ici, le serveur SCADA lit les débits et les niveaux mesurés et envoie les points de consigne aux automates. API1 compare le débit mesuré au point de consigne et contrôle la vitesse de la pompe. API2 compare le niveau mesuré au point de consigne et contrôle le débit à travers la vanne.

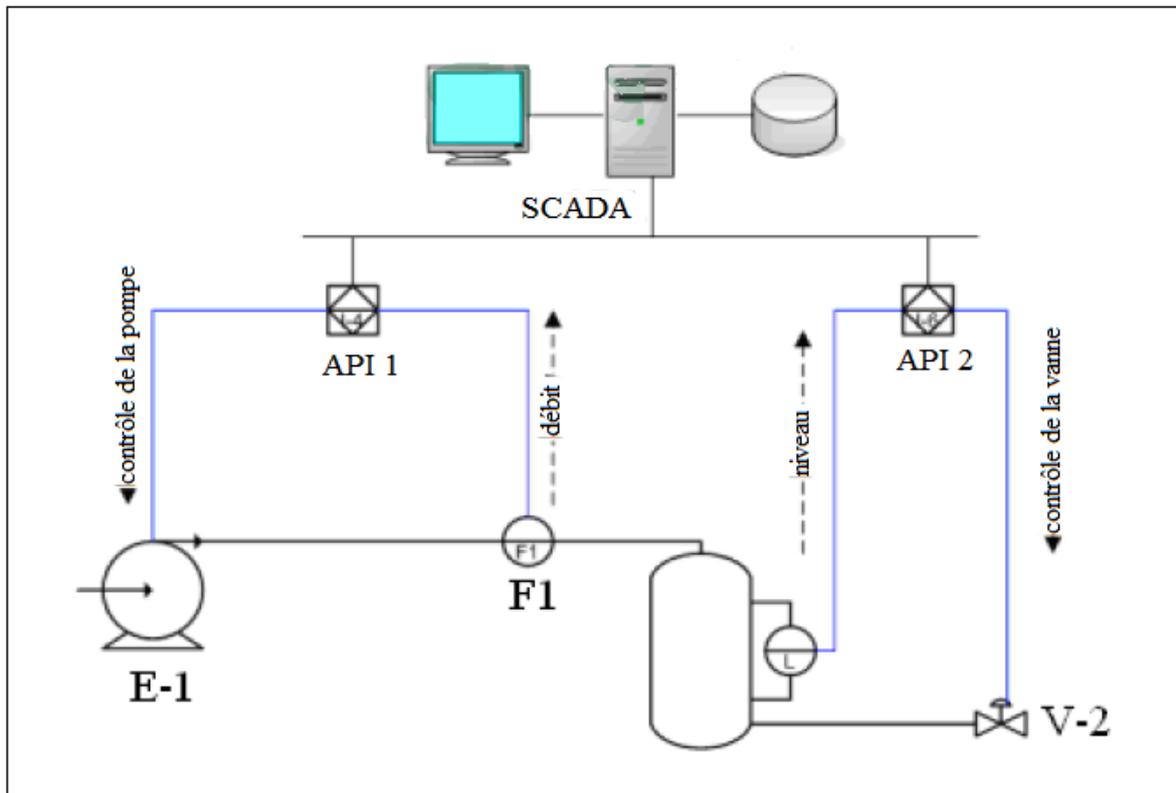


Figure III.9: Illustration de l'architecture d'un système SCADA simple.

Le RTU se connecte à l'équipement physique et lit les données d'état telles que l'état ouvert/fermé à partir d'un interrupteur ou d'une vanne, lit les mesures telles que la pression, le débit, la tension ou le courant. En envoyant des signaux à l'équipement, le RTU peut contrôler l'équipement, comme ouvrir ou fermer un interrupteur ou une soupape, ou régler la vitesse d'une pompe. Le RTU peut lire des données d'état numériques ou des données de mesure analogiques, et envoyer des commandes numériques ou des points de consigne analogiques.

Le terme "Master Station" fait référence aux serveurs et logiciels responsables pour communiquer avec les équipements de terrain (RTU, API, etc.), puis au logiciel HMI s'exécutant sur les postes de travail, des salles de contrôle, ou autre part. Dans les petits systèmes SCADA, le poste maître peut être composé d'un seul PC. Dans les grands systèmes SCADA, le poste maître peut inclure plusieurs serveurs et des applications logicielles distribuées. Le système SCADA présente généralement les informations au personnel d'exploitation sous forme de synoptique. Ce signifie que l'opérateur peut voir une représentation de l'installation contrôlée. Par exemple, une image d'une pompe reliée à un

tuyau peut montrer à l'opérateur que la pompe est en marche et la quantité de liquide qu'elle contient pompant à travers le tuyau pour le moment. L'opérateur peut alors basculer la pompe éteinte.

b) Communications

Les systèmes SCADA ont traditionnellement utilisé des combinaisons de RF (radio fréquence) et les connexions série ou modem pour répondre aux exigences de communication. Ethernet et IP (Internet Protocole) sont aussi fréquemment utilisés dans les grands sites comme les chemins de fer et les centrales électriques [8].

III.6 Conclusion

Ce chapitre consiste à présentation générale des réseaux industriels et du système de supervision industriel (SCADA). Au niveau du chapitre qui suit, nous décrivons la station étudiée et son automatisation.

CHAPITRE IV

Application à la station de tri

CHAPITRE IV

Application

IV.1. Présentation des systèmes MPS (Modular Production System):

IV.1.1 Introduction

Au département de l'électronique et de la télécommunication de l'université de 8 mai 1945 de Guelma, nous avons un ensemble des stations formant une simulation d'une mini-usine d'assemblage. Dans ce travail, nous étudions et programmons la station de tri, cette étude nous permet de connaître les différents aspects industriels tels que la pneumatique et l'utilisation d'une variété de capteurs.

Dans ce travail, nous suivons les étapes suivantes:

- L'étude du fonctionnement de la station et ses différentes parties.
- La réalisation du Grafcet correspondant au fonctionnement souhaité.
- Configuration matérielle de l'automate.
- Programmation du Grafcet en utilisant deux Langages de programmation, à savoir : le Langage contact et le langage Graphes de fonctions séquentielles.
- Tester le programme avec le logiciel PLCSIM et le transféré à la station.
- Réalisation d'un système de supervision basé sur SCADA [9].



Figure IV.1 : Exemple de quatre stations du système MPS

IV.1.2 Présentation des systèmes MPS :

Le système modulaire de production (MPS) permet la modélisation de systèmes d'automatisation industrielle de complexité variable et la création de contenu de formation. Il est universel, modulaire et ouvert à l'expansion, ce qui lui permet d'être adapté aux connaissances et à l'expérience antérieure des étudiants. En se basant sur des fonctions et séquence simples aux systèmes à petite échelle, le MPS peut être progressivement étendu à un système intégré.

La figure IV.1 représente un exemple d'une série du système MPS. Cette série se trouve au département de l'électronique et de la télécommunication de l'université de 8 mai 1945 de Guelma [9].

IV.1.3 Présentation des stations :

Dans le tableau IV.1, nous identifions les quatre stations et les caractéristiques et contenu de formation de chacune d'elle. Ces stations sont : Distribution, contrôle, manipulation et tri. Dans ce travail on s'intéresse à la station de tri (Figure IV.2) [9].



Figure IV.2 : La station de tri (Festo)

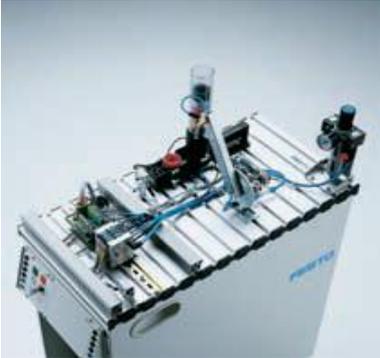
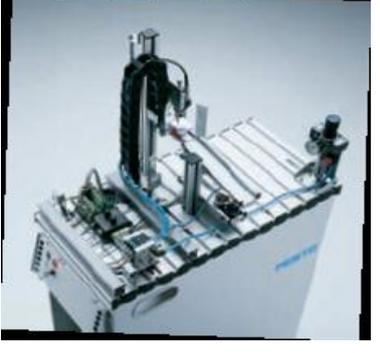
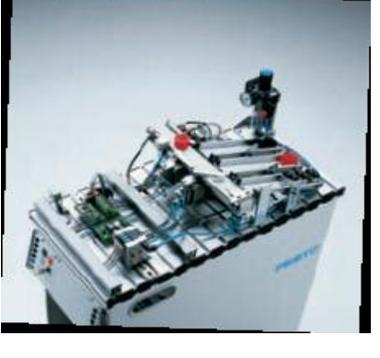
Stations	Equipement	Caractéristiques et contenu de la formation
	Station de distribution	<p>La station de distribution est au début de chaque ligne de production, elle extrait les pièces d'un magasin de stockage empilé (tube), et les transfère à la première station en aval en fonction de la demande de pièces. Ces dernières seront traitées dans le processus, tout comme dans les applications réelles.</p>
	Station de test	<p>La station de test vérifie les pièces de base insérées dans la chaîne de production pour déterminer leur aptitude au traitement ultérieur. Cette station est équipée de différents capteurs et d'un module de mesure, et sa technologie principale la reconnaissance des différentes couleurs dans un processus de simulation de production industrielle.</p>
	Station de manipulation	<p>L'utilisation de divers appareils de manutention est nécessaire à chaque fois que des pièces doivent être remplacés ou manipulés. Comme le contenu technologique de cette station est principalement lié à l'industrie de l'emballage, il se concentre sur les compétences de base de la manipulation et de préhension.</p>
	Station de tri	<p>Une application de tri est effectuée pour les pièces entièrement fabriquées en fin de production. La station de tri utilise différents types de reconnaissance des couleurs et des matériaux pour assurer un tri précis et fournir le contenu correspondant.</p>

Tableau IV.1 : Contenu et caractéristiques de la formation des stations MPS qui se trouve Au sein du département de l'électronique et télécommunication, université de Guelma

IV.2 La station de tri :

IV.2.1 Présentation:

Dans la station de tri (Figure IV.2), les pièces sont triées selon leur matériau et couleur. Les vérins pneumatiques transfèrent les pièces dans les glissières correspondantes.

La fonction de la station de tri est de :

- Trier les pièces selon leurs caractéristiques

La station de tri comprend les éléments suivants :

- Module de convoyeur de tri
- Module glissière
- Plaque de profil
- Chariot
- Console de commande
- API et module entrées/sorties

IV.2.2 Fonction de la station :

La station de tri sépare les pièces à l'aide de 3 glissières. Un capteur optique à réflexion détecte les pièces insérées au début du convoyeur.

Les caractéristiques des pièces (noir, rouge, argent) sont détectées par des capteurs devant le stopper, et les pièces sont triées sur les glissières appropriées via des branches. Les branches sont déplacées au moyen de vérins à course courte utilisant un mécanisme d'inversion.

Un capteur retro- réflexion surveille le niveau de remplissage des glissières [10].

