

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA
RECHERCHESCIENTIFIQUE

UNIVERSITE M'HAMED BOUGARA-BOUMERDES



Faculté de Technologie

Département Ingénierie des Systèmes Electriques

Mémoire de Master

Présenté par

GUETTARA Smail

BELKHIR Rabah

AGHA Mehdi

Filière : Automatique

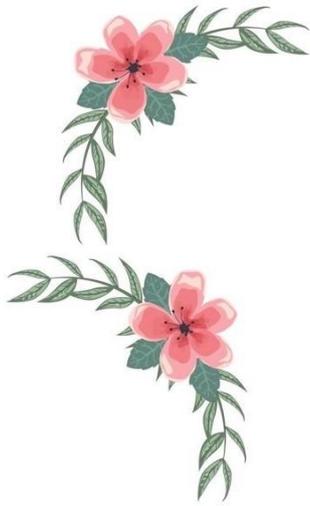
Spécialité : Automatique et Informatique Industrielle

**Etude et programmation d'une chaine de palettisation
de parpaing sur automate TWIDO de Schneider
électrique**

Soutenu le 13/06/2023 devant le jury composé de :

AHRICHE	Imad	UMBB	Président
AKROUM	Hamza	UMBB	Examineur
FELLAG	Sid Ali	UMBB	Rapporteur

Année Universitaire : 2022/2023



Remerciement

Notre remerciement s'adresse en premier lieu à Allah le tout puissant pour la volonté, la santé et la patience qu'il nous a donné durant toutes ces longues années.

Ainsi, nous tenons également à exprimer nos vifs remerciements à notre encadreur Ms. FELLAGUE qui nous a apporté une aide précieuse. Nous lui exprimons notre gratitude pour sa grande disponibilité ainsi que pour sa compréhension et les encouragements qu'elle nous a apportés.

Nos remerciements vont aussi à Ms. MESSAOUDI chef département et à tous nos enseignants qui ont contribué à notre formation et aux membres du jury qui ont accepté de juger notre travail.

Enfin, nous tenons à exprimer notre reconnaissance à tous nos amis et collègues pour leur soutien .





Dédicace

Je dédie ce modeste mémoire :

À ma très chère mère qui m'a toujours soutenu.

Et à mon très cher père aussi.

À ma chère épouse qui m'a toujours encouragé et soutenu

À mon fils MOUADH et mes filles Hadjer et Omaila

À mes frères et mes sœurs.

À toute tous nos amis et collègues.

Smail





Dédicace

Je dédie ce modeste mémoire :

À ma très chère mère qui m'a toujours soutenu.

Et à mon très cher père aussi.

À mes frères et mes sœurs.

À toute tous nos amis et collègues.

Rabah



ملخص

الدراسة المقدمة في هذه الأطروحة كمشروع التخرج لماجستير في التحكم الآلي والحوسبة الصناعية. الغرض من هذا المشروع هو دراسة وتصميم وظائف خط إنتاج كتل البناء، من أجل تحسين مردودية تشغيلها.

في الواقع، يتم التحكم في الآلة حاليًا من خلال نموذج قديم معقد نوعا ما نظرا للأعطال المتكرر التي تحدث يوميا.

وبالتالي، تم اقتراح استبدال النموذج القديم بنظام تحكم و مراقبة جديد و هو عبارة عن المبرمج الآلي: TWIDO-40E / STWDLCAA40DR من شنايدر إلكترونيك لحل هذه المشكلة.

لهذا ، كان من الضروري إجراء دراسة معمقة للتعرف بالتفصيل على المراحل المختلفة للآلة ، ووحدات التحكم والإعدادات من أجل التمكن من إنشاء لوحة التشغيل الخاصة بها ، وتحديد المداخل والمخارج المختلفة لقسم التحكم وبرمجتها والتي ستكون مسؤولة عن إدارة العملية بأكملها وتلبية متطلبات النظام.

الكلمات المفتاحية: نظام آلي ؛ أوتوماتون TWIDO ؛ شنايدر ، سيمنز

ABSTRACT

The study presented in this thesis was carried out as a final project of a MASTER in automatic control and industrial computing.

The purpose of this project is to study and design the functionalities of the concrete block palletizing chain, in order to improve its operation. Indeed, the control of this old model is currently provided by hard-wired logic.

The malfunctions encountered by the hard-wired logic were recurrent. Thus, it was proposed to replace the wired logic with a TWIDO-40E/STWDLCAA40DRF PLC from Schneider Electric in order to solve this problem.

However, a specialized study of the press was necessary to know in the smallest details the different phases of the machine, the control units and the settings in order to be able to establish its operating grafcet, define the different inputs and outputs of the section of control and program the API which will be responsible for managing the entire process and meeting the expectations of the specifications.

Mot clés : système automatisé ;API ;automateTWIDO ; Schneider ;siemens

RESUME

L'étude présentée dans ce mémoire a été réalisée en tant que projet final de MASTER en automatique et informatique industrielle

Le but de ce projet est d'étudier et de concevoir les fonctionnalités de la chaîne de palettisation de parpaing, afin d'améliorer son fonctionnement. En effet, le contrôle de cet ancien modèle est assuré pour le moment par la logique câblée.

Les dysfonctionnements que rencontrait la logique câblée étaient récurrents. Ainsi, il a été proposé de remplacer la logique filaire par un API TWIDO- 40E/STWDLCAA40DRF DE Schneider électrique afin de résoudre ce problème.

Cependant, une étude spécialisée de la chaîne de production s'imposait pour connaître dans les moindres détails les différentes phases de la machine, les unités de contrôle et les réglages afin de pouvoir établir son grafcet de fonctionnement, définir les différentes entrées et sorties de la section de contrôle et programmer l'API qui sera chargé de gérer l'ensemble du processus et de répondre aux attentes du cahier des charges.

Mot clés : système automatisé ;API ;TWIDO ; Schneider ;siemens

Table des matières

Table des matières

Chapitre 01 : présentation de l'automate TWIDO

Introduction générale.....	1
1.1. Introduction	2
1.2. Architecture des automates.....	2
1.2.1. Aspect extérieur.....	3
2. Généralités sur les Systèmes Automatisés de Production.....	4
2.1. Description des différentes parties.....	5
2.1.1. La partie opérative.....	5
2.1.2. La partie commande.....	5
2.1.3. La partie relation.....	6
3. Généralités sur les automates.....	6
4. La structure interne d'un API peut se représenter comme suit.....	7
4.1. Le microprocesseur.....	7
4.2. La zone mémoires	8
5. Les interfaces d'entrées/sorties	8
5.1. Interfaces d'entrées	8
5.2. Interfaces de sorties	9
6. FONCTIONNEMENT DE L'API	9
6.1. Traitement interne	10
6.2. Lecture des entrées	10
6.3. Ecriture des sorties	11
7. PROGRAMMATION.....	11
7.1. Langages de programmation.....	11
7.1.3 Langage littéral structuré.....	11
7.1.4 Langage à contacts.....	12
7.1.5 Blocs Fonctionnels.....	12

Table des matières

7.1.6 Programmation à l'aide du GRAFCET.....	13
8. Critères de choix d'un automate.....	14
9. Présentation de l'API TWIDO.....	15
10. Connexion à des modules de communication.....	15
11. Fonctionnalités avancées pour les bases compactes TWDLC••40DRF.....	16
12. Configuration matérielle maximale pour les bases compactes.....	16
12.1. Modules d'expansion d'E/S TOR.....	17
12.2. Modules d'expansion d'E/S analogiques.....	17
12.3. Modules de communication.....	18
13. Fonctions principales des automates.....	18
14. Présentation des communications.....	18
14.1. Modules d'affichage.....	19
14.2. Description physique d'un module d'affichage.....	19
14.2.1 Présentation de la gamme.....	19
14.2.2. Vue d'ensemble de la gamme XBT N 400.....	20
14.3. Présentation des commandes.....	21
14.4. Liaison Magelis - automate : exploitation.....	21
14.5. Organisation des fonctions Magelis.....	22
15. Architectures d'automatisme.....	22
15.1. Protocoles.....	22
15.2. Connexion point à point.....	23
16. Les autres matérielles de la chaîne de production.....	23
16.1. Les différents capteurs utilisés.....	23
16.2. Les pré-actionneurs.....	25
16.2.1 Les pré-actionneur électriques: les contacteurs.....	25
16.2.2 Les pré-actionneur hydraulique : les distributeurs.....	25
16.2.3 Les protections des moteurs électriques.....	26
16.3. Les actionneurs.....	26
16.3.1 Les vérins hydrauliques.....	26

Table des matières

16.3.2 Les moteurs électriques.....	27
17. Conclusion.....	28

Chapitre 02 : présentation du logiciel TwidoSoft et logiciel XBT L100

1. Introduction à TwidoSoft.....	29
2. Introduction aux langages Twido.....	29
2.1. Langage liste d'instructions.....	30
2.2. Langage schéma à contacts.....	30
3. Lancement du programme.....	31
3.1. Options du Lanceur d'applications.....	31
3.2. Fenêtre principale	32
3.2.1 Barre des étapes/sous-étapes d'une application.....	32
3.2.2. Projet.....	32
3.2.3. Description.....	32
3.2.4. Programme.....	33
3.2.5. Configurer.....	33
3.2.6. Programme.....	33
3.2.7. Mise au point.....	33
3.2.8. Rapport.....	33
3.2.9. Divers.....	33
3.2.10. Préférence.....	34
4. Création et édition d'un programme	34
4.1. Exemple d'application	34
4.2. Barres d'outils pour l'édition de programme	36
4.3. Barre d'outils du programme	36
4.4. Barre d'outils de la palette du langage schéma à contacts	37

Table des matières

4.5. Palette étendue du langage schéma à contacts.....	38
5. Comment éditer un programme	38
6. Tester le programme.....	41
6.1 Panneau de contrôle du Simulateur Twido.....	43
6.2 Transfert et exécution du programme.....	44
6.3 Exécution du programme.....	45
6.4 Animation d'un programme	46
6.5Afficher/masquer les table d'animation.....	46
6.6 Surveillance d'une application.....	47
7. Le logiciel XBT L100• pour l'afficheur XBTN 400	48
7.1. Présentation.....	48
7.2. Installation.....	48
7.3. Fonctionnalités.....	49
8. Principe de fonctionnement	49
9. Concepts pour la création des applications Magelis.....	50
9.1Présentation du logiciel.....	51
9.1.1Barre de menus.....	51
9.1.2 Barre d'outils.....	52
9.1.3 Fenêtre d'information des champs.....	52
9.1.4 Fenêtre de navigation.....	53
10. Création d'une nouvelle application.....	53
11. Présentation du langage de programmation GRAFCET.....	58
12. Structure graphique.....	58
12.1. Etapes	58
12.2. Actions associées à une étape.....	59
12.3 Transitions	59
12.4 La traduction algébrique du langage GEAFCEt en LADDER.....	60

Table des matières

13. Généralisation	61
13.1 Equation d'une initiale	61
13.2 Equation d'une étape non initiale	61
13.3. Equation des actions	61
14. Exemple d'application de la traduction de GRAFCET au LADDER	62
15. Conclusion.....	65

Chapitre3: Etude et programmation de procédés de palatisation

1. Introduction.....	66
2. Description de procédé	66
2. 1 Le descendeur des planches de parpaing.....	66
2.2 pousse planches de parpaing.....	67
2. 3 la pince.....	67
2.2.4 pousse planches latérale.....	67
2. 5 tourne planches.....	67
2.6. Pupitre	68
3. Programmation.....	68
3.1. Programmation avec GRAFCET.....	69
3.2. Programmation avec LADDER	70
3.2.1. L'affectation des entrées.....	70
3.2.2. L'étape: condition électrique.....	70
3.2.3. L'étape : commande démarrage groupe hydraulique.....	70
3.2.4. L'étape : condition hydraulique.....	71
4.2.5. L'étape : mode de fonctionnement.....	71
3.2.6. L'étape : commande marche descenseur et le pousse planche	72
4.2.6.1. Commande entrer pousse planche.....	73
4.2.6.2. Commande marche descenseur.....	73

Table des matières

4.2.6.3. Avance pousse planche.....	73
4.2.7. L'étape : déplacement, serrage et desserrage de la pince.....	74
4.2.7.1. Pince en haut.....	75
4.2.7.2. Pince en bas.....	75
4.2.7.3. Serrage pince.....	75
4.2.7.4. Pince en haut.....	75
4.2.7.5. Avance pince.....	75
4.2.7.6. Pince en bas.....	75
4.2.7.7. Desserrage pince.....	75
4.2.7.8. Pince en haut.....	75
4.2.7.9. Retour pince.....	75
4.2.8. L'étape pousse planche latéral.....	76
4.2.8.1. Pousse planche latéral arrière.....	76
4.2.8.2. Pousse planche latéral avance.....	76
4.2.9. L'étape : tourne planche.....	77
4.2.9.1. Tourne planche arrière.....	77
4.2.10. Mode manuelle.....	78
4.3. Les défauts : le traitement et l'acquittement des défauts.....	79
5. Description des sections du programme dans logiciel TwidoSoft	80
5.1. Mise à l'heur de l'XBT.....	80
5.2. Affectation des entrées.....	80
5.3. Les conditions électriques.....	80
5.4. Les conditions hydrauliques	80
5.5. Commande groupe hydraulique.....	80
5.6. Mode de fonctionnement.....	80
5.7. Mode manuelle.....	81
5.8. Mode automatique.....	81

Table des matières

5.9. Commande des sorties.....	81
5.10. Acquittance des défauts.....	81
5.11. Traitement des défauts.....	81
5.12. Affectation des sorties.....	81
6. Conclusion.....	82
CONCLUSION GENERALE.....	83
BIBLIOGRAPHIE.....	84
ANNEXE.....	85

Liste des figure

Liste des figures

<i>Figure 1.1. Automate compact LOGO-V8 de Siemens</i>	2
<i>Figure 1.2. Automates modulaires</i>	3
<i>Figure 1.3. Structure interne d'un automate programmable industriel (API)</i>	4
<i>Figure 1.4. Schéma de différentes parties d'un système automatisé</i>	5
<i>Figure 1.5. Schéma de communication entre PC et automate</i>	6
<i>Figure 1.6. ARCHITECTURE - CONSTITUTION D'UN API</i>	7
<i>Figure 1.7. Schéma électrique d'interface d'entrée</i>	9
<i>Figure 1.8. Schéma électrique d'interface de sortie</i>	9
<i>Figure 1.9. Le cycle de fonctionnement</i>	10
<i>Figure 1.10. Langage instruction List</i>	11
<i>Figure 1.11. Langage littéral structuré</i>	11
<i>Figure 1.12. Langage ladder diagram</i>	12
<i>Figure 1.13. Fonction bloc diagramme</i>	12
<i>Figure 1.14. Sequential finction chart</i>	13
<i>Figure 1.15. Afficheur XBTN 400</i>	20
<i>Figure 1.16. Liaison entre automate et Magelis</i>	21
<i>Figure 1.17. Organisation des fonctions Magelis</i>	22
<i>Figure 1.18. Connexion point à point</i>	23
<i>Figure 1.19. Les défèrent types des capteurs TOR</i>	23
<i>Figure 1.20. Liaison lumineuse du capteur cellule photo-électrique</i>	24
<i>Figure 1.21. Contacteur électrique</i>	24
<i>Figure 1.22. Distributeur hydraulique</i>	25
<i>Figure 1.23. Protecteur moteur électrique</i>	26
<i>Figure 1.24. Vérin hydraulique</i>	26
<i>Figure 1.25. Moteur asynchrone triphasé avec freinage magnétique</i>	27
<i>Figure 3.1 schéma du déroulement de la palettisation</i>	66

Introduction générale

Introduction générale :

Au cours des dernières décennies, les procédés industriels ont connu un intérêt très particulier, la mondialisation des économies et l'accentuation de la concurrence poussent davantage les entreprises à automatiser leurs processus de production, afin d'assurer leur fonctionnement idéal tout en améliorant les conditions de travail de leur personnel et supprimant les tâches pénibles et laborieuses,

Les automates programmables industriels apportent la solution sur mesure pour les besoins d'adaptation et de flexibilité de nombreuses activités industriels actuelles. Ils sont devenus aujourd'hui les constituants les plus répandus des installations automatisées.

Notre objectif consiste à remplacer une commande avec la logique câblée par un automate programmable afin de commander une palettisation d'une chaîne de production de parpaing et cette commande est assurée entièrement par un programme.

Le présent mémoire est organisé en trois chapitres :

- Le premier chapitre, commence par une introduction générale faite sur les types aillant d'automates programmables et leurs architectures ainsi que leurs constituants toute en détaillant les parties hard et soft.

- Le deuxième chapitre sera consacré à la modélisation de procédé à automatiser et sa programmation. Notre choix s'est tombée sur l'utilisation d'un automate nommée TWIDO. Pour la partie programmation nous allons utiliser TwidoSoft. Ce dernier sera présenté étape par étape.

- Dans le dernier chapitre nous allons parlée de l'application et de la simulation de notre programme.

En fin le travail se termine par une conclusion générale et une liste de références utilisées, ainsi que des annexes qui seront fourni pour enrichir le mémoire.

Chapitre 01

Présentation de l'automate TWIDO

1.1. Introduction :

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

Un automate programmable industriel, ou API (en anglais programmable Logic Controller, PLC), est un dispositif électronique programmable destiné à la commande de processus industriels par un traitement séquentiel. Il envoie des ordres vers les pré-actionneurs (partie opérative ou PO côté actionneur) à partir de données d'entrées (capteurs) (partie commande ou PC côté capteur), de consignes et d'un programme informatique. Là où les systèmes automatisés plus anciens employaient des centaines ou des milliers de relais et de cames, un simple automate suffit. On nomme automaticiens les programmeurs de ces API.

1.2. Architecture des automates [1] :

1.2.1. Aspect extérieur :

Les automates peuvent être de type compact ou modulaire :

A. Automate compact :

Comme les mini-automates « LOGO de Siemens, ZELIO de Schneider, MILENEUM de Crouzet », ils intègrent le processeur, l'alimentation, les entrées et les sorties. Selon les modèles et les fabricants, ils pourront réaliser certaines fonctions supplémentaires (comptage rapide, E/S analogiques ...) et recevoir des extensions en nombre limité. Ces automates, de fonctionnement simple, sont généralement destinés à la commande de petits automatismes (Fig1.1).



Figure 1.1. Automate compact LOGO-V8 de Siemens

B. Automate modulaire :

Les API modulaires sont les plus répandus dans l'industrie. Ils s'adaptent avec un maximum de souplesse à la machine ou au processus à automatiser. Le processeur, l'alimentation et les interfaces d'entrées/sorties résident dans des emplacements séparés (modules) et sont fixés sur un ou plusieurs racks contenant le fond de panier (bus plus connecteurs) (Fig 1.2).



Figure 1.2. Automates modulaires

1.2.2. Aspect intérieur :

Les API comportent quatre principales parties (Figure 1.3) :

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

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

Deux types de mémoire cohabitent :

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

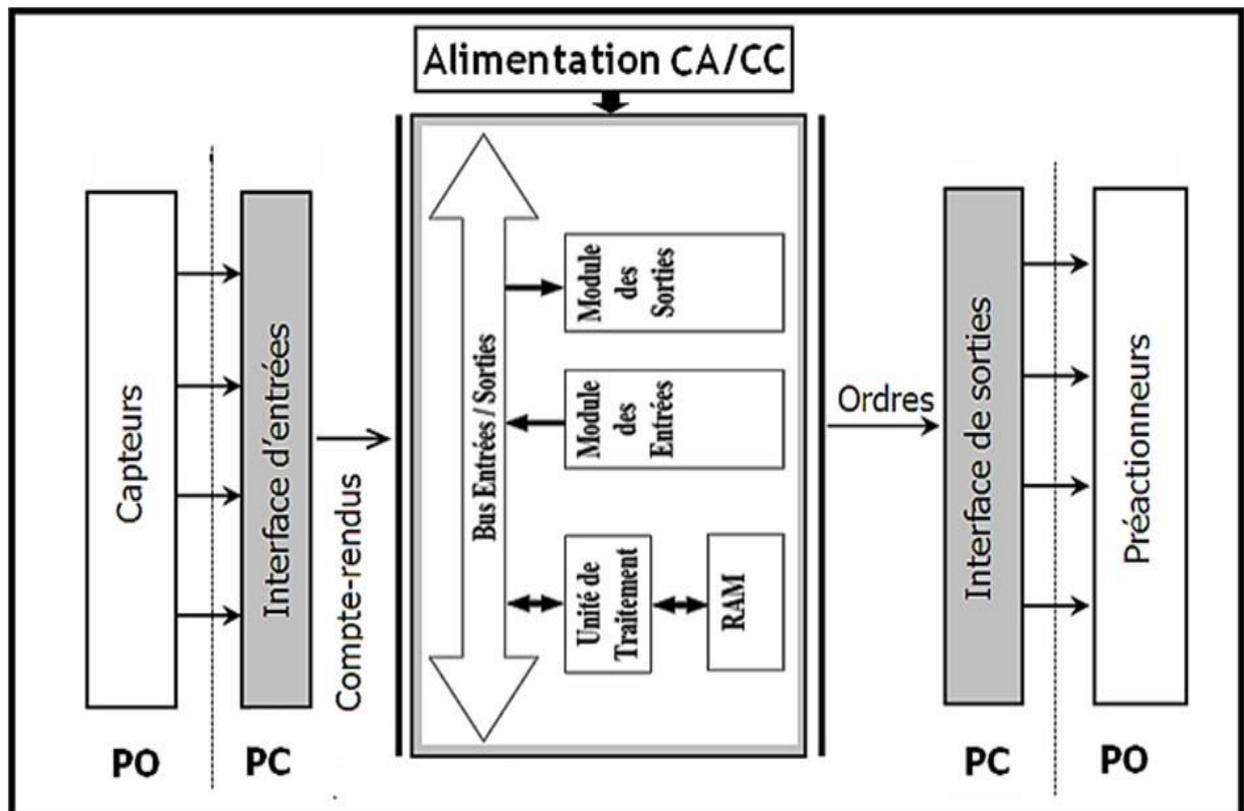


Figure 1.3. Structure interne d'un automate programmable industriel (API)

2. Généralités sur les Systèmes Automatisés de Production [2] :

Ce chapitre permet de comprendre la structure d'un Système Automatisé De Production et de définir les différentes parties de ce système. Un système de production est dit automatisé lorsqu'il peut gérer de manière autonome un cycle de travail pré établi qui se décompose en séquence set/ou en étapes.

Les systèmes automatisés, utilisés dans les secteurs industriels, possèdent une structure de base identique. Ils sont constitués de plusieurs parties plus ou moins complexes reliées entre elles :

- la partie opérative (PO).
- la partie commande (PC) ou système de contrôle/commande (SCC).
- la partie relation (PR) de plus en plus intégrée dans la partie commande.

2.1 Description des différentes parties [9] :

2.1.1 La partie opérative

C'est la partie visible du système, elle comporte les éléments du procédé, c'est à dire:

–des pré-actionneurs (distributeurs, contacteurs) qui reçoivent des ordres de la partie commandent.

–des actionneurs (vérins, moteurs, vannes) qui ont pour rôle d'exécuter ces ordres. Ils transforment l'énergie pneumatique (air comprimé), Hydraulique (huile sous pression) ou électrique en énergie mécanique;

–des capteurs qui informent la partie commandent de l'exécution du travail. Par exemple, on va trouver des capteurs mécaniques, pneumatiques, Electrique sou magnétiques montés sur les vérins. Le rôle des capteurs (ou détecteurs) est donc de contrôler, mesurer, surveiller et informer la PC sur l'évolution du système.

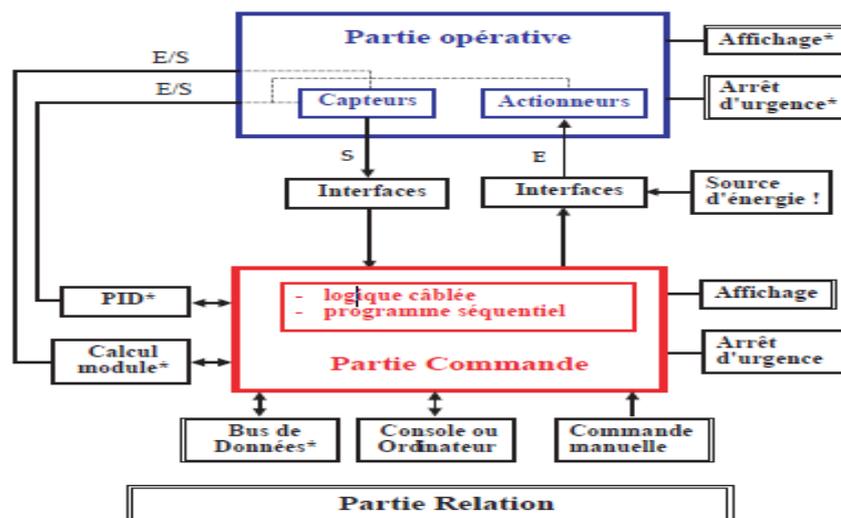


Figure 1.4. Schéma de différentes parties d'un système automatisé

2.1.2. La partie commande :

Ce secteur de l'automatisme gère selon une suite logique le déroulement Ordonné des opérations à réaliser .Il reçoit des informations en provenance des capteurs de la Partie Opérative, et les restitue vers cette même Partie opérative en direction des pré-actionneurs et actionneurs. L'outil de description de la partie commandes appelle le Graphe Fonctionnel de Commande

étape/Transition(GRAFCET).

2.1.3. La partie relation :

Sa complexité dépend de l'importance du système. Elle regroupe les différentes commandes nécessaires de bon fonctionnement du procédé, c'est-à-dire marche/arrêt,

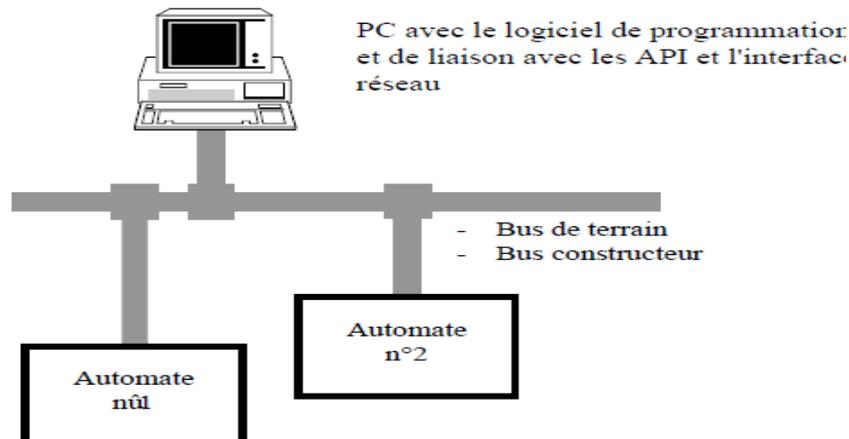


Figure 1.5. Schéma de communication entre PC et automate

Arrêt d'urgence, marche automatique, ...etc. L'outil de description s'appelle

Le guide d'études des modes de Marche set d'Arrêts (GEMMA).

Les outils graphiques, que sont le GRAFCET et le GEMMA, sont utilisés Par les automaticiens et les techniciens de maintenance.

3. Généralités sur les automates [8] :

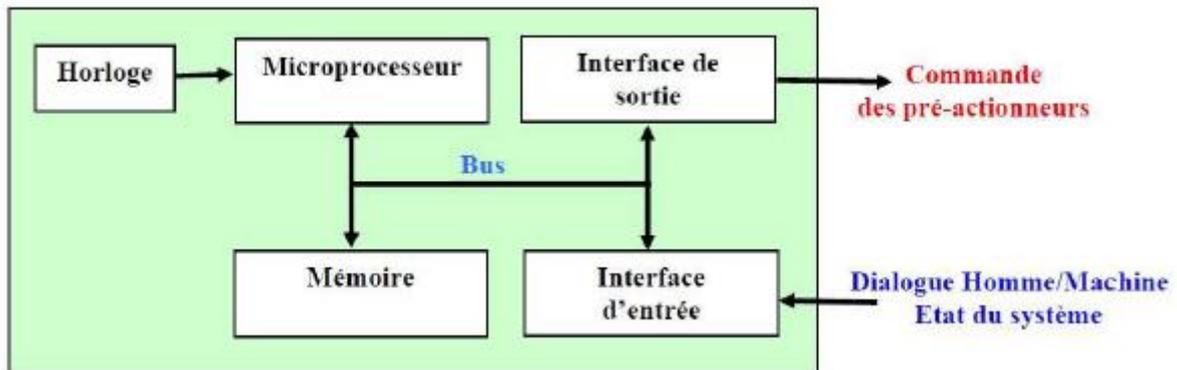
Un automate programmable industriel ou API, est un dispositif électronique programmable destiné à la commande de processus industriels par un traitement séquentiel. Il envoie des ordres vers les pré-actionneurs (partie opérative ou PO côté actionneur) à partir de données d'entrées (capteurs) (partie commande ou PC côté capteur), de consignes et d'un programme informatique. Lorsqu'un automate programmable remplit une fonction de sécurité, il est alors appelé automate programmable de sécurité ou APS.

Il ne faut pas confondre automate programmable et microordinateur, ces derniers peuvent néanmoins commander des appareillages par adjonction de cartes spécifiques dites Entrées/Sorties, mais ils ne sont pas aussi souples d'emploi que les A.P.I. spécialement étudiés.



Figure 1.6. ARCHITECTURE - CONSTITUTION D'UN API

4. La structure interne d'un API peut se représenter comme suit [9] :



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

Un API se compose donc de trois grandes parties :

- Le microprocesseur ;
- La zone mémoire ;
- Les interfaces Entrées/Sorties.

4.1. Le microprocesseur :

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

4.2. La zone mémoires :

a)- La Zone mémoire va permettre :

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

b)-Action possible sur une mémoire :

- ECRIRE pour modifier le contenu d'un programme.
- EFFACER pour faire disparaître les informations qui ne sont plus nécessaires.
- LIRE pour en lire le contenu d'un programme sans le modifier

c)- Technologie des mémoires :

- RAM (Random Acces Memory): mémoire vive dans laquelle on peut lire, écrire et effacer (contient le programme)
- ROM (Read Only Memory): mémoire morte accessible uniquement en lecture.
- EPROM mémoires mortes reprogrammables effacement aux rayons ultra-violets.
- EEPROM mémoires mortes reprogrammables effacement électrique.

Remarque :

La capacité mémoire se donne en mots de 8 BITS (Binary Digits) ou octets.

Exemple: Soit une mémoire de 8 Koctets = $8 \times 1024 \times 8 = 65\,536$ BITS. Cette mémoire peut contenir 65 536 informations binaires.

5. Les interfaces d'entrées/sorties :

Les entrées reçoivent des informations en provenance des éléments de détection (capteurs) et du pupitre opérateur (BP).

Les sorties transmettent des informations aux pré-actionneurs (relais, électrovannes ...) et aux éléments de signalisation (voyants) du pupitre.

5.1. Interfaces d'entrées :

Elles sont destinées à :

- Recevoir l'information en provenance des capteurs
- Traiter le signal en le mettant en forme (éliminer les parasites et en isoler électriquement l'unité de commande de la PO).

L'électronique de traitement de l'API fonctionne sous une tension de 5V.

Le capteur fournissant une tension de 24V, la carte des entrées doit adapter le signal.

Fonctionnement de l'interface d'entrée:

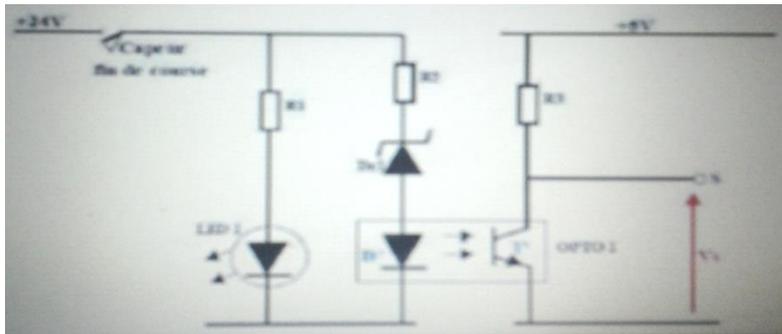


Figure 1.7. Schéma électrique d'interface d'entrée

Lors de la fermeture du capteur ;

- LED1 signale que l'entrée automate est actionnée.
- La led D d'optocoupleur s'éclaire.
- La photo transistor T de l'optocoupleur devient passante.
- La tension $V_s=0V$

Donc lors de l'activation d'une entrée automate, l'interface d'entrée envoie un 0 logique à l'unité de traitement et une 1 logique lors de l'ouverture du contact du capteur (entrée non actionnée).

5.2. Interfaces de sorties :

Elles sont destinées à :

- Commander les pré-actionneurs et éléments des signalisations du système
- Adapter les niveaux de tensions de l'unité de commande à celle de la PO du système en garantissant une isolation galvanique entre ces dernières.

