

République Algérienne Démocratique et Populaire
Ministère de L'Enseignement Supérieur et de la Recherche Scientifique
Université Mouloud MAMMARI DE Tizi Ouzou



Faculté Génie Electrique et Informatique
Département d'Automatique
Mémoire de Fin d'Etudes
En Vue de l'Obtention Du Diplôme de Master2 Académique
Spécialité : Automatique
Option : Commande des systèmes

Présenté par

DYHIA SOUALAH

SELIA LATEB

Mémoire dirigé par M^r KARA REDOUANE et M^{elle} TEBANI KARIMA

Thème

Commande par Automate Schneider M-340 et IHM d'un système de production en utilisant la technologie RFID

Mémoire soutenu publiquement le 09 Juillet 2015 devant les jurys :

M^r Touate Mouhnd ouachour

MCA, UMMTO, Président

M^r Allad Mourad

MAA, UMMTO, Examineur

M^{elle} Naît abdesslam Aldjia

MAA, UMMTO, Examinatrice

Mr Kara Redouane

MCA, UMMTO, Promoteur

Melle Tebani Karima

Magister, UMMTO, Co-promotrice

Le travail a été réalisé au sein du Centre de Développement des Technologies Avancées

Remerciement

Tout d'abord nous tenons à remercier Dieu le tout puissant pour la volonté et le courage qu'il nous a donné afin de pouvoir finir ce travail.

Nous tenons également, à remercier Monsieur GAHAM responsable du projet et M^{elle} TEBANI pour avoir assuré notre encadrement au sein du centre de développement des technologies avancées (CDTA) Baba Hassen. Nous les remercions pour leur orientation, la confiance et la patience qui ont été d'un apport considérable sans lequel ce travail n'aurait pas abouti.

Nous témoignons nos sincères gratitude à Monsieur KARA notre promoteur de l'université TIZIOUZOU qui a largement participé à notre encadrement et qui nous a donné une aide précieuse, à la fois par ses conseils sur le plan scientifique et sa disponibilité.

Nous tenons à exprimer nos sincères remerciements à nos professeurs de notre département GENIE ELECTRIQUE, qui nous ont aidé et soutenu dans la poursuite de nos études.

Nous remercions vivement Mesdames et Messieurs les membres de jury d'avoir accepté d'évaluer ce travail. Nous remercions nos parents, nos frères et sœurs pour leurs soutient moral, leurs encouragement et leurs patience durant les étapes difficiles de nos études.

Enfin, Ces remerciement ne seraient pas complets sans une pensée sincère à nos amis (es) proches et à tous ceux qui nous ont soutenus de près ou de loin, tout au long de cette année en particulier à notre promotion AUTOMATIQUE 2014/2015.



Dédicaces

Je dédie ce mémoire qui est le fruit de nombreuses années d'études et de travail, tout en exprimant ma profonde gratitude et sympathie à toutes les personnes qui ont participé de près ou de loin pour mener à bien ce projet et plus particulièrement :

A mes parents, à ma mère qui est la lumière de mes yeux qui me comble d'amour. A mon père qui m'a toujours épaulé, soutenu financièrement et moralement et donné son amour et son respect. Je tien aussi a le dédié à mes très chères sœurs FATIHA, MALIKA, SAMIA, OUARDA, LILA, KARIMA qui a été présente avec son soutien durant toute la période du stage et son mari CHABANE que je remercie infiniment pour son aide et son soutien, Je tien à les remercier pour leurs accueil pendant toute cette période. Et mes beaux-frères AMAR, AHMED, KAMAL, LOUNAS, OMAR, je tien a le dédié aussi à mes neveux MEZIANE, AHMED, MAHDI, MOUMOUH, GHILASSE, MASSIA, KLI, YANISSE, YASSINE, NONORE, AGHILESSSE et le petit YANISSE, mes nièces KAHINA, CELIA et SIHAME que je considère comme mes sœurs, mes chères NOUNOU, LILA, NINA, FARIELLE, LILI, ANAISSE et sans oublié la petite adorable RIMA.

Je dédie ce travail aussi à mes frères que je respecte beaucoup AZIZE, CHABANE ainsi KADER et mes belles sœurs ROSA et SADIA.

Sans oublié mon binôme ma sœur ma copine SELIA que j'aime beaucoup, à sa famille en tout particulier a na FARIZA A la mémoire de son père qui se repose en paix, sans oublié mes amis LYNDA, SAMIA, IMANE, SAMIRA, NADIA, FAREDJ, KHALEF, MOUHANDE, YOUNESSE, MOUKRANE, GHILASSE à la promotion automatique 2014/2015.

DYHIA



Dédicace

Je dédie ce modeste travail avant toute personne à l'homme de ma vie, mon exemple éternel, mon soutien moral et source de joie et de bonheur, celui qui s'est toujours sacrifié pour me voir réussir, que dieu te garde dans son vaste paradis, à toi mon *PERE*.

A la lumière de mes jours, la source de mes efforts, la flamme de mon cœur, ma vie et mon bonheur ; *MAMAN* que j'adore.

A mes frères *LAMARA* et *NASSIM* qui m'ont toujours encouragé et ma sœur *OURDIA* qui n'a cessé d'être pour moi un exemple de persévérance, de courage et de générosité

Aussi à mes chers grands parents, tantes et oncles, cousins et cousines.