IV.2.3 Séquence :

a) Pré-requis de démarrage

- Pièce au début du convoyeur

b) Position initiale

- Stopper étendu
- Vérin 1 rétracté
- Vérin 2 rétracté
- Arrêt du moteur du convoyeur

c) Séquence

1. Pièce détectée
2. Moteur de convoyeur activé

3. Identification des couleurs et des matériaux
4. Transfert de la pièce dans la glissière correspondante

Pièce noire détectée, dépôt sur glissière à l'extrémité du convoyeur

Stopper rétracté

Pièce éjectée

Étape d'attente

Pièce argentée détectée, déposé sur la glissière au centre du convoyeur

Le vérin 2 avance

Stopper rétracté

Pièce éjectée

Étape d'attente

Pièce rouge détectée, déposé sur la glissière au début du convoyeur

Le vérin 1 avance

Stopper rétracté

Pièce éjectée

Étape d'attente

5. Moteur de convoyeur éteint
6. Stopper avance
7. Le vérin 1/ vérin 2 se rétracte [10]



Figure IV.3 : Formes et couleurs des pièces utilisées

IV.2.4 Les composants de la station:

a) Module de transport :

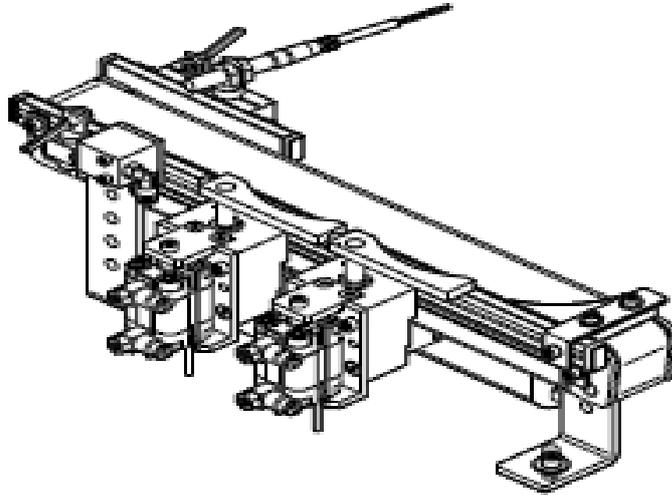


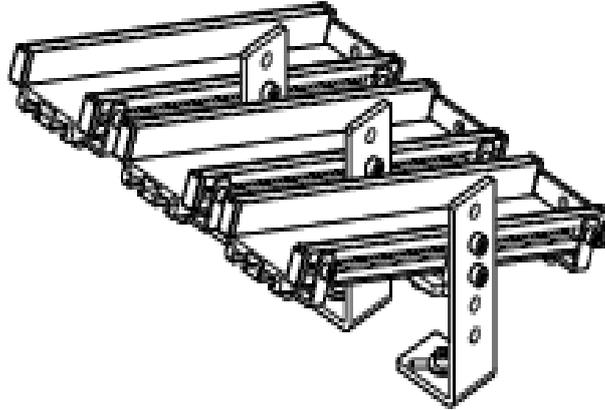
Figure IV.4 : Module de transport

Le module Convoyeur est utilisé pour transporter et éjecter les pièces. Deux branches peuvent être commutées au moyen des vérins à course courte fixés, où les pièces peuvent être triées selon leurs caractéristiques ou leur type. L'entraînement du convoyeur de tri s'effectue au moyen d'un moteur à engrenages CC.

Un capteur optique à réflexion détecte si une pièce est disponible au début du convoyeur. Le cycle du programme démarre et l'entraînement du convoyeur de tri est activé.

La pièce est arrêtée au moyen d'un stopper pneumatique. Un capteur diffus (capteur à réflexion diffuse) identifie la couleur de la pièce : rouge ou noire. Les pièces métalliques sont détectées par un capteur de proximité inductif.

Selon la pièce déterminée, les branches appropriées sont activées. Une fois qu'une pièce est libérée par le stopper, elle est ensuite transportée à la glissière appropriée [10].

b) Module glissière:**Figure IV.5 :** Module glissière

Le module glissière, dans la station de tri, est utilisé pour stocker les pièces. Ce module peut être appliqué universellement grâce à son inclinaison et sa hauteur réglable. Un module comporte trois glissières qui reçoivent les pièces arrivant du module Convoyeur. Un capteur retro-réflexion surveille le niveau de remplissage des glissières [10].

c) Capteur diffus pour la détection de pièces

Un capteur diffus est utilisé pour la détection des pièces. Un dispositif à fibre optique émet une lumière rouge visible. Le capteur diffus détecte la lumière réfléchiée par la pièce ; les différentes surfaces ou couleurs modifient la quantité de lumière réfléchiée.

d) Capteur diffus pour la distinction de couleurs

Ce capteur diffus est utilisé pour distinguer les couleurs des pièces. Le principe de fonctionnement est similaire à celui de Capteur diffus pour la détection de pièces. Cette fois-ci, il est réglé pour capter la différence de couleurs.

e) Capteur de proximité inductif pour la reconnaissance des matériaux :

Le capteur de proximité inductif est utilisé pour la reconnaissance des matériaux. Les capteurs de proximité inductifs détectent les objets métalliques. La distance de commutation est en fonction du matériau et de la finition de surface.

f) Capteurs de proximité pour les vérins

Des capteurs de proximité sont utilisés pour détecter la position finale des vérins. Ce type de capteur est sensible grâce à un aimant permanent monté sur le piston du vérin [10].

g) Capteur rétro-réfléchissant pour le niveau de remplissage :

Le capteur rétro-réfléchissant est utilisé pour surveiller le niveau de remplissage de l'espace de travail des glissières. Un capteur rétro-réfléchissant se compose d'un émetteur et d'un récepteur dans le même boîtier. Le capteur rétro-réfléchissant émet une lumière rouge visible. La lumière est réfléchiée par un réflecteur externe. Si le faisceau lumineux est interrompu par un objet, l'état de commutation des rétro-réfléchissants changes [10].

IV.2.5. Les entrées sorties de la station de tri :

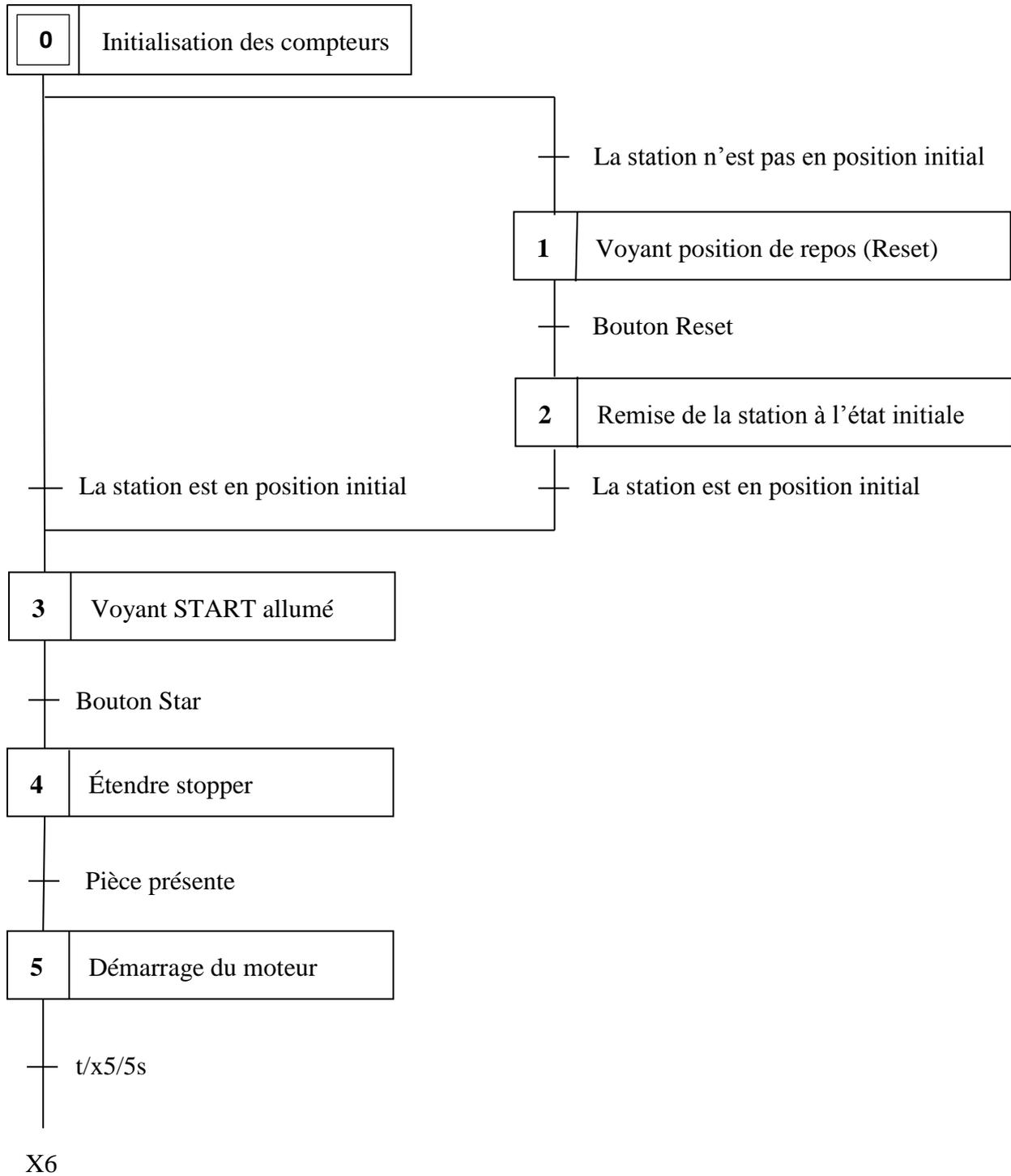
Dans le tableau IV.2, on présente les différents capteurs et actionneurs de la station de tri avec leurs désignations et codes :