Fonctionnement de l'interface de sortie :

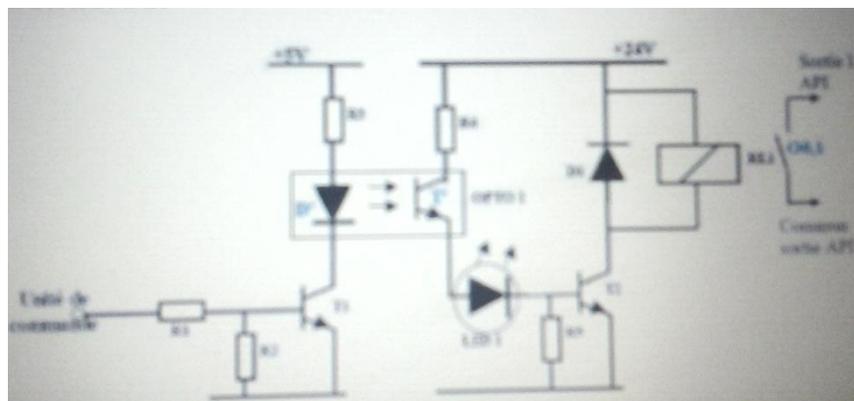


Figure 1.8. Schéma électrique d'interface de sortie

Lors de la commande d'une sortie automate ;

- L'unité de commande envoie un 1 logique (5V).
- T1 devient passant, donc D' s'éclaire.
- La photo transistor T' de l'optocoupleur devient passant.
- LED 1 s'éclaire et nous informe de la commande de la sortie O0,1.
- T2 devient passant.
- La bobine RL1 devient sous tension et commande la fermeture du contact de la sortie O0,1.

Donc pour commander une sortie automate l'unité de commande doit envoyer :

- Une 1 logique pour actionner une sortie API.
- Une 0 logique pour stopper la commande d'une sortie API.

6. FONCTIONNEMENT DE L'API [9] :

Le cycle de fonctionnement de l'API est décrit ci-dessous :

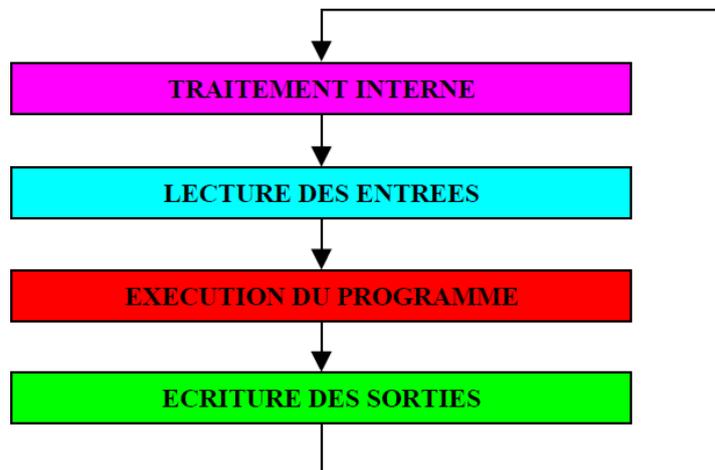


Figure 1.9. Le cycle de fonctionnement

6.1. Traitement interne:

L'automate effectue des opérations de contrôle et met à jour certains paramètres systèmes (détection des passages en RUN / STOP, mises à jour des valeurs de l'horodateur, ...).

6.2. Lecture des entrées :

L'automate lit les entrées (de façon synchrone) et les recopie dans la mémoire image des entrées.

Exécution du programme :

L'automate exécute le programme instruction par instruction et écrit les sorties dans la mémoire image des sorties.

6.3. Ecriture des sorties :

L'automate bascule les différentes sorties (de façon synchrone) aux positions définies dans la mémoire image des sorties.

7. PROGRAMMATION [8] :

7.1. Langages de programmation :

Il existe 4 langages de programmation des automates qui sont normalisés au plan mondial par la norme CEI 61131-3. Chaque automate se programme via une console de programmation propriétaire ou par un ordinateur équipé du logiciel constructeur spécifique.

7.1.2 Liste d'instructions (IL : Instruction List) :

Langage textuel de même nature que l'assembleur (programmation des microcontrôleurs). Très peu utilisé par les automaticiens.

1	U(
2	U(
3	O	"WE11D001 ORD (ADC) S11 ARRET"	\$E16.4
4	O	"ORD HMI Arret WE11D001"	\$M70.1
5)		
6	U	"M WE11D001 MARCHE"	\$M75.0
7	S	"Tag_6"	\$M80.7
8	U	"M WE11D001 Ord-Arret"	\$M16.3
9	R	"Tag_6"	\$M80.7
10	U	"Tag_6"	\$M80.7
11)		
12	U	"M Cadence 5"	\$M300.5
13	UN	"WE11S003 CFC S1 / FERME"	\$E0.4
14	UN	"Défaut WE11S003"	\$M0.0
15	=	"WE11S003 ORD FERM"	\$A0.5

Figure 1.10. langage instruction List

7.1.3 Langage littéral structuré (ST: Structured Text) :

Langage informatique de même nature que le Pascal, il utilise les fonctions comme if ... then ... else ... (si ... alors ... sinon ...) Peu utilisé par les automaticiens.

```
#statVelocity := 0.0; // Correct, cause assignment with
// Correct --> Working with constants
//default value of data type 0.0
IF (ABS(#velocity) < #MAX_VELOCITY) THEN
  #statVelocity := #velocity;
ELSIF (#velocity < 0) THEN
  #statVelocity := -1.0 * #MAX_VELOCITY;
ELSE
  #statVelocity := #MAX_VELOCITY;
END_IF;
// Wrong --> Working with numerical values
IF (ABS(#velocity) < 10.0) THEN
  #statVelocity := #velocity;
ELSIF (#velocity < 0) THEN
  #statVelocity := -10.0;
ELSE
  #statVelocity := 10.0;
END_IF;
```

Figure 1.11. Langage littéral structuré

7.1.4 Langage à contacts (LD : Ladder diagram) :

Langage graphique développé pour les électriciens. Il utilise les symboles tels que : contacts, relais et blocs fonctionnels et s'organise en réseaux (labels). C'est le plus utilisé.

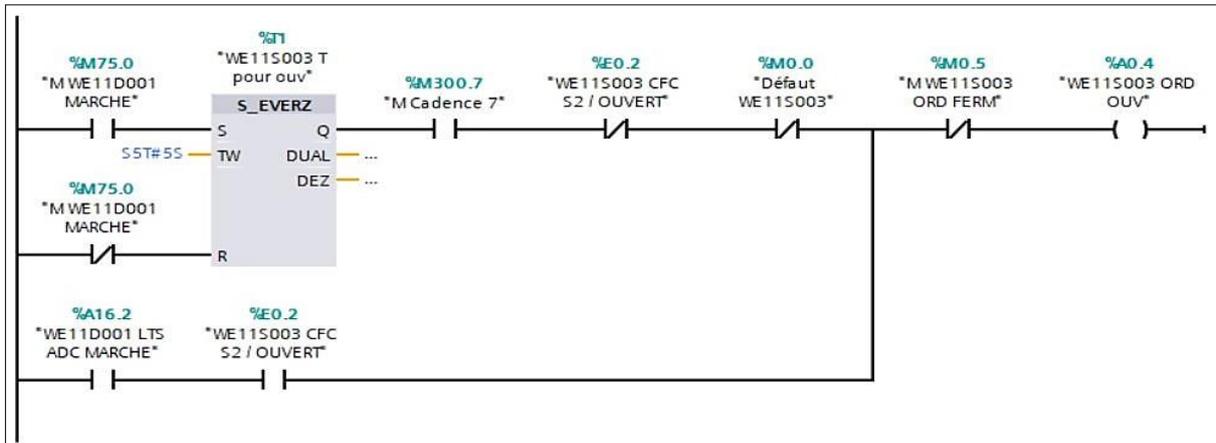


Figure 1.12. Langage ladder diagram

7.1.5 Blocs Fonctionnels (FBD : Function Bloc Diagram) :

Langage graphique où des fonctions sont représentées par des rectangles avec les entrées à gauche et les sorties à droites. Les blocs sont programmés (bibliothèque) ou programmables. Utilisé par les automaticiens.

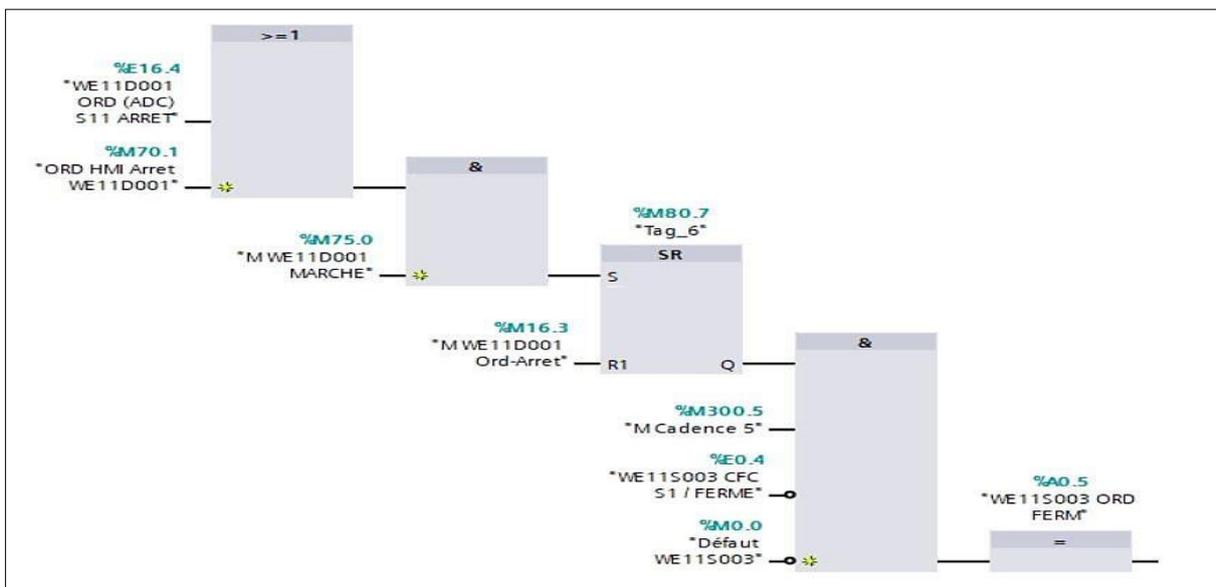


Figure 1.13. Fonction bloc diagramme

7.1.6 Programmation à l'aide du GRAFCET (SFC : Sequential Function Chart) :

Le GRAFCET, langage de spécification, est utilisé par certains constructeurs d'automate (Schneider, Siemens) pour la programmation. Parfois associé à un langage de programmation, il permet une programmation aisée

des systèmes séquentiels tout en facilitant la mise au point des programmes ainsi que le dépannage des systèmes.

On peut également traduire un grafcet en langage en contacts et l'implanter sur tout type d'automate.

Certains logiciels permettent une programmation totale en langage GRAFCET et permettent de s'adapter à la plupart des automates existants (logiciels CADEPA ou AUTOMGEN).

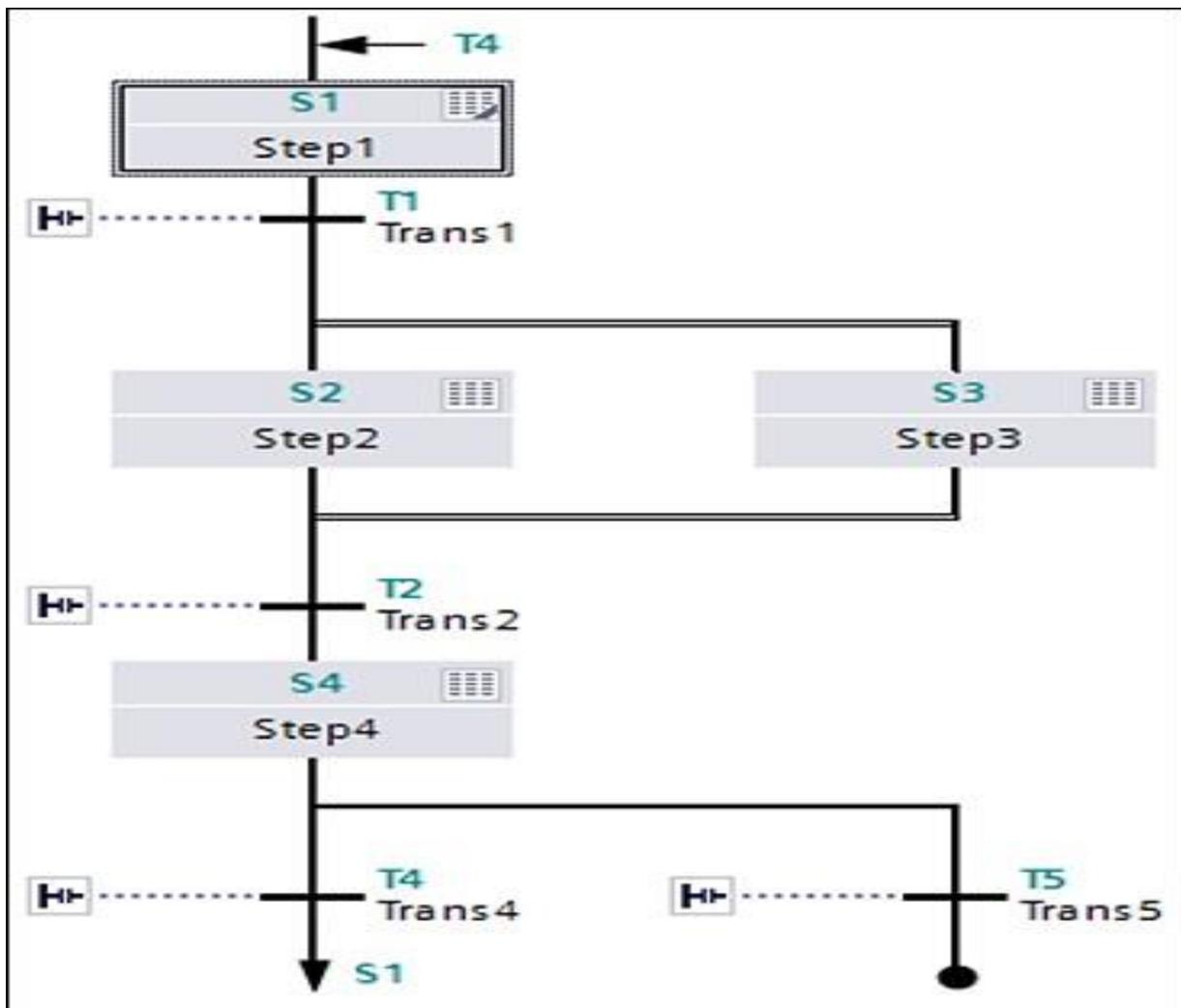


Figure 1.14. Sequential finction chart

8. Le choix d'un automate [9] :

Le choix d'un automate programmable est en premier lieu lié :

Au choix d'une société ou d'un groupe.

Et les contacts commerciaux et expériences vécues sont déjà un point de départ.

En effet, es grandes sociétés privilégieront deux fabricants pour faire jouer la concurrence et pouvoir se retourner en cas de perte de vitesse de l'une d'entre elles.

Le personnel automatique et de maintenance doit toutefois être formé sur ces matériels et une trop grande diversité des matériels peut avoir de graves répercussions.

La possession d'un logiciel de programmation est aussi source d'économies (achat du logiciel également souhaitables. Il faut ensuite quantifier les besoins :

Nombre d'entrées/sorties : le nombre de cartes peut avoir une incidence sur le nombre de racks dès que le nombre d'entrées/sorties nécessaires devient élevé.

Type de processeur : la taille mémoire, la vitesse de traitement et les fonctions spéciales offertes par le processeur permettront le choix dans la gamme souvent très étendue.

Spécificité des modules E/S : par exemple avec système de diagnostic et alarmes de processus paramétrables (selon besoin).

Fonctions de communication : l'automate doit pouvoir communiquer avec les autres systèmes de commande (API, supervision...) et offrir des possibilités de communication avec des standards normalisés (PROFINET, Profibus,...).

Dans notre projet, l'automate envisagé pour l'automatisation de la chaîne de production de parpaing a été imposé par le client.

9. Présentation de l'API TWIDO [3] :

Compact 40E/STWDLCAA40DRF, Type de voles 16 Entrées 24Sorties ,24Vcc Relais X14, Transistors X2, alimentation 100/240Vca

Twido est un Automates programmables, une nouvelle gamme de produits de Schneider Electric, il est très compétitif avec des fonctions puissantes et faible encombrement, facile à utiliser et extrêmement flexible.

En option, un petit écran est capable de statut de contrôleur d'affichage et version de l'OS afficher et modifier des données (bits, mots, les valeurs, les prestes...), possibilité de modifier les données peut être désactivée par l'application configurer les liaisons série protocole et adresse, accès date et l'heure.

Il existe deux modèles de bases Twido:

- Bases compactes
- Bases modulaires

Les bases compactes sont disponibles avec 10, 16, 24 ou 40E/S.

Les bases modulaires sont disponibles avec 20 ou 40E/S.

Il est possible d'ajouter des E/S supplémentaires aux bases à l'aide de modules

D'expansion d'E/S. Il s'agit des modules suivants:

- 15modules d'expansion de type E/S TOR ou relais;
- 10modules d'expansion de type E/S analogique.

Il est également possible d'ajouter plusieurs options aux bases automates, comme le montre le tableau de la section Options de bases ci-après:

Outres ces options, vous pouvez également ajouter celles mentionnées ci-après:

- câbles de programmation ;
- câbles d'E/S TOR;
- systèmes pré-câblés Telefast avec interfaces d'E/S.

10. Connexion à des modules de communication

La connexion à un module interface de bus AS-Interface permet également de gérer jusqu'à 62équipements esclaves. Utilisez le module suivant:

- module maître interface de bus AS-Interface V2: TWDNOI10M3.

Chapitre 01 : Systèmes automatisés et présentation de l'automate TWIDO

Les bases compactes 24E/S et 40E/S et toutes les bases modulaires peuvent se connecter à un module interface de bus de terrain CANopen. Le module maître CANopen peut gérer jusqu'à 16 équipements esclaves CANopen (sans dépasser 16 Transmit-PDOs (TPDO) et 16 Receive-PDOs (RPDO)).

Utilisez le module suivant :

-module maître interface de bus de terrain CANopen: TWDNCO1M.

11. Fonctionnalités avancées pour les bases compactes TWDLCAE40DRF :

Des fonctionnalités intégrées avancées sont proposées sur les bases compactes

TWDLCAE40DRF:

port réseau Ethernet intégré 100Base-TX: TWDLCAE40DRF et TWDLCAE40DRF uniquement; horodateur (RTC) intégré: TWDLCAE40DRF; quatrième compteur rapide (FC): TWDLCAE40DRF; prise en charge de la pile externe: TWDLCAE40DRF.

12. Configuration matérielle maximale pour les bases compactes : [3]

Le tableau suivant répertorie le nombre maximal d'éléments de configuration

Élément de la base	Base compacte
TWD...	LCAA40DRF
Ports série	2
Port Ethernet	1
Emplacements cartouche	1
Taille maximale application/ sauvegarde (Ko)	64
Cartouche mémoire facultative (Ko)	32 ou 64 ²
Cartouche RTC facultative	Horodateur intégré ³
Afficheur facultatif	oui
2ème port série facultatif	oui
Module interface Ethernet facultatif	oui

Note :

1. Une base compacte peut être dotée d'une cartouche mémoire ou d'une cartouche RTC.
2. Cartouche mémoire uniquement, car RTC intégré.
3. Les bases compactes TWDLC••40DRF sont équipées d'un RTC intégré. Par conséquent, il est impossible d'ajouter une cartouche RTC sur ces automates ; seule une cartouche mémoire peut être insérée.

12.1. Modules d'expansion d'E/S TOR :

Élément de la base	Base compacte
TWD...	LCAA40DRF
Entrées TOR standard	24
Sorties TOR standard	16 (14 sorties à relais + 2 sorties transistor)
Modules d'expansion d'E/S max. (TOR ou analogiques)	7
Entrées TOR max. (automate E/S + exp E/S)	$24+(7 \times 32)=248$
Sorties TOR max. (automate E/S + exp E/S)	$16+(7 \times 32)=240$
E/S TOR max. (automate E/S + exp E/S)	$40+(7 \times 32)=264$
Sorties à relais max.	14 sur base + 96 sur expansion
Potentiomètre	2

12.2. Modules d'expansion d'E/S analogiques :

Élément de la base	Base compacte
TWD...	LCAA40DRF
Entrées analogiques intégrées	0
E/S analogiques max. (automate E/S + exp E/S)	56 entrées/14 sorties

12.3. Modules de communication :

Elément de la base	Base compacte
TWD...	LCAA40DRF
Modules interface de bus AS-Interface max.	2
E/S max. avec modules AS-Interface (7 E/S par esclave)	40+(2x62x7)=908
Modules d'interface de bus de terrain CANopen max.	1
T/R-PDO max. avec équipement CANopen	16 TPDO 16 RPDO
Automates distants	7

13. Fonctions principales des automates :

Par défaut, toutes les E/S des bases sont configurées en tant qu'E/S TOR. Cependant, certaines E/S dédiées peuvent être affectées à des tâches spécifiques pendant la configuration comme:

- Entrée RUN/STOP
- Entrées à mémorisation d'état
- Compteurs rapides (FC):
- Compteurs/décompteurs simples: 5kHz (monophasés)
- Compteurs très rapides (VFC): compteurs/décompteurs - 20kHz (bi-phases)
- Sortie état de l'automate
- PWM (Pulse Width Modulation, modulation de largeur)
- Sortie générateur d'impulsions (PLS)

Les automates Twido sont programmés à l'aide du logiciel TwidoSuite, qui permet d'utiliser les fonctions PID et PID Auto-tuning accessibles sur certains automates

14. Présentation des communications [4] :

Les automates Twido sont dotés d'un port série, ou d'un second port série facultatif, utilisé pour des services en temps réel ou de gestion système.

Quatre types de communication sont utilisables avec les automates Twido:

- Connexion du bus AS-Interface
- Connexion au bus de terrain CANopen

-Connexion par réseau Ethernet

-Connexion Modem

Les services temps réel fournissent des fonctions de distribution de données afin d'échanger des données avec les équipements d'E/S et des fonctions de messagerie pour communiquer vers les équipements externes. Les services de gestion système gèrent et configurent la base via le logiciel TwidoSuite. L'un des ports série est utilisé pour ces services, mais seul le port série 1 est dédié aux communications avec TwidoSuite.

Pour fournir ces services, trois protocoles sont disponibles sur chacune des bases:

-Liaison distante

-Modbus

-ASCII

En outre, les bases compactes TWDLCAE40DRF et TWDLCDE40DRF proposent un port de communication Ethernet RJ45 intégré permettant de réaliser des communications en temps réel et des tâches de gestion système via le réseau. La communication Ethernet implémente le protocole suivant:

-Modbus TCP/IP

14.1. Modules d'affichage

L'afficheur est un module facultatif qui peut être ajouté à toutes les bases compactes. Il s'installe sur une base compacte comme un module d'affichage

(TWDXCPODC).

L'afficheur fournit les services suivants :

-informations sur l'état de l'automate

-contrôle de la base par l'utilisateur

-surveillance et réglage des objets donnés d'application par l'utilisateur

L'afficheur dispose de deux états :

-affichage de l'état - visualise les données

-modification de l'état - permet à l'utilisateur de modifier les données

14.2. Description physique d'un module d'affichage [4] :

14.2.1 Présentation de la gamme

Les terminaux compacts Magelis ont pour fonctions principales de :

- Visualiser des données issues de l'automatisme.

Chapitre 01 : Systèmes automatisés et présentation de l'automate TWIDO

- Modifier des paramètres de l'automatisme.
- Commander un procédé à l'aide de commandes Tout ou Rien.

Ces terminaux communiquent avec les automates par une liaison série intégrée en mode point à point (ou multipoints, multi-drop pour les XBT N401, NU400, R410 et R411). Les protocoles de communication sont ceux des automates de Schneider Electric (Uni-Telway, Modbus maître et esclave).

14.2.2. Vue d'ensemble de la gamme XBT N 400 :

GAMME	CARACTERISTIQUES	AFFICHAGE	MODE	FACE AVANT
XBT NU400	<ul style="list-style-type: none">- Alimentation 24V DC- Application préchargée⁽¹⁾- Liaison mode point à point, multipoints,- Protocole Modbus maître,- Rétro-éclairage vert.	4X20 matriciel	Saisie ou Commande	

Les terminaux comportent :

En face avant

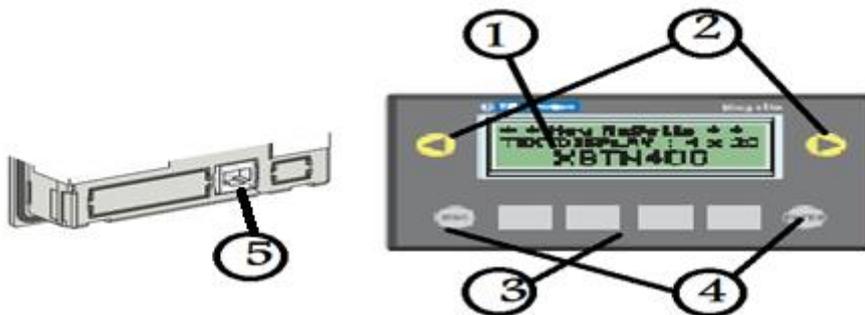


Figure 1.15. Afficheur XBTN 400

1-Ecran LCD Rétro éclairé

2- Touches de liens ou de commande contextuelle : Mode commande

Touches de liens ou de commande contextuelle :

- Changer de page dans un menu
- Visualiser les alarmes courantes
- Activer la fonction associée à un lien fonctionnel:
- Commande impulsionnelle
- Commande "pousser/pousser"
- Ecriture variable

3-Touches de service : Touches de fonction statiques

- Accès à une page
- Commande impulsionnelle
- Commande "pousser/pousser"

4-Touches de fonctions / de service (Selon configuration du XBT N) : Touches de service

- Annuler une saisie ou une action
- retourner à la page précédente
- Valider un choix ou une saisie
- Acquitter une alarme

En face arrière :

5-RJ45 : Liaison série + alimentation automate

14.3. Présentation des commandes :

Les terminaux disposent, en face avant, de touches de fonctions et/ou de touches de service.

Les terminaux XBT R disposent des deux possibilités, par contre, pour les terminaux XBT N, cela dépend de leur personnalisation en mode "commande" ou "saisie".

L'ergonomie des terminaux est la suivante :

- XBT N :
- 4 touches de service (mode saisie) personnalisables et configurables en touches de fonctions (mode commande)
- 2 touches de service non configurables
- 2 touches de liens ou de commande contextuelle

14.4. Liaison Magelis - automate : exploitation :

Mode d'alimentation : alimentation par automate

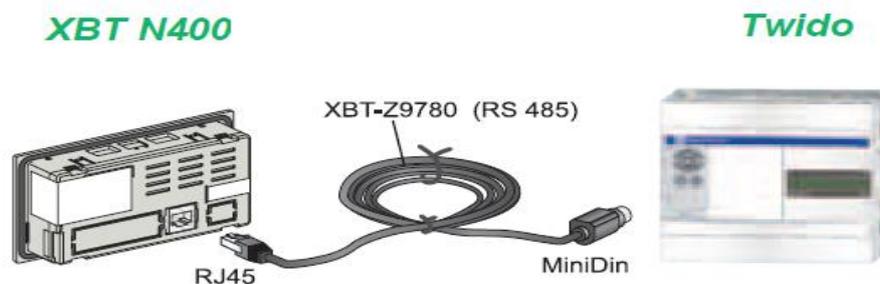


Figure 1.16. Liaison entre automate et Magelis

Figure-16- liaison Magelis- automate en fonction du protocole utilisé, l'affichage de "XBT N400" en lieu des valeurs ou la persistance de la connexion indique un problème de communication. Cela peut provenir du câble utilisé.

14.5. Organisation des fonctions Magelis [4] :

Les terminaux Magelis (ou XBT) disposent d'un certain nombre de fonctions.

L'organigramme ci-après présente ces différentes fonctions

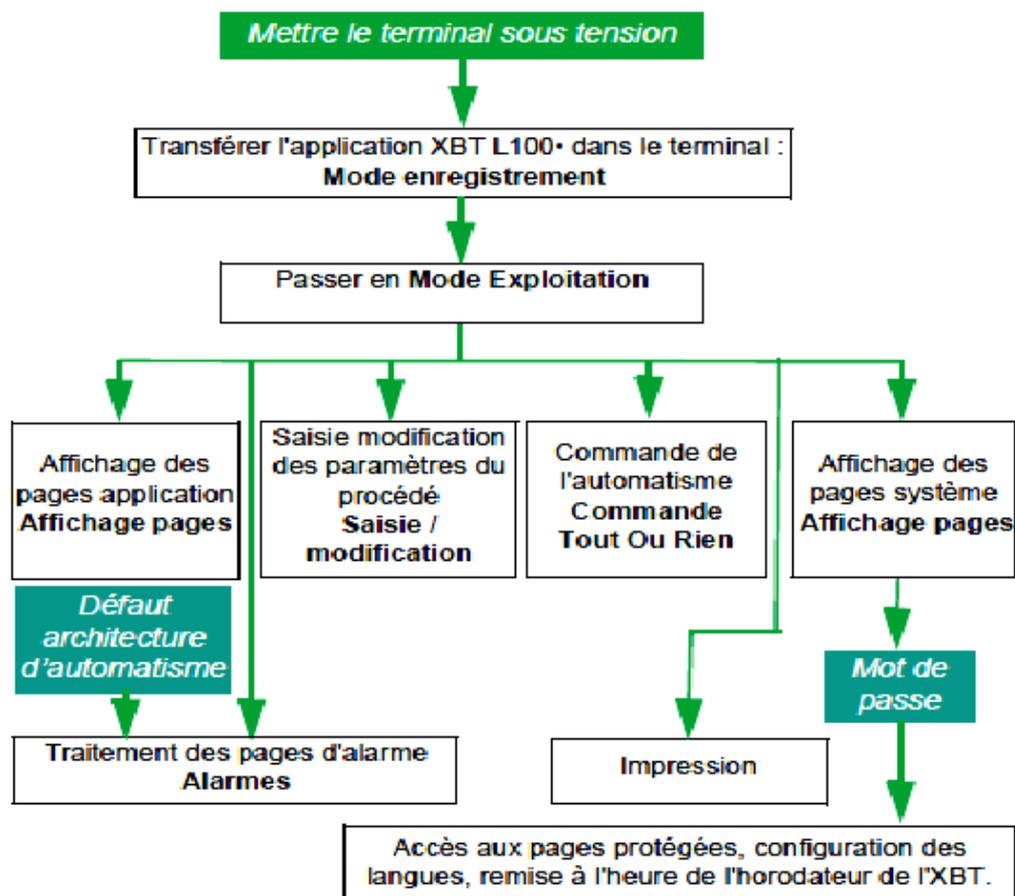


Figure 1.17. Organisation des fonctions Magelis

15. Architectures d'automatisme

15.1. Protocoles

La communication entre le terminal et les équipements connectés est réalisée par l'intermédiaire d'un protocole de communication qui est choisi lors de la création de l'application sous XBT L100• :

Les protocoles disponibles pour la gamme permettent de communiquer avec la gamme des automates (API) Schneider, des équipements spécifiques (variateurs de vitesse), etc....

Ces protocoles sont Uni-Telway et Modbus (maître et esclave).

Il existe plusieurs types d'architectures permettant de relier un terminal à plusieurs équipements ou plusieurs terminaux à un seul équipement.

15.2. Connexion point à point

Un terminal relié à un équipement.



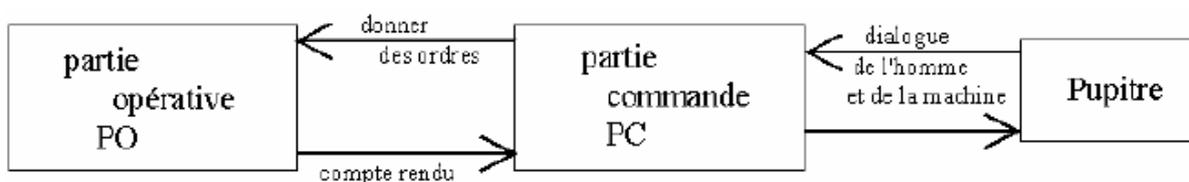
Figure 1.18. Connexion point à point

16. Les autres matérielles de la chaîne de production :

16.1. Les différents capteurs utilisés :

Tout automatisme comporte des éléments qui renseignent la partie commande de l'état de la machine automatisée.

Pour réaliser la fonction « détecter » plusieurs technologies existent.



Et dans notre chaîne de production les Capteurs utilisés sont de type de détection avec contact, capteurs à commande mécanique, ce sont les capteurs de positions TOR qui donnent une information dite "Tout Ou Rien c'est à dire qu'ils sont dans un état logique 1 ou 0".