A celle qui m'a soutenu et encouragé durant ces cinq années et durant ce projet, mon binôme, ma sœur *DYHIA* et a toute sa famille surtout *DA NORDINE* et *NANA FATMA*, sans oublié sa sœur *KARIMA* et son époux *CHABANE*, leurs enfants *YANISSE* et *RIMA*, qui nous ont accueilli vivement pendant notre stage,

Aux personnes qui m'ont accompagné durant mon chemin d'études supérieures, mes aimables amis(es) : *IMANE*, *NAWAL*, *AMINA*, *MANEL*, *DIHIA*, *LYNDA*, *LILIA*, *REZIKA* ainsi *FAREDJ*, *KHALEF*, *MOHAND*, *MOKRANE*, *YOUNESSE*, *ACHOUR*, sans oublié toute la promotion automatique 2014/2015.

SELIA

Résumé

De nos jours l'exploitation de la technologie RFID commence à se généraliser sur des axes d'applications très variés : passeport biométrique, cartes de crédits, cabines de péages, badges sécurisés, systèmes de stockage sécurisés et identification.

Ce travail vise à exploiter cette nouvelle technologie dans les systèmes de production. Aussi, on a introduit un nouvel automate SCHNEIDER M340 avec une plateforme de communication JAVA-OPC (IHM) tout en intégrant le robot GT-6A pour l'évacuation des produits dans la cellule flexible du CDTA.

Mots clés :

RFID, OPC, IHM

Abstract

Nowadays the use of RFID technology is beginning to spread on a wide variety of application areas: biometric passports, credit cards, toll booths, secure badges, secure and stock identification systems. The present work aims to exploit this new technology in a real production systems by introducing a new controller SCHNEIDER M340 with a communication platform JAVA-OPC and integrating the GT-6A robot for removing products in CDTA production cell.

Keywords:

RFID, OPC, IHM

SOMMAIRE

Introduction générale.....	1
CHAPITRE I : Généralités.....	3
I.1 Les systèmes automatisés :	3
I.1.1 Objectifs de l'automatisation des chaines de productions :	3
I.1.2 Structure d'un système automatisé :	3
I.2 Intégration des robots dans les systèmes automatisés :	6
I.2.1 Les trois générations de robots:	6
I.2.2 Utilité des robots:	6
I.3 Systèmes d'identifications automatiques :	6
I.3.1 Code barre :	7
I.3.1.1 Structure d'un code barre :	7
I.3.1.2 Description du code barre 39 (3 of 9)	7
I.3.2 La gravure :	8
I.3.3 Les encres invisibles:	8
I.3.4 Les caméras en logistique :	8
I.3.5 La Technologie RFID :	8
I.4 Conclusion :	8
CHAPITRE II : IDENTIFICATION DES ELEMENTS DE LA CELLULE FLEXIBLE DU CDTA.....	10
II.1 INTRODUCTION :	10
II.2 DESCRIPTION DE LA CELLULE FLEXIBLE (état des lieux) :	10
II.2.1 Présentation de l'automate GE FANUC 90-30 IC 693CPU323 :	11
II.2.2 variateur de vitesse (SSD-582):	11
II.2.3 Présentation du robot GT 6A :	12
II.2.4 Capteurs :	13
II.2.5 Convoyeur :	15
II.3 FONCTIONNEMENT DE LA CELLULE:	15
II.3.1 Le cahier de charge de cette application :	15
II.3.2 Diagramme de fonctionnement du système :	16

II.3.2.1	Interprétation du schéma	16
II.4	Le protocole de communication OPC.....	17
II.4.1	HISTORIQUE DE L'OPC :	17
II.4.2	Définition d'OPC:.....	17
II.4.3	<i>SPECIFICATIONS DE LA NORME OPC</i> :.....	17
II.4.4	Structure de données d'un serveur OPC Data Access :.....	18
Objet OPC Item	18
Objet OPC Branche	18
II.4.5	Côté serveur :	19
II.4.6	Structure de données vues du côté client SERVEURS OPC :.....	19
Objet OPC Server	19
Objet OPC Group	19
II.4.7	OPC Data Access côté client :.....	20
II.4.8	Composant Object COM :	20
II.4.9	Le modèle COM distribué ou DCOM :	20
II.4.10	Avant OPC :	21
II.4.11	Depuis OPC :.....	22
II.5	Introduction au langage Java.....	22
II.5.1	Environnement Java	22
Compilation	23
II.5.2	Programmation orientée-objet :.....	23
1)Classe	:	24
2)Objet	:	24
II.6	Travail demandé.....	24
I.6.1	Identification du câblage de l'ancienne armoire:.....	24
II.7	Conclusion :	25
CHAPITRE III : PRESENTATION DE LA TECHNOLOGIE RFID		26
III.1	Introduction :	26
III.2	Bref historique :	26
III.3	Présentation de la RFID :.....	26
III.4	Composition d'un système RFID	27
III.4.1	Le tag :.....	27