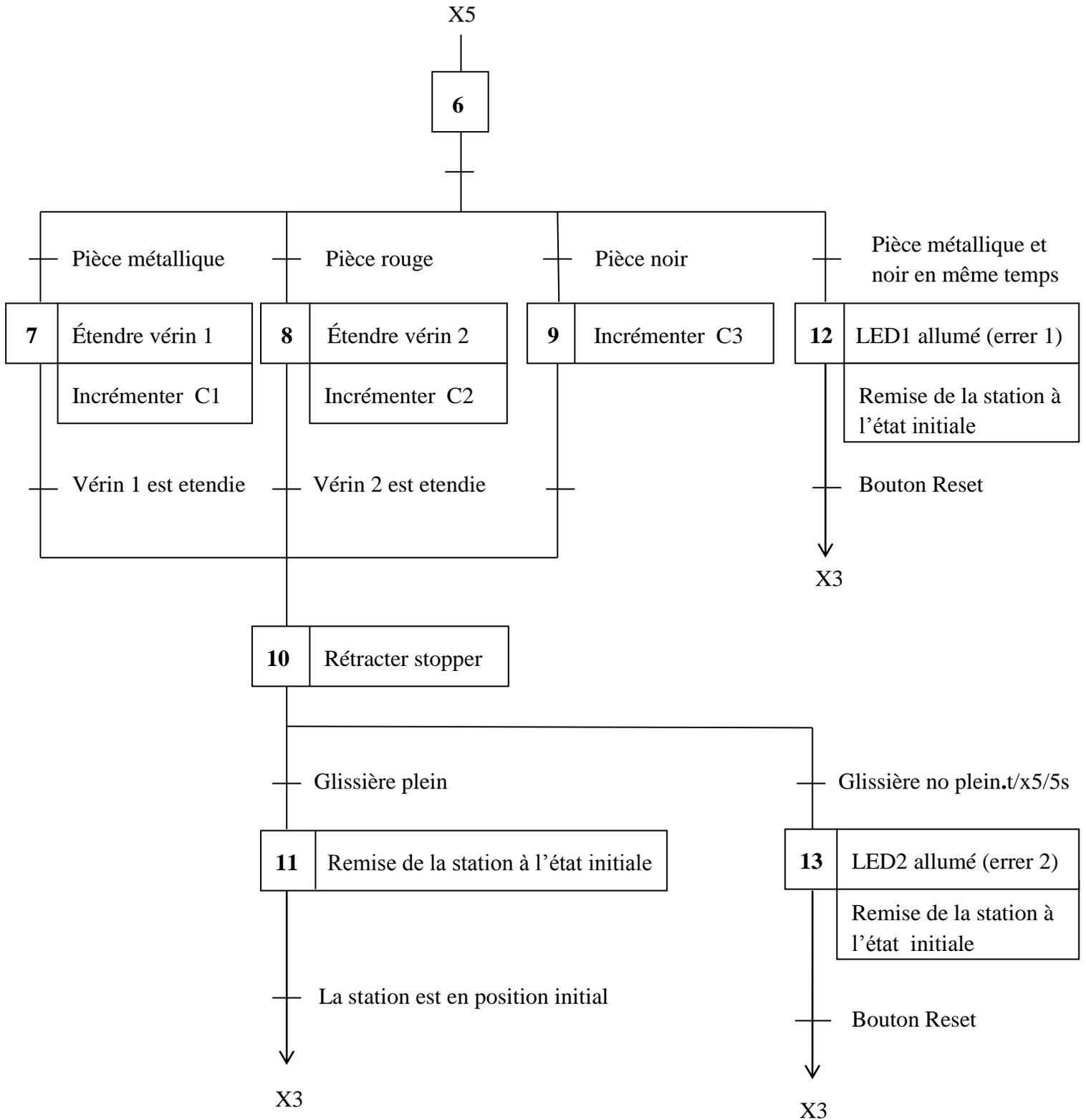
Variable	Désignation	Utilité
K1	Actionneur	Moteur à courroie activé
1M1	Actionneur	Etendre vérin 1
1M2	Actionneur	Etendre vérin 2
1M3	Actionneur	Rétracté stopper
P1	Actionneur	Voyant START allumé
P2	Actionneur	Voyant position de repos (Reset)
LED1	Actionneur	Led allumé pour erreur 1
LED2	Actionneur	Led allumé pour erreur 2
Part_AV	Capteur optique de proximité	Pièce présente
B2	Capteur optique de proximité	La Pièce est métallique
B3	Capteur optique de proximité	La pièce à usiner n'est pas noire.
B4	Capteur rétro-réfléchissant	Glissière plein
1B1	Capteur électromagnétique	Vérin 1 rétracté
1B2	Capteur électromagnétique	Vérin 1 étendu
2B1	Capteur électromagnétique	Vérin 2 rétracté
2B2	Capteur électromagnétique	Vérin 2 étendu
S1	Capteur électromagnétique	Touche START
S2	Capteur électromagnétique	Touche STOP (contact à ouverture)
S3	Capteur électromagnétique	Sélecteur automatique/manuel
S4	Capteur électromagnétique	Touche mise en référence/RESET

Tableau IV.2 : Capteur et actionneur de la station de tri.

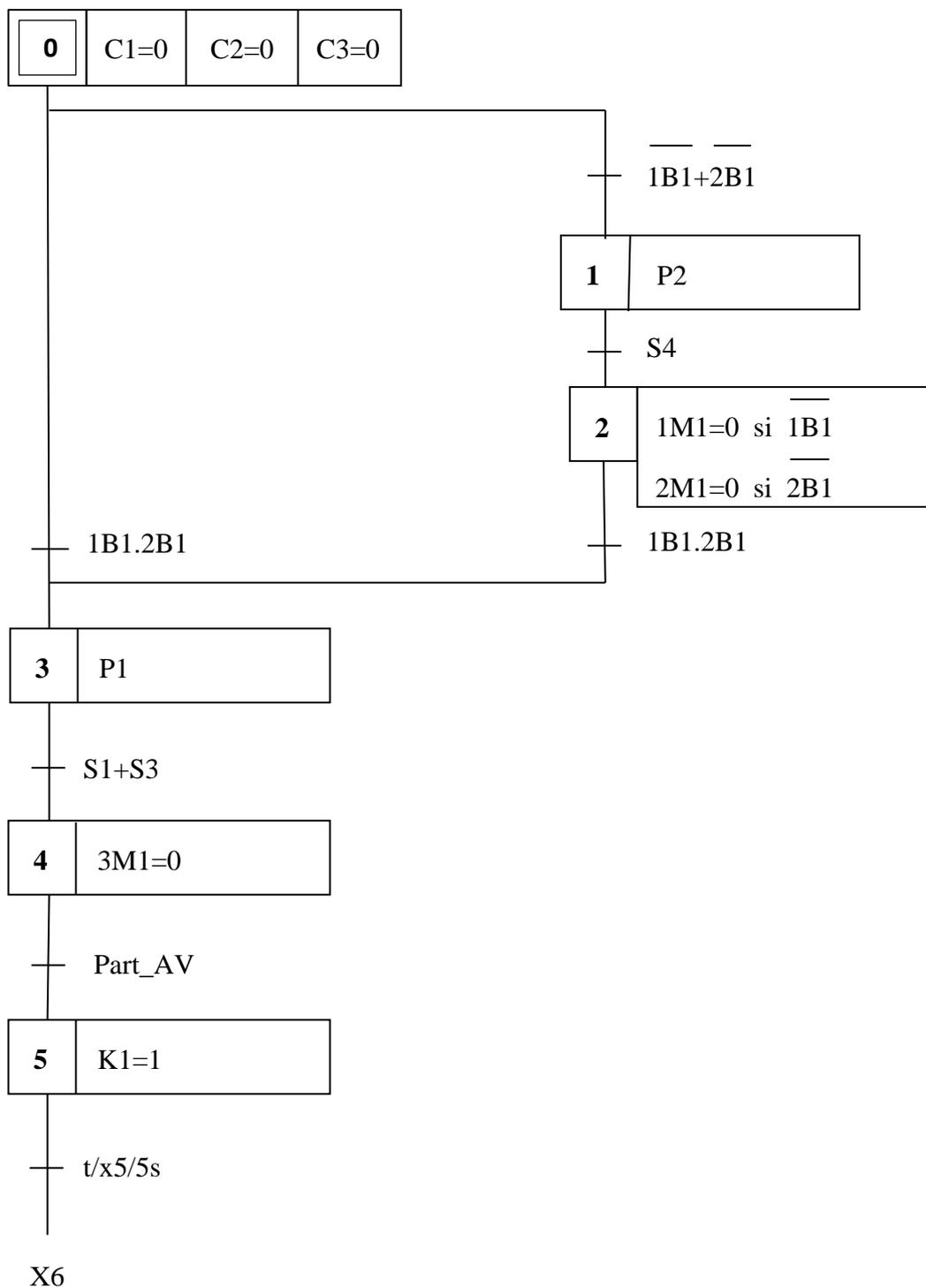
IV.3 Grafset de la station

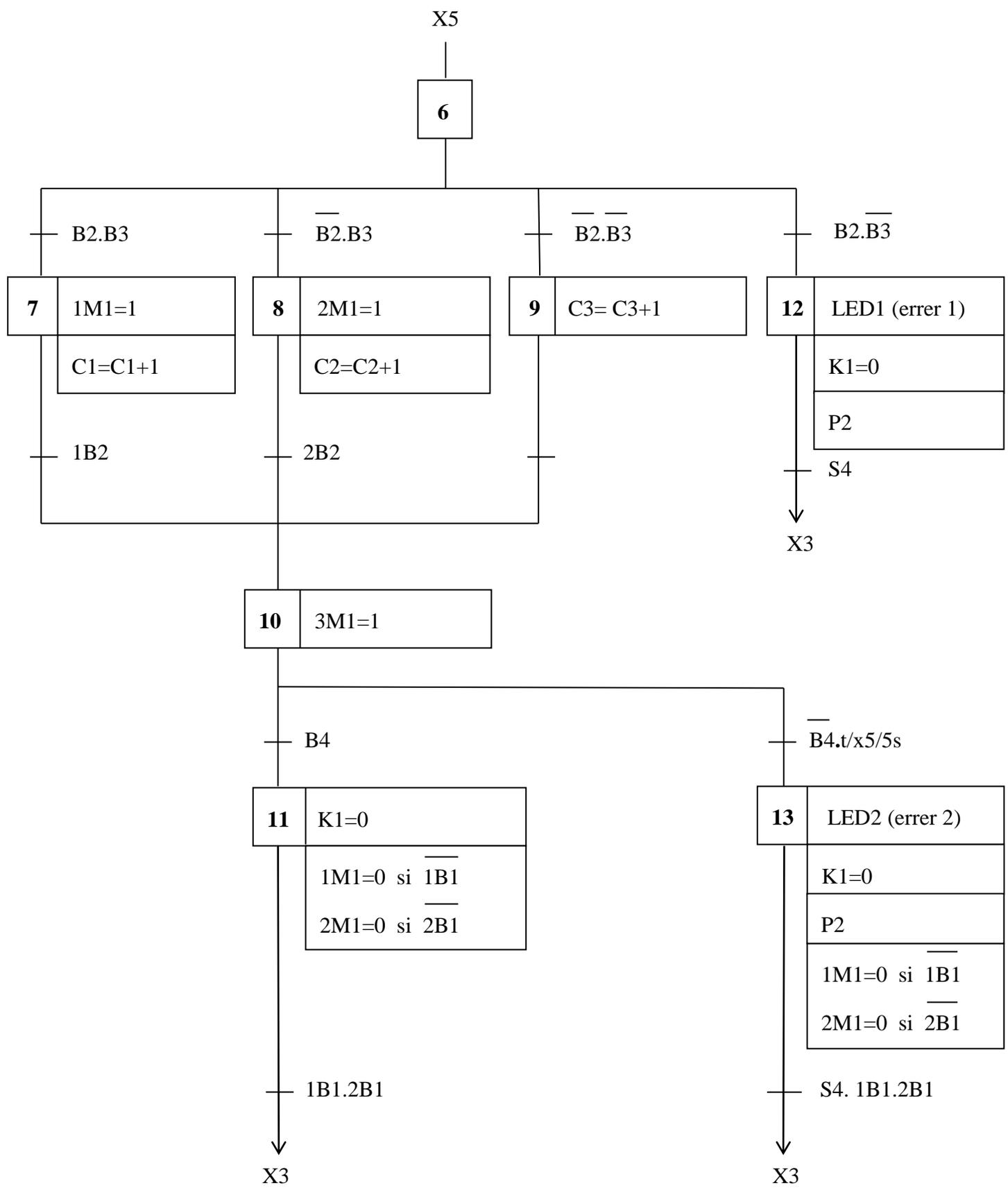
IV.3.1 Grafset niveau 1 :





IV.3.2 Grafcet niveau 2 :





IV.4 Configuration et programmation de l'automate en utilisant grafcet

IV.4.1 Présentation du de step7 :

STEP 7 est le progiciel de base pour la configuration et la programmation de systèmes d'automatisation SIEMENS. Il fait partie de l'industrie logicielle SIMATIC. STEP 7 offre les fonctions suivantes pour l'automatisation d'un processus industriel :

- Configuration et paramétrage du matériel
- Paramétrage de la communication
- Programmation
- Test, mise en service et maintenance
- Documentation, archivage
- Fonctions de diagnostic et d'exploitation
- Un diagnostic d'interface utilisateur qui répond aux connaissances ergonomiques modernes avec un apprentissage est très facile [11].

IV.4.2 Configuration matérielle :

Dans cette étape nous réalisons la configuration matérielle : la station utilisée comporte un CPU 314 et un module d'entrée/sortie digital SM 323DI16/DO16×24V/0.5A. La figure IV.6 donne un aperçu de la configuration de l'automate utilisé.