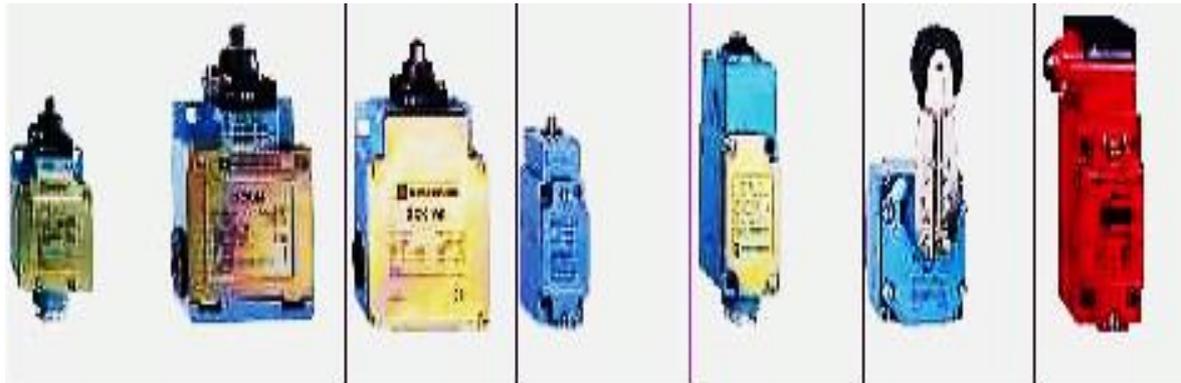


Figure 1.19. Les différents types des capteur TOR

Et ce type de capteur sont installés pour détecter les débuts et les fins de courses pour :

- le pousse planche ;
- monte et descend de la pince et aussi la trajectoire de gauche à droite ;
- le pousse planche latérale;
- le tourne planche ;

Il y'a aussi un capteur à cellule photo-électrique C'est un capteur qui comporte 2 parties:

- un émetteur de lumière,
- un récepteur de lumière

La méthode de Reflex, un boîtier émetteur, récepteur et un miroir (catadioptré), pour des distances moyennes.

Le capteur donne un signal ouverture ou fermeture d'un circuit, quand la liaison lumineuse entre l'émetteur et le récepteur est rompue.

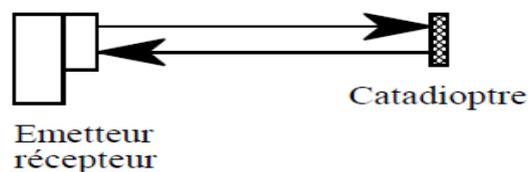


Figure 1.20. Liaison lumineuse du capteur cellule photo-électrique

Il est installé pour détecter la présence du produit sous la pince.

16.2. Les pré-actionneurs [9] :

16.2.1 Les prés-actionneur électriques: les contacteurs:



Figure 1.21. Contacteur électrique

Le contacteur est un appareil de commande capable d'établir ou d'interrompre le passage de l'énergie électrique.

Il a donc un pouvoir de coupure. Ce type de constituant pré actionneur puisqu'il se trouve avant L'actionneur dans la chaine des énergies. Ce dernier peut être commandé à distance au moyen de contacts actionnés manuellement (bouton poussoir) ou automatiquement (asservi à une grandeur physique : pression, température, vitesse, etc.).

Il a un pouvoir de coupure de 10 KA (kilo ampère : 10000 A) permettra de couper un circuit où circule 10000 A max et de supprimer l'arc électrique qui résulte de l'ouverture de ce dernier.

Les principales défaillances d'un contacteur sont :

- le collage des contacts
- une connexion défectueuse
- une bobine grillée
- un ressort de rappel cassé

16.2.2 Les prés-actionneur hydraulique : les distributeurs :



Figure 1.22. Distributeur hydraulique

Le distributeur est constitué de 3 parties : le corps, le tiroir, les éléments de commande.

Les éléments de commande agissent sur le tiroir et le déplacent vers la droite ou vers la gauche.

En se déplaçant, le tiroir met en communication les orifices.

Donc le pré-actionneur hydraulique, les distributeurs sert à aiguiller le débit vers l'une ou l'autre partie du circuit, autoriser ou bloquer le passage du débit.

16.2.3 Les protections des moteurs électriques : GV2ME07 [9] :

La fonction "protection contre les courts circuits" a pour but de protéger le moteur et le câble reliant celui-ci au départ-moteur contre les très fortes surintensités.

Vu les très grands courants rencontrés, le temps d'intervention de l'organe de protection doit être très court, de l'ordre de quelques millisecondes. Deux technologies existent: la protection par fusibles et la protection par disjoncteur. Les fusibles ne présentant que très peu d'avantages

pour la protection d'un moteur, un disjoncteur protège un moteur en interrompant de manière très rapide le courant de court-circuit qui risque de le traverser.

On utilise le terme "pouvoir de coupure" pour caractériser le courant maximal présumé que le disjoncteur peut couper.



Figure 1.23. Protecteur moteur électrique

16.3. Les actionneurs :

Pour les actionneurs on a des vérins hydrauliques, est des moteurs électriques.

16.3.1 Les vérins hydrauliques :

Le vérin hydraulique est un appareil qui transforme une énergie hydraulique en énergie mécanique animée d'un mouvement rectiligne.

Et les types des vérins hydrauliques utilisés dans notre chaîne de production sont les vérins à double effet c'est un récepteur linéaire dont le piston est en mesure de recevoir le débit en provenance de la pompe sur chacune de ses faces.



Figure 1.24. Vérin hydraulique

Les deux surfaces réceptrices du piston étant inégales, les forces développées ainsi que les vitesses, en rentrée et sortie de tige, sont différentes.

Et ce type de vérins sont installer pour :

- le pousse planche.
- mante et décent de la pince.
- pousse planche latérale.
- tourne planche.

16.3.2 Les moteurs électriques:

Les moteurs électriques utilisés dans cette chaine de production sont des moteurs asynchrone triphasé avec freinage magnétique et de grande puissance, qui sont installé pour :

- le descenseur des planches
- pour le déplacement de la pince de gauche à adroit



Figure 1.25. Moteur asynchrone triphasé avec freinage magnétique

17. Conclusion :

Dans ce chapitre, on a présenté un aperçu sur les systèmes automatisés et les API., nous avons exposé le choix et les caractéristiques de l'API proposé TWIDO : TWDLCAA40DRF, 24 entrées, 16 sorties et. La constitution modulaire, la facilité de réalisation d'architecture décentralisée, la facilité d'emploi font du système TWDLCAA40DRF, la solution idéale pour notre tâche d'automatisation.

Chapitre 02

Présentation du logiciel TwidoSoft et logiciel XBT L100

1. Introduction à TwidoSoft

Le logiciel de programmation TwidoSuite est un programme 32 bits peut s'exécuter sur différents systèmes d'exploitation Windows 2000/XP/Vista. Le présent chapitre décrit la configuration informatique requise pour installer et exécuter le logiciel. Vous y trouverez également des informations sur le lancement du logiciel TwidoSoft est un environnement de développement graphique permettant de créer, configurer et gérer des applications pour automates programmables Twido.

TwidoSoft vous permet de créer des programmes avec différents types de langage (Langage liste d'instructions, Langage Grafcet, Langage schéma à contacts), puis de transférer l'application en vue de son exécution sur un automate.

2. Introduction aux langages Twido : [7]

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

Les langages suivants peuvent être utilisés pour créer des programmes d'automates Twido :

-Langage liste d'instructions :

Un programme liste d'instructions est constitué d'une série d'expressions logiques, rédigées sous la forme d'une séquence d'instructions booléennes.

-Langage schéma à contacts :

Un schéma à contacts est une représentation graphique d'une expression Logique

-Langage Grafcet :

Le langage Grafcet est constitué d'une succession d'étapes et de transitions.

Twido comprend les instructions liste Grafcet, mais pas les objets de représentation graphique Grafcet.

Les opérations de création et d'édition de programmes Twido à l'aide de ces langages de programmation peuvent être réalisées depuis un ordinateur personnel (PC).

Une fonctionnalité de réversibilité liste d'instructions / schéma à contacts vous permet de convertir un programme en langage liste d'instructions dans le langage schéma à contacts, et vice-versa.

2.1. Langage liste d'instructions

Un programme rédigé en langage liste d'instructions consiste en une série d'instructions exécutées de manière séquentielle par l'automate. Vous trouverez ci-dessous un exemple de programme en langage liste d'instructions.

0BLK%C8

1LDF%I0.1

2R

3LD%I0.2

4AND%M0

5CU

6OUT_BLK

7LDD

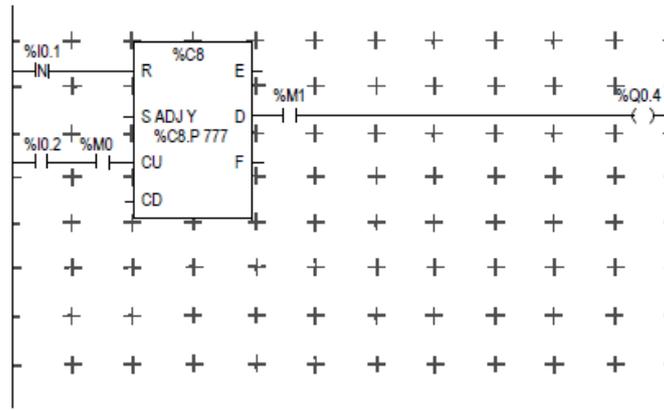
8AND%M1

9ST%Q0.4

10END_BLK

2.2. Langage schéma à contacts

Les schémas à contacts utilisent la même représentation graphique que celle des circuits de relais en logique programmée. Dans ces schémas, les éléments graphiques, tels que des bobines, des contacts et des blocs représentent les instructions du programme. Ci-dessous un exemple de schéma à contacts.



3. Lancement du programme :

Vous pouvez démarrer le programme via le **Lanceur d'applications TwidoSuite** : en suivant le chemin **Démarrer** → **Programmes** → **Schneider Electric** → **TwidoSuite** → **Lanceur d'applications** ; ou à partir du raccourci bureau du Lanceur d'applications installé avec le programme :

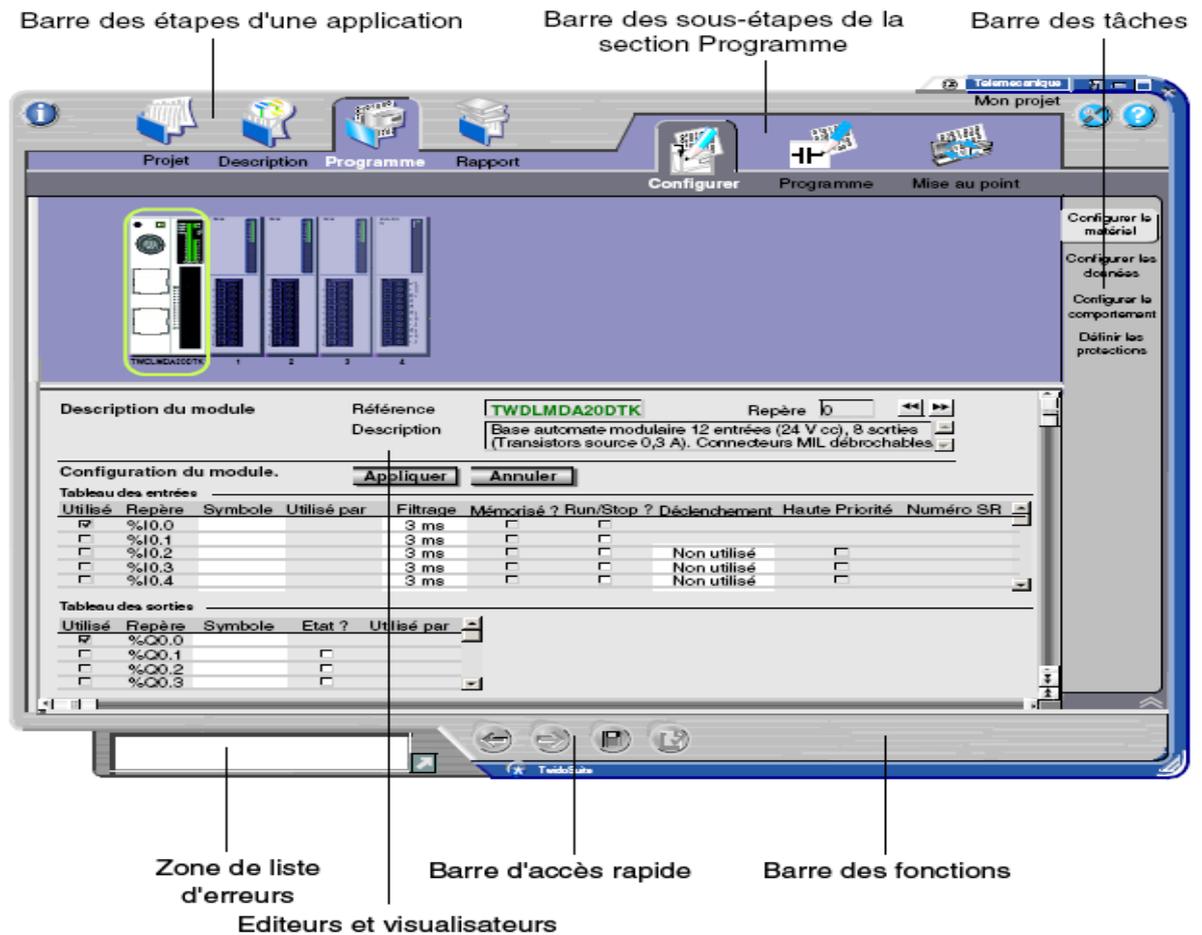


3.1. Options du Lanceur d'applications

La figure suivante présente un exemple d'écran du Lanceur d'applications TwidoSuite Application pour la paire de langues anglais-français :



3.2. Fenêtre principale :



3.2.1 Barre des étapes/sous-étapes d'une application

3.2.2. Projet :



Il s'agit de la première à exécuter lors de la création/ l'ouverture d'une application, le nom de l'application ouverte apparaît dans le coin supérieur droit de la barre des étapes/ sous-étapes de l'application.

3.2.3. Description :



Il s'agit de la deuxième étape à exécuter lors de la construction d'une application d'automatisation Twido. L'étape description vous permet de sélectionner des ressources matérielles dans le catalogue et de décrire l'interconnexion des périphériques via des réseaux.

3.2.4. Programme :



Il s'agit de la troisième étape à exécuter lors de la configuration d'une application Twido. Elle permet d'accéder aux 3 sous-étapes suivantes :

3.2.5. Configurer :

Cette sous-étapes vous permet de visualiser/ paramètre :



-la configuration matérielle de la base automate, des modules d'expansion, des modules de communication et des options matérielles,

-la configuration des objets de données pour les objets simples, les blocs fonction, les objets d'E/S et les objets professionnels,

-le comportement de l'automate

-le programme d'application/ les protections par mot de passe

3.2.6. Programme :

Ouvre l'éditeur de programme par défaut (ladder ou list) et permet d'accéder à l'éditeur de symboles



3.2.7. Mise au point :

Permet de gérer la connexion entre TwidoSuite et l'automate et d'accéder à l'éditeur de tables d'animation et aux taches de surveillance.



3.2.8. Rapport :

Il s'agit de dernière étape exécuté lors de la configuration d'une application. Cette étape ouvre la fenêtre gestionnaire de rapport dans laquelle vous pouvez sélectionner les composants du rapport, configurer les options d'impression et exporter le rapport de l'application au format HTML.



3.2.9. Divers :

Permet d'accéder à l'aide générale et aux informations contenus dans a propos de TwidoSuite.

3.2.10. Préférence :

Ouvre la fenêtre préférences de l'utilisateur.

Affiche les pages d'aide rapide relative aux tâches.



4. Création et édition d'un programme :



Cette section vous explique comment créer et éditer un programme avec TwidoSuite.

Vous devez ouvrir le lanceur d'application pour démarrer TwidoSuite. Cliquez sur

l'icône  située sur le bureau de votre PC

La boîte de dialogue lanceur d'application s'affiche, comme illustré ci-dessous :



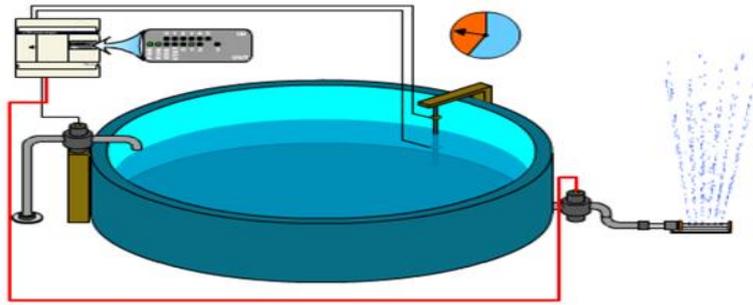
Le lanceur d'application TwidoSuite vous permet d'accéder aux modes suivants :

- mode de programmation TwidoSuite
- mode de programmation de surveillance TwidoSuite
- utilitaire de mise à jour du firmware de l'automate

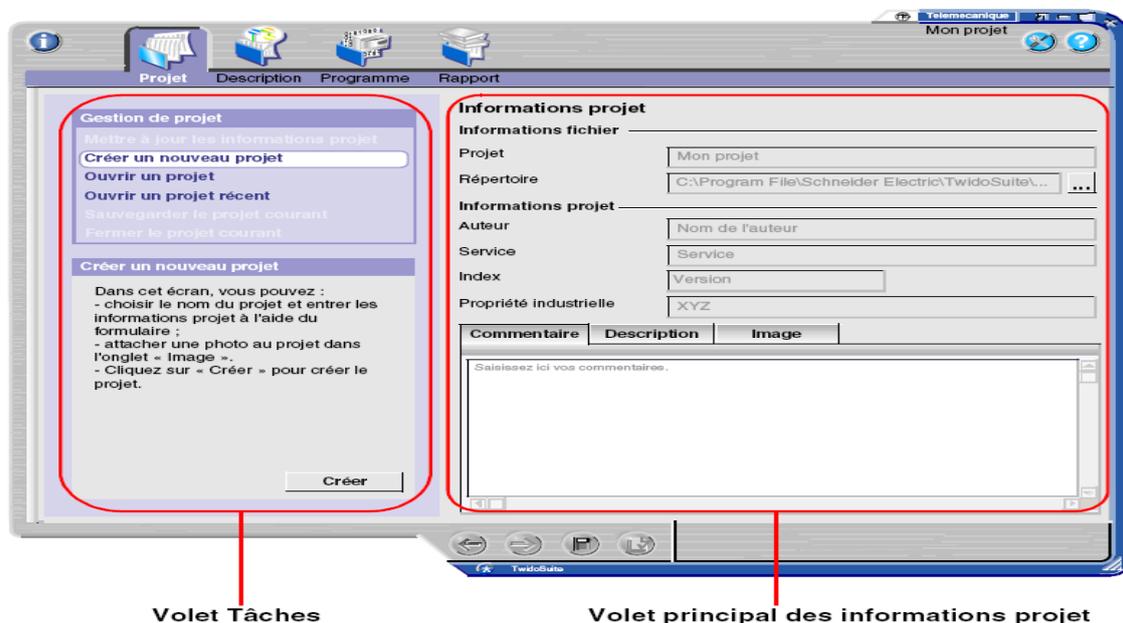
4.1. Exemple d'application :

Les sections suivantes vous expliquent, étape par étape, les notions fondamentales requises pour la programmation avec TwidoSuite, à commencer par l'exemple d'application suivant :

gestion du niveau de l'eau d'un réservoir utiliser pour l'irrigation



Cliquer sur l'icône projet  dans la barre des étapes de l'espace de travail TwidoSuite afin d'afficher la fenêtre projet et de créer, ouvrir ou modifier un projet TwidoSuite.



1. Cliquez sur créer un nouveau projet dans le volet "gestion de projet".
2. Saisissez un nom explicite pour le projet dans la section information fichier. Un nom de projet valide peut contenir au maximum 36 caractères alphanumériques (les espaces sont autorisées).
3. Indiquez le nouveau chemin du répertoire de projet ou utilisez le chemin par défaut. Dans la zone de texte répertoire, saisissez directement le chemin du répertoire de stockage ou cliquez sur le bouton parcourir situé à droite de la zone de texte répertoire afin de localiser le dossier ou les fichiers de projet sont enregistrés.
4. Entrez les informations relatives à la création du projet dans la section information projet : nom de l'auteur, nom du service, numéro de version du projet dans le champ

index et propriété industrielle.

5. Dans l'onglet commentaires, saisissez le nom du projet et une courte description, utilisez la barre de défilement à droite pour lire l'intégralité du texte.
6. Si vous mettez à jour un projet existant, cliquez sur l'onglet description pour afficher une représentation miniature de l'architecture du système.
7. Cliquez sur l'onglet image, puis sur modifier pour personnaliser l'image du projet ou utiliser l'image par défaut, m'image du projet est imprimée sur la page de couverture du rapport du projet. Un fichier image valide doit être au format. BMP.
8. Cliquez sur créer pour enregistrer vos modifications dans le nouveau fichier de projet.

Remarque : si vous mettez à jour les informations d'un projet existant, cliquez sur application pour enregistrer les modifications

4.2. Barres d'outils pour l'édition de programme :

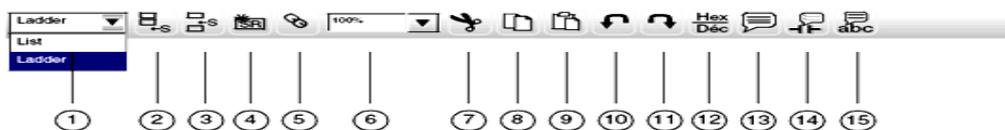
Trois barres d'outils de programmation en langage schéma à contacts (voir la figure ci-dessus) sont disponibles et vous permettent ainsi d'éditer des programmes en toute facilité, ainsi que de saisir graphiquement des instructions en langage schéma à contacts :

-barre d'outils du programme

-barre d'outils de la palette du langage schéma à contacts

-palette étendue du langage schéma à contacts

4.3. Barre d'outils du programme :



1-basculer en mode Ladder ou List

2-ajouter une action

3-insérer une section

4-ajouter un sous-programme

5-diviser en 2 sections

6-facteur de zoom (le pourcentage par défaut est de 100%)

7-couper

8-copier

9-coller

10-annuler

11-rétablir

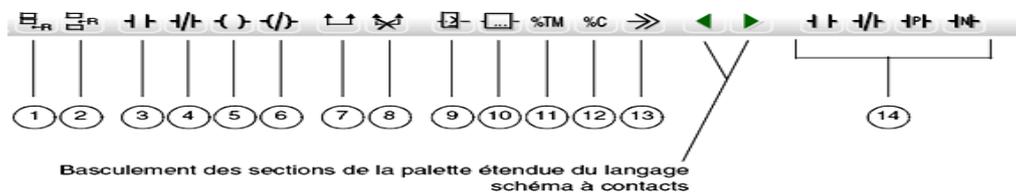
12-afficher au format hexadécimal ou décimal

13-afficher/masquer les commentaires relatif au réseau

14-afficher/masquer les commentaires relatifs aux éléments du langage schéma à contacts

15-afficher/masquer les symboles

4.4. Barre d'outils de la palette du langage schéma à contacts :



1-ajouter un réseau

2-insérer un réseau

3-ajouter un contact normalement ouvert

4-ajouter un contact normalement clos

5-ajouter une bobine

6-ajouter une bobine inverse

7-ajouter une liaison

8-supprimer une liaison

9-ajouter un bloc comparaison

10-ajouter un bloc opération

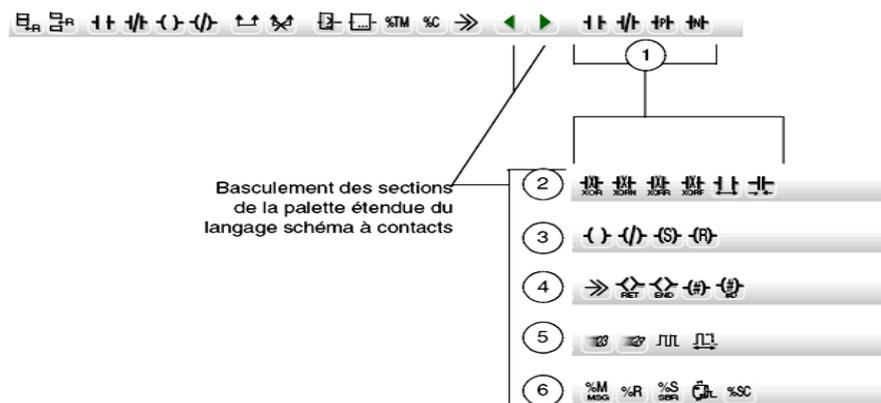
11-ajouter un bloc temporisateur

12-ajouter un bloc compteur

13-ajouter un appel de saut ou sous-programme

14-boutons de la palette étendue du langage schéma à contacts (voir la figure ci-dessous)

4.5. Palette étendue du langage schéma à contacts :



1-contacts : normalement ouvert, normalement clos, à front montant descendant

2-contacts spéciaux : XOR, XORN, XORR, XORF, OPN, SHORT

3-bobines : normale, inverse, d'enclenchement, de déclenchement

4-bobines spéciales : JMP/SR, RET, END, Grafcet1, Grafcet2

5-blocs fonction élémentaires : %FC, %VFC, %PLS, %PWM

6-blocs fonction avancés : %MSG, %R, %SBR, %DR, %SC

5. Comment éditer un programme :

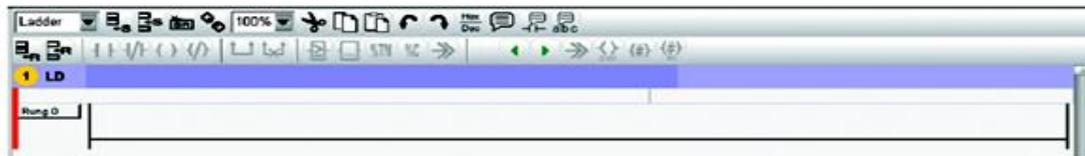
La procédure ci-après vous indique les étapes de base pour démarrer l'édition d'un programme en langage schéma à contacts.

1-lorsque vous commencez à écrire un nouveau programme, cliquez sur l'icône ajouter une

section  dans la barre d'outils programme, à l'invite suivant.



Résultat : la première section est insérée dans l'éditeur Ladder et le premier réseau vide s'affiche, comme illustré ci-dessous. (Un programme doit comporter au moins une section)



2-cliquer deux fois sur l'en-tête de section pour saisir les informations suivantes :

-titre de section ;

-commentaires ;

3-cliquez deux fois sur l'en-tête de réseau pour saisir les informations suivantes :

-titre du réseau ;

-commentaires ;

-déclaration du type de réseau : standard, étiquette (%Li) ou étape Grafcet (étape initiale, étape de début ou étape de fin).

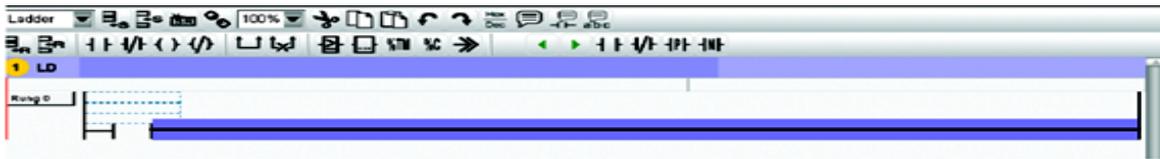
4-sélectionnez la liaison horizontale vide qui se situe entre les circuits d'alimentation gauche et droite du réseau.

Résultat : le lien sélectionné est surligné en bleu.



5-cliquez sur l'icône contact normale  (par exemple) dans la palette du langage schéma contacts afin d'insérer l'instruction graphique d'entrée.

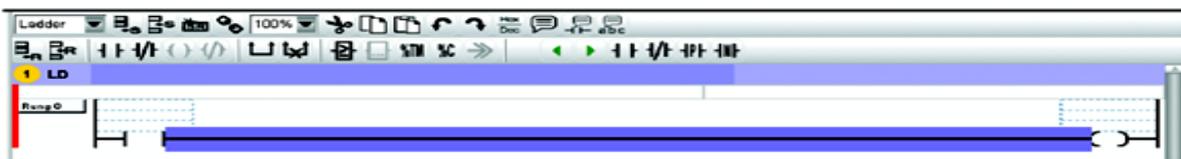
Résultat : le contact normal est inséré dans la partie la plus à gauche du réseau.



6-répétez les étapes 4 à 5 pour insérer d'autres instructions graphiques si nécessaire.

7-un réseau valide doit toujours comporter un élément de sortie (bobine ou bloc opération).

Cliquez sur l'icône bobine



Remarque :

A ce stade, la barre de couleur rouge indique que le réseau n'est toujours pas valide.

8-vous devez spécifier un repère pour chaque élément du réseau. Cliquez sur le champ de repère au-dessus de chaque élément et saisissez le repère souhaité. Saisissez : %I0.0 pour le contact d'entrée et %Q0.0 pour la bobine de sortie (par exemple).

Remarque : vous avez également la possibilité de finir un symbole et un commentaire pour chaque élément.



Résultat : la barre de couleur jaune indique que le réseau est valide, mais qu'il n'a pas encore été analysé (compilé avec le programme).

-cliquez sur l'icône analysé pour lancer le processus de validation du programme et rechercher les erreurs de programmation éventuelles dans la zone de liste d'erreurs

6. Tester le programme :

TwidoSuite comprend une fonction de simulation qui vous permet de tester un nouveau programme sans avoir à le charger sur un automate. Le simulateur Twido vous permet d'exécuter le programme et de tester différents paramètres afin de voir si vous obtenez les réponses attendues.

Pour démarrer le simulateur Twido, sélectionnez programme → programme et cliquez sur le bouton simulateur Twido  en bas à droite de l'écran TwidoSuite.

Résultat :

-TwidoSuite passe automatiquement à programme → mise au point → animer le programme.

-la fenêtre de présentation du simulateur Twido s'ouvre.

-le panneau de contrôle apparaît.

-si un horodateur (RTC)

Est configuré, le panneau de gestion du temps s'affiche aussi.

Le simulateur charge la configuration que vous avez créée dans l'écran description.

Une fois le simulateur lancé, vous pouvez utiliser les fonctions disponibles dans le mode de mis au point telles que :

-animer le programme ;

-gérer les tables d'animation ;

-contrôler l'automate ;

-Surveiller la configuration matérielle ;

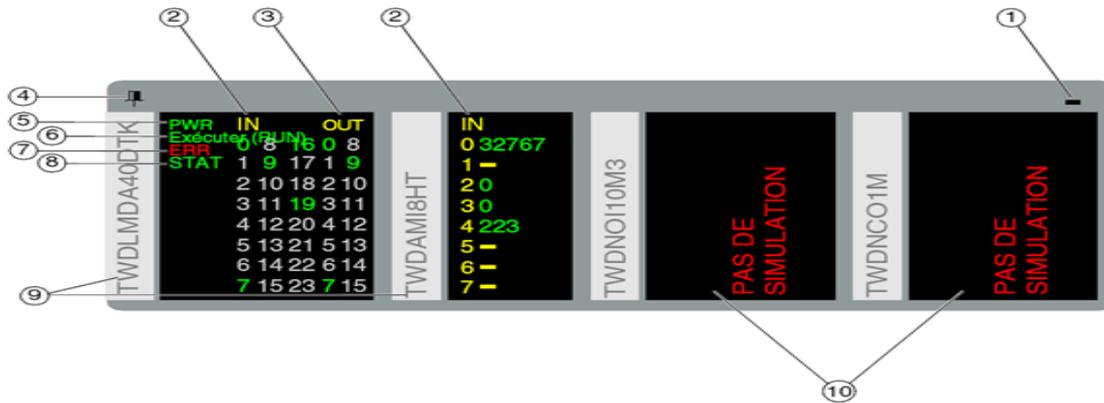
-Surveiller la configuration logicielle ;

-surveiller la configuration décrite ;

-contrôler le comportement.

La fenêtre de présentation du simulateur Twido indique l'état des entrées/sortie de tous les modules de base automate et d'expansion décrits.

Vous pouvez voir la fenêtre de présentation ci-dessous :



1-cliquez pour réduire la fenêtre de présentation.

Lorsque la fenêtre de présentation est réduite, l'icône du simulateur Twido  apparait dans la zone d'état (en bas à droite de votre écran). Cliquez deux fois sur cette icône pour restaurer la fenêtre de présentation.

2-état des entrées :

-cliquez pour définir/réinitialiser l'entrée correspondante ;

-cliquez deux fois pour ouvrir la fenêtre de paramétrage des sorties TOR.

3-état des sorties : cliquez deux fois pour ouvrir la fenêtre de traçage des sortie ;

4-cliquez pour toujours afficher la fenêtre de présentation en haut de l'écran.

5-voyant PWR : indique si l'automate simulé est en marche.