III.4.1.1 Type des tags :	28
II.4.1.2 Types de lecture d'étiquette :	31
II.4.2 le lecteur.....	32
III.5 Principe de fonctionnement :	33
III.6 La communication :	34
III.6.1 La communication par champs électromagnétiques :	34
III.6.2 La communication par ondes radio :	34
III.7 Distance de communication en technologie RFID :	35
III.7.1 Très courte distance :	35
III.7.2 Courte distance :	35
III.7.3 Proximité :	35
III.7.4 Voisinage :	35
III.7.5 Longue distance :	35
III.8 la fréquence :	36
III.9 Application de la technologie RFID :	36
Dans les entreprises :	36
En logistique :	37
III.10 Avantages et inconvénients :	38
Avantages	38
Inconvénients	38
III.11 Boite de raccordement réseaux :	38
III.12 Câblage d'un réseau Modus Ethernet TCP/IP :	39
III.12.1 Schéma du montage :	39
➤Présentation du Modbus :	39
III.13 Configuration du transpondeur RFID :	40
III.13.1 Démarche à suivre :	40
III.13.2 La programmation de lecture /écriture sous Unity Pro :	40
Utiliser des blocs fonctions :	41
III.13.3 La zone mémoire :	42
III.14 Avantages :	45
III.15 Conclusion :	45

IV.1 INTRODUCTION :	47
IV.2 Câblage des éléments :	47
IV.3 Réalisation d'un système de pilotage automatisé :	48
IV.3.1 Description de l'application :	48
IV.3.2 Cahier de charge :	48
IV.3.3 Diagramme de fonctionnement du système :	49
IV.3.4 Câblage de l'armoire électrique :	49
IV.3.5 La configuration matérielle sous UNITY-PRO :	51
IV.3.6 Jeux d'instruction sous unity-pro :	51
Instruction de transfert de données :	51
Instruction de temporisation :	52
Instruction de conversion d'adresse :	52
IV.3.7 Editeur de données :	53
IV.4 Phase d'implémentation :	53
IV.4.1 programme sous unity-pro:	53
IV.4.1.1 Programme automate :	53
a)Langage LADDER :	54
IV.4.2 Logicielle KEPServer :	59
IV.4.3 Programmation des tâches du robot :	62
IV.4.3.1 Les différentes tâches sous MS-DOS :	62
Conclusion générale	66
ANNEXE	67
ANNEXE 1 :	67
Caractéristiques de l'automate GE FANUC séries 90-30 :	67
Constitution de l'API GE FANUC 90-30 (IC693CPU323) :	67
Les Modules d'entrées sorties (TOR).....	68
Les modules d'entrées/ sorties analogiques :	68
Interface de communication	68
ANNEXE 2 :	69
L'automate programmable industriel Schneider M340 :	69
Définition d'un automate programmable industriel :	69
Structure d'un API :	69

Présentation de l'automate Schneider M340 :	70
Caractéristique de l'automate Schneider M340 :	71
Caractéristiques mécanique :	71
Caractéristiques techniques des principaux constituants L'automate :	71
Description de l'automate Modicon M340 :	71
Structure mémoire de l'automate M340 :	73
RAM interne :	73
Carte mémoire :	73
Alimentation :	73
Communication :	73
logiciel UNITY PRO :	73
1) Présentation :	74
BIBLIOGRAPHIE	80
Bibliographie	80

Table des figures

FIGURE 1. 1 LES PARTIES D'UN SYSTEME AUTOMATISE	4
FIGURE 1.2 : STRUCTURE D'UN SAP	5
FIGURE 1.1 : STRUCTURE D'UN CODE A BARRE	7
FIGURE 2.1 : LA CELLULE FLEXIBLE DE PRODUCTION	11
FIGURE 2.2 : VARIATEUR DE VITESSE SSD-582	12
FIGURE2.3 : ROBOT PUMA	12
FIGURE 2.4 : LE ROBOT GT 6A	13
FIGURE 2.5 : SCHEMA FONCTIONNEL D'UN CAPTEUR	13
FIGURE 2.6 : CAPTEUR INFRAROUGE (ACEL, 9405)	14
FIGURE 2.7 : SYSTEME BARRAGE	14
FIGURE 2.8 : CAPTEUR INFRAROUGE	14
FIGURE2.9 : CONVOYEUR	15
FIGURE 2.10 : DIAGRAMME DE FONCTIONNEMENT DU SYSTEME	16
FIGURE 2.11 : STRUCTURE OPC COTE SERVEUR	19
FIGURE 2.12 : STRUCTURE OPC COTE CLIENT	19
FIGURE 2.13 : STRUCTURE D'UN MODELE COM DISTRIBUE.....	21
FIGURE 2.14 : AVANT OPC	21
FIGURE 2.15 : DEPUIS OPC	22
FIGURE2.16 : ENVIRONNEMENT JAVA	23
FIGURE 2.17 : L'ANCIENNE ARMOIRE	25
FIGURE 3.1 : LA COMMUNICATION ENTRE LE LECTEUR ET LE TAG.....	27
FIGURE 3.2 : LA STRUCTURE DU TAG	27
FIGURE 3.3 : ALIMENTATION DU TAG PASSIF	28
FIGURE 3.4 : TAG RFID ACTIF	29
FIGURE 3.5 : TRANSMISSION DES DONNEES VERS LE LECTEUR.....	30
FIGURE 3.6 : L'ETIQUETTE RFID	32
FIGURE 3.7 : LE LECTEUR	32
FIGURE 3.8 : PRINCIPE DE FONCTIONNEMENT DE RFID	33
FIGURE 3.9 : LA GESTION DE LA CHAINE D'APPROVISIONNEMENT	36
FIGURE 3.10 : BOITIER OSITRACK	38
FIGURE 3.11: SCHEMA DE RACCORDEMENT RESEAU	39
FIGURE 3.12: PRESENTATION DU MODBUS	39
FIGURE 3.13: BADGE DE CONFIGURATION DU RFID	40
FIGURE 3.14: CREATION D'UNE SECTION SOUS UNITY -PRO	40
FIGURE 3.15: BLOC FONCTION READ-VAR	41
FIGURE 3.16: BLOC FONCTION WRITE-VAR	42
FIGURE 3.17: ZONE D'ADRESSES DES MOTS	43
FIGURE 3.18: LA ZONE MEMOIRE STATION	44
FIGURE 4.1: LA NOUVELLE CELLULE FLEXIBLE	45
FIGURE 4.2: DIAGRAMME DE FONCTIONNEMENT DU SYSTEME	47
FIGURE 4.3: CABLAGE DE L'AUTOMATE M-340	48
FIGURE 4.4: CABLAGE DU VARIATEUR ET DU TRANSFORMATEUR	48
FIGURE 4.5: LA CONFIGURATION MATERIELLE DE L'AUTOMATE	49
FIGURE 4.6: BLOC POUR VARIATEUR DE VITESSE	50