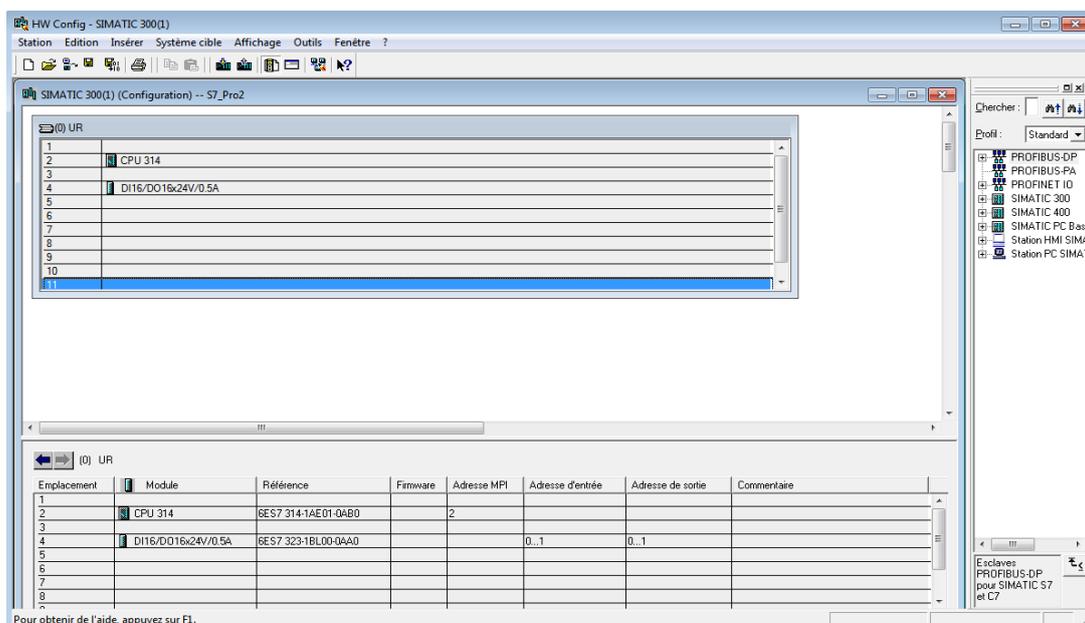


Figure IV.6 : Configuration matérielle du CPU utilisé pour l'automatisation de la station de tri

IV.4.3 Programmation

Le programme souhaité est d'abord créé sur le PC en utilisant le logiciel STEP7. Plus précisément, le programme S7 doit être créé dans un projet contenant la station correspondante. Une fois le PC est relié avec l'interface MPI de l'automate, le programme peut alors être chargé à l'aide de la fonction de chargement dans la mémoire de l'automate programmable (figure IV.7).

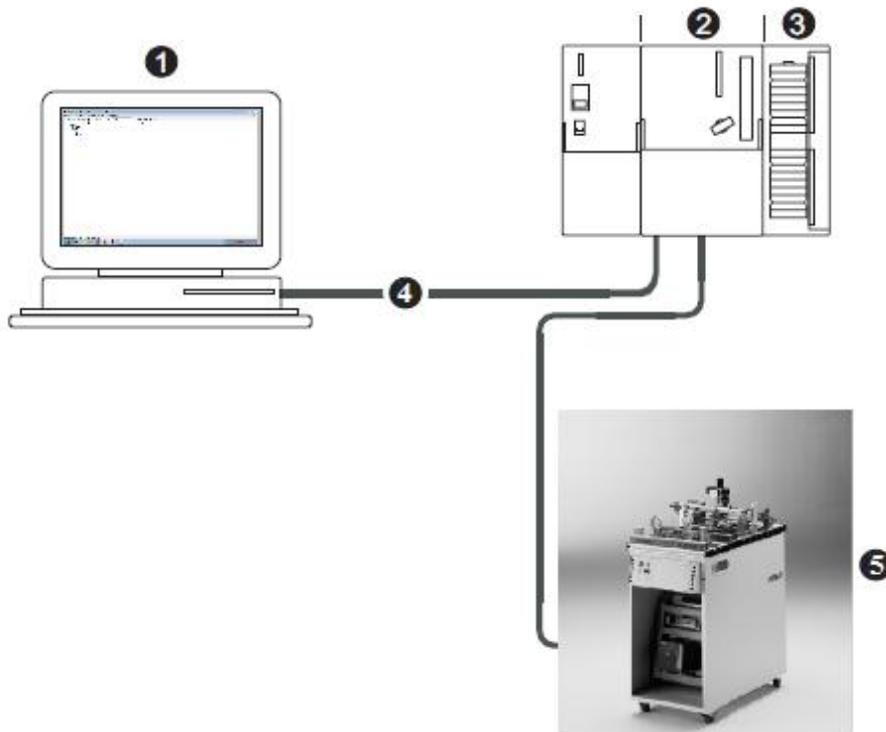


Figure IV.7: Interaction logiciel-matériel :

- (1) PC avec le logiciel STEP 7
- (2) CPU
- (3) Module d'entrées / sorties
- (4) Câble PG
- (5) Station étudiée

Dans ce projet, nous avons programmé le Grafcet en utilisant deux langages de programmation, à savoir : le langage contact et le langage Graphes de fonctions séquentielles. Ces programmes sont présentés dans l'annexe.

IV.4 Supervision avec SCADA

Dans ce travail, nous avons programmé une interface SCADA (avec le logiciel WinCC flexible) pour la supervision et le contrôle du système étudié. L'interface que nous avons créée (Figure IV.8) affiche l'état de tous les composants du système :

- Capteurs : les trois capteurs optique de proximité (pour la détection de présence, la couleur et métallique ou non) et le capteur rétroréfléchissant (détection glissières pleines)
- Actionneurs : convoyeur, stopper et les deux vérins.
- Les boutons poussoir : Start, Stop et Reset ainsi que le sélecteur de mode.
- Les deux Leds d'erreurs

Nous avons ajouté des compteurs pour compter le nombre de pièces dans chaque glissière. Nous avons également ajouté des temporisateurs correspondants aux erreurs.

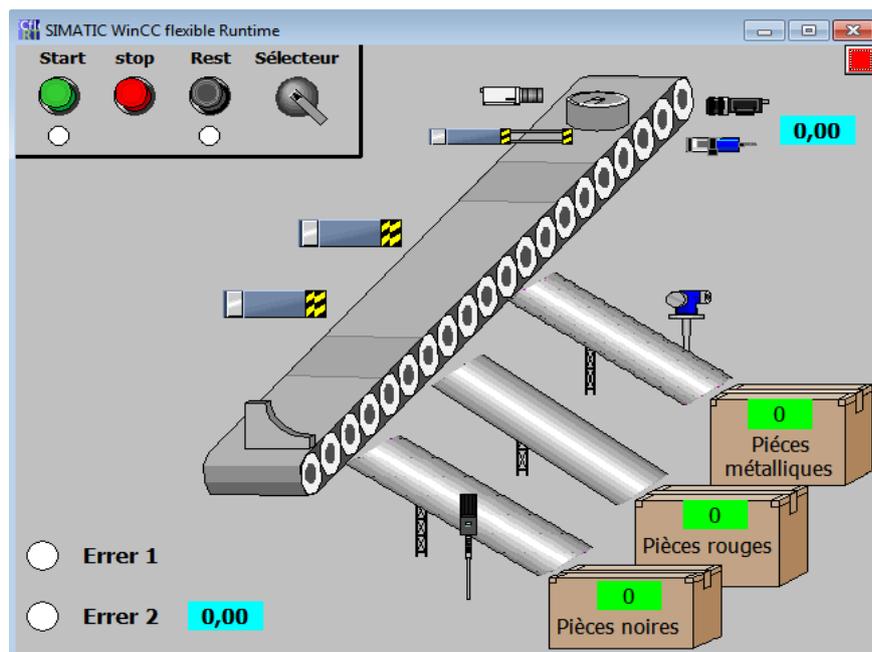


Figure IV.8 : Interface SCADA

IV.4 Conclusion

Dans ce chapitre, nous avons donné une description générale sur la station de tri, et programmé le Grafcet en utilisant deux langages de programmation, à savoir : le langage contact et le langage Graphes de fonctions séquentielles. Et puis nous avons programmé une interface SCADA (avec le logiciel WinCC flexible)

Conclusion générale

Conclusion générale

Dans ce mémoire, nous avons étudié un système didactique au niveau du laboratoire pédagogique du département d'électronique et télécommunication. Il s'agit d'une station de tri faisant partie d'un système de formation pour les automaticiens.

Nous avons commencé par effectuer une étude générale sur les différents aspects technologiques de cette station. En effet, cette dernière comporte une diversité de composants, à savoir : mécanique, pneumatique (avec différents types de vérins et distributeurs) et optique (capteur de proximité et de couleurs). Nous avons ensuite étudié le cahier des charges de son fonctionnement. Nous avons élaboré une modélisation de la commande à l'aide du GRAFCET. Nous avons ensuite programmé GRAFCET en utilisant deux langages de programmation : langage à contacts (langage Ladder) et langage Graphe de fonctions séquentielles. Pour ce faire nous avons utilisé le STEP7.

Une fois que la solution programmable dans l'automate S7-300 a été établie, nous avons testé le programme de commande à l'aide du logiciel PLCSIM. Nous avons finalement transféré le programme à la station et l'avons mise en œuvre, puis nous avons programmé une interface SCADA (avec le logiciel WinCC flexible) pour superviser et contrôler le système étudié.

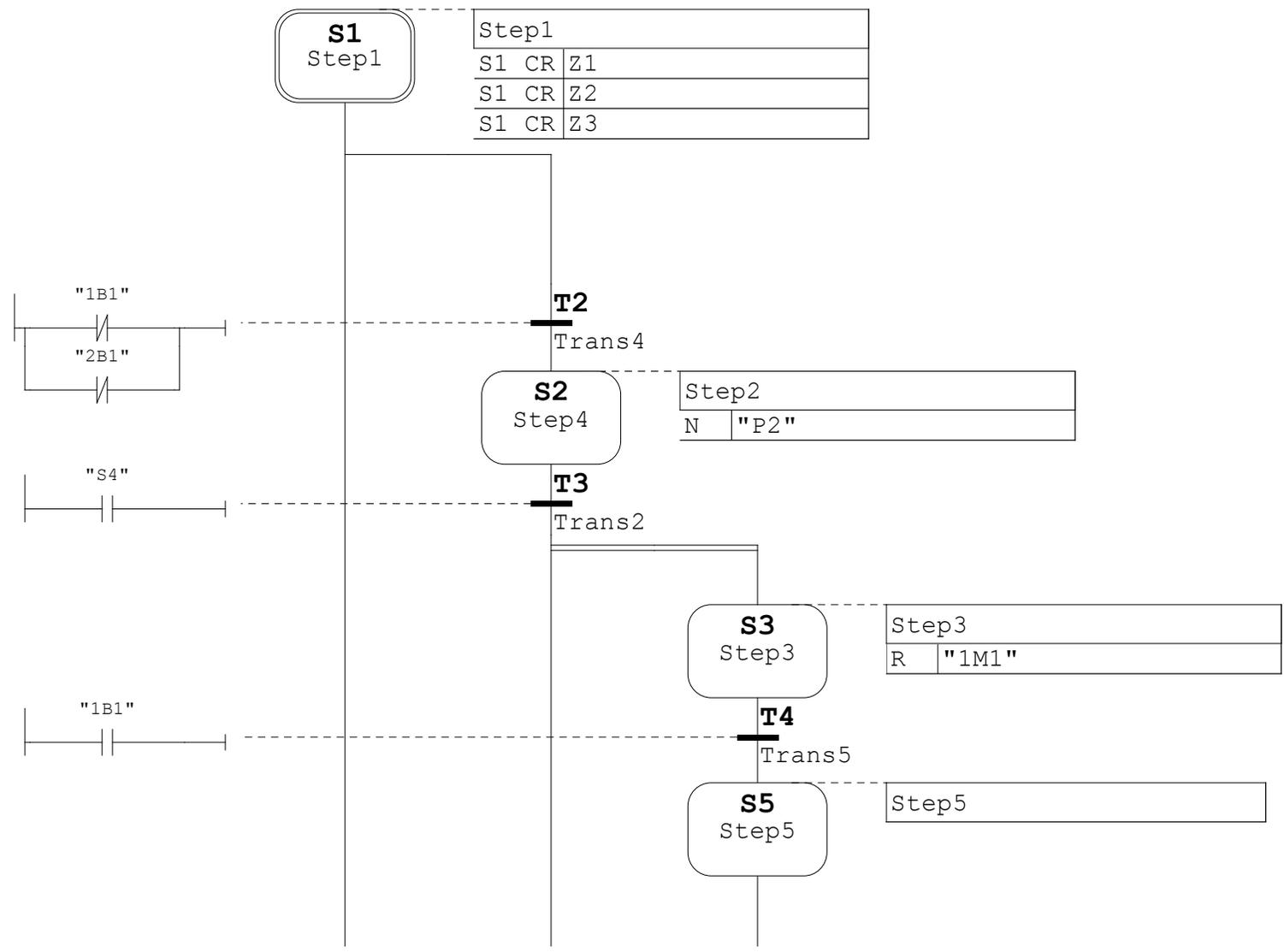
Ce travail nous a permis de tester et de confirmer mes compétences en automatisation et d'évaluer nos connaissances pratiques. Nous espérons que notre travail sera bénéfique pour les promotions à venir.