6-voyant RUN : indique l'état de l'automate simulé.

7-voyant ERR : indique l'état de l'automate simulé.

8-voyant STAT : l'action du voyant STAT est définie par la logique utilisateur

9-références de la base automate et des modules d'expansion

10-modules d'expansion qui ne peuvent pas être simulés.

Cliquez à l'aide du bouton droit sur la fenêtre de présentation pour ouvrir le menu contextuel suivant :

-gestion du temps : ouvre le panneau gestion du temps.

-traçage des sorties : ouvre la fenêtre de traçage des sorties.

Pour accéder aux paramètres des entrées analogiques, cliquez deux fois sur le voyant des entrées analogiques.

Pour accéder aux paramètres des entrées TOR, cliquez deux fois sur le voyant des entrées TOR.

6.1. Panneau de contrôle du Simulateur Twido :

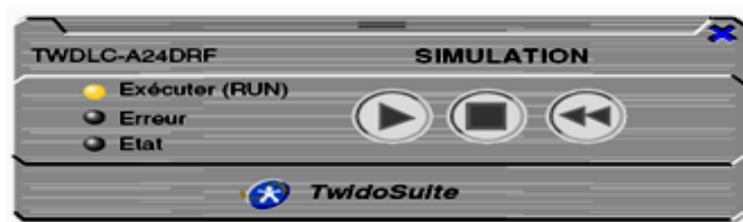
Vous pouvez utiliser le panneau de commande du simulateur Twido pour envoyer les commandes suivantes à l'automate :

-exécuter (en mode simulation) 

-stop 

-initialiser 

Lorsque l'application TwidoSuite est en mode simulation, ces commandes peuvent être sélectionnées à partir du panneau de contrôle du simulateur Twido indiqué ci-dessous :



Les voyants RUN, ERR, et STAT sont simulés dans le panneau de contrôle du simulateur Twido comme ils apparaîtraient sur une base automate connectée.

Pour fermer le panneau de commande du panneau de contrôle

Le panneau de commande se développe vers le bas (voir la figure ci-dessous). Vous pouvez le réduire en cliquant le même bouton de développement/réduction.

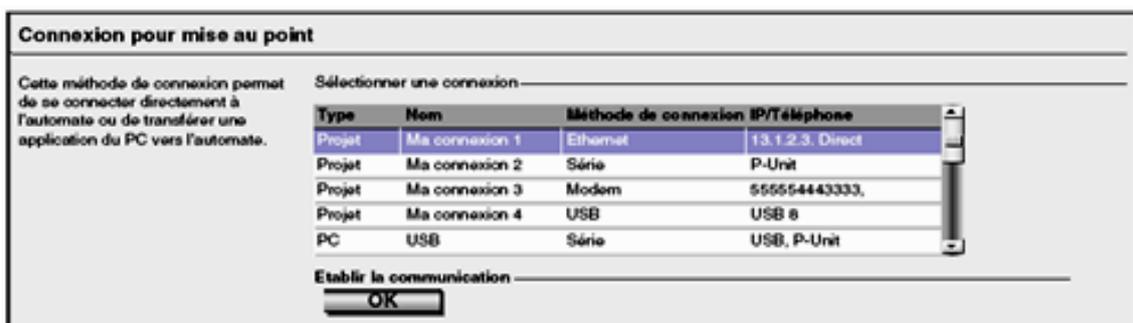


6.2. Transfert et exécution du programme :

Pour mettre au point votre application, vous devez d'abord la transférer dans la mémoire vive de l'automate.

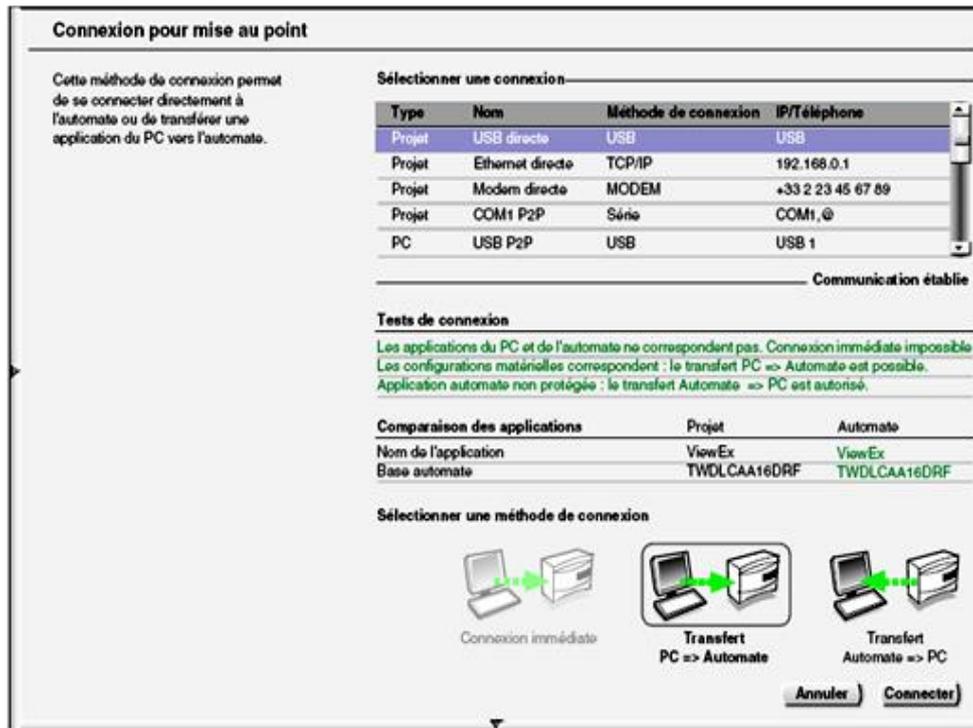
Suivez les étapes ci-après pour télécharger le programme de votre application dans l'automate :

1-si l'ordinateur n'est pas encore connecté à l'automate, sélectionnez une connexion de la table des connexions dans la tâche programme → mise au point → connecter, puis sur exécuter.



Résultat : TwidoSuite tente d'établir une connexion avec l'automate et procède à des contrôles de synchronisation entre les applications de l'ordinateur et de l'automate.

-le tableau de test de connexion affiche les résultats de 3 des contrôles



2-si les applications de l'ordinateur et de l'automate différent, sélectionnez transfert PC



3-si l'application de l'automate est protégée par un mot de passe, vous êtes invité à saisir ce mot de passe. Saisissez le mot de passe, puis cliquez sur OK pour continuer le transfert.

4-si l'automate est en cours d'exécution, une boîte de dialogue d'avertissement apparaît et indique que vous devez arrêter l'automate pour continuer le transfert.

5-cliquez sur OK pour continuer le transfert.

6.3. Exécution du programme :

Utilisez le panneau de contrôle de l'automate pour exécuter, arrêter ou initialiser un programme qui a été téléchargé dans l'automate.

Pour exécuter le programme que vous avez transféré vers l'automate, procédez comme suit :

1-si la connexion entre TwidoSuite et l'automate n'est pas active, sélectionnez une connexion de la table des connexions dans le programme → mise au point → connecter, puis cliquez sur OK pour établir la connexion.

2-une fois la connexion établie entre TwidoSuite et l'automate, la boîte de dialogue du panneau de contrôle de l'automate s'affiche toujours en haut de l'espace de travail de TwidoSuite, sous la forme d'un panneau de commande.



Celui-ci comporte les boutons de commande suivants :

-  exécuter
-  stop
-  initialiser

3-cliquez sur exécuter pour exécuter le programme d'application TwidoSuite présent dans la mémoire vive de l'automate.

Remarque : chaque commande doit être confirmée avant de pouvoir être exécuté

Résultat : trois voyants sur le panneau (RUN, Error, Stat) fournissent des diagnostics en direct.

6.4. Animation d'un programme :

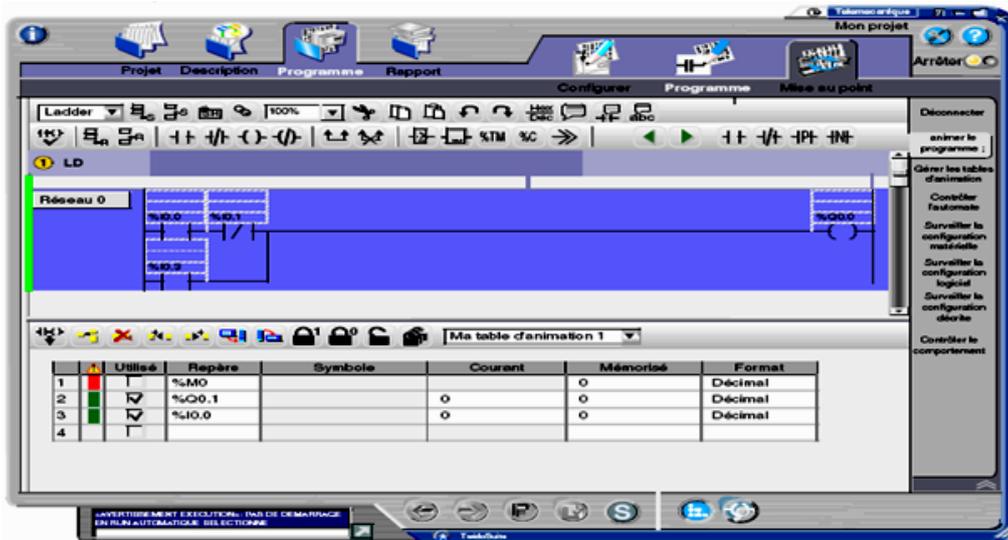
En ligne, vous pouvez animer de programme en langage schéma à contacts et list d'instruction. Cette section décrit l'animation d'un programme en langage schéma à contacts.

6.5. Afficher/masquer les table d'animation :

Cliquer sur l'icône  dans la barre des fonctions pour afficher/masquer les table d'animation dans la partie inférieure de la fenêtre.

Pour copier des objets de réseau dans la table d'animation sélectionnée, cliquez sur 

Bouton le plus à gauche dans la barre d'outil de la table d'animation comme indiqué ci-dessous.



6.6. Surveillance d'une application :

Pour démarrer l'utilitaire de surveillance TwidoSuite :

1-ouvrez le lanceur d'application en cliquant sur l'icône située sur le bureau de votre PC

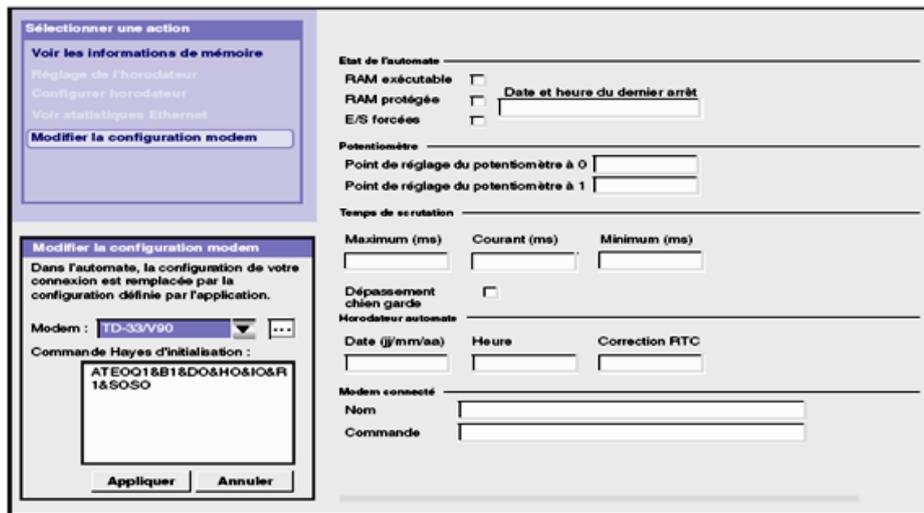


Résultat : la boîte de dialogue du lanceur d'application s'affiche, comme illustré ci-dessous.



2- à partir du lanceur d'application TwidoSuite, cliquez sur Mode de surveillance.

Sélectionnez contrôler l'automate :



L'utilitaire de surveillance vous permet d'effectuer les tâches suivantes :

- 1-voir les informations de mémoire.
- 2-définir l'heure de l'horodateur (RTC).
- 3-configurer le coefficient de correction du RTC.
- 4-voir les statistiques Ethernet.
- 5-modifier la configuration modem.

7. Le logiciel XBT L100• pour l'afficheur XBTN 400 : [5]

7.1. Présentation

Le logiciel XBT L100• est le logiciel de configuration des terminaux d'exploitation de la gamme Magelis.

Ce logiciel fonctionne sur des ordinateurs équipés du logiciel d'exploitation Windows 98SE, 2000 et XP.

7.2. Installation

Pour pouvoir utiliser le logiciel XBT L100•, la configuration minimum nécessaire est la suivante :

- PC Pentium II 350 MHz
- CD-ROM 4x minimum.
- Microsoft 98SE, 2000 ou XP.
- 128 Mo RAM.

- 30 Mo d'espace libre sur le disque dur.
- Carte vidéo SVGA ou supérieur.

7.3. Fonctionnalités

Le logiciel de configuration XBT L100• permet la création d'applications utilisables par les terminaux Magelis, dans lesquelles il sera possible de :

- créer différents types de pages :
- pages application,
- pages alarme.

Ces pages peuvent contenir toutes sortes de variables prédéfinies dans le logiciel XBT L100•. Différentes propriétés peuvent leur être affectées.

- configurer les touches de fonctions pour effectuer des commandes sur la machine, ou appeler des pages application.
- importer la liste des symboles automates de type PL7 (Junior/Micro/Pro), Concept et TwidoSoft, pour éviter la double saisie des variables.

Le logiciel XBT L100• offre, en outre, au bureau d'études, la possibilité de contrôler les applications réalisées, sans utilisation d'un terminal ou d'un automate programmable.

Le programme de simulation combiné aux touches du clavier de l'ordinateur permet de tester :

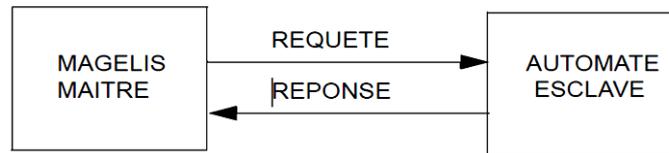
- la navigation entre les pages,
- la saisie des variables,
- la visualisation de variables,
- la gestion des alarmes,
-

8. Principe de fonctionnement

XBT a le statut de maître. Le principe de fonctionnement de l'XBT repose sur une

« Table de dialogue » qui se trouve dans l'automate (esclave). En tant que maître, l'XBT réalise 3 types d'actions : - des actions initiées par l'automate, - des actions initiées par l'opérateur, - des actions auto-initiées. Le protocole SYSMAC-WAY est le protocole de communication utilisé par tous les automates OMRON. Il est possible de connecter l'XBT en point à point sur l'automate via le connecteur RS232 ou d'utiliser un réseau RS422 sur lequel peuvent être raccordés jusqu'à 32 automates. (Le réseau RS422 n'est pas disponible pour les terminaux XBT N/R.) Le terminal MAGELIS XBT (maître) envoie des requêtes de

lecture/écriture à l'automate (esclave) qui exécute ces requêtes et renvoie les réponses. Sous le protocole SYSMAC-WAY, le terminal dialogue en mode ASCII.



Remarque : Le branchement d'un XBT qui exige des requêtes d'écriture en mémoire à un automate en mode de service (RUN) entraîne le basculement de l'automate en mode surveillance (MONITOR)

Configuration de la table de dialogue

Pour configurer la table de dialogue :

Etape	Action
1	Sélectionnez Configuration/Table de dialogue.
2	Indiquez l'adresse de début de la table et le temps de scrutation.
3	Configurez la table en ajoutant ou en supprimant les fonctions souhaitées pour votre application.

9. Concepts pour la création des applications Magelis

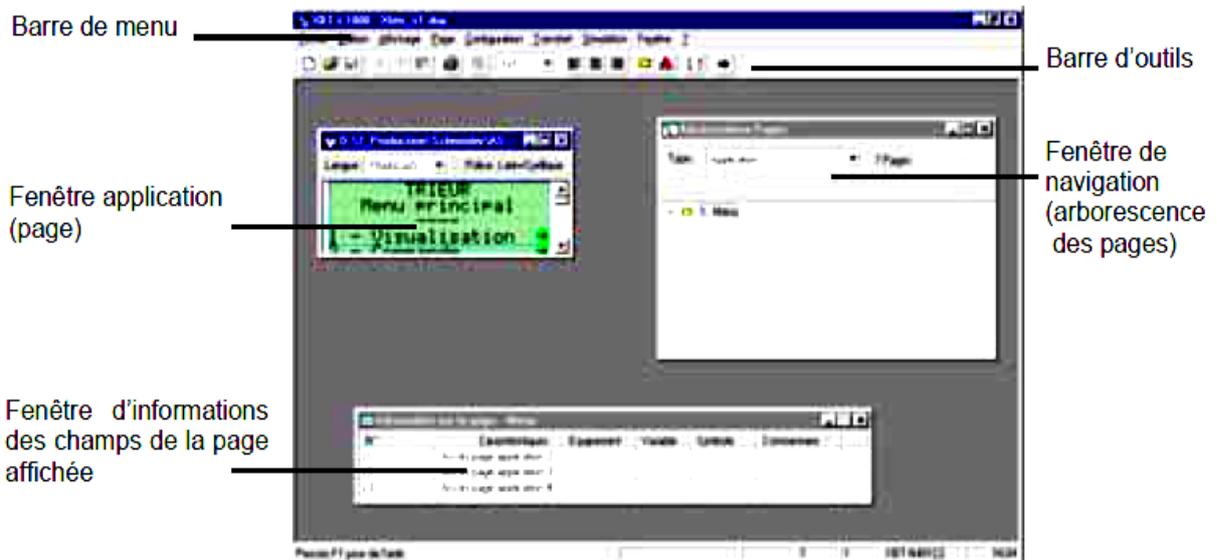
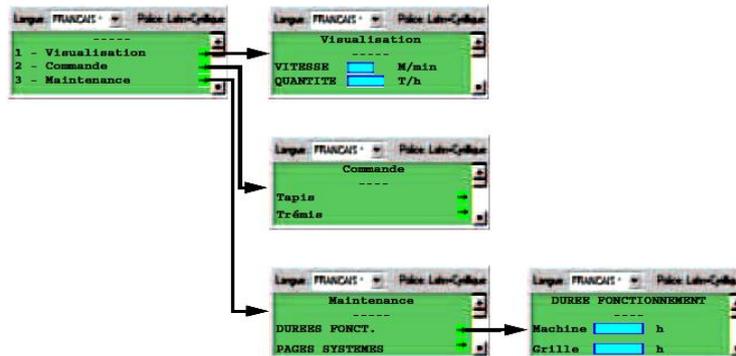
Une application Magelis constitue l'ensemble du dialogue entre l'opérateur et le procédé automatisé. Cet ensemble doit prendre en compte :

- les critères liés à l'automatisme :
 - suivi de production,
 - maintenance préventive,
 - maintenance corrective,
 - commande du procédé.
- les critères des utilisateurs :
 - ergonomie,
 - niveau d'intervention.
- les critères de réalisation de l'application de dialogue elle-même :
 - programmation,
 - mise au point,
 - évolution.

Remarque :

Ces contraintes nécessitent donc de structurer votre application.

Une application sera composée d'un ensemble de pages, pouvant constituer une structure arborescente telle que la montre la figure suivante :



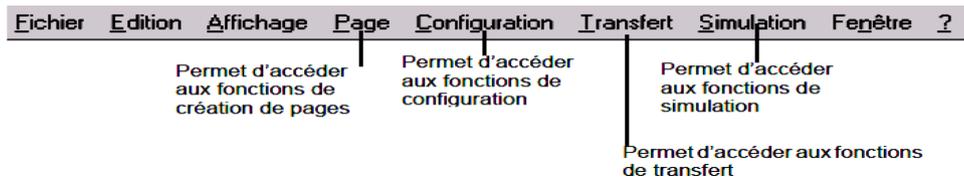
9.1 Présentation du logiciel

9.1.1 Barre de menus

Si aucune application n'est ouverte, la barre de menus affichée est la suivante :

Fichier Affichage Transfert ?

Si une application est ouverte, la barre de menus est la suivante :

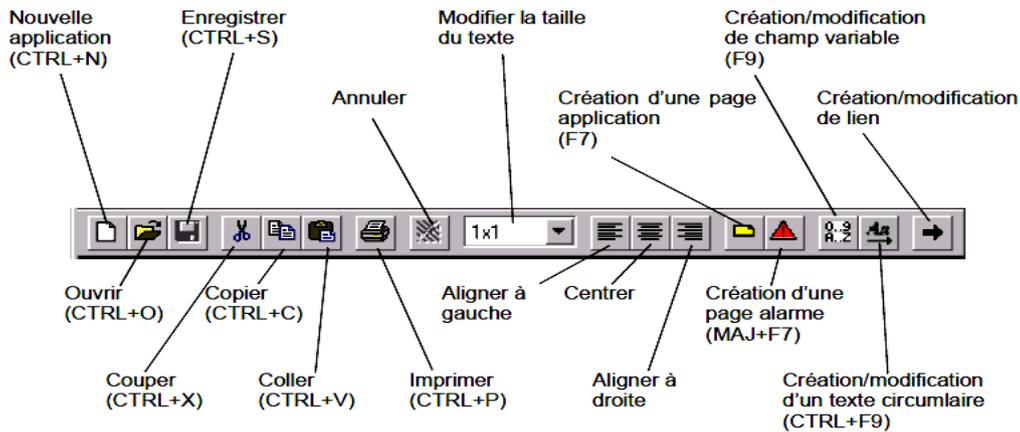


9.1.2 Barre d'outils

Si aucune application n'est ouverte, la barre d'outils de la fenêtre principale comporte les boutons suivants :



Lors de l'édition d'une page application ou d'une page alarme la barre d'outils de la fenêtre principale comporte les boutons suivants :



9.1.3 Fenêtre d'information des champs

La fenêtre d'information des champs permet de visualiser les propriétés des champs et des liens de la page application en cours.

Information sur la page : Durée					
N°	Caractéristiques	Equipement	Variable	Symbole	Commentaire
1	9999999	API_PLC	40021	CST9	
9	9999999	MASTER	40121		
L2	Accès page application 4				

La colonne "N°" indique le numéro de champ ou de lien.

La colonne "*Caractéristiques*" indique le type et la longueur du champ.

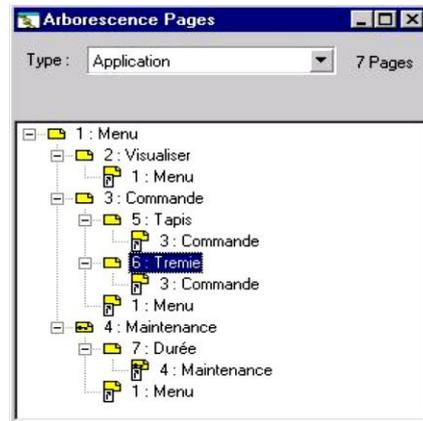
Dans l'exemple ci-dessus, on peut voir que le champ n°1 est de type décimal et d'une longueur de sept caractères.

La colonne "*Symbole*" indique le symbole équipement auquel est liée la variable.

La colonne "*Commentaire*" indique le commentaire lié au symbole.

9.1.4 Fenêtre de navigation

La fenêtre de navigation contient la structure des pages de l'application active et précise la page en cours.



10. Création d'une nouvelle application

Pour créer une application sous XBT L100, cliquer sur le bouton  de la barre d'outils ou bien sélectionner *Nouveau* dans le menu *Fichier*.

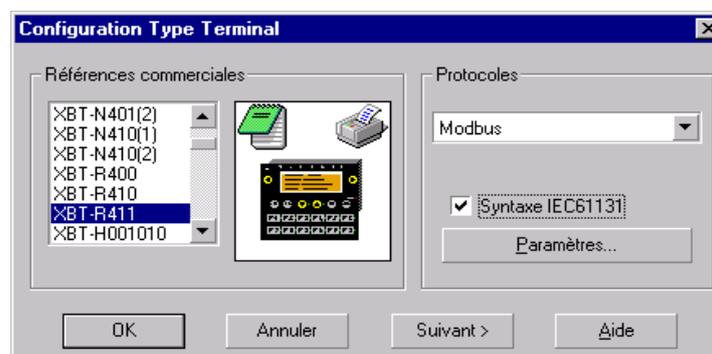
La création d'une application passe par les étapes suivantes :

- Etape 1 : choix du type de terminal,
- Etape 2 : configuration des paramètres du terminal,
- Etape 3 : configuration des paramètres du protocole,
- Etape 4 : configuration des équipements,
- Etape 5 : choix des langues applicatives,
- Etape 6 : configuration de la table de dialogue.

Etape 1 - Choix du type de terminal

Une première fenêtre permet de sélectionner le terminal et le protocole associés à l'application à créer.

La fenêtre de sélection du terminal est la suivante :

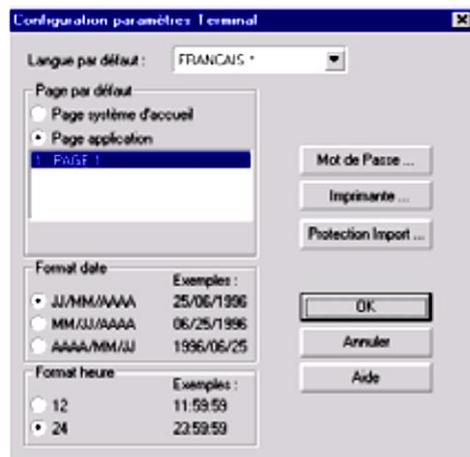


Pour les terminaux XBT N, il est possible de sélectionner soit un terminal en mode saisie, soit un terminal en mode commande. En fonction de la sélection, le clavier du terminal sera différent.

- La validation par le bouton "OK" permet de créer une nouvelle application sans passer par l'assistant de création.
- Un "clic" sur le bouton "Suivant >" guide le concepteur afin de créer une nouvelle application grâce à l'enchaînement de plusieurs fenêtres.
- Un "clic" sur le bouton "Paramètre..." permet d'accéder à la configuration des paramètres du protocole sélectionné
- La boîte à cocher notée "Syntaxe IEC61131" permet pour le protocole Modbus de choisir la syntaxe employée pour les adresses des variables.

Etape 2 - Configuration des paramètres du terminal

Cette fenêtre apparaît après sélection de ***Paramètres Terminal*** depuis le menu ***Configuration***.



Cette fenêtre permet de configurer les paramètres du terminal suivants :

- la langue par défaut (cette langue sera utilisée à la mise sous tension du terminal.),
- la page par défaut (le choix se fait dans une liste correspondant à la liste des pages application existantes),
- la protection de l'import de l'application,
- les mots de passe,
- l'imprimante (Uniquement XBT N401 et XBT R411)
- le format de date utilisée par le terminal,
- le format de l'heure utilisée par le terminal.

Etape 3 - Configuration des paramètres du protocole

Cette fenêtre apparaît après sélection de **Paramètres Protocole** depuis le menu **Configuration**.

Cette fenêtre permet :

- d'indiquer les paramètres de ligne du protocole
- de saisir des paramètres spécifiques au protocole



Etape 4 - Configuration des équipements

Cette fenêtre apparaît après sélection de **Symboles équipements** depuis le menu **Configuration**.

Principe

Un automatisme est composé de plusieurs équipements qui peuvent être reliés à un ou plusieurs terminaux. Un équipement peut être un automate, un ordinateur, un variateur de vitesse, ... Le terminal lui-même est considéré comme un équipement, ce qui lui permet d'avoir ses propres variables, telles que la date, l'heure, ...

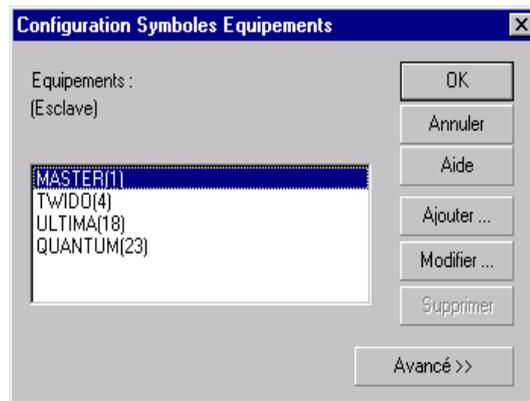
Du point de vue de XBT L100*, un équipement est défini par deux caractéristiques : l'adresse et le symbole de l'équipement. L'adresse dépend uniquement du protocole utilisé.

Un terminal peut communiquer avec quinze équipements différents au maximum.

Fenêtre de configuration des équipements

En sélectionnant **Symboles Equipement** depuis le menu

Configuration, il est possible de créer, modifier ou supprimer des symboles d'équipement.



Par défaut, la liste comporte un équipement avec une adresse par défaut, son symbole est "MASTER". Cet équipement ne peut pas être supprimé, mais son nom et son adresse peuvent être modifiés.

Le bouton "Ajouter fichier" permet au concepteur de choisir le fichier symboles de variables.

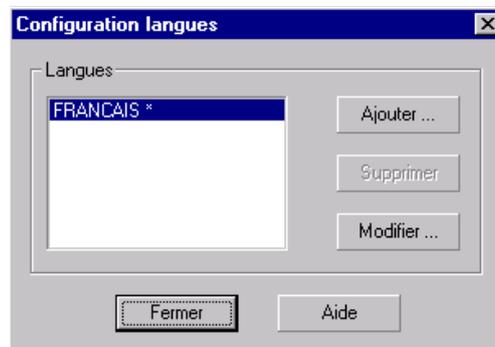
Les extensions de ces fichiers sont ".SCY"

(PL7), ".TXT" (Concept) ou ".TWD" (TwidoSoft).

Etape 5 - Choix des langues applicatives

En sélectionnant ***Langues de l'Application*** depuis le menu

Configuration, il est possible de modifier la liste des langues utilisées par l'application.



Le bouton "Supprimer" supprime la langue sélectionnée. La langue de référence ne peut être supprimée.

Le bouton "Ajouter" provoque l'affichage de la fenêtre suivante :



Cette fenêtre permet d'ajouter des langues supplémentaires. Elle comporte :

- une liste permettant de choisir la police (Latin+Cyrillique, Latin+Kata kana, Cyrillique, Grec ou Chinois),
- une zone permettant d'indiquer le nom de la langue,
- une liste comprenant les 6 langues disponibles pour les pages systèmes (Français, Anglais, Espagnol, Allemand, Italien et Chinois).

Tant que la police chinoise n'a pas été choisie, le choix du Chinois n'apparaît pas dans la liste des langues des pages systèmes.

Lorsqu'il y a plusieurs langues dans l'application, par défaut, les pages des langues supplémentaires sont identiques aux pages de la langue de référence, excepté si la langue de référence est le chinois. Dans ce cas, les caractères chinois sont remplacés par des '?' (Textes statiques, textes circulaires et textes des listes énumérées) pour les langues où la police chinoise n'a pas été utilisée.

Toute modification de champs variables ou de liens dans une page dans la langue de référence (première de la liste) est systématiquement répercutée sur les pages des autres langues. Par contre, toute modification de texte statique n'a aucun effet sur le texte statique des pages des autres langues applicatives. Il en est de même pour les textes des listes énumérées

La saisie dans les pages des autres langues est limitée au texte statique, au texte circulaire et au texte des listes énumérées. La modification des textes dans une langue autre que celle de référence n'a aucune incidence sur les pages des autres langues. Il n'est pas possible de supprimer, modifier, déplacer ou ajouter un champ variable ou un lien depuis une page éditée dans une langue autre que celle de référence.

Une liste déroulante dans la barre d'outils de la fenêtre principale de l'application indique le nom de la langue courante de l'application. Si une seule langue applicative est configurée, cette liste est inactive. Lorsque la langue sélectionnée est la langue de référence, un astérisque ("*") suit le nom de la langue.

La langue de référence est la langue de développement de l'application.

Une langue par défaut peut être définie à l'ouverture de l'application

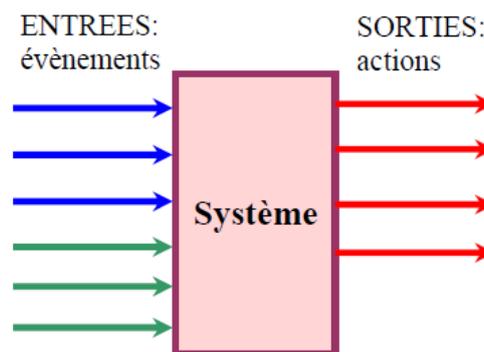
Etape 6 - Configuration de la table de dialogue

En sélectionnant **Table Dialogue** depuis le menu **Configuration**, il est possible de configurer la table de dialogue et les éventuelles tables d'alarmes supplémentaires qui permettent le déclenchement d'alarmes par plusieurs équipements.