FIGURE 4.7:BLOC DE TEMPORISATION	50
FIGURE 4.8:BLOC DE CONVERSION D'ADRESSE	51
FIGURE 4.9:EDITEUR DE DONNEES	51
FIGURE 4.10:TABLE D'ANIMATION	52
FIGURE 4.11 :ILLUSTRATION DU GRAFCET	55
FIGURE 4.12: CONFIGURATION DU PORT	58
FIGURE 4.13 : CHOIX DE L' AUTOMATE	60
FIGURE 4.14 : CREATION D'UN GROUP	61
FIGURE 4.15 : CRÉATION D'ITEM	61
FIGURE 4.16: TABLE DES ITEMS	60
FIGURE 4.17: STATION DU ROBOT	61
FIGURE 4.18: INITIALISATION DU ROBOT	61
Figure 4.19: SELECTION DE L'ETAPE APPRENTISSAGE	61
Figure 4.20:CREATION DES TACHES	62
Figure 4.21:EXECUTION D'UNE TACHE	62
FIGURE A.1 : L' AUTOMATE SCHNEIDER M-340	66
FIGURE A.2 : LE SIGNALE TOR ET ANALOGIQUE	67
FIGURE A.3 : CONFIGURATION MONORACK	68
FIGURE A.4 : CONFIGURATION MULTIRACK	68
FIGURE A.5 : LA CPU M-340	68
FIGURE A.6 : CPU M-340 AVEC UNE CARTE MEMOIRE	70
FIGURE A.7 : CREATION D'UN NOUVEAU PROJET	71
FIGURE A.8 : CONFIGURATION TYPE DU BAC	72
FIGURE A.9 : CHOIX DES MODULES D'ENTREES /SORTIES.....	72
FIGURE A.10 : CONFIGURATION DES CARTES D'ENTREES /SORTIES	73
FIGURE A.11 : CONFIGURATION DU PROCESSEUR	73
FIGURE A.12 : CONFIGURATION DES MODULES	74
FIGURE A.13 : EDITION DES ADRESSES SYMBOLIQUES DES ENTREES/SORTIES	74
FIGURE A.14 : EDITION DES ADRESSES SYMBOLIQUES DES MOTS INTERNES	75
FIGURE A.15 : EDITION DES ADRESSES SYMBOLIQUES D'UN BIT INTERNE	75
FIGURE A.16 : METHODOLOGIE DE SAISIE DU LADDER	76
FIGURE A.17 : METHODOLOGIE DE SAISIE DU GRAFCET	76

Liste des tableaux

Tableaux 1 : Comparaison entre les différents systèmes des étiquettes30

Tableau 2 : Les plages mémoire des étiquettes.....44

Acronymes et abréviations

AC : Alternatif Current

ADSL : Asymmetric Digital Subscriber Line

API : Automate Programmable Industriel

CAN : Convertisseur Analogique/ Numérique

CDTA : Centre de Développement des Technologies Avancées

CNA : Convertisseur Numérique/ Analogique

COM : Component Object Model

CPU: Central Processing Unit

DA: Data Access

DC: Direct Current

DCOM: Distributed Component Object Model

DDE : Dynamic Data Exchange

DDR :Dispositif Differentiel Résiduel

DEL: Diode Electro-lumineuse

DFB: Diagram Fonction Bloc

EAN: European Article Number

EPC: Electronic Product Code

EPROM: Erasable Programmable Read Only Memory

FTDI : Future Technology Devices International

FTP : File Transfer Protocol

GRAFCET : GRAPhe Fonctionnel de Commande Etape-Transition

HTTP: Hypertext Transfer Protocol

IEC: Israel Electric Corporation

IEEE: International Equipement Electric Electronic

IFF: Identifie Friendly Foe

IP: Internetwork Protocol

ISO: International Organisation for Standardization

JDK: Java Developpement Kit

LD: Ladder Diagram

MSDOS: Microsoft Disk Operating System

NTIC :Nouvelles Technologies de l'Information et la Communication

OLE: Object Linking Embedding

OPC: Ole for Process Control

PC : Personnel Computer

PC : Partie de Commande

PLC : Programmable Logic Controller

PO : Partie Opérative

RAM: Random Access Memory

ROM: Read Only Memory

RFID: Radio Frequency Identification

RS232: Recommanded Standard 232

SAP : Système Automatisé de Production

SD card : Secure Digital card

TCP : Transmission Control Protocol

TOR : Tout ou Rien

UPC: Universal Product Code

USB: Universal Serial Bus

Centre de Développement des Technologies Avancées (CDTA) :