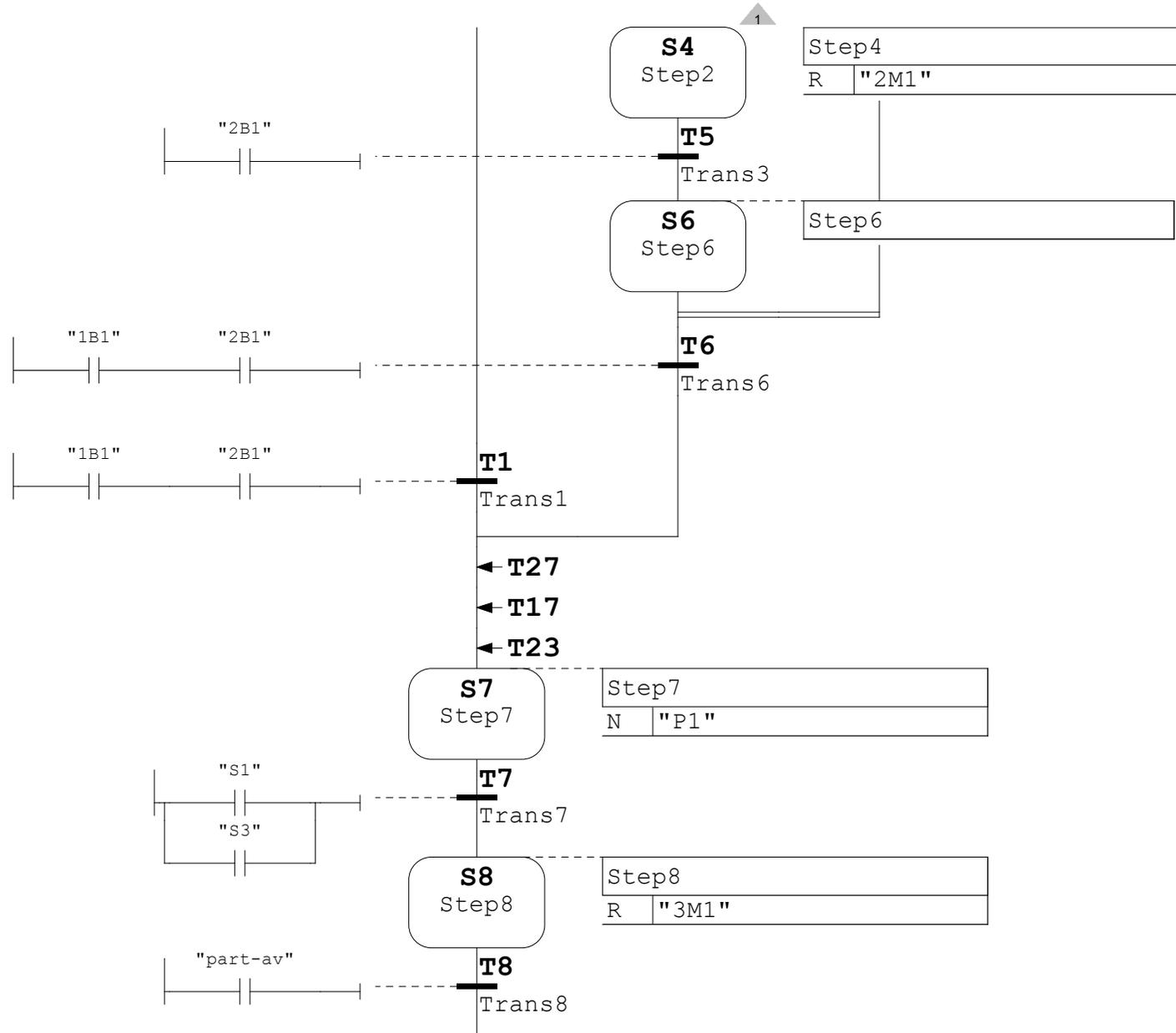
Références

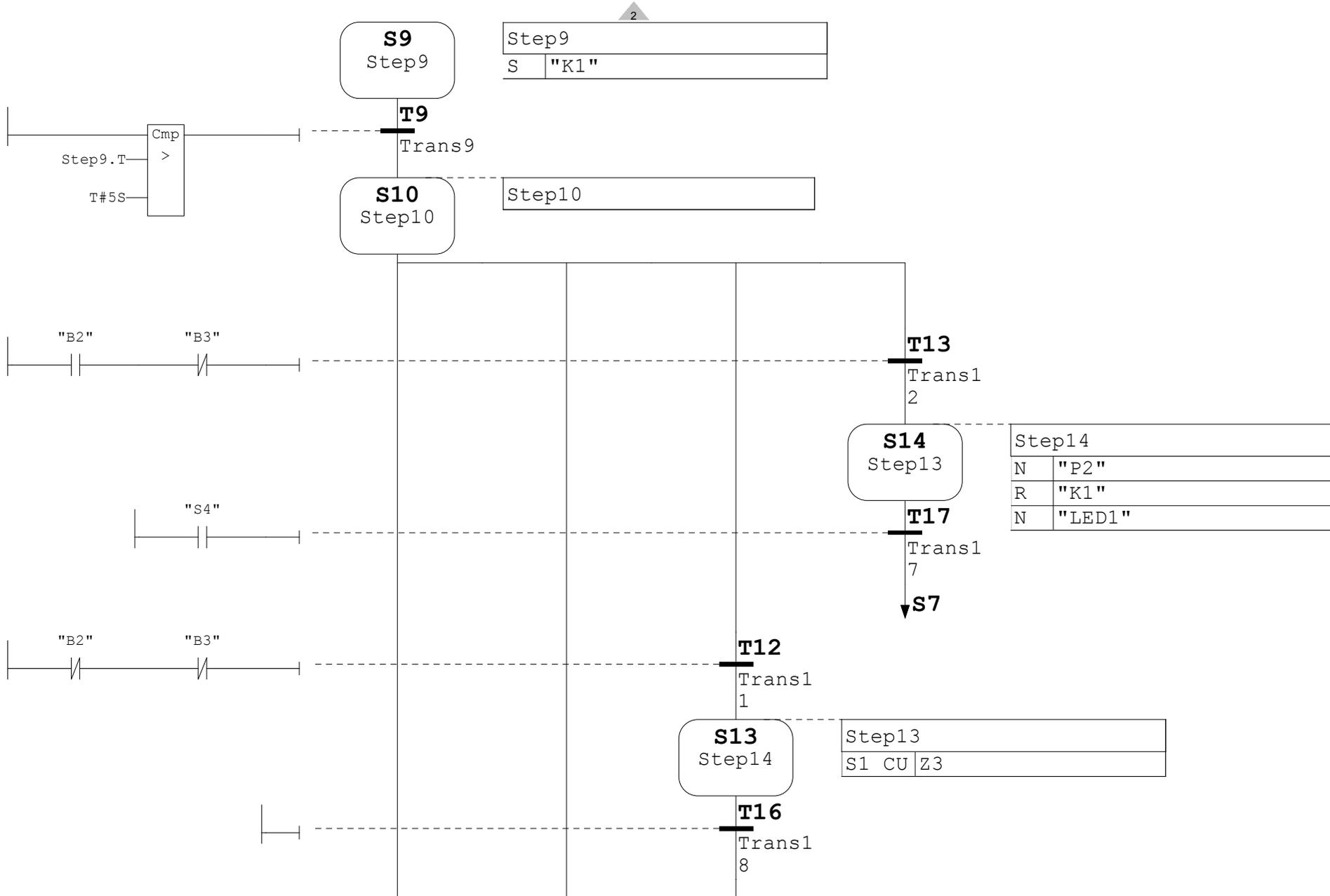
- [1] Lamb, F. (2013). *Industrial automation: hands-on*. McGraw-Hill Education.
- [2] Manesis, S., & Nikolakopoulos, G. (2018). *Introduction to industrial automation*. CRC Press.
- [3] Bolton, W. (2019). *Automates programmables industriels-2e éd*. Dunod.
- [4] Hanssen, D. H. (2015). *Programmable logic controllers: a practical approach to IEC 61131-3 using CODESYS*. John Wiley & Sons.
- [5] Berger, H. (2014). *Automating with SIMATIC S7-400 inside TIA portal: configuring, programming and testing with STEP 7 Professional*. John Wiley & Sons.
- [6] Cours Réseaux Locaux Industriels. (2010). Retrieved May 22, 2023, from http://www.abdelhamid-djeffal.net/web_documents/polycope_rli_10.pdf
- [7] Jones, C. T. (2009). *STEP 7 in 7 Steps: A Practical Guide to Implementing S7-300/S7-400 Programmable Logic Controllers*. Brilliant-Training.
- [8] El-Saba, M. H. (2017). Supervisory Control And Data Acquisition (SCADA). ResearchGate.https://www.researchgate.net/publication/353142664_Supervisory_Control_And_Data_Acquisition_SCADA
- [9] Catalogue FESTO Teciam part P « parallel production »
- [10] Manuel de la station de tri «The Sorting station »
- [11] Manuel mise en route du logiciel Step 7 V5.2 siemens