11. Présentation du langage de programmation GRAFCET : [9]

Le langage GRAFCET (GRAPhe Fonctionnel de Commandes Etapes Transitions)

Il s'agit d'un langage graphique permettant de définir le comportement séquentiel d'un système automatisé à partir de la connaissance des actions à entreprendre, associées à des variables de sorties, et des événements qui peuvent permettre le passage d'une situation à une autre, associés à des variables d'entrée



12. Structure graphique : [6]

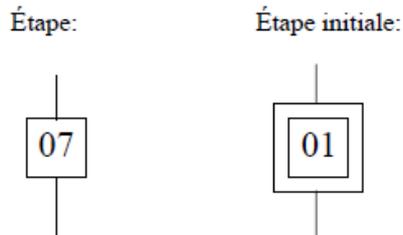
12.1. Etapes :

Une étape correspond à une situation dans laquelle le comportement de toute partie du système par rapport à ses entrées et ses sorties est invariant.

A un instant donné et suivant l'évolution du système :

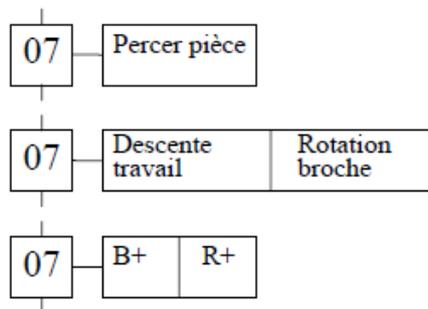
- une étape est soit active, soit inactive ;
- l'ensemble des étapes actives définit la situation du système ou du sous-système.

Les étapes qui sont actives au début du processus sont appelées étapes initiales



12.2. Actions associées à une étape :

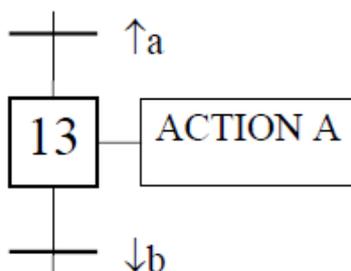
Les actions associées à une étape indiquent ce qui doit être fait chaque fois que l'étape à laquelle elles sont associées est active. Elles sont choisies parmi les sorties du système. Leur nature diffère donc légèrement en fonction du point de vue adopté.



12.3. Transitions :

Les transitions indiquent les possibilités d'évolution entre 2 étapes. On associe à chaque transition une condition logique appelée réceptivité qui permet de distinguer parmi toutes les informations d'entrée possible, uniquement celle qui sont susceptibles de faire évoluer la partie commande.

- soit littéralement ;
- soit par une expression booléenne ;
- soit symboliquement.



12.4. La traduction algébrique du langage GRAFCET en LADDER : [6]

Soit la partie de grafcet représentée la figure ci-dessus, pour décrire l'activation de l'étape n, nous utiliserons la notation suivante :

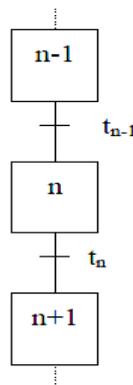
- $X_n=1$ si l'étape n est active ;

- $X_n=0$ si l'étape n est inactive

La réceptivité t_n étant une variable binaire, a pour valeur

- $t_n=0$ si la réceptivité est fausse ;

- $t_n=1$ si la réceptivité est vraie.



Une transition est soit validée, soit non validée elle est validée lorsque toutes les étapes immédiatement précédentes sont actives, elle ne peut être franchie que :

La traduction de cette règle donne la condition d'activation de l'étape n :

$$CDX_n = X_{n-1} \cdot t_{n-1} = X_{n+1}$$

Si la CA et la CD de l'étape ne sont fausses, l'étape reste dans son état, c'est ce qu'on appelle l'effet mémoire c'est-à-dire que l'état de X_n à l'instant t est l'état précédent de X_n à l'instant t ,

d'après ces trois points précédents, on peut donc écrire : $X_n = f(CAX_n, X_n, CDX_n)$

A l'initialisation du GRAFCET, toutes les étapes autre que les étapes initiales sont désactivées, seules sont activées les étapes initiales.

Soit la variable Init telle que :

-Init=1 : initialisation du GRAFCET : mode arrêt

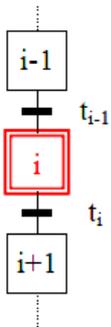
-Init=0 : déroulement du cycle : mode marche

-AUDur=1 : désactivation de toutes les étapes

-AUDoux=1 : désactivation des actions, les étapes restant actives.

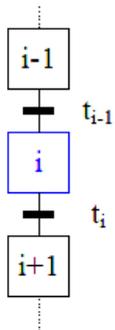
13. Généralisation :

13.1. Equation d'une i initiale :



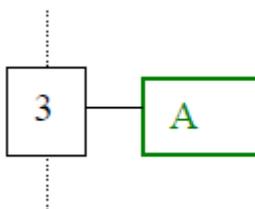
CAXi	CDXi	Equation de Xi
$X_{i-1}.t_{i-1} + Init$	$X_{i+1}.Init$	$X_i = (CAX_i + \overline{CDX_i}.X_i + Init) . \overline{AUdur}$

13.2. Equation d'une étape i non initiale :



CAXi	CDXi	Equation de Xi
$X_{i-1}.t_{i-1}.Init$	$X_{i+1}.Init$	$X_i = (CAX_i + \overline{CDX_i}.X_i) . Init . \overline{AUdur}$

13.3. Equation des actions :



$$A = X3 . \overline{AUdoux}$$

14. Exemple d'application de la traduction de GRAFCET au LADDER :

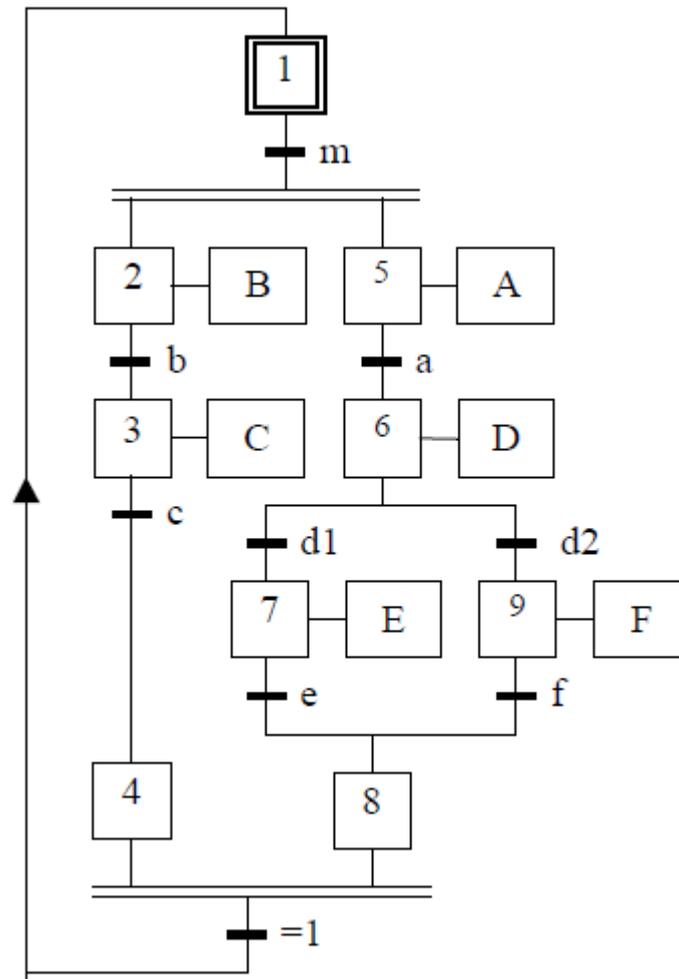
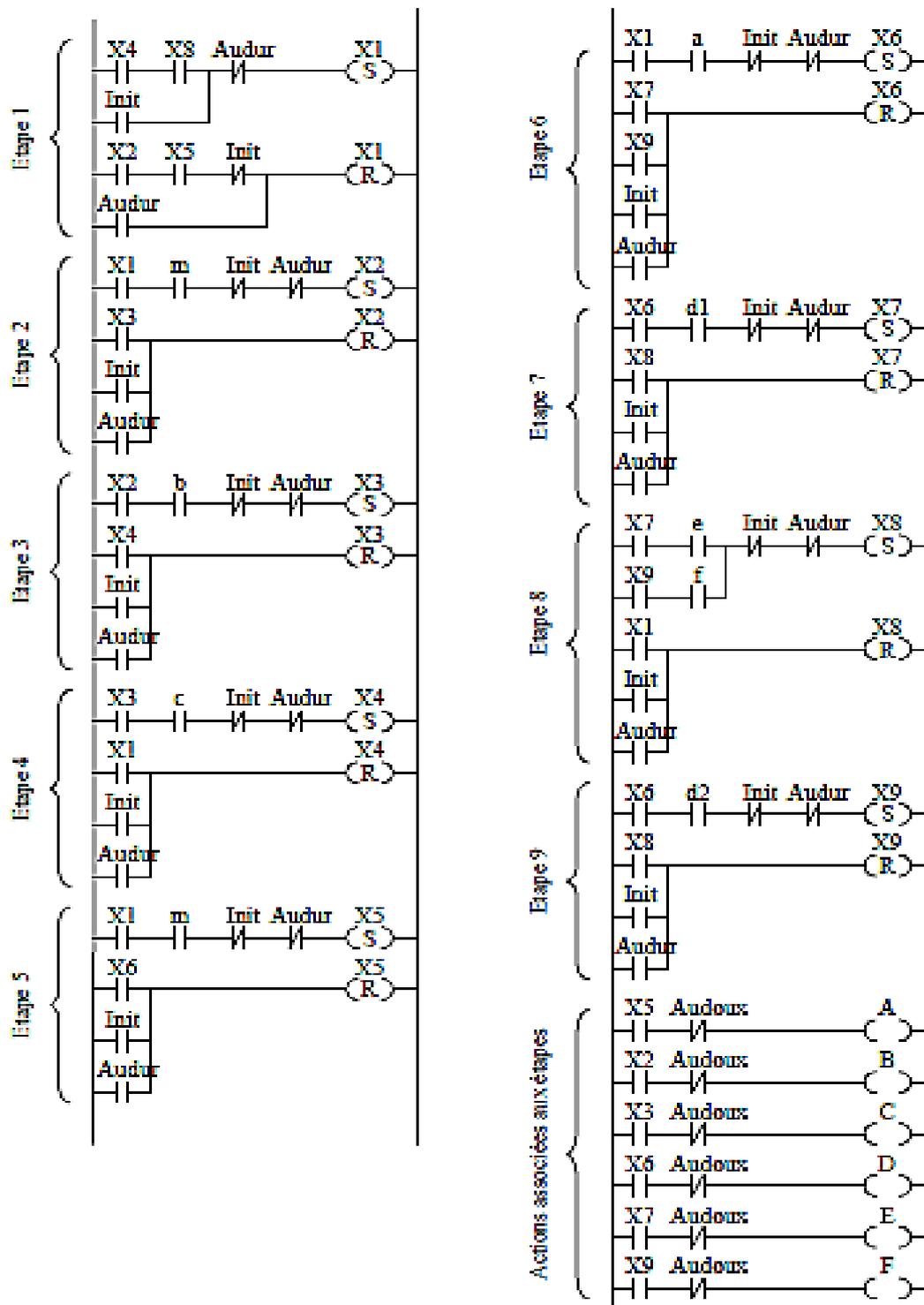


Table des conditions d'activation et de désactivation des étapes :

Xi	CAX_i	CDX_i
1	$X_4 \cdot X_8 + Init$	$X_2 \cdot X_5 \cdot \overline{Init}$
2	$X_1 \cdot m \cdot \overline{Init}$	$X_3 + Init$
3	$X_2 \cdot b \cdot \overline{Init}$	$X_4 + Init$
4	$X_3 \cdot c \cdot \overline{Init}$	$X_1 + Init$
5	$X_1 \cdot m \cdot \overline{Init}$	$X_6 + Init$
6	$X_5 \cdot a \cdot \overline{Init}$	$X_7 + X_9 + Init$
7	$X_6 \cdot d1 \cdot \overline{Init}$	$X_8 + init$
8	$(X_7 \cdot e + X_9 \cdot f) \cdot \overline{Init}$	$X_1 + Init$
9	$X_6 \cdot d2 \cdot \overline{Init}$	$X_8 + init$

La traduction en LADDER :



15. Conclusion :

Dans ce chapitre, nous avons exposé le choix et les caractéristiques de logiciel de programmation associé **TwidoSoft**, et le logiciel XBT L100• pour l'afficheur XBTN 400.

.La présentation précédente de API TWIDO : TWDLCAA40DRF et du logiciel de programmation **TwidoSoft**, prépare le terrain en vue de l'application « Etude et Automatisation de procédé de palatisation » qui fera l'objet du prochain chapitre.

Chapitre 03

Etude et programmation de procèdes de palatisation

1. Introduction :

Dans ce chapitre on va entamer la partie de programmation qui va faire fonctionner notre système nous allons utiliser pour cette fin le logiciel TwidoSoft (automate) et logiciel XBT L100• (afficheur).

Après l'identification des entrées/ sorties qu'on va les utiliser pour les commandes, L'idée étant d'intégrer l'automate en question dans un dispositif permettant de répondre à un besoin spécifique En effet, notre projet consiste à commander une chaîne de production avec l'automate Twido. La commande est entièrement assurée par la programmation de l'automate de façon à répondre exactement à nos besoins.

2. Description de procédé :

Le système de palettisation se compose en six parties comme indique dans le schéma suivant :

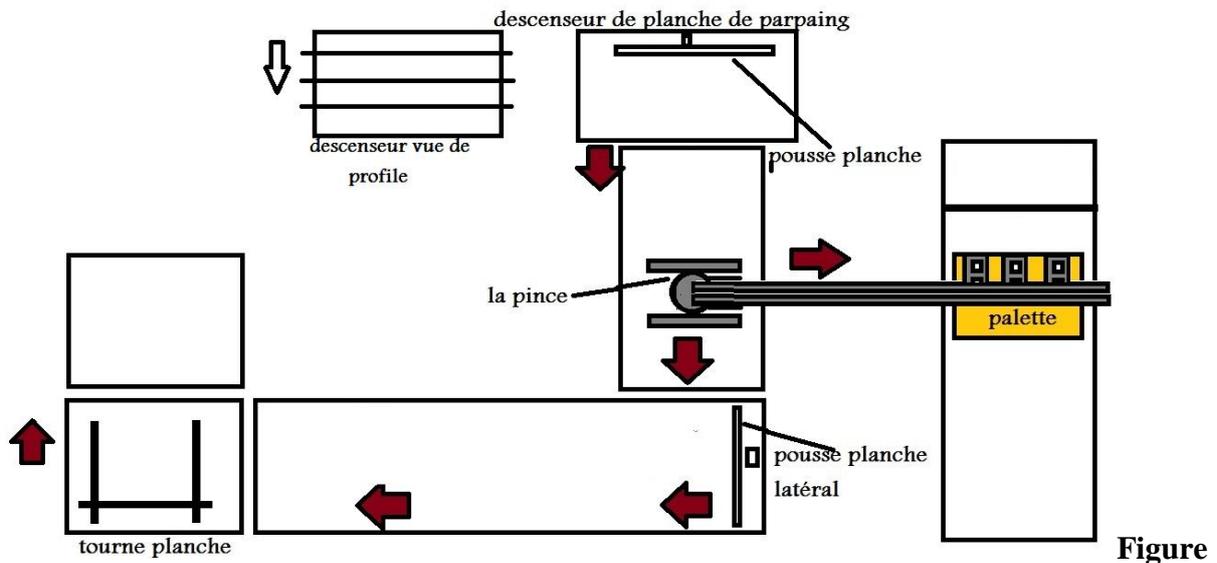


Figure3.1 schéma du déroulement de la palettisation

2. 1 Le descendeur des planches de parpaing :



Le Clarke transporte les planches de dépôt, 06 en tout, vers le descendeur. le descendeur commence à descendre puis le pousse-planche les pousse vers la pince.

2.2 pousse planches de parpaing :



Le pousse planche commence à pousser les planches pleines vers la pince à chaque fois que le descendeur fait descendre une planche.

2. 3 la pince :



Le rôle de La pince est de descendre pour presser le parpaing, se trouvant sur les planches, et l'enlever, puis elle monte et le transporte vers une palette.

2.2.4 pousse planches latérale :



Pousse les planches vide latéralement vers la tourne planche.

2. 5 tourne planches :



Elle fait tourner les planches vides arrivant de la pousse planche latérale, pour les ranger.

2.6. Pupitre :

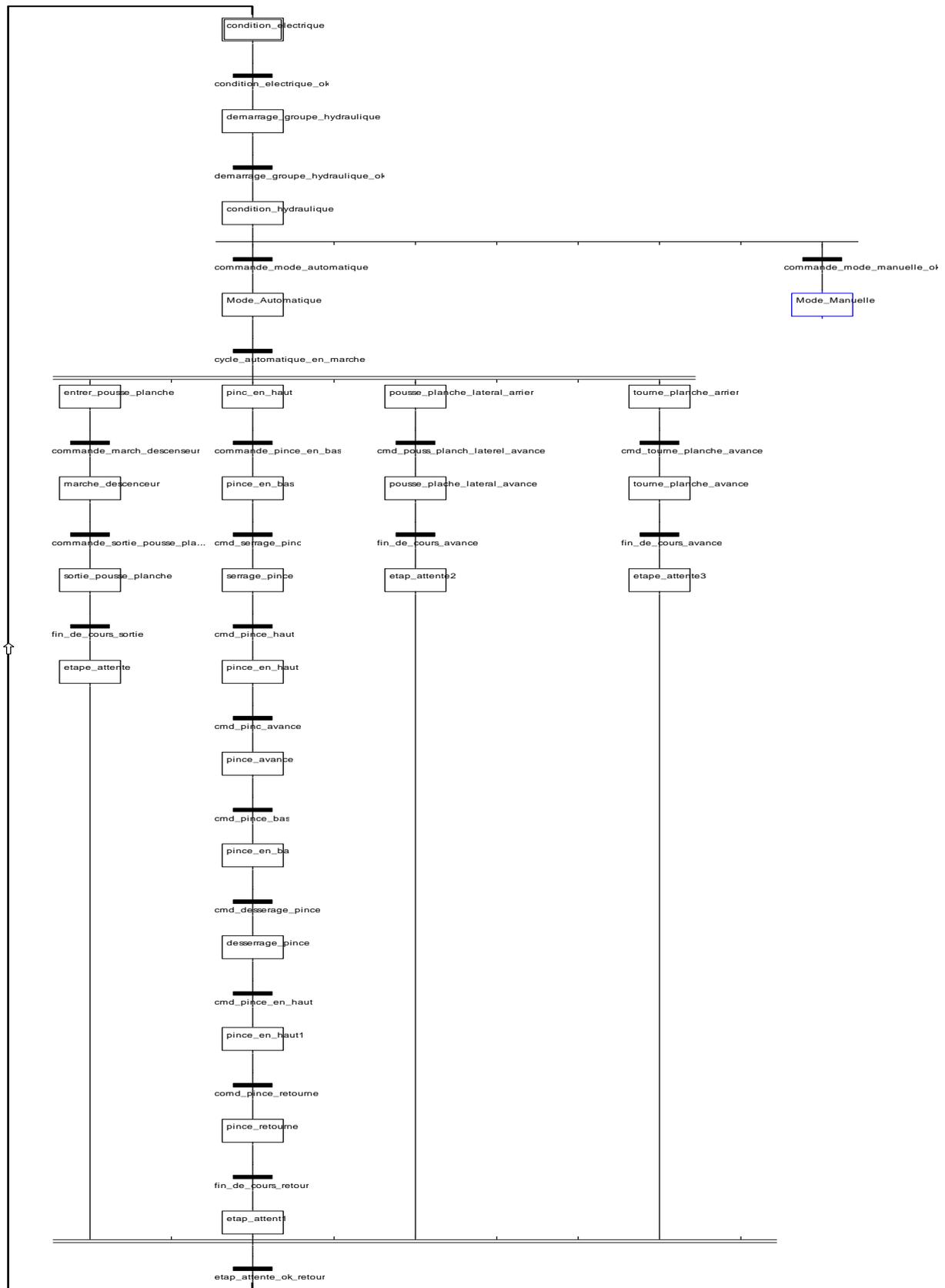
Le pupitre assure la commande en manuel et en automatique, Il contient l'automate TWIDO et 4 modules d'entrée sortie qui sont branchés, il y'a aussi, les disjoncteurs des moteurs électrique, il contient aussi des contacteurs, les protecteurs moteur électrique, et ses derniers sont brancher selon un schéma électrique (voir annexe)



3. Programmation :

Après l'identification des E/S qu'on va utiliser dans la programmation de la commande, on commencera la rédaction du programme, et pour faciliter la programmation on l'écrit en langage GRAFCET pour mieux comprendre les étapes d'action et transitions de chaque actionneur de la commande du mode automatique après en le traduit en langage LADDER sur logiciel TwidoSoft pour le charger dans l'automate.

3.1. Programmation avec GRAFCET :

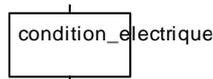


3.2. Programmation avec LADDER :

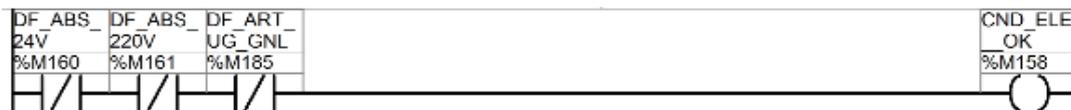
3.2.1. L'affectation des entrées :

Copier les entrées physique %i dans les entrées analogique %M, en cas de décalage ou changement des entrés physique il sera ajoure (voir annexe).

3.2.2. L'étape: condition électrique

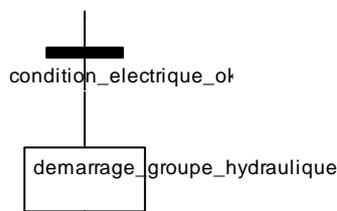


Les conditions électrique sont exécutables si il y pas défaut absence 24v et défaut absence 220v et le défaut d'arrêt d'urgence générale.

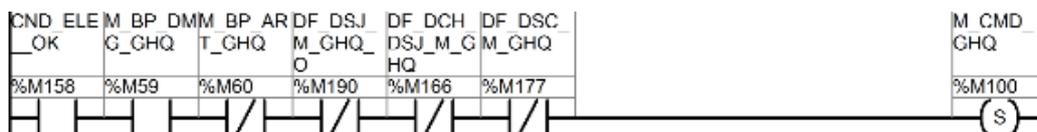


-si %M160=0 absence 24v et %M161=0 absence220v c.-à-d. le courant passe normale et il y'a pas arrêt urgence général contacteur inverser, alors la condition électrique %M158=1 est exécutables

3.2.3. L'étape : commande démarrage groupe hydraulique

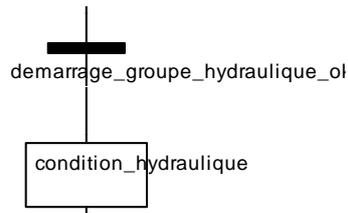


Pour que la commande du groupe hydraulique soit exécutable il faut : les conditions électrique, marche BP groupe hydraulique,

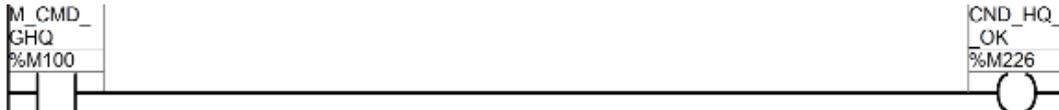


-si la condition électrique s'exécute %M158=1, appui sur marche bp de démarrage du groupe hydraulique %M59=1, il y'a pas un appui sur l'arrêt du groupe hydraulique%M60=0, pas des défauts disjoncteur du groupe hydraulique%M190=0, %M166=0, %M177=0 alors %M100 =1 exécutable

3.2.4. L'étape : condition hydraulique :

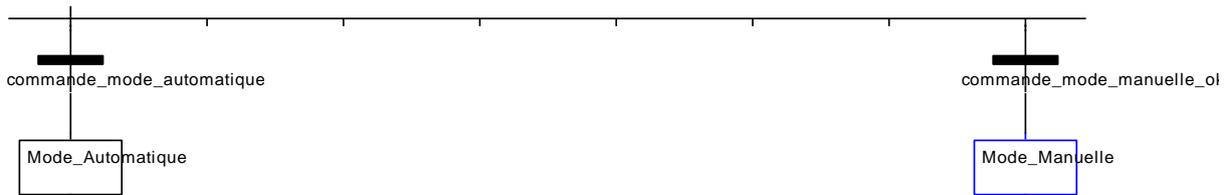


Commande de démarrage du groupe hydraulique.



Si %M100=1 la commande groupe hydraulique et exécutable alors %M226 =1 la condition hydraulique et exécutable pour passer à l'étape de choix du mode

4.2.5. L'étape : mode de fonctionnement



-le choix de fonctionnement (mode automatique ou mode manuelle)

a. Mode manuelle :



-si on choisit le mode manuelle c.-à-d. %M61=1 marche commutateur manuelle, le commutateur marche automatique %M62=0, pour que le mode manuelle %M156=1 soit exécutable.

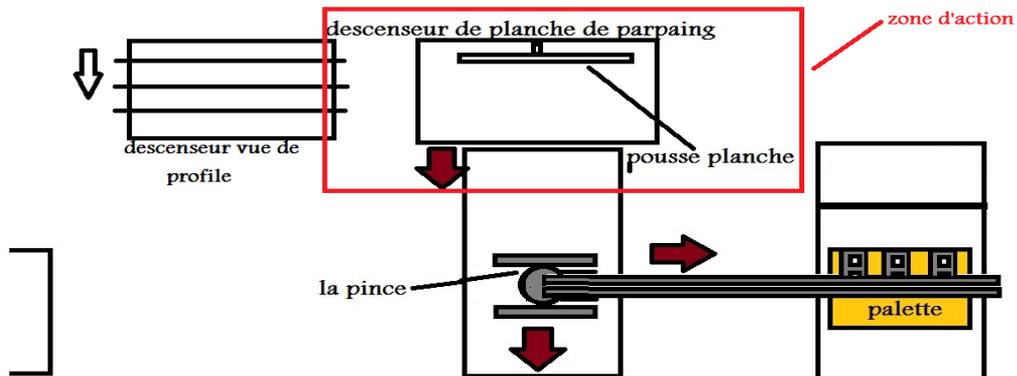
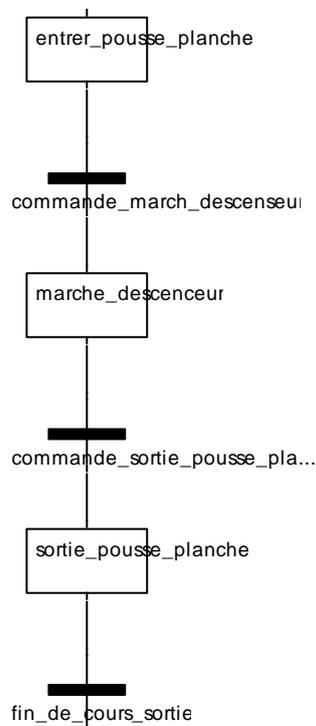
b. Mode automatique

-démarrage du cycle automatique.

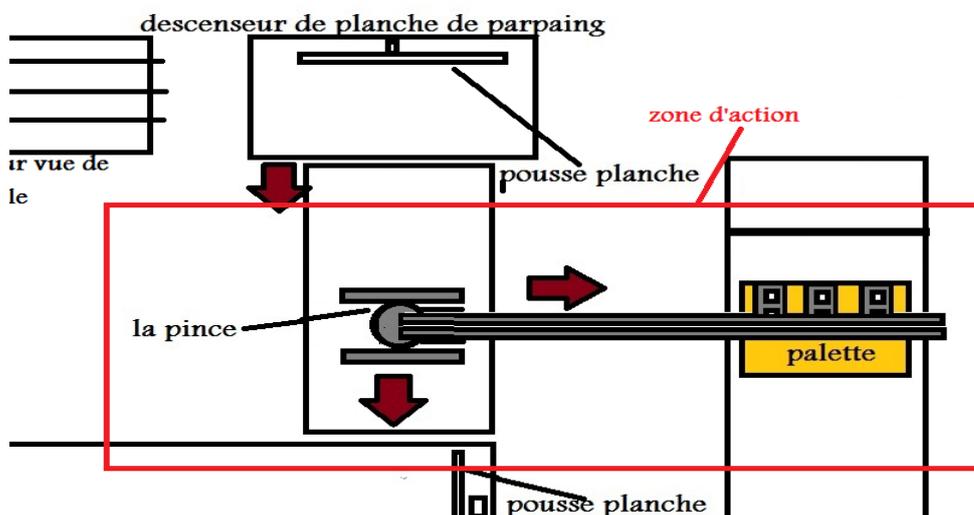
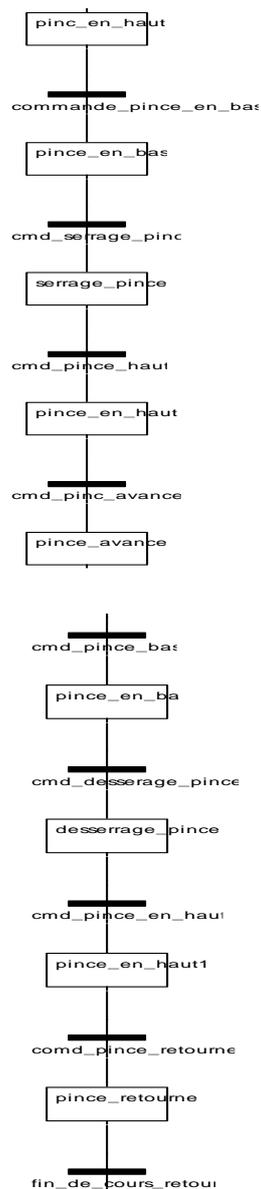


-si on choisit le mode automatique, %M62=1 commutateur commande automatique est en 1 et l'inverse pour le commutateur commande manuelle %M61=0, alors %M64=1 démarrage cycle automatique.

3.2.6. L'étape : commande marche descenseur et le pousse planche



4.2.7. L'étape : déplacement, serrage et desserrage de la pince



4.2.7.1. Pince en haut

Pour que la pince soit en haut il faut, la commande de la pince en haut %M108=1, il y'a pas de fin cours en haut %M41=0,

4.2.7.2. Pince en bas

Si il y'a une présence de produit sous la pince, %M32=1 capteur photocellule, le pousse planche revient en arrière, puis la pince décent pour serrer le produit

4.2.7.3. Serrage pince

Une fois la pince il y'a un fin de cours de la pince en bas %M40=1, la pince commence le serrage de produit

4.2.7.4. Pince en haut

Il y'a pas de fin de cours pince en haut M%41=0, la pince monte en haut si le serrage et terminer, alors la commande pince en haut s'exécute

4.2.7.5. Avance pince

La pince monte, une détection de fin de cours en haut %M41, et avance vers la palette

4.2.7.6. Pince en bas

La pince décent en pas quand il y'a une détection de fin de cours avance la pince %M41=1 elle s'exécute la commande de pince en bas

4.2.7.7. Desserrage pince

Il y'a un fin de cours de pince en bas, elle s'exécute la commande de desserrage le produit

4.2.7.8. Pince en haut

La pince monte en haut

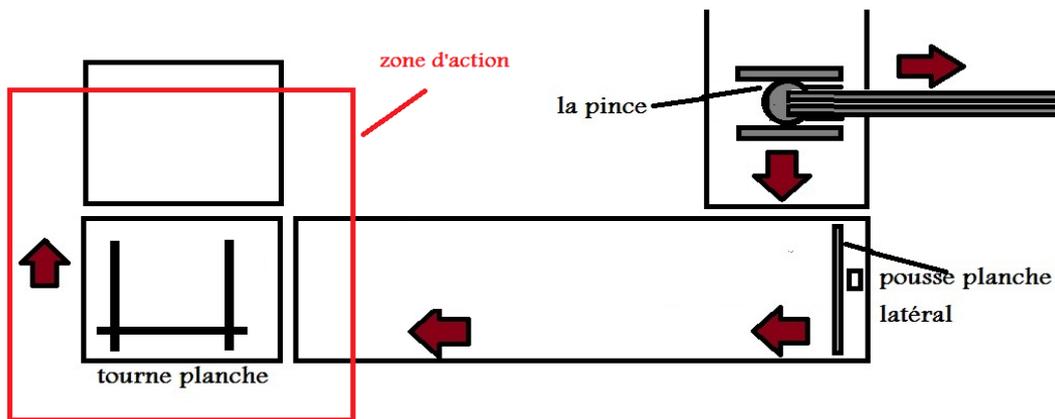
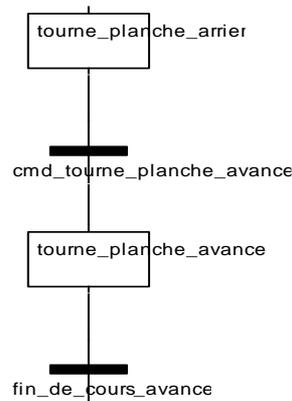
4.2.7.9. Retour pince

La pince retourne dans sont état initiale et attendre une présence de produit

Chapitre3: Etude et programmation de procédés de palatisation

Cette commande s'exécute si, il y'a pas fin de cours avance pousse planche latéral %M34, pas de présence une planche dans le tourne planche %M36, et il y'a un fin retour tourne planche %M35, alors le pousse planche avance et pousse la planche vers le tourne planche

4.2.9. L'étape : tourne planche

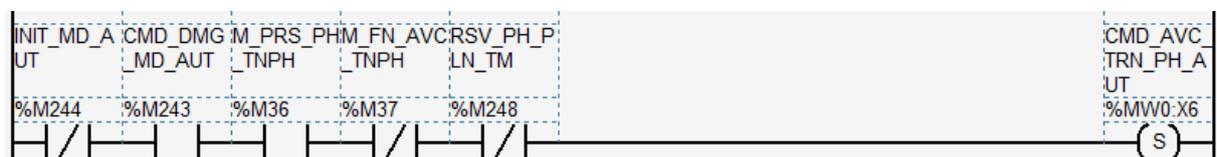


4.2.9.1. Tourne planche arrière



Le tourne planche revient en arrière si, il y'a fin avance tourne planche, capteur avance %M37=1 et il y'a pas fin retour tourne planche en arrière, capteur retour %M38=0

4.2.9.2 Tourne planche avance



Chapitre3: Etude et programmation de procédés de palatisation

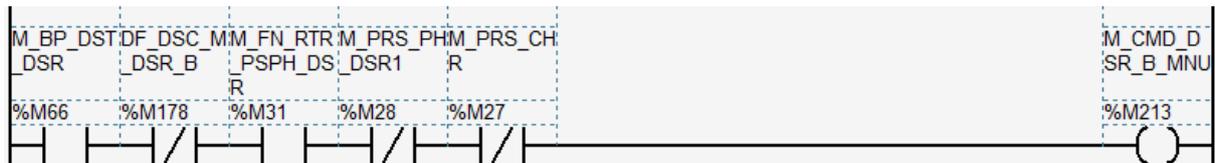
Si il y'a une présence de planche sur le tourne planche %M36=1, et le capteur de fin avance tourne planche à l'état 0, %M37=0, alors le tourne planche avance pour ranger la planche.