Le Centre de Développement des Technologies Avancées (CDTA) est un établissement public à caractère scientifique et technologique (EPST), il a été créé au sein du Commissariat aux Energies Nouvelles en 1982 en tant que Centre de Développement des Techniques Avancées. . Il a pour mission de mener des actions de recherche scientifique, d'innovation technologique, de valorisation et de formation dans les domaines des sciences et des technologies de l'information, des technologies industrielles et de la robotique, des dépôts et des traitements des matériaux, des applications et des technologies des lasers. A travers ses missions, le CDTA contribue activement au développement du savoir, à sa transformation en savoir-faire et en produits nécessaires au développement économique et sociétal. Ainsi, le CDTA constitue un véritable réceptacle en matière d'appropriation et de diffusion des technologies avancées.

Dans le cadre de l'accomplissement de ces missions, les actions menées par le CDTA s'articulent autour des axes suivants :

- **La microélectronique et la nanotechnologie.**
- **L'architecture des systèmes et le multimédia.**
- **La productique et la robotique.**
- **Les lasers.**
- **Les milieux ionisés.**

Les projets de recherche émanant des axes sus-cités sont exécutés par vingt (20) équipes de recherche organisées au sein de (04) divisions de recherche et soutenues par trois (03) départements de soutien technique et administratif.

INTRODUCTION GENERALE :

De jours en jours la technologie évolue progressivement en apportant des avantages ainsi que des inconvénients pour l'Homme. Cette évolution impact l'homme sur le plan économique et/ou social. Parmi les avantages de l'évolution technologique on cite la réduction des risques de travail dans l'industrie ainsi que le remplacement de l'Homme dans des tâches rudes par des automates. Parmi les inconvénients, il y a le chômage dû à la substitution de l'Homme par des robots dans l'industrie.

L'utilisation croissante de l'automatisation a influencé en profondeur la vie quotidienne et l'évolution générale de la société. Tout au long de l'histoire industrielle, cette automatisation, ainsi que les systèmes robotisés ont permis une augmentation dans la quantité des volumes ainsi qu'une meilleure qualité des produits.

Auparavant ils utilisent différentes technologies d'identification citons à titre d'exemple les encres invisibles, les caméras en logistique.....etc. mais dans le temps le système le plus adopté a été le code à barre ; Aujourd'hui, après des années de recherche, de miniaturisation et d'efforts de normalisation, la technologie RFID vit une étape majeure de son développement.

La technologie RFID (Radio Frequency Identification) — ou identification par fréquence radio — fait partie des technologies d'identification automatique, au même titre que la reconnaissance optique de caractères ou de codes barre. Le but de ces technologies est de permettre l'identification d'objets ou d'individus par des machines.

La technologie RFID a la particularité de fonctionner à distance, sur le principe suivant : un lecteur émet un signal radio et reçoit en retour les réponses des étiquettes— ou tags — qui se trouvent dans son champ d'action. Il existe une variété presque infinie de systèmes RFID ; différents types de mémoire, différentes fréquences, différents types d'alimentation la technologie RFID permet la lecture des identifiants même sans ligne de vue directe et peut traverser de fines couches de matériaux (peinture, neige, boîtes etc).

L'introduction des robots dans les cellules de production a permis un énorme progrès pour les systèmes automatisés et afin de réaliser une communication entre les automates et les robots, on a opté pour l'utilisation des Interfaces Homme Machine (IHM).

La problématique de ce mémoire consiste à introduire la technologie RFID dans un système de production réel qui est la cellule flexible de production du CDTA. Nous avons aussi pour objectif d'introduire un automate SCHNEIDER de la gamme MODICON M340 et d'inclure un robot dans le but d'avoir un système autonome.

Le travail de recherche réalisé durant ces trois mois est mené au sein du centre du développement des technologies avancées a eu pour but d'améliorer un système de production tout en remplaçant un automate par une nouvelle gamme de SCHNEIDER, ainsi le code à barre par technologie RFID qui a été établie pour la première fois en Algérie, pour le stockage du produit on a introduit le robot GT6A avec une interface de communication JAVA-OPC.

Ce mémoire décrit l'ensemble de nos travaux. Il est constitué de la présente introduction, de quatre chapitres et d'une conclusion.

Dans le premier chapitre, on a donné un aspect général sur les systèmes automatisés, ainsi l'intégration des robots dans ces derniers tout en présentant les différents systèmes d'identification.

Le second chapitre, concerne l'identification des éléments de la cellule qui est constitué d'un automate GE FANUC, un robot à six axes, un variateur de vitesse SSD 582, un convoyeur avec deux capteurs.

Dans le troisième chapitre nous présentons la technologie RFID et ces différents domaines d'application.