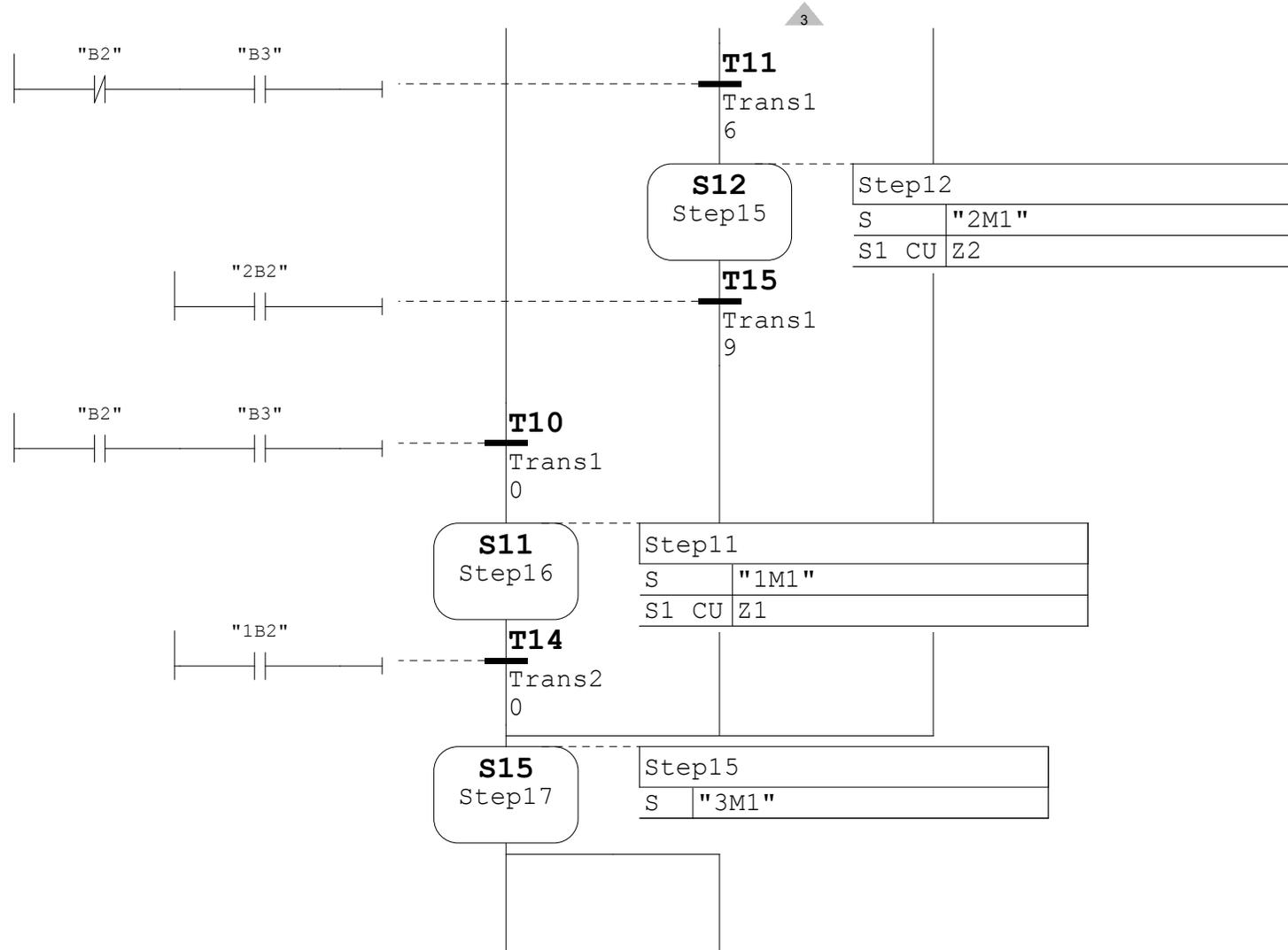
Annexe 01

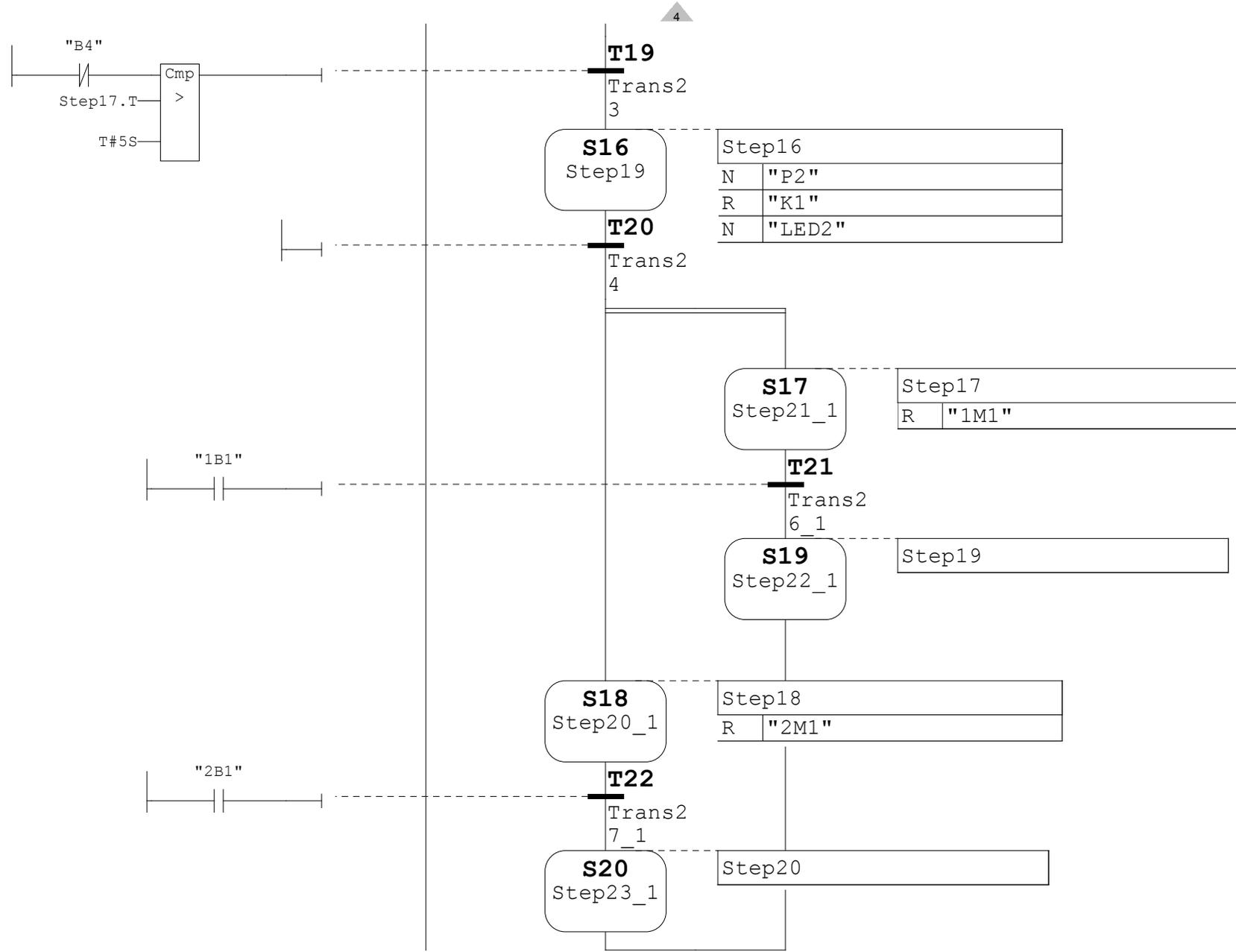
Commentaire de bloc





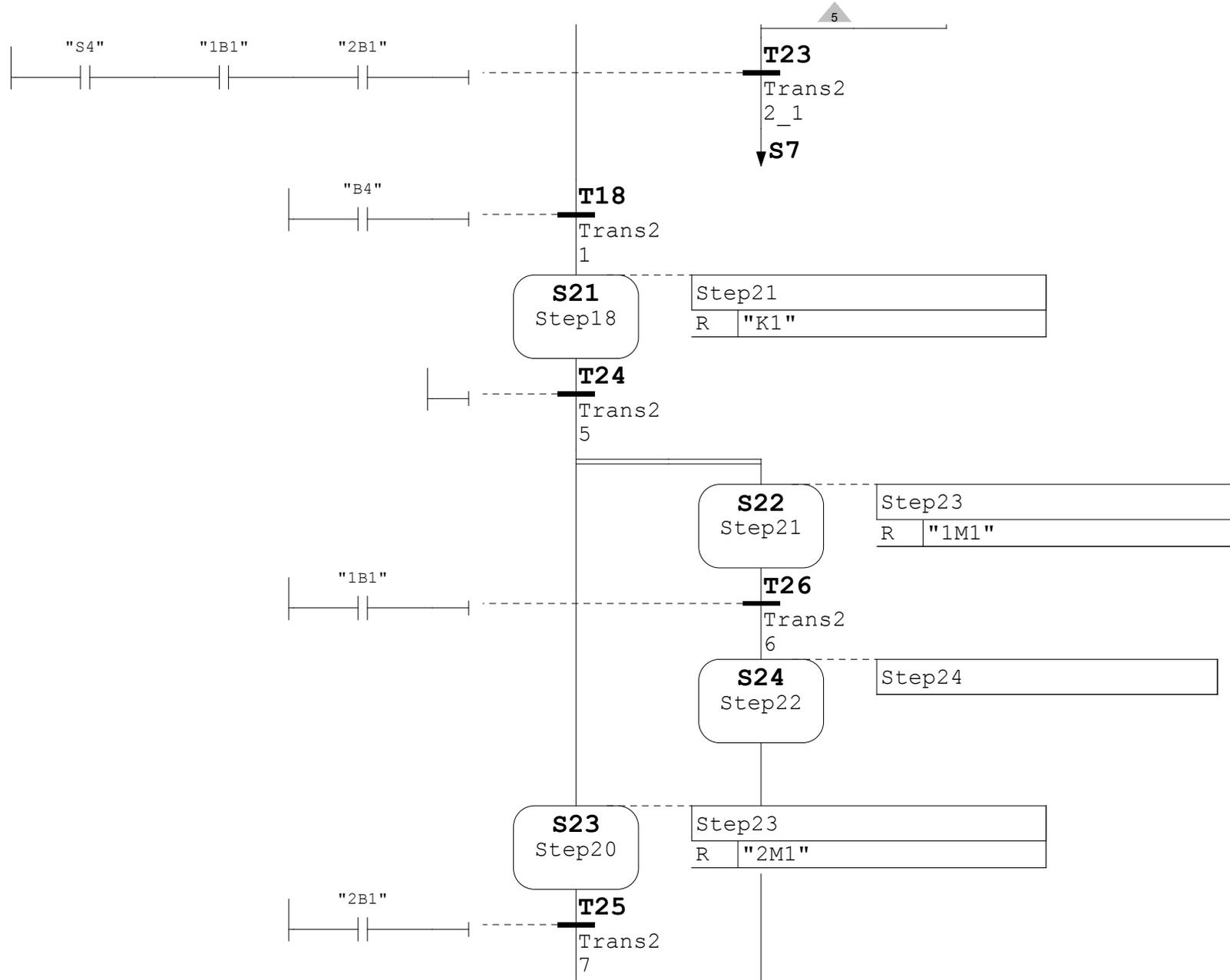


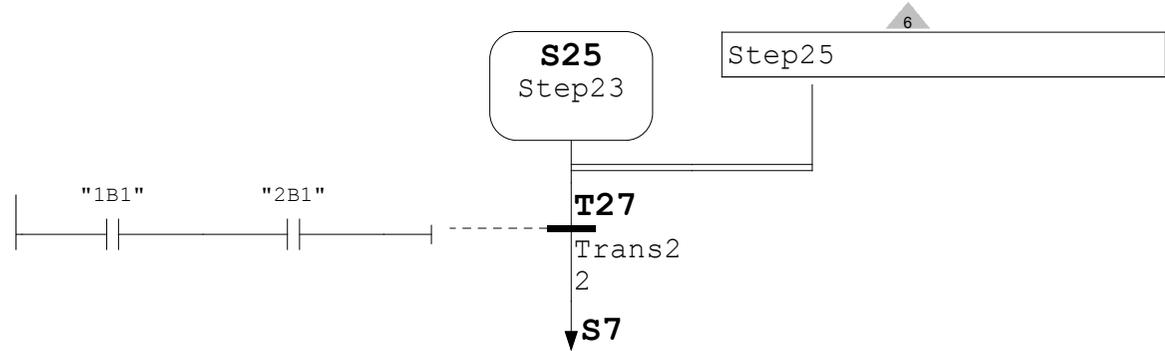




4

6





Annexe 02

FC1 - <hors ligne>

""