4.2.10. Mode manuelle :

Dans la commande du mode manuelle, on a le démarrage et arrêt de chaque actionneur.

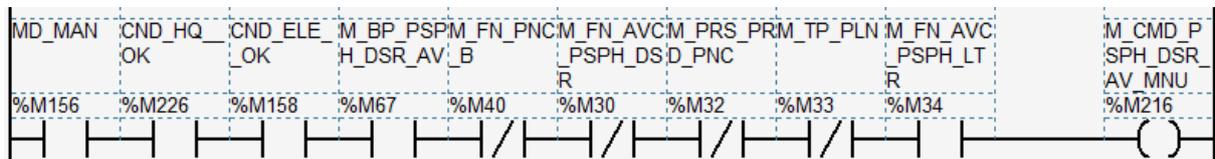
Exemple de quelque commande manuelle :

Commande descenseur en bas



Pour commander le descenseur en bas manuel, on pousse sur le bouton poussoir de commande en bas %M66=1,

Commande avance pousse planche :



Un appui sur le bouton marche avance pousse planche %M67, les conditions hydraulique et électrique sont exécutable, le pousse planche avance

Chapitre3: Etude et programmation de procédés de palatisation

4.3. Les défauts : le traitement et l'acquittement des défauts :

Pour le traitement des défauts, on va créer des mémoires %M les conditions pour le déclenchement des défauts, et ces mémoires seront adressé dan l'afficheur sur des adresses variables %MW* : X*, on à 22 défaut à traiter

DEFAUT ABSENCE 24 VDC
DEFAUT ABSENCE 220 VAC
DEFAUT DECLANCHEMENT DISJONCTEUR MOTEUR TRANSLATION
DEFAUT DECLANCHEMENT DISJONCTEUR MOTEUR GROUPE HYDRAULIQUE
DEFAUT DECLANCHEMENT DISJONCTEUR MOTEUR DESCENSEUR
DEFAUT DECLANCHEMENT DISJONCTEUR MOTEUR TOURNE PLANCHE
DEFAUT DISCORDANCE MOTEUR TRANSLATION AVANT
DEFAUT DISCORDANCE MOTEUR TRANSLATION ARRIERE
DEFAUT DISCORDANCE MOTEUR GROUPE HYDRAULIQUE
DEFAUT DISCORDANCE MOTEUR DESCENSEUR BAS
DEFAUT DISCORDANCE MOTEUR TOURNE PLANCHE AVANT
DEFAUT DISCORDANCE MOTEUR TOURNE PLANCHE ARRIERE
DEFAUT DISJONCTEUR MOTEUR TRANSLATION
DEFAUT DISJONCTEUR MOTEUR GROUPE HYDRAULIQUE OUVERT
DEFAUT DISJONCTEUR MOTEUR DESCENSEUR OUVERT
DEFAUT DISJONCTEUR MOTEUR TOURNE PLANCHE OUVERT
DEFAUT INCOHERENCE POUSSE PLANCHE DESCENSEUR
DEFAUT INCOHERENCE POUSSE PLANCHE LATERALE
DEFAUT INCOHERENCE TOURNE PLANCHE
DEFAUT INCOHERENCE MONTER DESCENTE PINCE
DEFAUT INCOHERENCE TRANSLATION
DEFAUT INCOHERENCE MODE FONCTIONNEMENT

La zone de traitement de ces défauts commencent (de %M160 à %M201) il y'a des mémoires vide entre ces 40 %M.

Et le défaut ce traite comme suit : (exemple défaut absence 24v)



Chapitre3: Etude et programmation de procédés de palatisation

Ce défaut sera affiché sur l'afficheur IHM, si on appui sur le bouton acquittement sur l'afficheur un acquittement de reset s'effectuer comme suit :



5. Description des sections du programme dans logiciel TwidoSoft : (voir le programme complet dans l'annexe)

5.1. Mise à l'heure de l'XBT :

Pour régler l'heure de l'afficheur, envoyée les bits système de l'automate Twido vers l'afficheur

5.2. Affectation des entrées :

Copier les entrées physique dans les entrées analogique, en cas de décalage ou changement des entrés physique il sera ajoure

5.3. Les conditions électriques :

Les conditions électrique sont exécutables si il y pas défaut absence 24v et défaut absence 220v et le défaut d'arrêt d'urgence générale.

5.4. Les conditions hydrauliques :

Commande de démarrage du groupe hydraulique.

5.5. Commande groupe hydraulique :

Pour que la commande du groupe hydraulique soit exécutable il faut : les conditions électrique, marche BP groupe hydraulique,

5.6. Mode de fonctionnement :

-le choix de fonctionnement (mode automatique ou mode manuelle)

-démarrage du cycle automatique.

-l'initialisations du mode automatique.

5.7. Mode manuelle :

Démarrage et arrêt de chaque actionneur.

5.8. Mode automatique :

Démarrage du cycle automatique, le processus.

5.9. Commande des sorties :

Commande des actionneurs dans le mode automatique ou manuelle, et le verrouillage de sécurité avec un temporisateur.

5.10. Acquiescement des défauts :

L'effacement des défauts et sauvegardé.

5.11. Traitement des défauts :

Les conditions pour le déclenchement d'un défaut.

5.12. Affectation des sorties :

Affectation les sortie physiques.

6. Conclusion :

On peut maintenant considérer que nos recherches et nos études théoriques sont assez suffisantes pour aborder la partie pratique de notre projet. En effet, on a pu bien enrichir nos connaissances sur les automates et la manipulation de programmation et plus précisément avec l'automate TWIDO : TWDLCAA40DRF, 24 entrées, 16 sorties et, évidemment le minimum de connaissance pour une bonne réalisation du typon.

Donc il ne nous reste qu'à disposer de composants adaptée à notre réalisation et attaqué la réalisation de la commande de cette chaine de production.

A travers ce projet on a su acquérir et l'esprit du travail en groupe avec tout l'échange d'idées qui vient avec, et la capacité de bien coordonner et répartir les tâches.

Conclusion générale

Conclusion générale

Ce projet de fin d'étude s'inscrit dans le cadre de la matérialisation de notre cursus de la spécialité et ce par le choix d'un thème très intéressant sur l'automatisation d'un procédé industriel. Nous avons choisi une chaîne de production de palettisation de parpaing qui se trouve à la société HM Automatismes- DYLLES.

Nous avons remplacé la commande existante à base de logique câblée qui n'est plus utilisée de nos jours, et qui présente l'inconvénient majeur pour la commande de l'installation, vu le manque de pièces de rechange et des constituants associés, par une commande automatisée programmable.

Dans ce travail nous avons entrepris l'étude et l'analyse de tous les modes de fonctionnement du procédé. Ensuite nous avons présenté l'automate envisagé pour la tâche d'automatisation qui est le Compact 40E/STWDLCAA40DRF de la marque Schneider. Le logiciel de supervision à base de TwidoSuite a été aussi utilisé. En se focalisant sur la partie de programmation de l'automate et l'élaboration des interfaces HMI pour la supervision, nous avons développé avec réussite une application assurant les différentes fonctionnalités de processus, répondant ainsi parfaitement aux exigences de cahier de charges.

Ce projet nous a permis de voir les différents avantages offerts par les automates programmables et les systèmes de supervision de la famille Twido de Schneider et de les appliquer sur le procédé de chaîne de palettisation. Munis d'un logiciel de programmation très performant et simple à utiliser, TwidoSuit, nous avons élaboré notre programme utilisateur et configuré les fonctions du diagnostic. Nous avons développé les interfaces de supervision HMI avec XBT N 400 pour une meilleure exploitation de système.

Ce mémoire constitue pour nous un complément indispensable pour la formation de Master, car il nous a permis une transition facile de la formation universitaire au domaine professionnel. C'est dans cet environnement de la diversité technologique des différentes installations que nous avons travaillé tout en profitant de l'expérience des travailleurs de l'entreprise pour mieux comprendre le processus.

Ayant atteint l'objectif assigné, nous espérons que notre projet verra sa concrétisation sur le terrain et apportera un plus au savoir et au progrès au sein de notre université.

Enfin, nous espérons que ce travail servira comme base de départ pour notre vie professionnelle et sera bénéfique aux promotions futures.

Bibliographie

Bibliographie

Ouvrage :

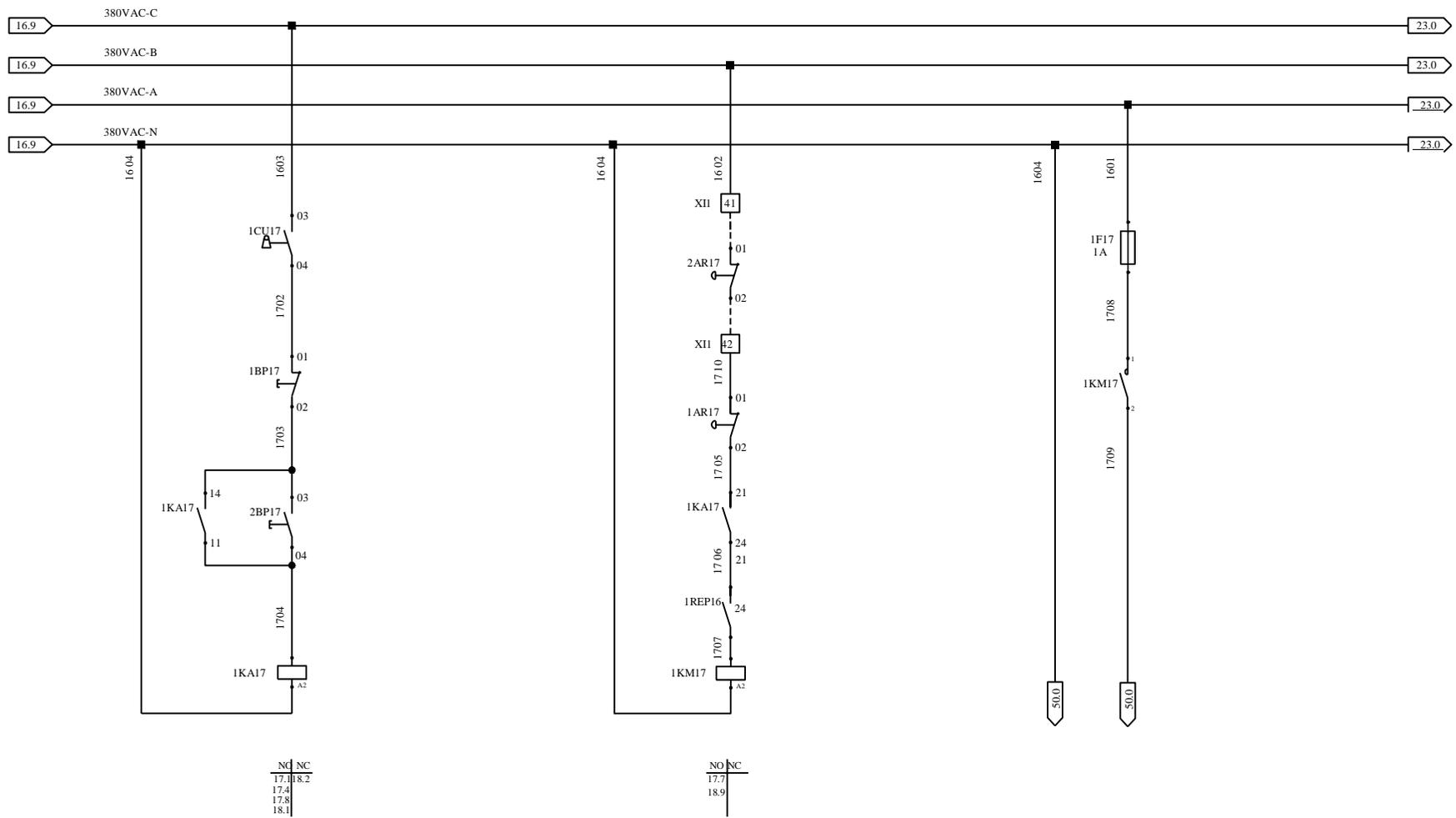
- [1] « G.MICHEL » Les API, architecture et application des automates programmables industriels. Edition 1988. DUNOD.
- [2] Norme CEI 1131-3. Automates programmables - Partie 1 : Les automates programmables, Commission électrotechnique internationale, 1993.
- [3] Twido automate programmable, Guide Matériel -Bases Compactes & Modulaires, Schneider électrique, 09/2007.
- [4] terminaux compact Magelis XBT N/XBT R, guide d'exploitation, Schneider électrique octobre 2006.
- [5] Télémécanique protocole Mitsubishi FX, guide instruction de service XBT L1000, Télémécanique, 07/2006.
- [6] ADEPA/AF CET. Le GRAFCET : ouvrage de référence. Collection Automatisation et Production, Cepadues Editions, 1991.
- [7] TwidoSuite V2.0 logiciel de programmation, Mise en route, Schneider électrique, 09/2007
- [8] Automatisation et supervision d'une station de dessalement d'eau de mer à base d'API S7-400 et du WinCC-RT-Professional du système SCADA- Mémoire de fin d'études-Master- (Ali ALAMRANI-FHC 2020).

[9] Sites internet :

- <http://www.automation-sense.com>.
- <https://support.industry.siemens.com/cs/start?lc=fr-FR>
- <https://www.wika.fr/>
- <https://www.dz.endress.com/fr>
- <https://realpars.com/>
- <http://www.schneiderlectric.com/>
- <http://www.telemecanique.com/>

ANNEXE

MIS SOUS TENTION	ARRET D'URGENCE GENERALE	ALIMENTATION DES SORTIE 230VAC
------------------	--------------------------	--------------------------------



Algerie Solution Automatismes

	DATE	Dessiner par :	Approuvé par :		
B					
A					
0	29/12/2013	S.GUETTARA	A.MEBTOUCHE	OK	

CIRCUIT DE COMMANDE

PUPITRE DE COMMANDE

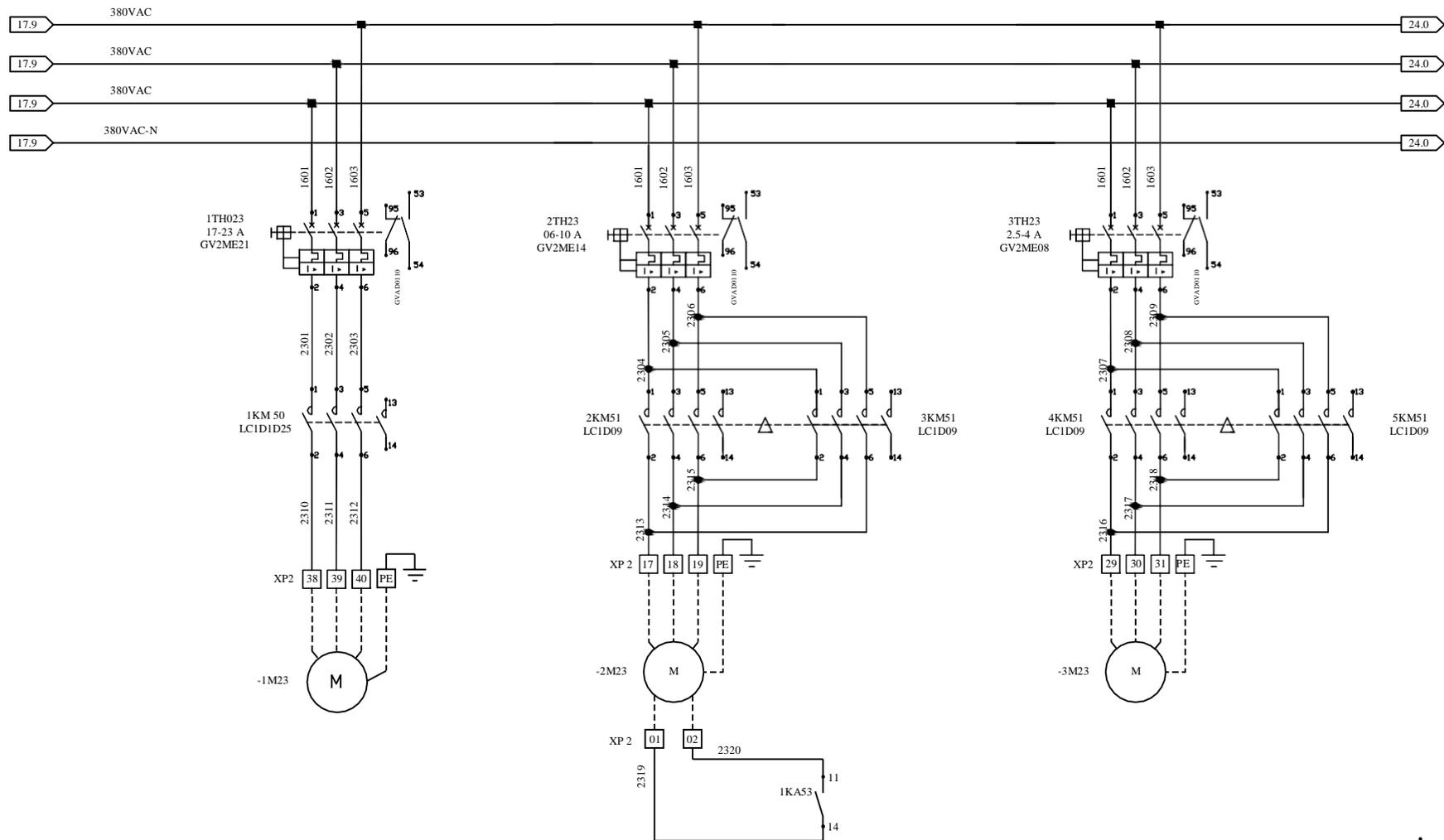
SAPCI BAGHLIA			
RENOVATION ARMOIRE DE COMMANDE			
VERSION	0	COMMANDE N°	SCHEMA N°

FOLIO
17

MOTEUR DE LA POMPE
HYDRAULIQUE

MOTEUR DE
DESCENSEUR

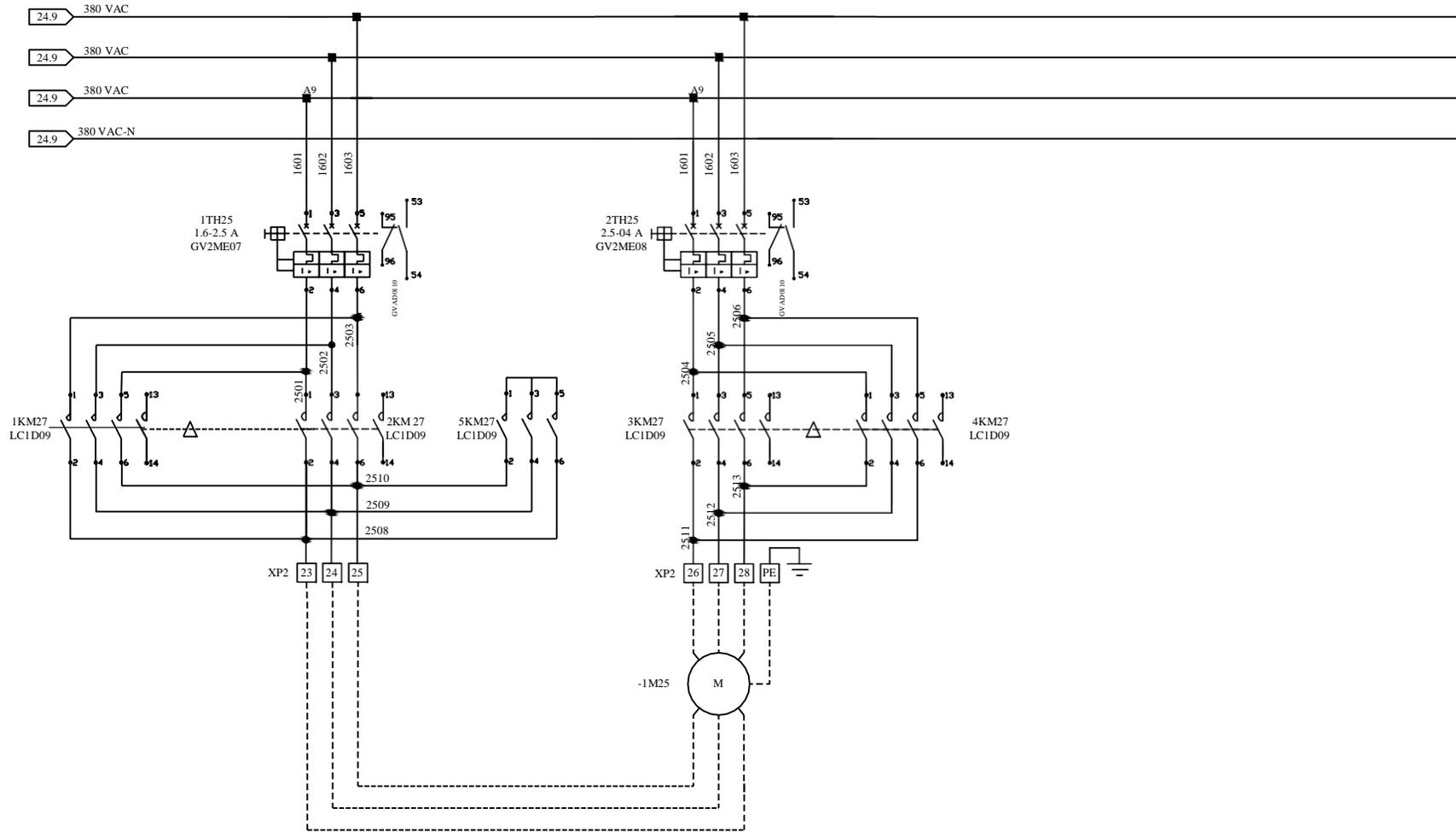
MOTEUR DE TOURNE
PLANCHE



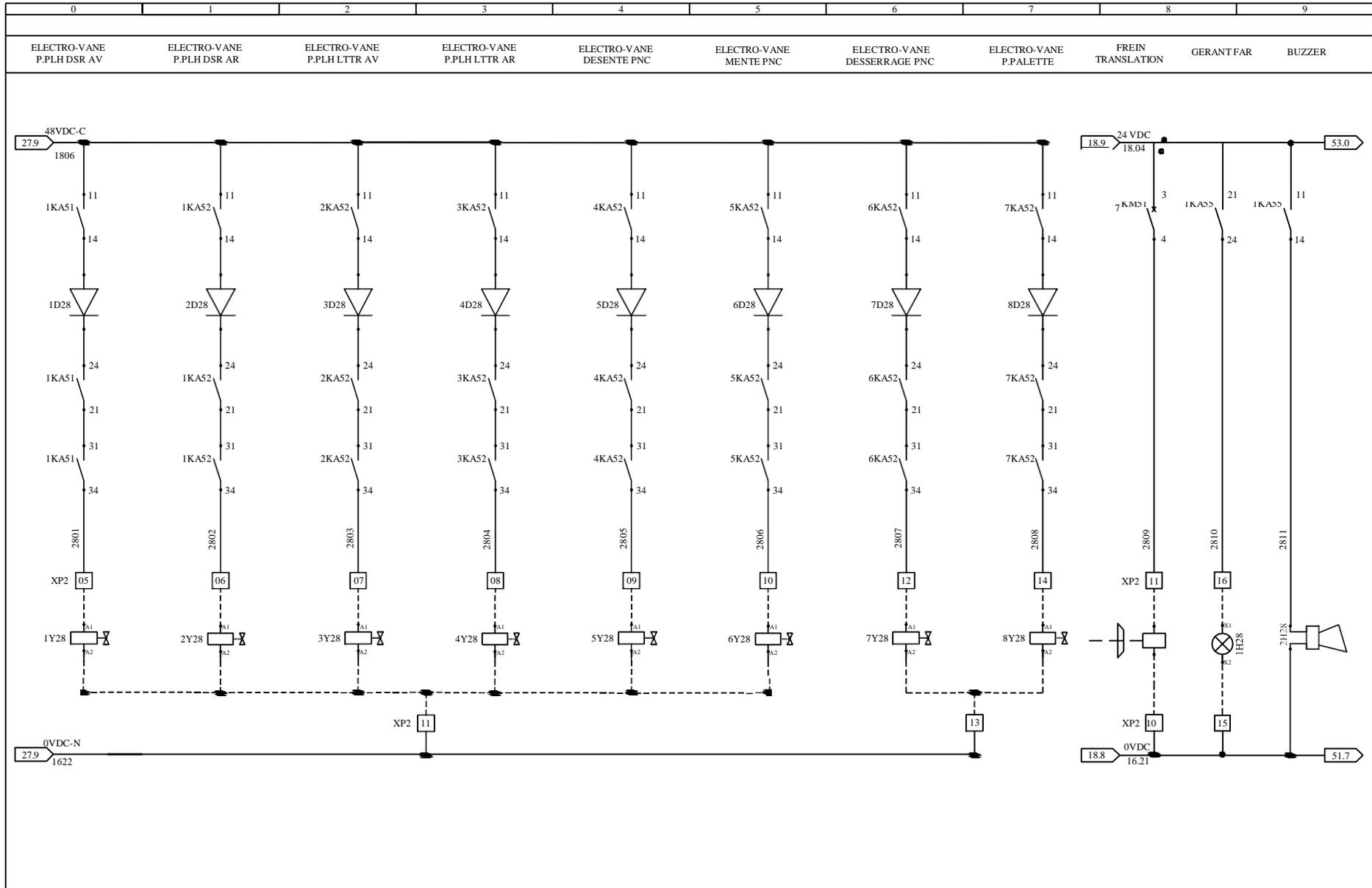
 Algerie Solution <i>Automatisme</i> <small>Algerie Solution Automatismes</small>	DATE	Dessiner par :	Approuvé par :	SAPCI BAGHLIA				FOLIO	
	B			CIRCUIT DE PUISSANCE				23	
A			PUPITRE DE COMMANDE				RENOVATION ARMOIRE DE COMMANDE		
0	29/12/2013	S.GUETTARA	A.MEBTOUCHE	OK	VERSION	0	COMMANDE N°	SCHEMA N°	24

MOTEUR DE TRANSLATION
LENTE

MOTEUR DE TRANSLATION
ACCELERER



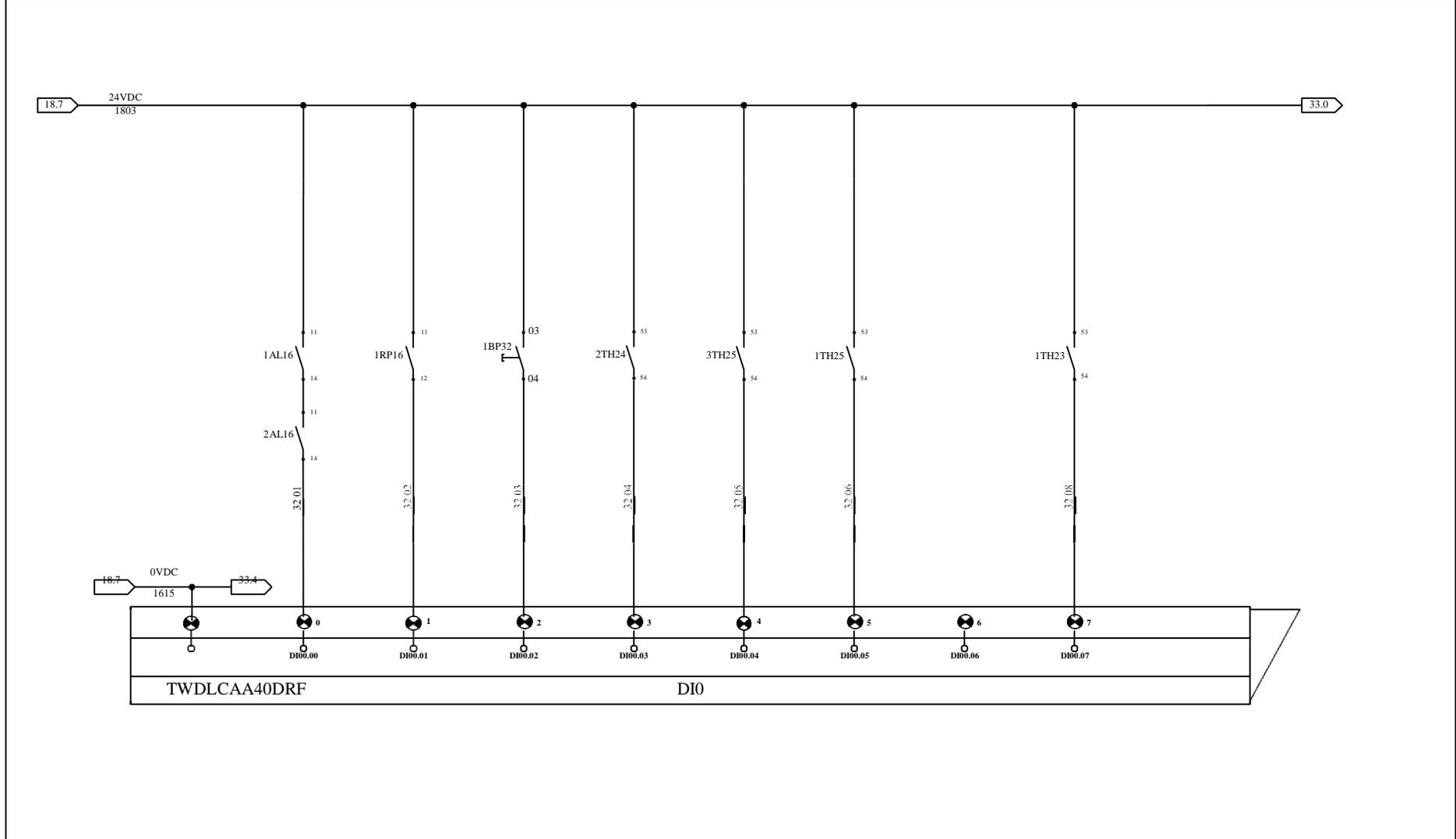
<p style="font-weight: bold; margin-top: 5px;">Algerie Solution Automatisme</p>		DATE	Dessiner par :	Approuvé par :	CIRCUIT DE PUISSANCE	SAPCI BAGHLIA				FOLIO	
	B						RENOVATION ARMOIRE DE COMMANDE				25
	A										
CE DOCUMENT EST LA PROPRIETE DE ASA IL NE PEUT ETRE NI COPIE NI COMMUNIQUÉ A DES TIERS SANS NOTRE AUTORISATION	0	29/12/2013	S.GUETTARA	A.MEBTOUCHE	OK	PUPITRE DE COMMANDE	VERSION	0	COMMANDE N°	SCHEMA N°	27