Dans le quatrième chapitre nous avons développé notre application qui concerne la mise en place de tous les éléments avec leurs câblages puis la présentation du langage LADDER sous UNITY-PRO avec le programme de ce dernier ainsi que la modélisation de notre système avec l'outil GRAFCET et pour introduire le robot dans notre chaîne de production on a utilisé une plateforme de communication qu'on appelle JAVA-OPC tout en présentant le serveur KEPWAR

CHAPITRE I : Généralités

I.1 Les systèmes automatisés :

"Depuis toujours l'homme est en quête de bien être". Cette réflexion (qui rejoint la notion de **besoin**) peut paraître bien éloignée d'un cours de Sciences Industrielles, pourtant c'est la base de l'évolution des sciences en général, et de l'automatisation en particulier. L'homme a commencé par penser, concevoir et réaliser. Lorsqu'il a fallu multiplier le nombre d'objets fabriqués, produire en plus grand nombre, l'automatisation des tâches est alors apparue : remplacer l'homme dans des actions pénibles, délicates ou répétitives.

Le développement des connaissances, et des outils mathématiques, ont conduit à un formidable essor des **systèmes automatisés**, et des **systèmes asservis**, dans la deuxième moitié du 20ème siècle. Certains se hasardent à rapprocher l'**Automatique** et la **philosophie**, observant d'étranges similitudes entre les processus propres à l'homme et l'approche technologique. Alors un système automatisé réalise des opérations pour lesquels l'homme n'intervient que dans la programmation et dans le réglage et aussi il exécute le même travail après avoir reçu les consignes d'un opérateur.

I.1.1 Objectifs de l'automatisation dans les chaîne de production productions :

Les objectifs d'une automatisation peuvent être assez variés. On peut retenir quelques-uns :

- La recherche de coûts plus bas, par réduction des frais de main-d'œuvre, d'économie de matière, d'économie d'énergie,...
- La suppression des travaux dangereux ou pénibles pour l'homme et l'amélioration des conditions de travail.
- La réalisation d'opérations impossibles à contrôler manuellement.

La compétitivité (concurrence) d'un produit final peut être définie comme sa capacité à être bien vendu. La compétitivité résulte essentiellement des résultats obtenus sur les facteurs suivants : coût, qualité, innovation, disponibilité. L'automatisation des équipements de production peut améliorer les coûts, la qualité et même la disponibilité des produits. Il est cependant important de vérifier que le produit sur lequel s'applique cette automatisation soit optimisé et répond aux besoins du marché.

L'expérience montre qu'une automatisation conduit souvent à remettre en cause le processus de fabrication et donc le produit. [1]

I.1.2 Structure d'un système automatisé :

On peut distinguer trois parties dans un système automatisé :

- Partie commande (PC)
- Partie opérative (PO)
- Partie dialogue (PD)

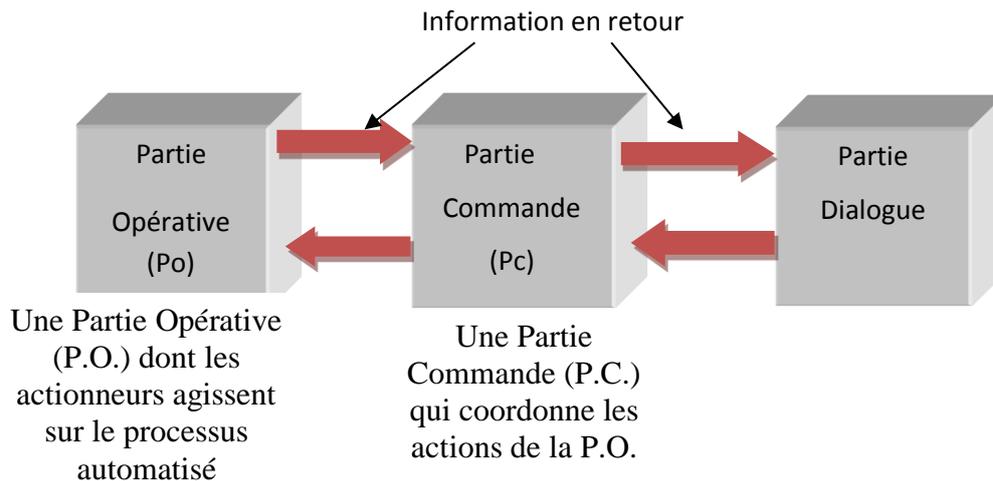


Figure 1. 1 les parties d'un système automatisé

a- La partie commande PC :

Elle permet à l'opérateur d'établir le programme qui commande l'automate. La partie commande reçoit les consignes d'un opérateur. Elle adresse des ordres à la partie opérative. Inversement le PC reçoit des comptes rendus de la partie opérative et envoie des signaux à l'opérateur. [1]

b- La partie dialogue PD :

Elle permet la communication entre l'homme et la machine.

c- La partie opérative PO :

La **partie opérative** d'un automatisme est le sous-ensemble qui effectue les actions physiques (déplacement, émission de lumière...), mesure des grandeurs physiques (température, humidité, luminosité...) et rend compte à la partie commande. Elle est généralement composée d'actionneurs (vérins et moteur), de pré actionneurs (distributeurs et contacteurs) et des éléments de détection (capteurs, détecteurs). Plus simplement, la partie opérative reçoit les ordres de la partie commande et les exécute. [1]

Structure d'un Système automatisé de production :