Nom :
Auteur :
Horodatage Code :
Interface :
Longueur (bloc/code /données locales) :

Famille :
Version : 0.1
Version de bloc : 2
 17/05/2023 22:10:57
 04/02/2023 12:04:16
 00834 00638 00002

Nom	Type de données	Adresse	Commentaire
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Bloc : FC1

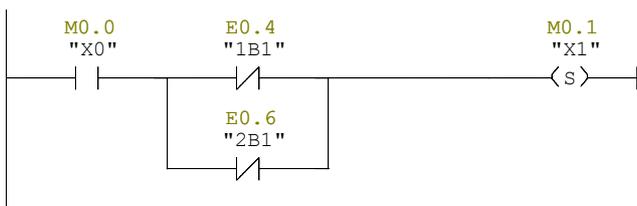
Réseau : 1 X0 Act



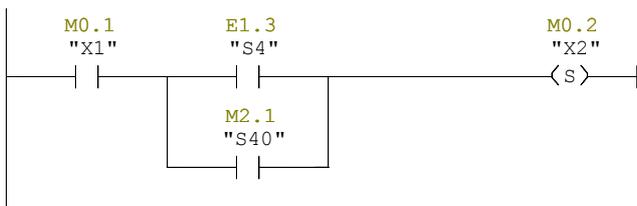
Réseau : 2 X0 Désact



Réseau : 3 X1 Act



Réseau : 4 X2 Act



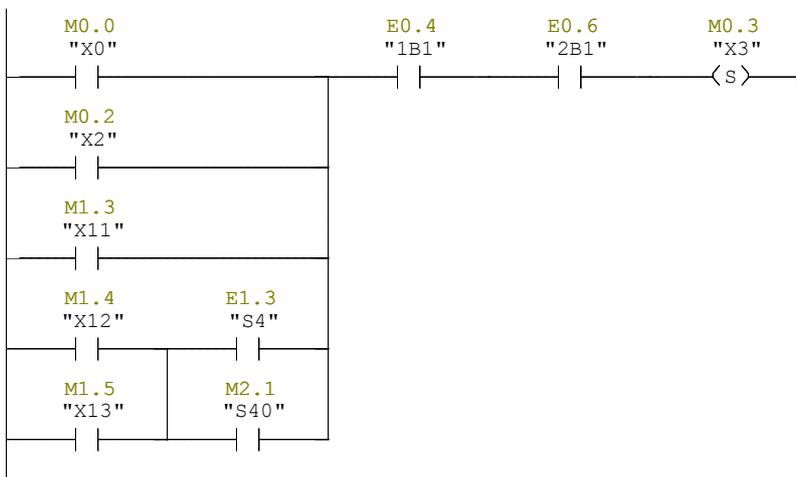
Réseau : 5 X1 désact



Réseau : 6 X2 désact



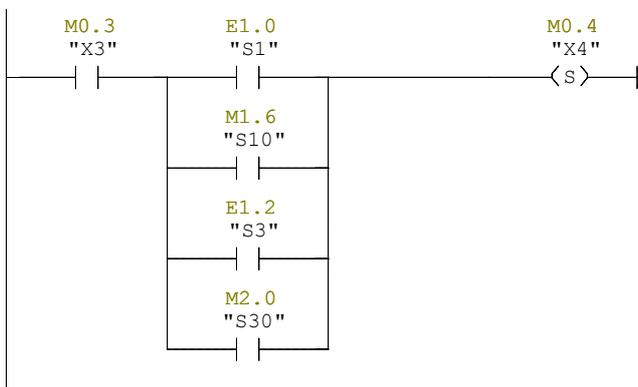
Réseau : 7 X3 Act



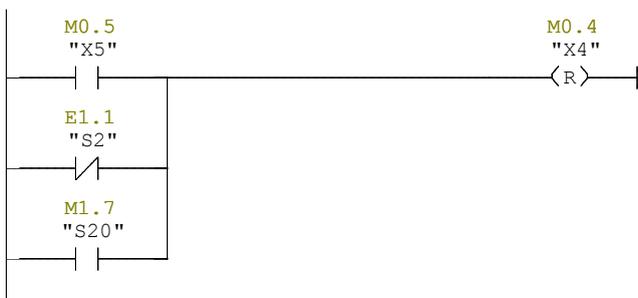
Réseau : 8 X3 Désact



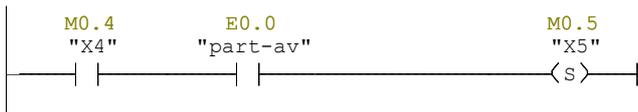
Réseau : 9 X4 Act



Réseau : 10 X4 Désact



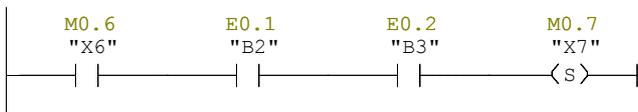
Réseau : 11 X5 Act



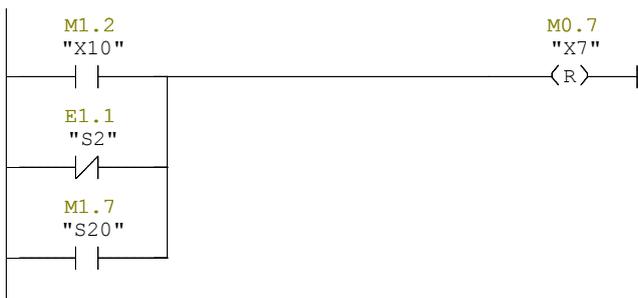
Réseau : 12 X5 Désact



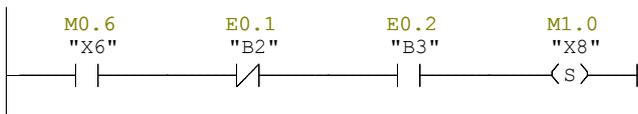
Réseau : 13 X7 Act



Réseau : 14 X7 Désact



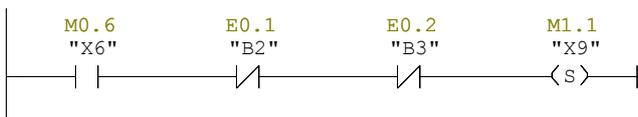
Réseau : 15 X8 Act



Réseau : 16 X8 Désact



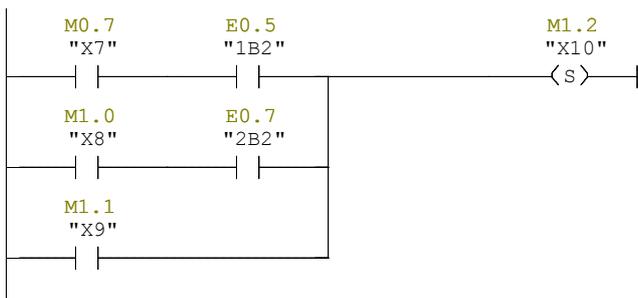
Réseau : 17 X9 Act



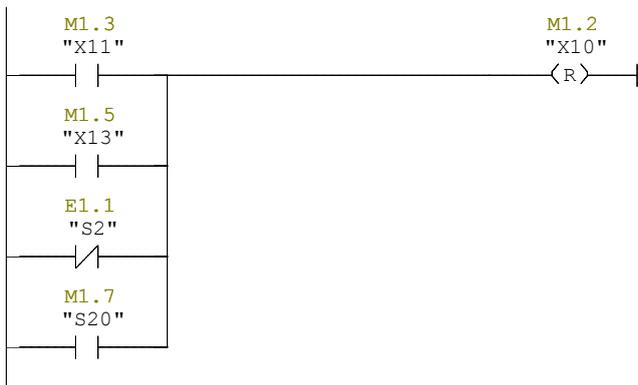
Réseau : 18 X9 Désact



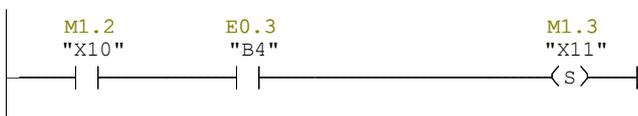
Réseau : 19 X10 Act



Réseau : 20 X10 Désact



Réseau : 21 X11 Act



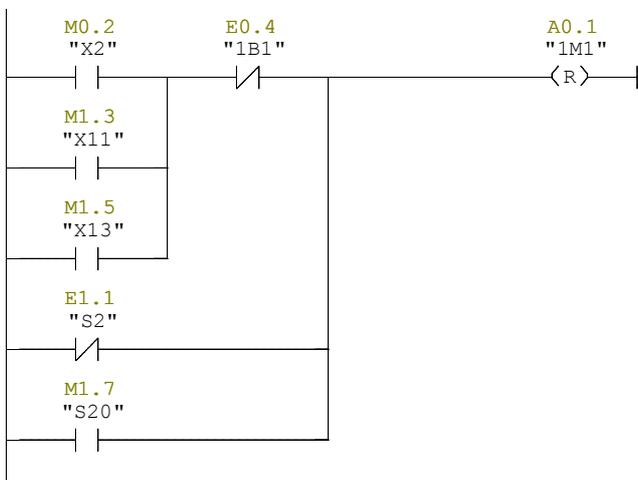
Réseau : 22 X11 Désact



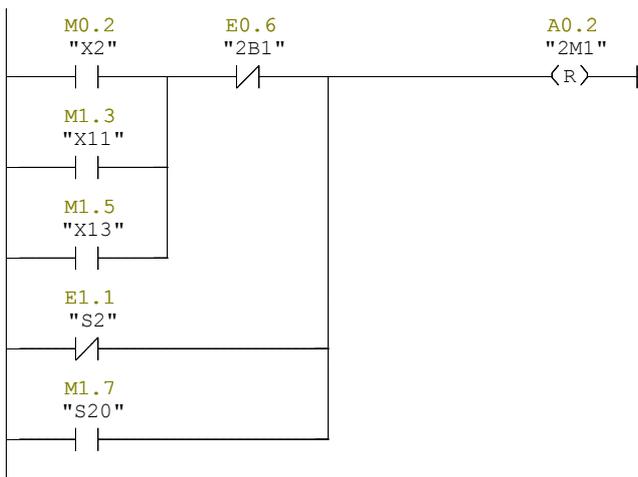
Réseau : 23



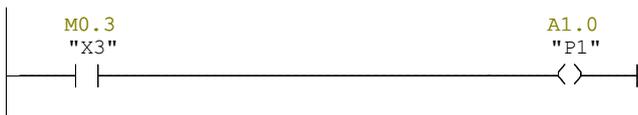
Réseau : 24



Réseau : 25



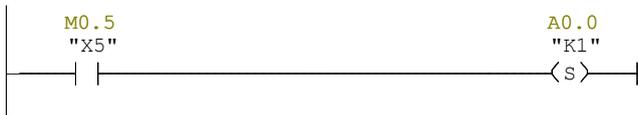
Réseau : 26



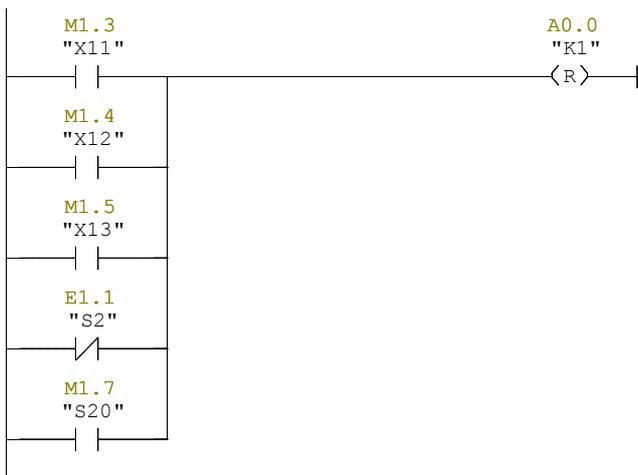
Réseau : 27



Réseau : 28



Réseau : 29



Réseau : 30



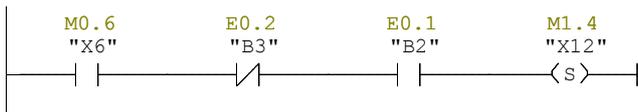
Réseau : 31



Réseau : 32



Réseau : 33 X12 Act



Réseau : 34 X12 Désact



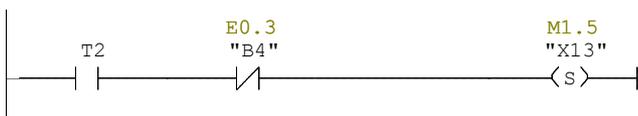
Réseau : 35



Réseau : 36



Réseau : 37 X13 Act



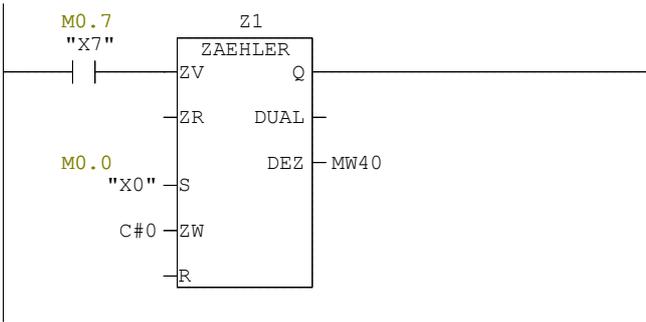
Réseau : 38 X13 Désact



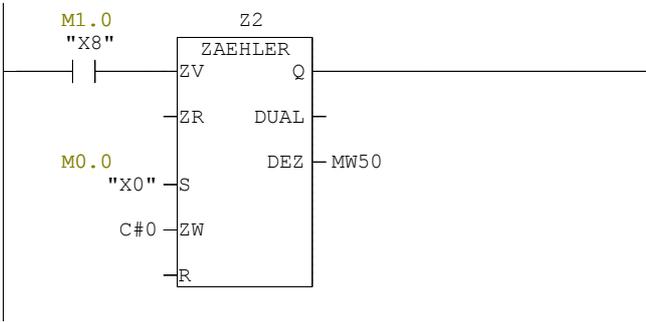
Réseau : 39



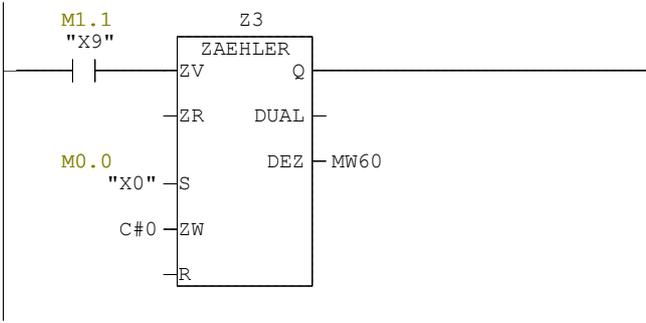
Réseau : 40 C1



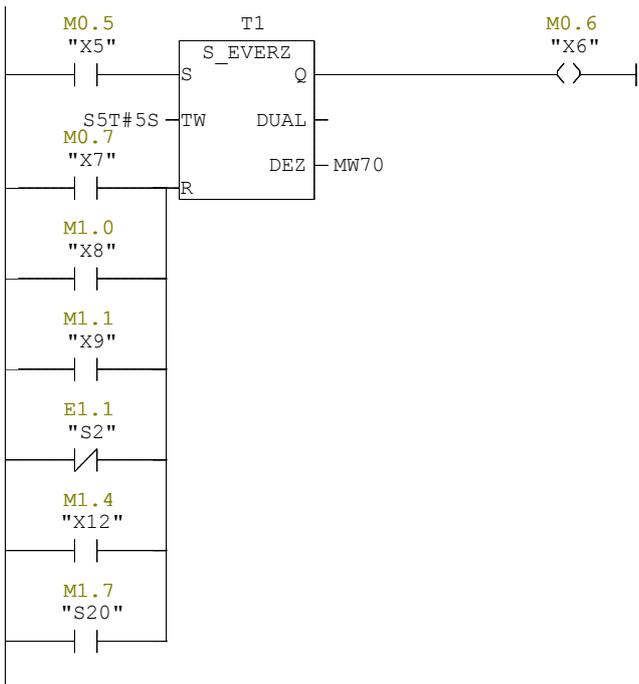
Réseau : 41 C2



Réseau : 42 C3



Réseau : 43 X6 Act & Désact



Réseau : 44



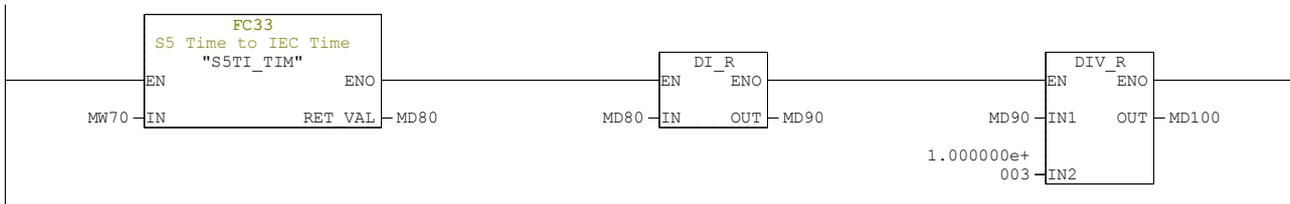
Réseau : 45



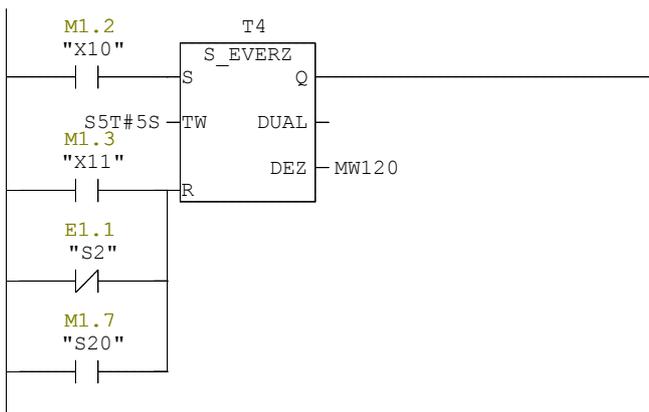
Réseau : 46