 Algerie Solution Automatism Algerie Solution Automatism	DATE	Dessiner par :	Approuvé par :	CIRCUIT DE COMMANDE	SAPCI BAGHLIA				FOLIO					
	B					RENOVATION ARMOIRE DE COMMANDE				28				
	A					PUPITRE DE COMMANDE	VERSION	0	COMMANDE N°	SCHEMA N°	30			
CE DOCUMENT EST LA PROPRIETE DE ASA IL NE PEUT ETRE NI COPIE NI COMMUNIQUE A DES TIERS SANS NOTRE AUTORISATION					0	29/12/2013	S.GUETTARA	A.MEBTOUCHE	OK					

0	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---

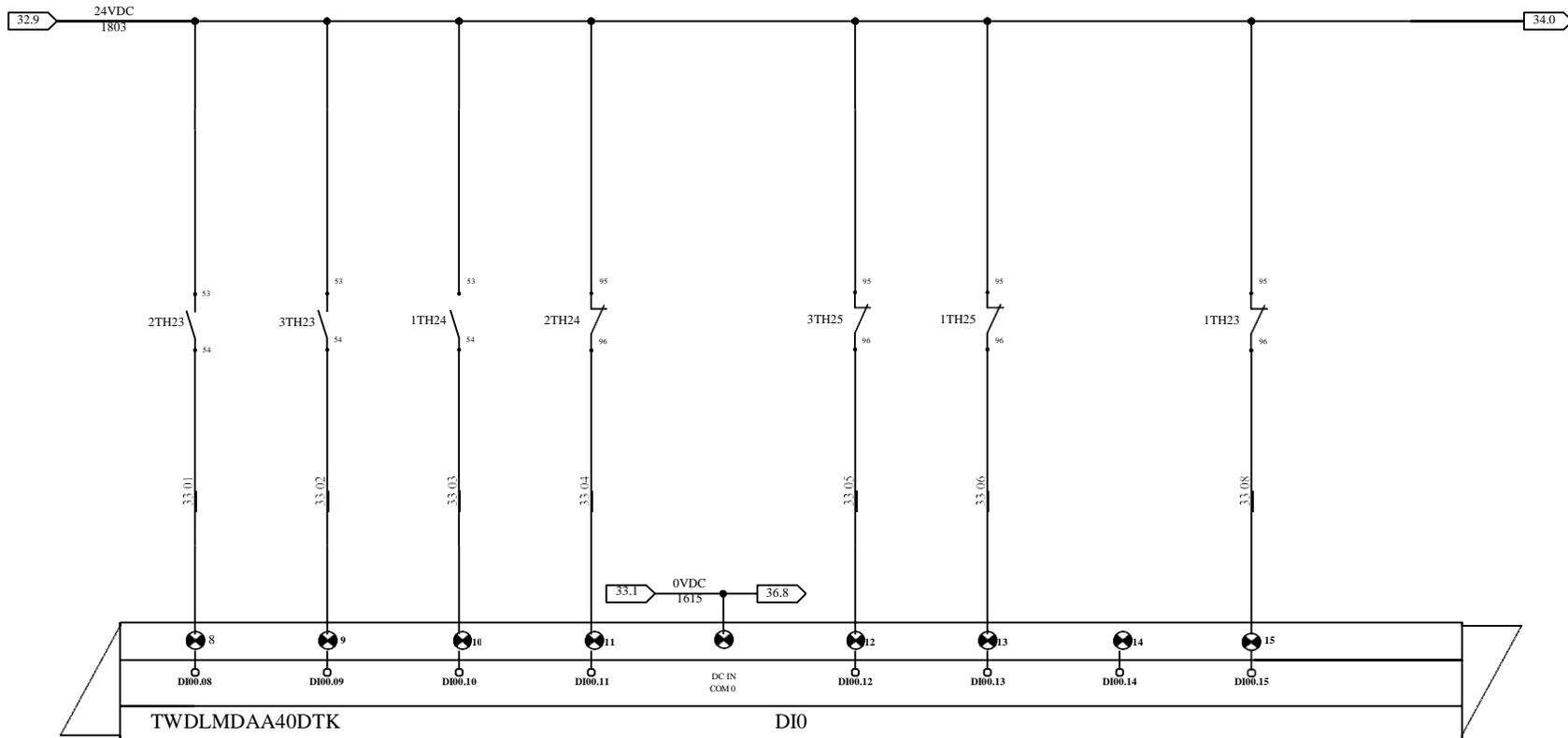
ABSENCE 24V DC	ABSENCE 220V AC	TESTE LAMPES	DISJONCTEUR MOTEUR DE ROTATION OUVERT/FERMER	DISJONCTEUR MOTEUR TRANSLATION LENTE OUVERT/FERMER	DISJONCTEUR MOTEUR TRANSLATION ACCELERER OUVERT/FERMER	RESERVE	DISJONCTEUR MOTEUR HYDROLIQUE OUVERT/FERMER
-------------------	--------------------	--------------	---	--	--	---------	--



 Algerie Solution Automatisme <small>Algerie Solution Automatismes</small>	DATE		Dessiner par :		Approuvé par :		SAPCI BAGHLIA				FOLIO					
	B						RENNOVATION ARMOIRE DE COMMANDE				32					
	A						PUPITRE DE COMMANDE				33					
<small>CE DOCUMENT EST LA PROPRIETE DE ASA IL NE PEUT ETRE NI COPIE NI COMMUNIQUE A DES TIERS SANS NOTRE AUTORISATION</small>							0	29/12/2013	S.GUETTARA	A.MEBTOUCHE	OK	VERSION	0	COMMANDE N°	SCHEMA N°	33

0	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---

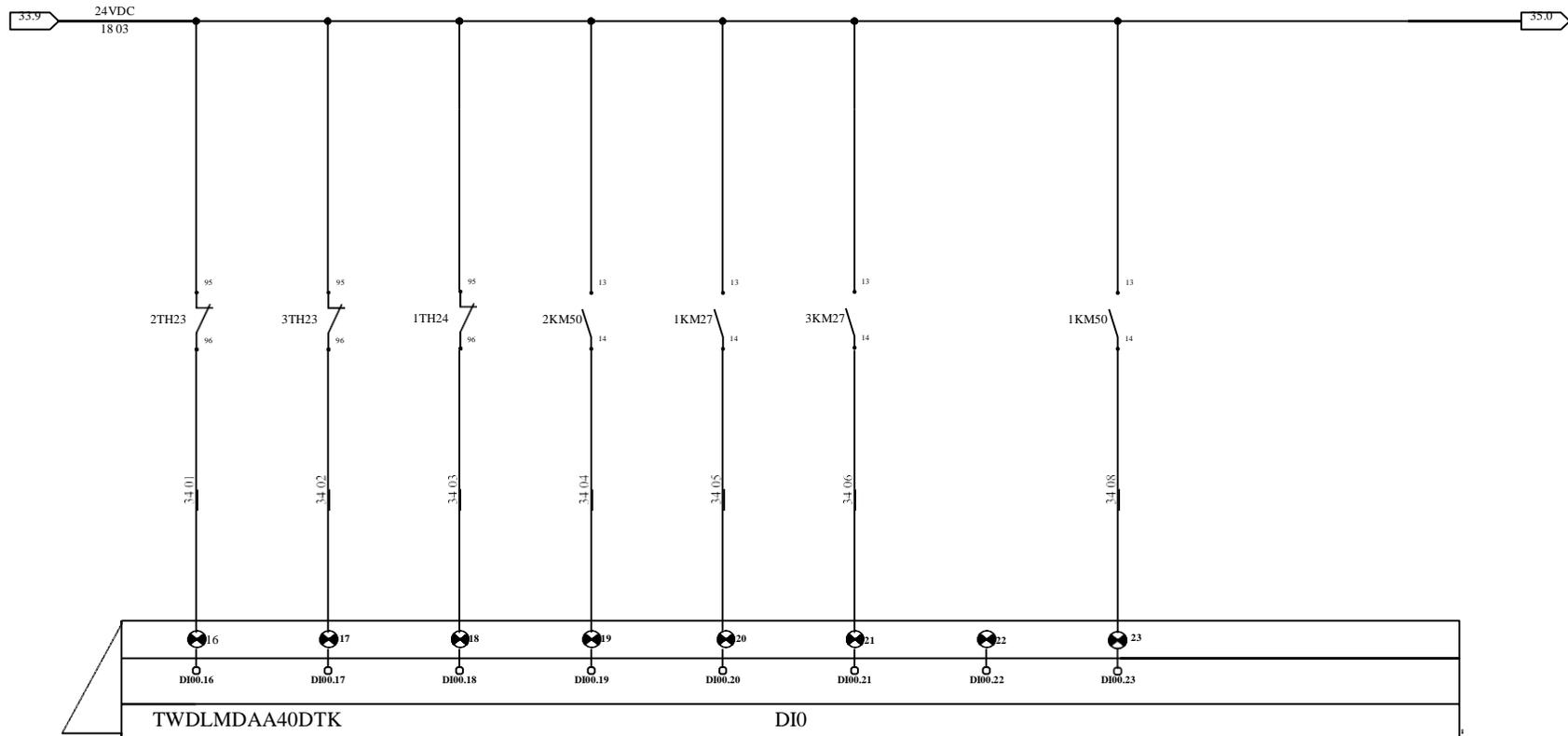
DISJONCTEUR MOTEUR DESCENSEUR OUVERT/FERMER	DISJONCTEUR MOTEUR TOURNE PLANCHE OUVERT/FERMER	DISJONCTEUR MOTEUR BROSSE OUVERT/FERMER	DECLENCHEMENT DISJONCTEUR MOTEUR DE ROTATION	DECLENCHEMENT DISJONCTEUR MOTEUR TRANSLATION LENTE	DECLENCHEMENT DISJONCTEUR MOTEUR TRANSLATION ACCELERER	RESERVE	DECLENCHEMENT DISJONCTEUR MOTEUR G.HYDROLIQUE
--	--	--	---	--	--	---------	--



 Algerie Solution Automatismes <small>Algerie Solution Automatismes</small>	<table border="1"> <tr> <td></td> <td>DATE</td> <td>Dessiner par :</td> <td>Approuvé par :</td> </tr> <tr> <td>B</td> <td></td> <td></td> <td></td> </tr> <tr> <td>A</td> <td></td> <td></td> <td></td> </tr> </table>		DATE	Dessiner par :	Approuvé par :	B				A				<table border="1"> <tr> <td colspan="2">LES ENTRES DI0 D'AUTOMATE</td> <td colspan="2">SAPCI BAGHLIA</td> <td>FOLIO</td> </tr> <tr> <td colspan="2"></td> <td colspan="2">RENOVATION ARMOIRE DE COMMANDE</td> <td>33</td> </tr> </table>				LES ENTRES DI0 D'AUTOMATE		SAPCI BAGHLIA		FOLIO			RENOVATION ARMOIRE DE COMMANDE		33
			DATE	Dessiner par :	Approuvé par :																						
		B																									
A																											
LES ENTRES DI0 D'AUTOMATE		SAPCI BAGHLIA		FOLIO																							
		RENOVATION ARMOIRE DE COMMANDE		33																							
<table border="1"> <tr> <td>0</td> <td>29/12/2013</td> <td>S.GUETTARA</td> <td>A.MEBTOUCHE</td> <td>OK</td> </tr> </table>				0	29/12/2013	S.GUETTARA	A.MEBTOUCHE	OK	<table border="1"> <tr> <td colspan="2">PUPITRE DE COMMANDE</td> <td>VERSION</td> <td>0</td> <td>COMMANDE N°</td> <td>SCHEMA N°</td> <td>34</td> </tr> </table>				PUPITRE DE COMMANDE		VERSION	0	COMMANDE N°	SCHEMA N°	34								
0	29/12/2013	S.GUETTARA	A.MEBTOUCHE	OK																							
PUPITRE DE COMMANDE		VERSION	0	COMMANDE N°	SCHEMA N°	34																					
<small>CE DOCUMENT EST LA PROPRIETE DE ASA IL NE PEUT ETRE NI COPIE NI COMMUNIQUE A DES TIERS SANS NOTRE AUTORISATION</small>																											

0	1	2	3	4	5	6	7	8	9
---	---	---	---	---	---	---	---	---	---

DISJONCTEUR MOTEUR DESCENSEUR OUVERT/FERMER	DISJONCTEUR MOTEUR TOURNE PLANCHE OUVERT/FERMER	DISJONCTEUR MOTEUR BROSSE OUVERT/FERMER	RETOUR MARCHE MOTEUR DE ROTATION GAUCHE	RETOUR MARCHE MOTEUR TRANSLATION LENTE AVANT	RETOUR MARCHE MOTEUR TRANSLATION ACCELERER ANANT	RESERVE	RETOUR MARCHE MOTEUR G.HYDROLIQUE
--	--	--	--	---	--	---------	---



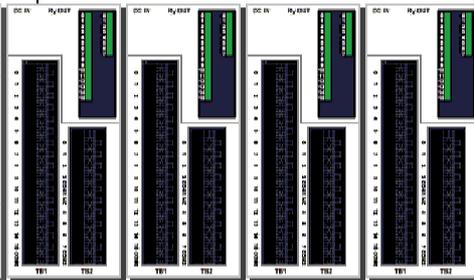
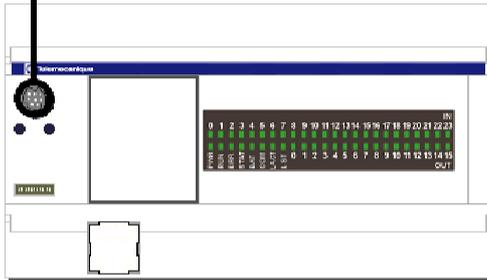
 Algerie Solution Automatisme <small>Algerie Solution Automatismes</small>	DATE		Dessiner par :		Approuvé par :		SAPCI BAGHLIA			FOLIO			
	B						LES ENTREES DI0 D'AUTOMATE			34			
	A						RENOVATION ARMOIRE DE COMMANDE						
0		29/12/2013	S.GUETTARA		A.MEBTOUCHE	OK	PUPITRE DE COMMANDE		VERSION	0	COMMANDE N°	SCHEMA N°	35

CE DOCUMENT EST LA PROPRIETE DE ASA IL NE PEUT ETRE NI COPIE NI COMMUNIQUE A DES TIERS SANS NOTRE AUTORISATION

TwidoSuite PROGRAMME PAUSE

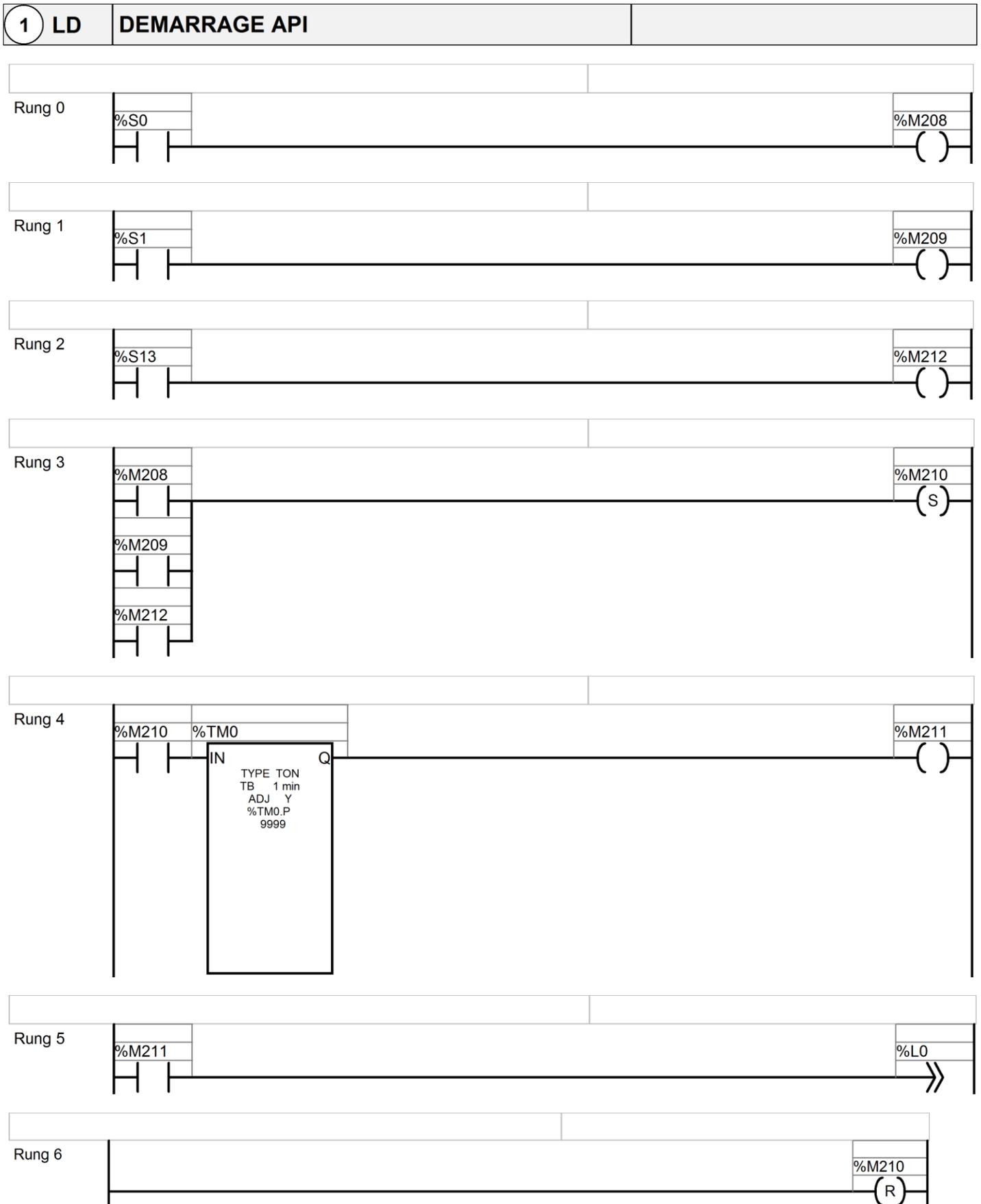


Maitre Mon Twido





Programme listes et diagrammes





2	LD	MIS A L'HEUR DE L'XBT	
----------	-----------	------------------------------	--

MIS A L'HEUR DE L'XBT	
------------------------------	--

Rung 0

SHORT

	%MW102 := SHL(%SW50, 8)
	%MW102 := SHL(%SW50, 8)

	%MW103 := %SW51
	%MW103 := %SW51

	%MW104 := %SW52
	%MW104 := %SW52

	%MW105 := %SW53
	%MW105 := %SW53

3	LD	AFFECTATION DES ENTREES	
----------	-----------	--------------------------------	--

Rung 0

ABS_24_V

%I0.0



	M_ABS_24_V
	%M0



Rung 1

ABS_220V

%I0.1



	M_ABS_220V
	%M1



Rung 2

TEST_LMP

%I0.2



	M_TEST_LMP
	%M2



Rung 3

DSJ_M_RTN_O_F

%I0.3



	M_DSJ_M_RTN_O_F
	%M3



Rung 4

DSJ_M_TL_O_F

%I0.4



	M_DSJ_M_TL_O_F
	%M4



Rung 5

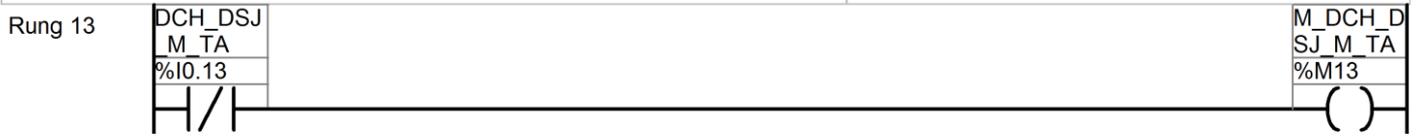
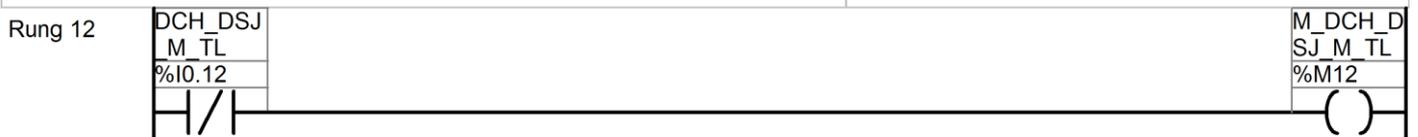
DSJ_M_TAO_F

%I0.5



	M_DSJ_M_TAO_F
	%M5



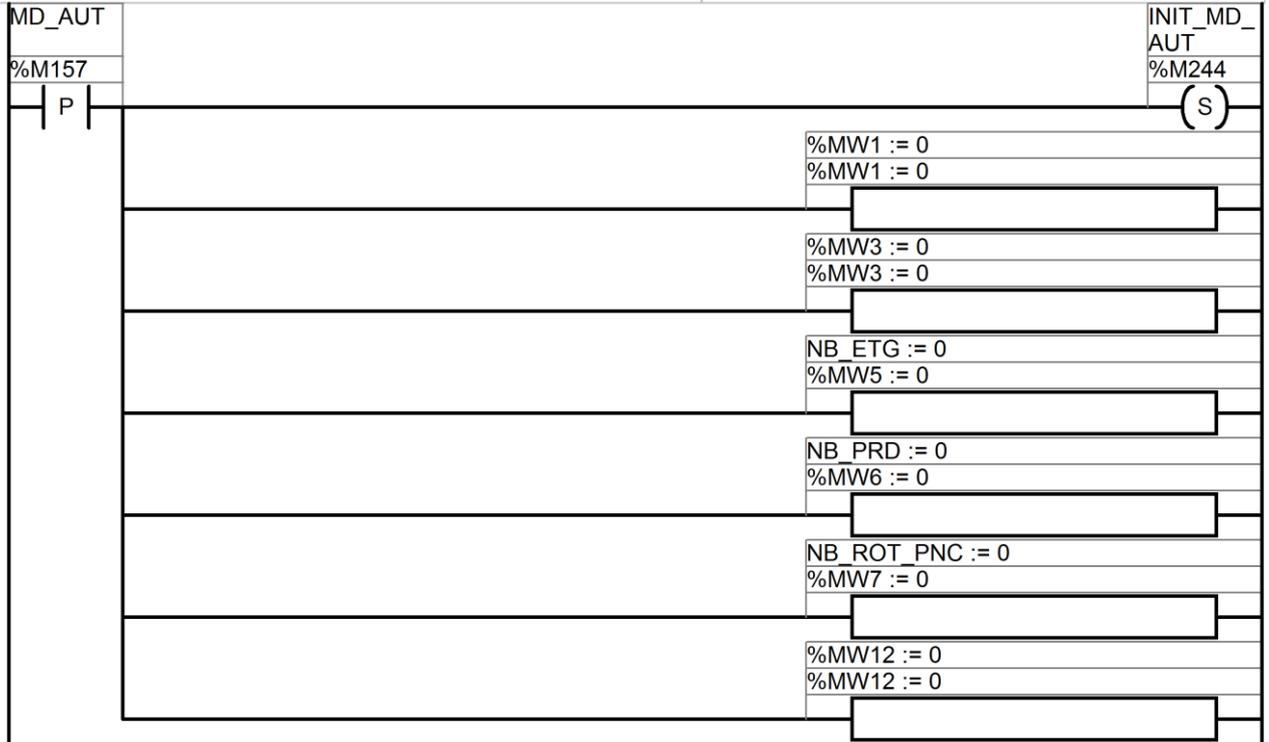


**ARRET DU MODE AUTO**

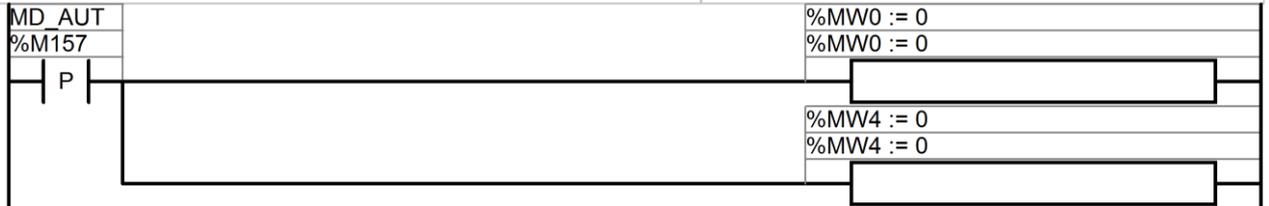
Rung 4

**INITIALISATION MODE AUTOMATIQUE**

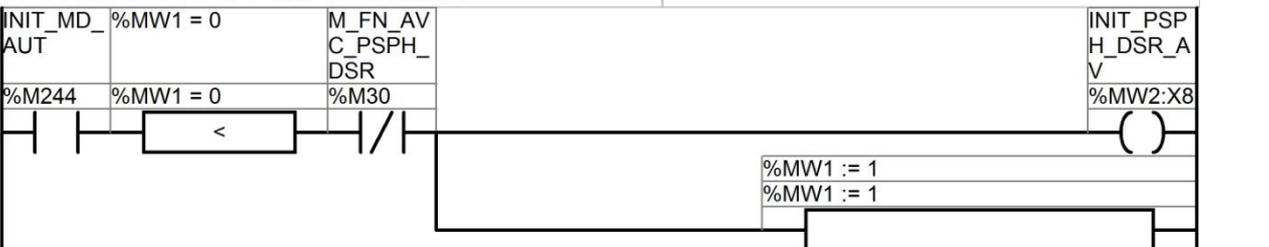
Rung 5

**INITIALISATION MODE AUTOMATIQUE**

Rung 6

**INIT POUSSE PLANCHE DESCENSEUR AVANT**

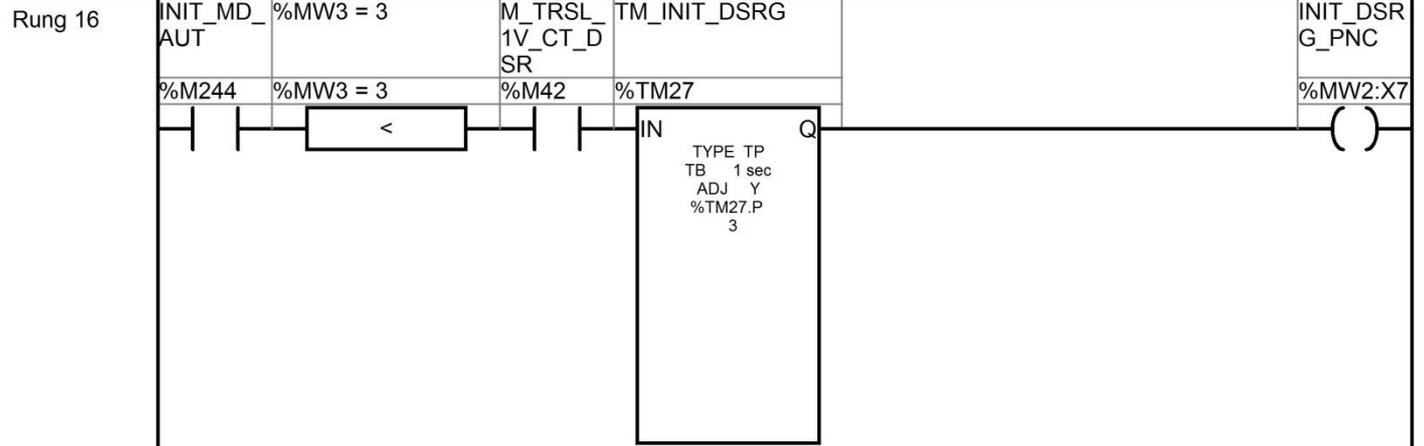
Rung 7



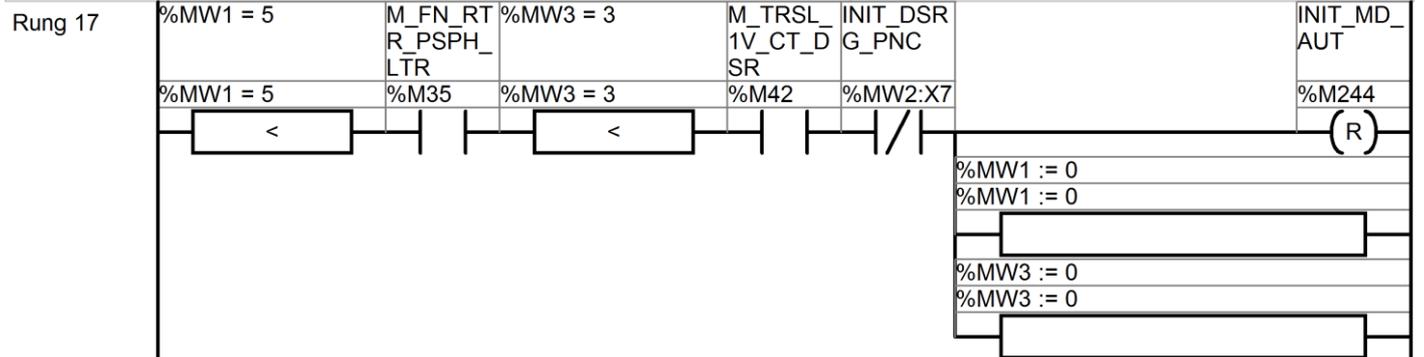


8	LD	MODE MANUEL	
----------	-----------	--------------------	--

INIT PINCE DESSERRAGE



FIN INIT



**COMMANDE DESCENSEUR EN HAUT**

Rung 1	CND_MR	M_BP_MN	DF_DSC	M_PRS_C							M_CMD_DSR_H_M
	H_DSR	T_DSR	M_DSR_HHR								NU
	%M215	%M65	%M179	%M27							%M214
	H	H	H	H	/	/	/	/	/	/	()

COMMANDE DESCENSEUR EN BAS

Rung 2	CND_MR	M_BP_DS	DF_DSC	M_FN_RT	M_PRS_P	M_PRS_P	M_PRS_C				M_CMD_DSR_B_M
	H_DSR	T_DSR	M_DSR_B	R_PSPH_DSR	H_DSR1	H_DSR2	HR				NU
	%M215	%M66	%M178	%M31	%M28	%M29	%M27				%M213
	H	H	H	H	/	/	/	/	/	/	()

COMMANDE AVANCE POUSSE PLANCHE DESCENSEUR

Rung 3	MD_MAN	CND_HQ	CND_ELE	M_BP_PS	M_PNC_H	M_FN_AV	M_PRS_P	M_TP_PL	M_FN_RT		M_CMD_P
		_OK	_OK	PH_DSR_AV		C_PSPH_DSR	RD_PNC	N	R_PSPH_LTR		SPH_DSR
	%M156	%M226	%M158	%M67	%M41	%M30	%M32	%M33	%M35		AV_MNU
	H	H	H	H	H	/	/	/	/	/	()

COMMANDE RETOUR POUSSE PLANCHE DESCENSEUR

Rung 4	MD_MAN	CND_HQ	CND_ELE	M_BP_PS	M_FN_RT						M_CMD_P
		_OK	_OK	PH_DSR_AV	R_PSPH_DSR						SPH_DSR
	%M156	%M226	%M158	%M67	%M31						AR_MNU
	H	H	H	H	/	/	/	/	/	/	()

COMMANDE AVANCE POUSSE PLANCHE LATERALE

Rung 5	MD_MAN	CND_ELE	CND_HQ	M_BP_PS	M_FN_AV	M_PRS_P	M_FN_RT	M_TP_PL			CMD_AV
		_OK	_OK	PH_LTR_AV	C_PSPH_LTR	H_TNPH	R_TNPH	N			PSPH_L
	%M156	%M158	%M226	%M69	%M34	%M36	%M38	%M33			TR_MNU
	H	H	H	H	/	/	/	/	/	/	()

COMMANDE RETOUR POUSSE PLANCHE LATERALE

Rung 6	MD_MAN	CND_ELE	CND_HQ	M_BP_PS	M_FN_RT						CMD_RTR
		_OK	_OK	PH_LTR_AV	R_PSPH_LTR						PSPH_L
	%M156	%M158	%M226	%M69	%M35						TR_MNU
	H	H	H	H	/	/	/	/	/	/	()

COMMANDE AVANCE TOURNE PLANCHE

Rung 7	MD_MAN	CND_ELE	M_BP_TN	DF_DSJ	DF_DSC	DF_DCH	M_PRS_P	M_FN_AV	M_RSV_P	M_FN_RT	M_CMD_A
		_OK	PH_AV	M_TNPH_O	M_TNPH_AV	DSJ_M_TNPH	H_TNPH	C_TNPH	H_PLN	R_PSPH_LTR	VC_TNPH_MUN
	%M156	%M158	%M71	%M192	%M180	%M168	%M36	%M37	%M39	%M35	%M220
	H	H	H	H	/	/	/	/	/	/	()

COMMANDE RETOUR TOURNE PLANCHE

Rung 8	MD_MAN	CND_ELE	M_BP_TN	DF_DSJ	DF_DSC	DF_DCH	M_FN_RT				M_CMD_RTR_TNP
		_OK	PH_AV	M_TNPH_O	M_TNPH_AV	DSJ_M_TNPH	R_TNPH				H_MUN
	%M156	%M158	%M71	%M192	%M180	%M168	%M38				%M227
	H	H	H	H	/	/	/	/	/	/	()