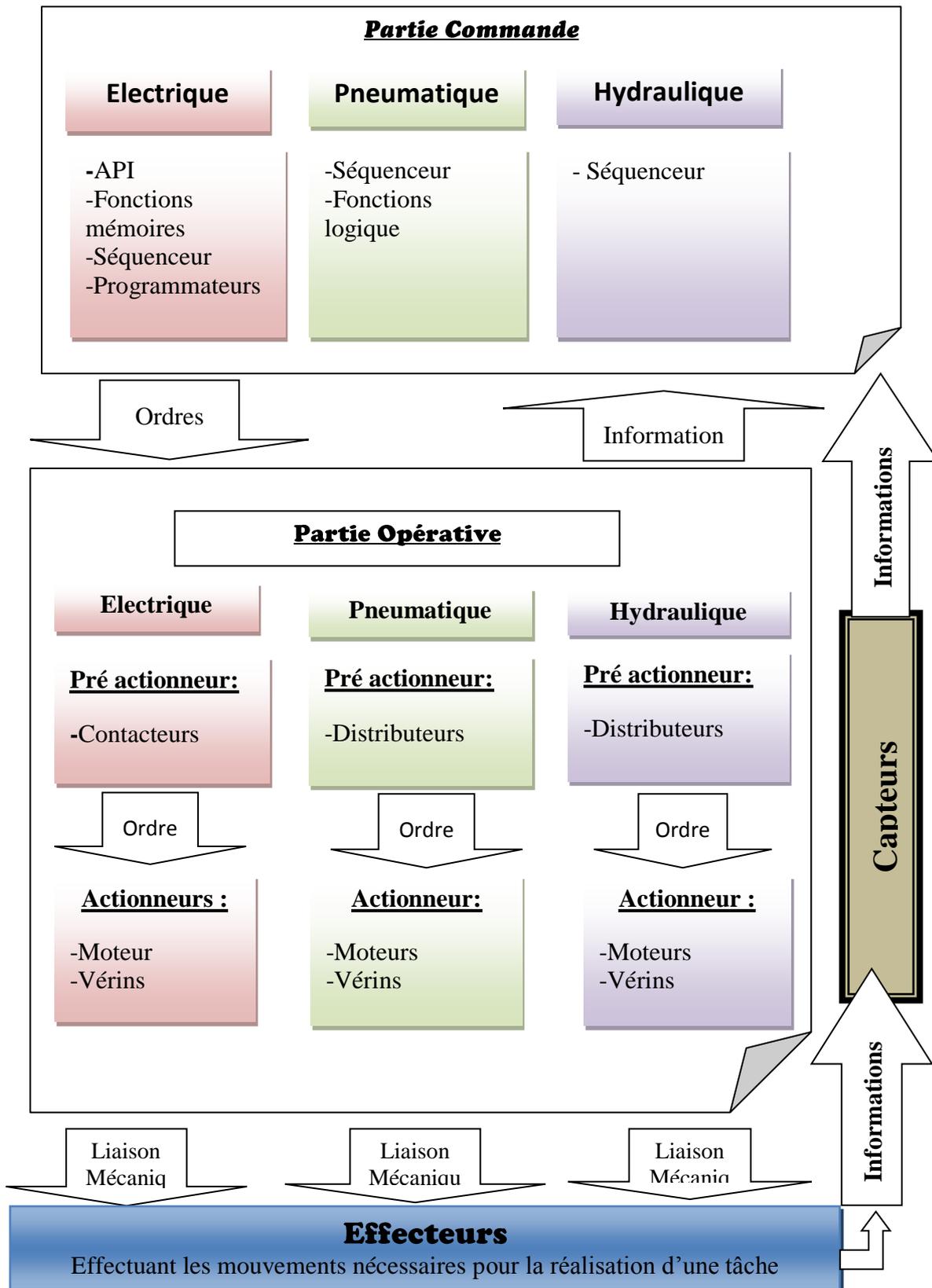


Figure 1. 2 : structure d'un SAP

I.2 Intégration des robots dans les systèmes automatisés :

Les robots industriels sont apparus aux Etats-Unis vers 1960. De nos jours, ils sont très utilisés dans les usines. Les robots peuvent être programmés de différentes façons. Ils peuvent reproduire les opérations autrefois effectuées par l'homme.

Un robot peut être une machine qui doit être capable d'accéder par ses sens à des informations autour d'elle et effectuer une action physique (ex: bouger, manipuler des objets ...etc).

En l'an 2000 ,90% des robots étaient utilisés dans les industries. Les robots peuvent travailler dans les usines, les laboratoires et les hôpitaux.

Plus de 50% des robots industriels travaillent dans les usines de fabrications de voitures. Il y a tellement de robots dans les industries de voitures que le travail des humains est surtout de s'occuper du maintien et du fonctionnement de ces robots. Les robots peuvent être utilisés pour assembler des pièces, pour analyser et pour vérifier certains produits, pour manipuler et utiliser des matériaux dangereux. [2]

I.2.1 Les trois générations de robots:

Les premiers robots sont appelés des automates, ils réalisent des actions répétitives sur des chaînes de montages.

Les robots de la deuxième génération sont dotés de capteurs, ils peuvent réagir à l'environnement qui les entoure.