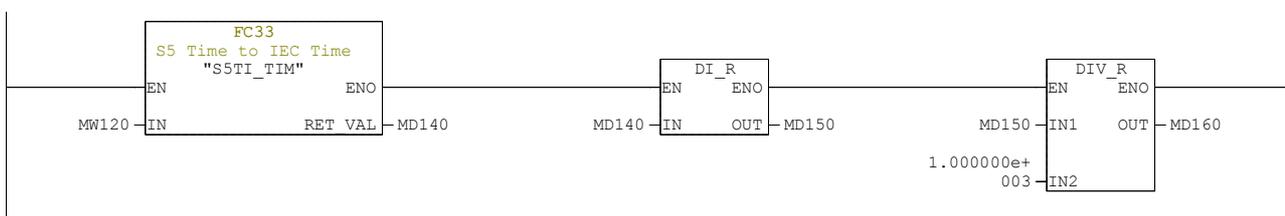
Réseau : 47



Réseau : 48



Réseau : 49



OB1 - <hors ligne>

""

Nom :
Auteur :
Horodatage Code :
Interface :
Longueur (bloc/code /données locales) :

Famille :
Version : 0.1
Version de bloc : 2
 16/04/2023 13:10:13
 15/02/1996 16:51:12
 00122 00008 00020

Nom	Type de données	Adresse	Commentaire
TEMP		0.0	
OB1_EV_CLASS	Byte	0.0	Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
OB1_SCAN_1	Byte	1.0	1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1)
OB1_PRIORITY	Byte	2.0	Priority of OB Execution
OB1_OB_NUMBR	Byte	3.0	1 (Organization block 1, OB1)
OB1_RESERVED_1	Byte	4.0	Reserved for system
OB1_RESERVED_2	Byte	5.0	Reserved for system
OB1_PREV_CYCLE	Int	6.0	Cycle time of previous OB1 scan (milliseconds)
OB1_MIN_CYCLE	Int	8.0	Minimum cycle time of OB1 (milliseconds)
OB1_MAX_CYCLE	Int	10.0	Maximum cycle time of OB1 (milliseconds)
OB1_DATE_TIME	Date_And_Time	12.0	Date and time OB1 started

Bloc : OB1 "Main Program Sweep (Cycle)"

Réseau : 1

FC1
<CALL>—|

Propriétés de la table des mnémoniques

Nom : Mnémoniques
Auteur :
Commentaire :
Date de création : 18/02/2023 11:52:20
Dernière modification : 24/05/2023 13:52:51
Dernier filtre sélectionné : Tous les mnémoniques
Nombre de mnémoniques : 44/44
Dernier tri : Opérande ordre croissant

Etat	Mnémonique	Opérande	Type de données	Commentaire
	K1	A 0.0	BOOL	
	1M1	A 0.1	BOOL	
	2M1	A 0.2	BOOL	
	3M1	A 0.3	BOOL	
	P1	A 1.0	BOOL	
	P2	A 1.1	BOOL	
	LED1	A 1.2	BOOL	
	LED2	A 1.3	BOOL	
	part-av	E 0.0	BOOL	
	B2	E 0.1	BOOL	
	B3	E 0.2	BOOL	
	B4	E 0.3	BOOL	
	1B1	E 0.4	BOOL	
	1B2	E 0.5	BOOL	
	2B1	E 0.6	BOOL	
	2B2	E 0.7	BOOL	
	S1	E 1.0	BOOL	
	S2	E 1.1	BOOL	
	S3	E 1.2	BOOL	
	S4	E 1.3	BOOL	
	S5TI_TIM	FC 33	FC 33	S5 Time to IEC Time
	G7_STD_3	FC 72	FC 72	
	X0	M 0.0	BOOL	
	X1	M 0.1	BOOL	
	X2	M 0.2	BOOL	
	X3	M 0.3	BOOL	
	X4	M 0.4	BOOL	
	X5	M 0.5	BOOL	
	X6	M 0.6	BOOL	
	X7	M 0.7	BOOL	
	X8	M 1.0	BOOL	
	X9	M 1.1	BOOL	
	X10	M 1.2	BOOL	
	X11	M 1.3	BOOL	
	X12	M 1.4	BOOL	
	X13	M 1.5	BOOL	
	S10	M 1.6	BOOL	
	S20	M 1.7	BOOL	
	S30	M 2.0	BOOL	
	S40	M 2.1	BOOL	
	ALARM_SQ	SFC 17	SFC 17	Generate Block-Related Messages with Acknowledgment
	ALARM_S	SFC 18	SFC 18	Generate Permanently Acknowledged Block-Related Messages
	WR_USMSG	SFC 52	SFC 52	Write a User-Defined Diagnostic Event to the Diagnostic Buffer
	TIME_TCK	SFC 64	SFC 64	Read the System Time