9	LD	MODE AUTOMATIQUE	
----------	-----------	-------------------------	--

COMMAMDE MARCHE ROTATION PINCE DROITE

Rung 22	CND_T_R	M_BP_RT	M_FN_RT	DF_DCH	DF_DSC	DF_DSJ		CMD_RTN
	TN_PNC	N_DT_PN	N_DT	DSJ_M_R	M_RTN_D	M_RTN_O		DT_MNU
	%M229	%M78	%M47	%M162	%M171	%M186		%M234

CONDITION DEMARRAGE MODE AUTOMATIQUE 2

Rung 1	DF_DCH	DF_DSC	DF_DSC	CND_DM								
	DSJ_M_R	DSJ_M_T	DSJ_M_T	DSJ_M_V	DSJ_M_G	DSJ_M_D	DSJ_M_T	DSJ_M_B	M_RTN_D	M_RTN_G	G_MD_AU	
	TN	L	A	T	HQ	SR	NPH	RS	T	H	T 2	PSP
%M162	%M163	%M164	%M165	%M166	%M167	%M168	%M169	%M171	%M170	%M239		/_MN
											(S)	

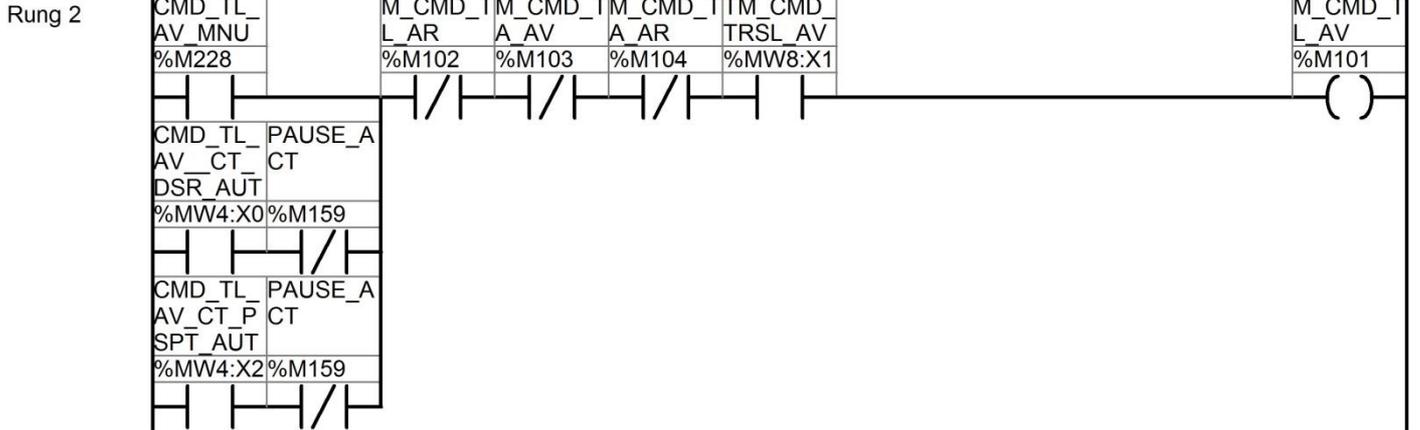
COMMANDE MARCHE TRANSLATION ACCELERER ARRIERE

Rung 19	CND_T_R	CMD_TL	M_TRSL	DF_DCH	DF_DSC	DF_DSJ		CND_TA
	TN_PNC	AR_MNU	2V_CT_P	DSJ_M_T	M_TA_AR	M_TA_O		AR_MNU
	%M229	%M231	%M45	%M164	%M175	%M188		%M232

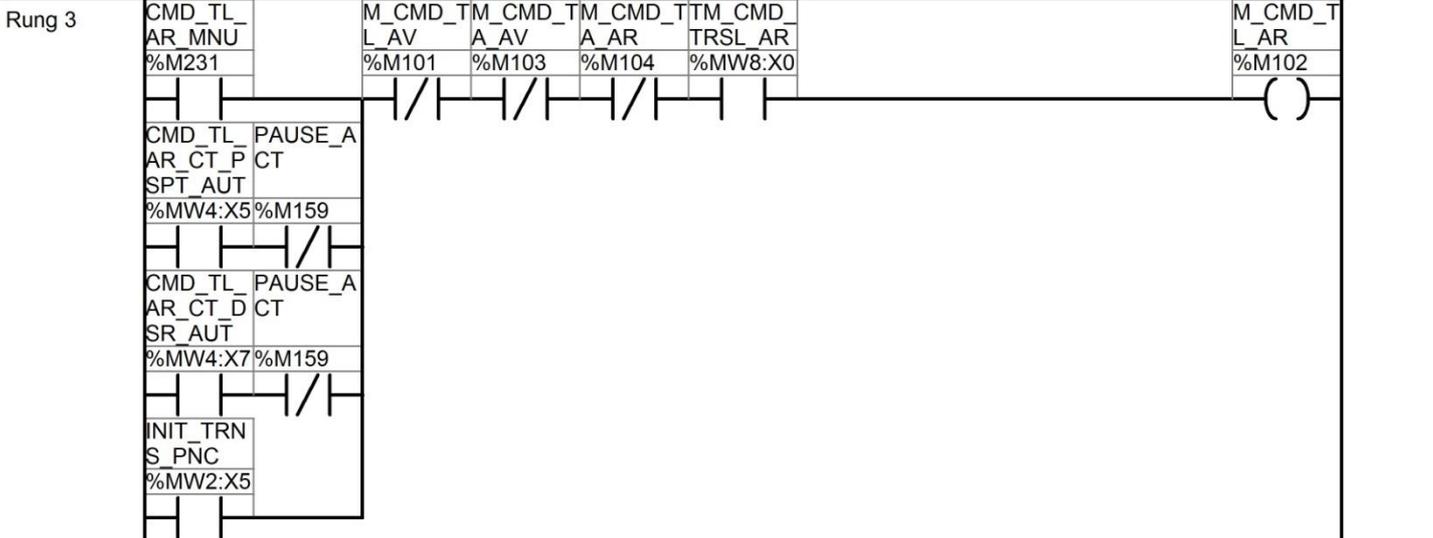


10	LD	COMMANDE DES SORTIES	
-----------	-----------	-----------------------------	--

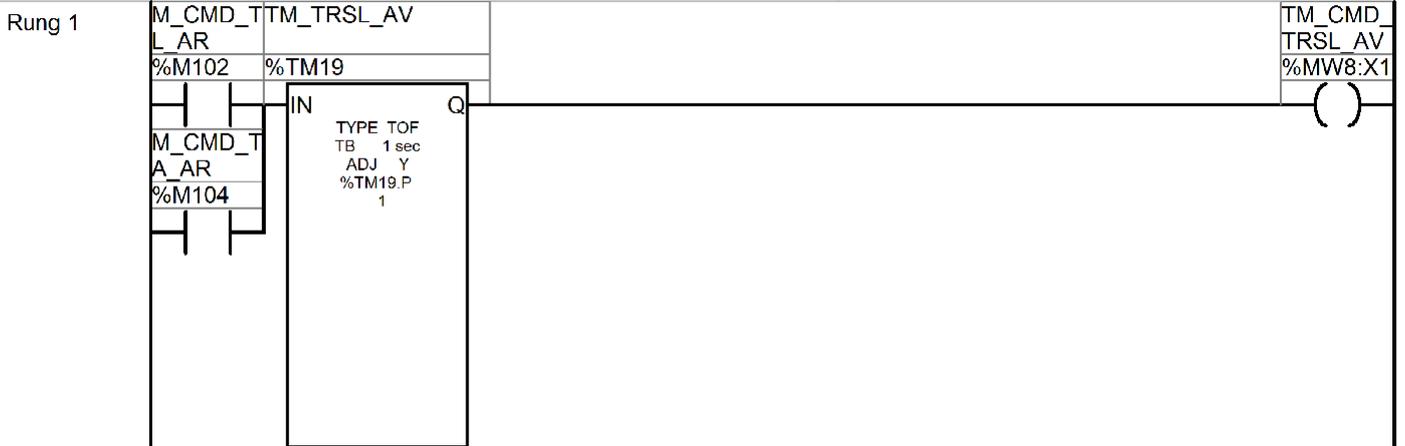
COMMANDE TRANSLATION LENTE AVANT	
---	--

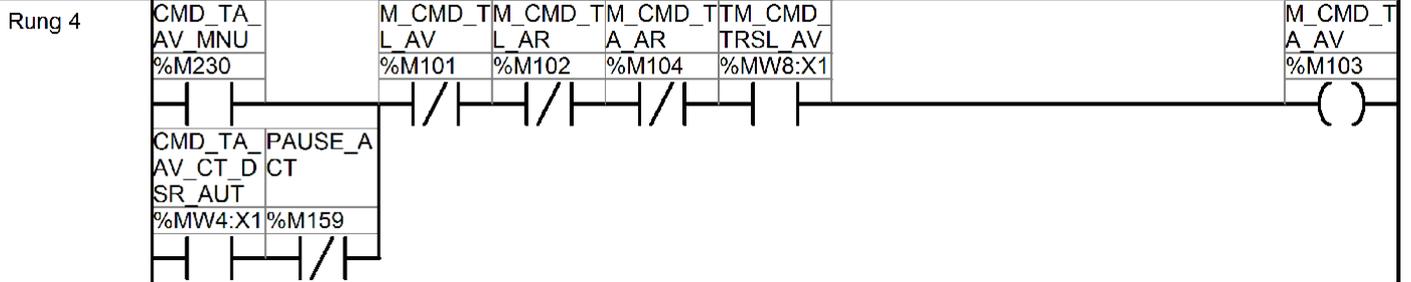
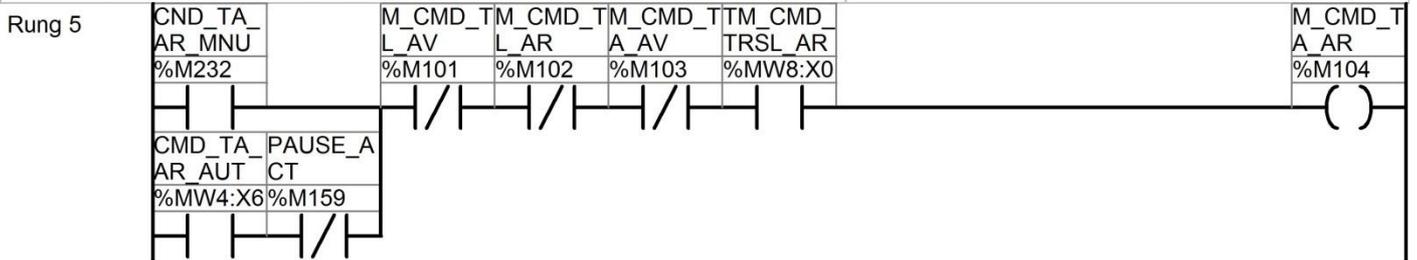
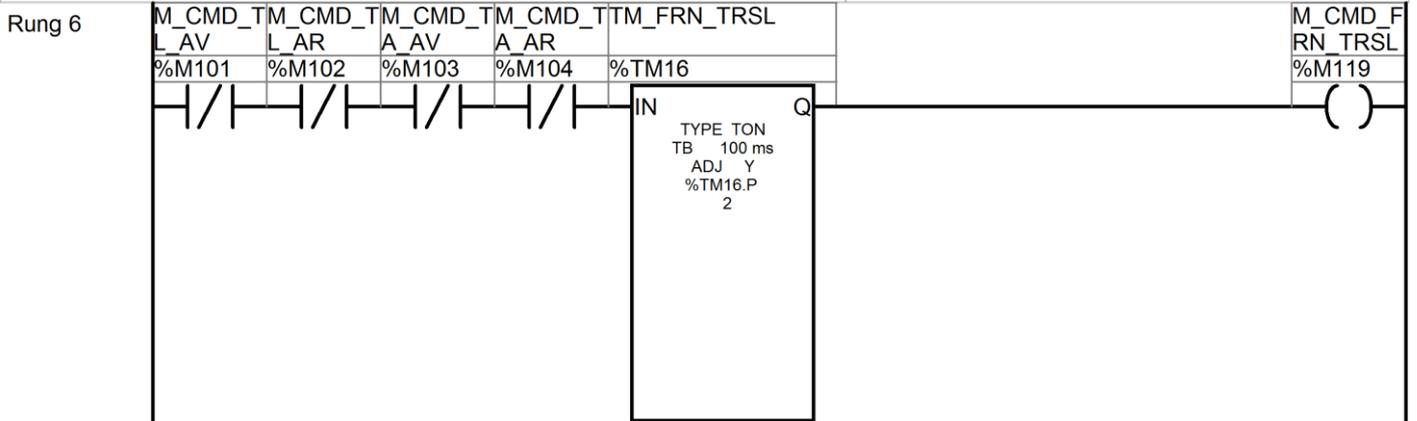
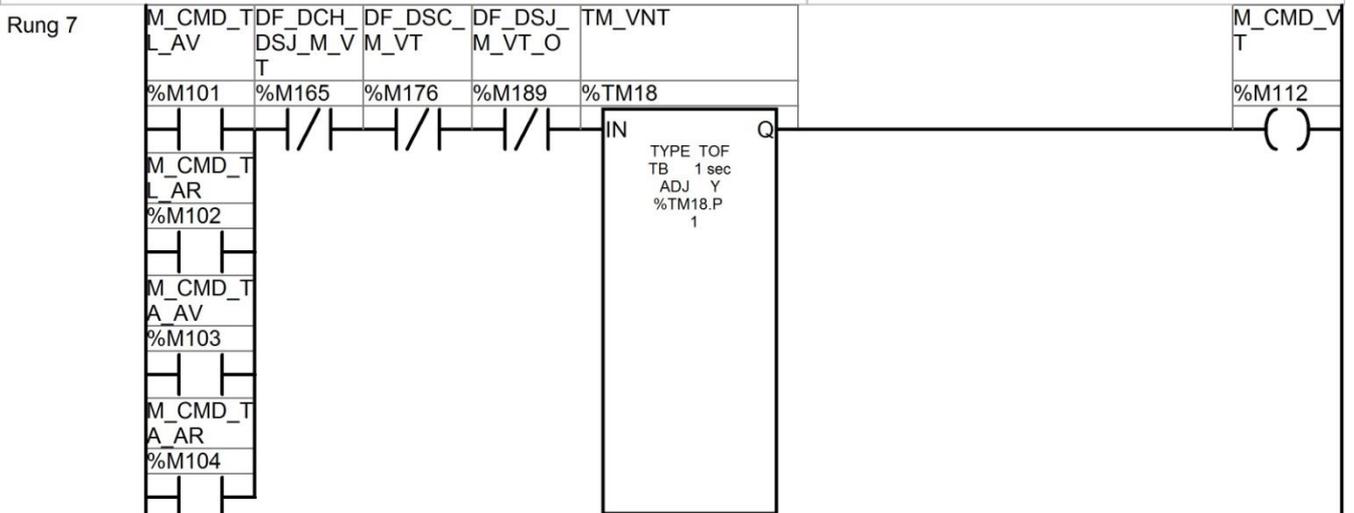


COMMANDE TRANSLATION LENTE ARRIERE	
---	--



TEMPORISATION TRANSLATION AVANT	
--	--

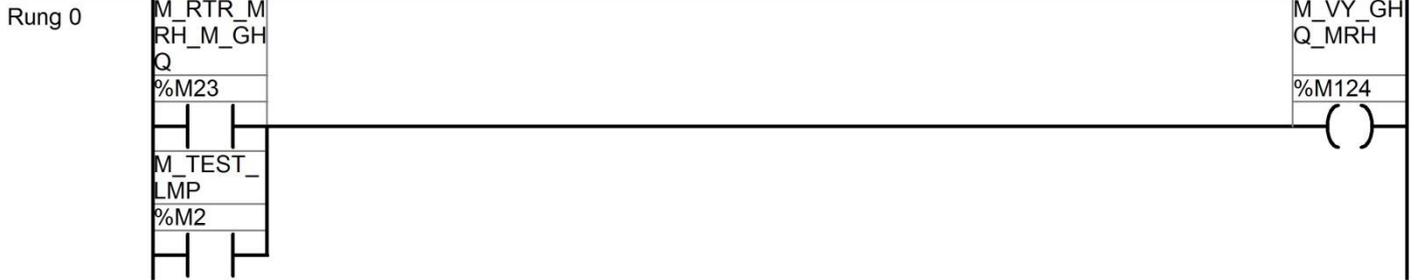


**COMMANDE TRANSLATION ACCELERER AVANT****COMMANDE TRANSLATION ACCELERER ARRIERE****COMMANDE FREINAGE DE TRANSLATION****COMMANDE VENTILATEUR**



11 LD COMMANDE DES VOYENT

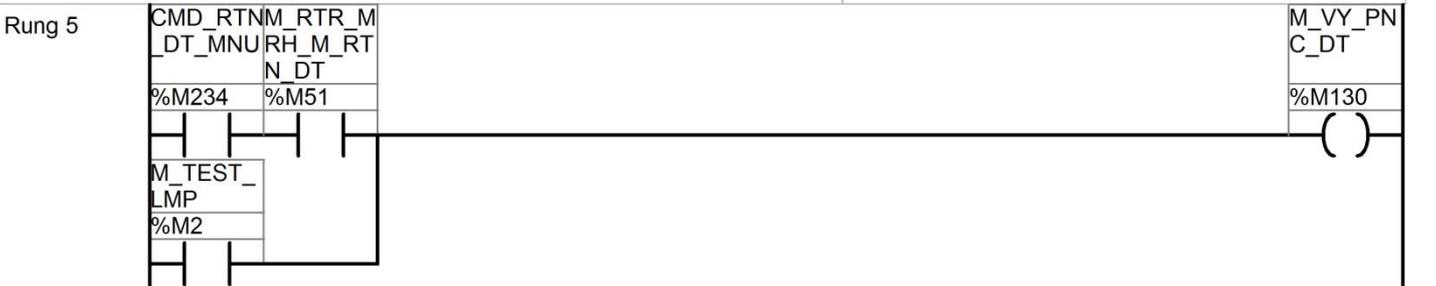
VOYENT GROUPE HYDROLIQUE MARCHE



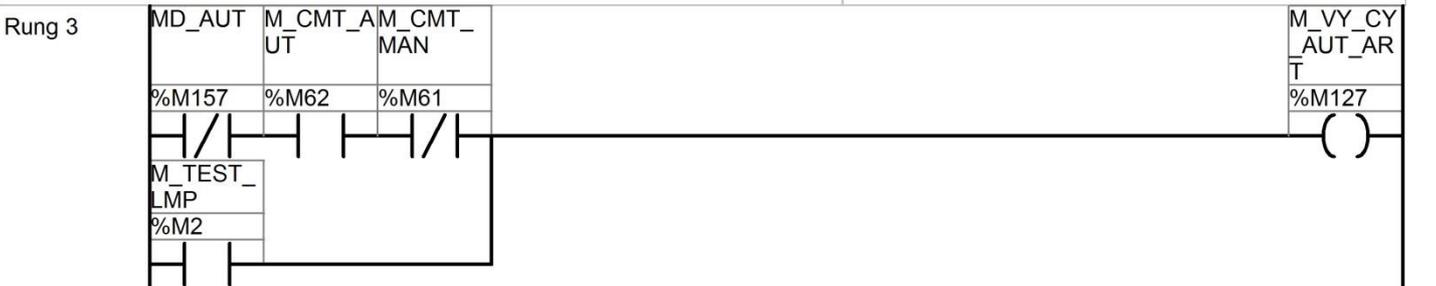
VOYENT GROUPE HYDROLIQUE ARRET



VOYENT ROTATION PINCE VERS LA DROITE



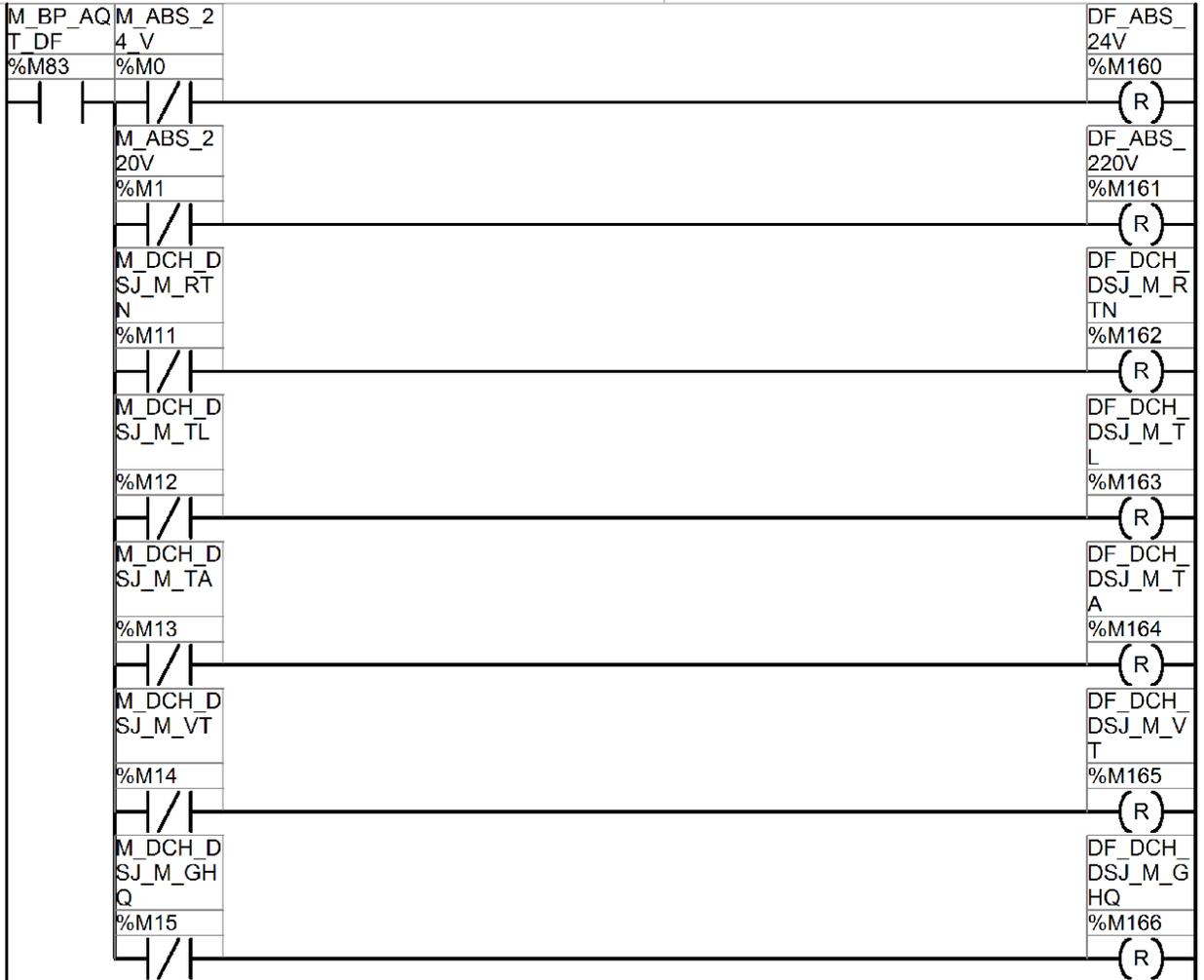
VOYENT ARRET CYCLE AUTOMATIQUE





12 LD ACQUITEMENT DES DEFAUT

Rung 0

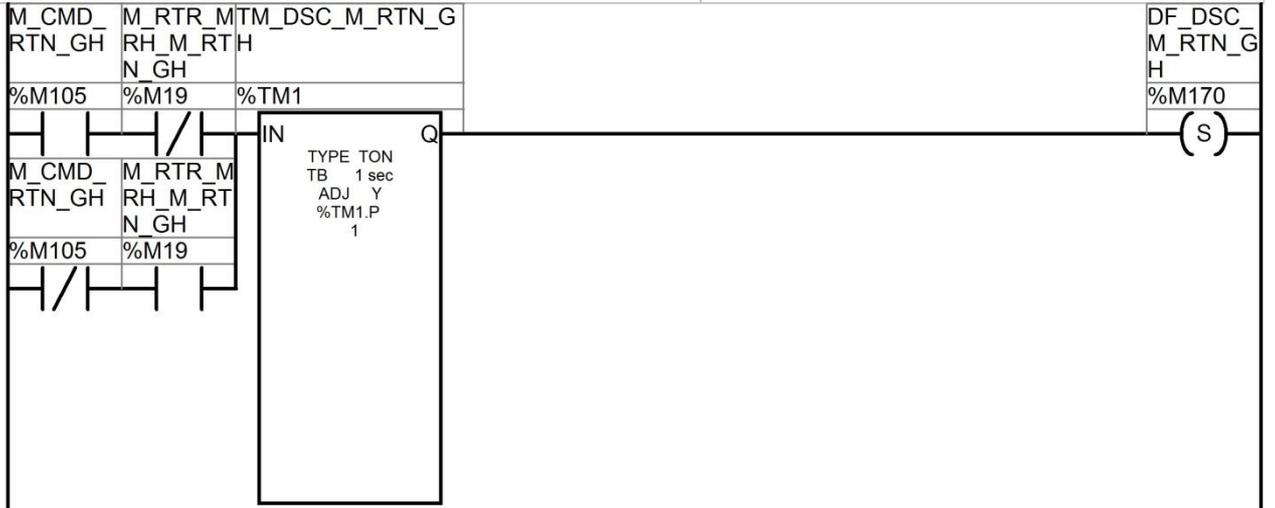


**13 LD TRAITEMENT DES DEFAUTS****DEFAUT DECLANCHEMENT DISJONCTEUR MOTEUR DE VENTILATION TRANSLATION****DEFAUT DECLANCHEMENT DISJONCTEUR MOTEUR DE GROUPE HYDROLIQUE****DEFAUT 24V****DEFAUT DECLANCHEMENT DISJONCTEUR MOTEUR DE DESCENSEUR****DEFAUT DECLANCHEMENT DISJONCTEUR MOTEUR DE TOURNE PLANCHE**



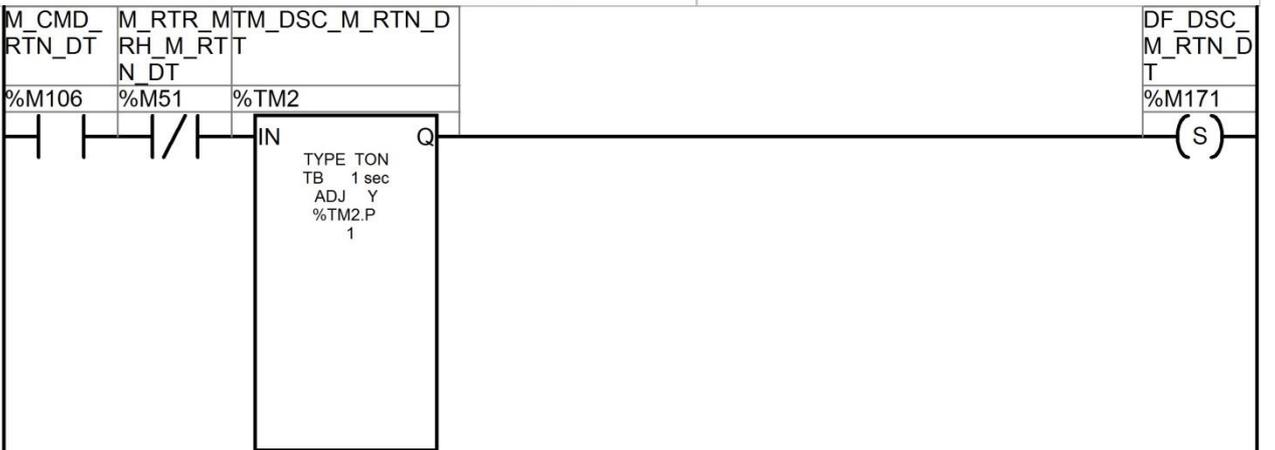
DEFAUT DISCORDANCE MOTEUR ROTATION GAUCHE

Rung 10



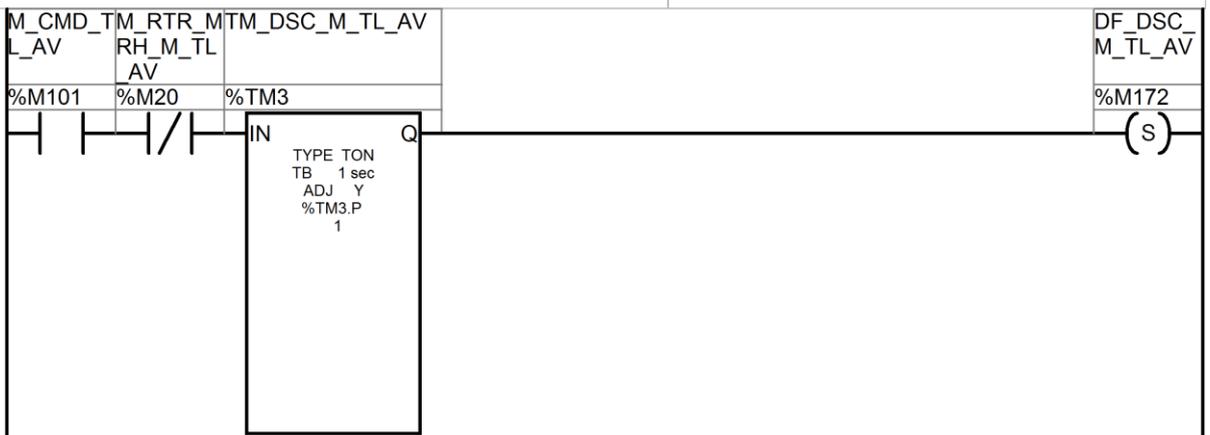
DEFAUT DISCORDANCE MOTEUR ROTATION DROITE

Rung 11



DEFAUT DISCORDANCE MOTEUR TRANSLATION LENTE AVANT

Rung 12

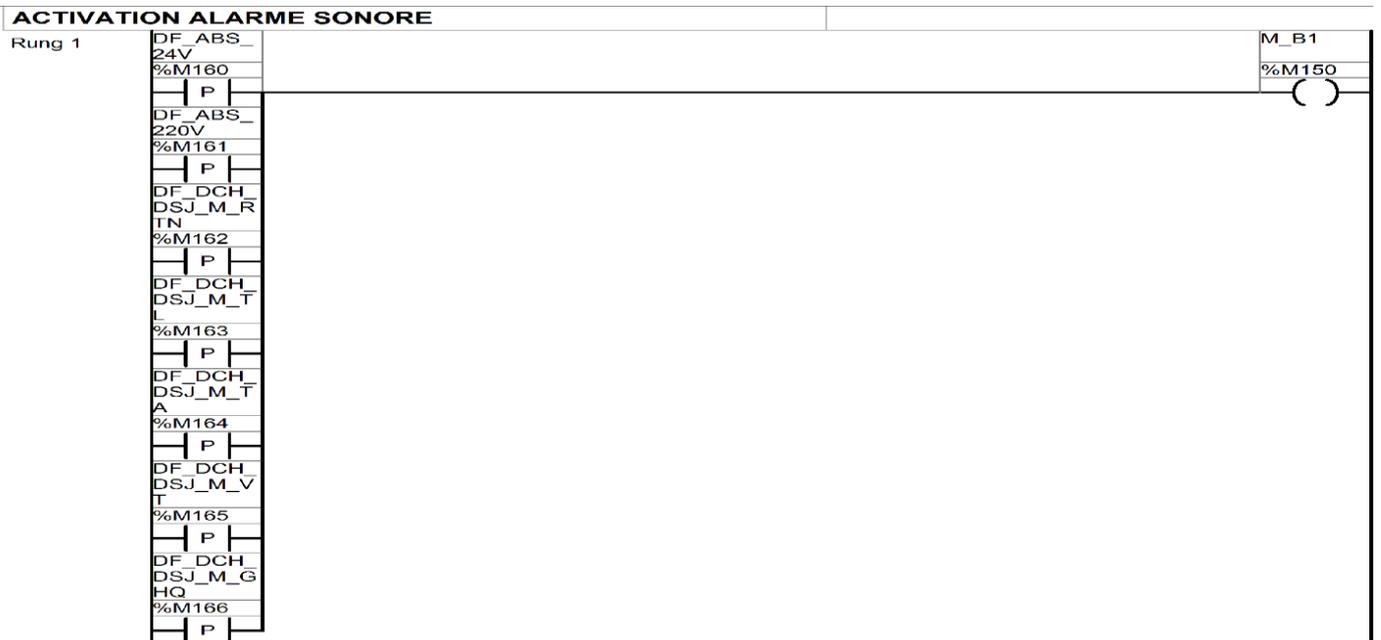




14	LD	MISE EN MEMOIRE DEFAUT	
-----------	-----------	-------------------------------	--



15	LD	CONSEQUENCES DES DEFAUTS	
-----------	-----------	---------------------------------	--





16 LD AFFECTATION DES SORTIES