Les robots de la troisième génération sont équipés d'intelligence artificielle et peuvent alors prendre des décisions. [2]

I.2.2 Utilité des robots:

Les robots sont idéals pour faire des tâches répétitives ou des tâches précises. Ils remplacent les hommes quand les conditions de travail sont trop dangereuses, ils ont des qualités importantes par rapport aux humains

I.3 Systèmes d'identifications automatiques :

Les nouvelles technologies de l'information et de la communication (les NTIC) ont un rôle fondamental dans notre société moderne. Elles participent à sa transformation par différents effets sur les plans économiques et sociaux. Le développement de ces technologies est initié par des découvertes scientifiques, lesquelles permettant de nouvelles applications technologiques, elles-mêmes participant au partage de la connaissance. L'introduction de ces technologies de l'information participe ainsi à la croissance économique en facilitant l'échange et la diffusion des informations. Les technologies d'identification font partie de ces technologies de l'information. Elles trouvent leurs applications dans des domaines très divers tels que la distribution, la logistique, la traçabilité, la sécurité ou les loisirs. Jusqu'alors les technologies d'identifications étaient soit passives : marquage, code-barres... soit nécessitaient un contact : carte bancaire, carte d'appels téléphoniques... Grâce au développement récent des systèmes sans fils et de la micro-électronique, de nouvelles technologies d'identifications sans contacts ont vu le jour : les technologies de radio-identification (ou RFID pour Radio-Frequency IDentification). Ces nouvelles technologies,

par leur plus grande souplesse, rendent l'échange d'information nettement plus rapide et efficace. Elle devrait remettre en question les structures de télécommunications existantes en ouvrant de nouvelles perspectives à des modèles de gestion inédits. [3]

I.3.1 Code barre :

Le code barre est une technologie plus utilisée à l'heure actuelle et il n'est en fait qu'une forme de cryptage (action de coder un message) et décryptage automatique d'une quantité plus au moins grande d'information. Leur principale apport est de réduire la surface de stockage ce qui permet d'envisager que le code à barre puisse rester sur la pièce tout au long du processus de fabrication.



Figure 1.3 : structure d'un code à barre

I.3.1.1 Structure d'un code barre :

Le code barre est un code binaire représenté par une séquence de barres vides et de barres pleines, larges ou étroites, disposées parallèlement. La séquence peut être interprétée numériquement ou alpha numériquement. Elle est lue par balayage optique au laser, c'est-à-dire d'après la différence de réflexion du rayon laser par les barres noires et les espaces blancs. On utilise actuellement une dizaine de types de codes barre différents, sans compter les codes à barres à deux dimensions mais s'agit-il encore de codes barre ? Le code barre le plus courant est le code EAN (European Article Number) donné en Figure 1.3, créé pour répondre aux besoins de l'industrie alimentaire en 1976. Le code EAN est une évolution de l'UPC (Universal Product Code) américain, introduit aux Etats-Unis dès 1973 ; UPC et EAN sont compatibles entre eux. [4]

I.3.1.2 Description du code barre 39 (3 of 9)

Le code 39 qui représente le premier code à barre alphanumérique créé, et le plus communément utilisé. Aussi connu sous l'appellation code USD-3 ou code 3/9, le code 39 est très utilisé dans de nombreuses industries, et il représente la norme pour de nombreuses spécifications de codes à barres gouvernementales, y compris le ministère américain de la défense.

Le code barre le plus courant est le code 39 (appelé aussi code 3 parmi 9). Il est composé de 9 barres et espaces ; trois sont larges et six sont étroites. Dans un code 39, 3 éléments parmi les 9 barres et espaces sont larges, d'où le nom de code 3 parmi 9.

Le code 39 sert souvent à l'identification, à l'inventaire et au suivi des produits en cours, parce que le jeu de caractères est alphanumérique, la longueur de la chaîne d'entrée est

variable (bien qu'il soit difficile de concilier plus de 25 caractères), et le symbole peut varier en hauteur et en largeur. Il est bidirectionnel, comprend une option de contrôle, et il est discret. [5]

I.3.2 La gravure :

Le principe est de réaliser une gravure unique par micro percussion (collision) ou par laser sur chaque pièce. D'un point de vue fonctionnel, la gravure offre les mêmes avantages et inconvénients que le code à barres. A partir d'une gravure, de la même manière qu'en scannant un code à barres, on peut retrouver soit à partir d'une base de données, soit en lisant directement l'information contenue dans le symbole, toutes les informations relatives à la pièce. Le gros avantage de cette technique est l'absence de consommable. En effet, la marque d'identification est directement usinée dans la matière et ne nécessite pas l'ajout d'un autre composant donc Le seul investissement réside dans l'achat de la machine de gravure et des dispositifs de lecture optique. [4]

I.3.3 Les encres invisibles:

Cette technologie offre les mêmes possibilités tout en éliminant la contrainte de visibilité. En effet comme ces encres ne sont pas visibles dans la plage de longueur d'onde que l'œil humain peut voir, le marquage pourra donc être de taille et de forme quelconque. Cette technologie est principalement utilisée par l'intermédiaire d'un opérateur humain pour la lutte contre les contrefaçons. Actuellement aucun dispositif de lecture n'existe sur le marché pour une utilisation automatique en production. Mais cette technologie possède encore des inconvénients comme le risque d'effacement de l'encre ou encore le marquage peut être recouvert d'une couche de finition ou par un emballage. [4]

I.3.4 Les caméras en logistique :

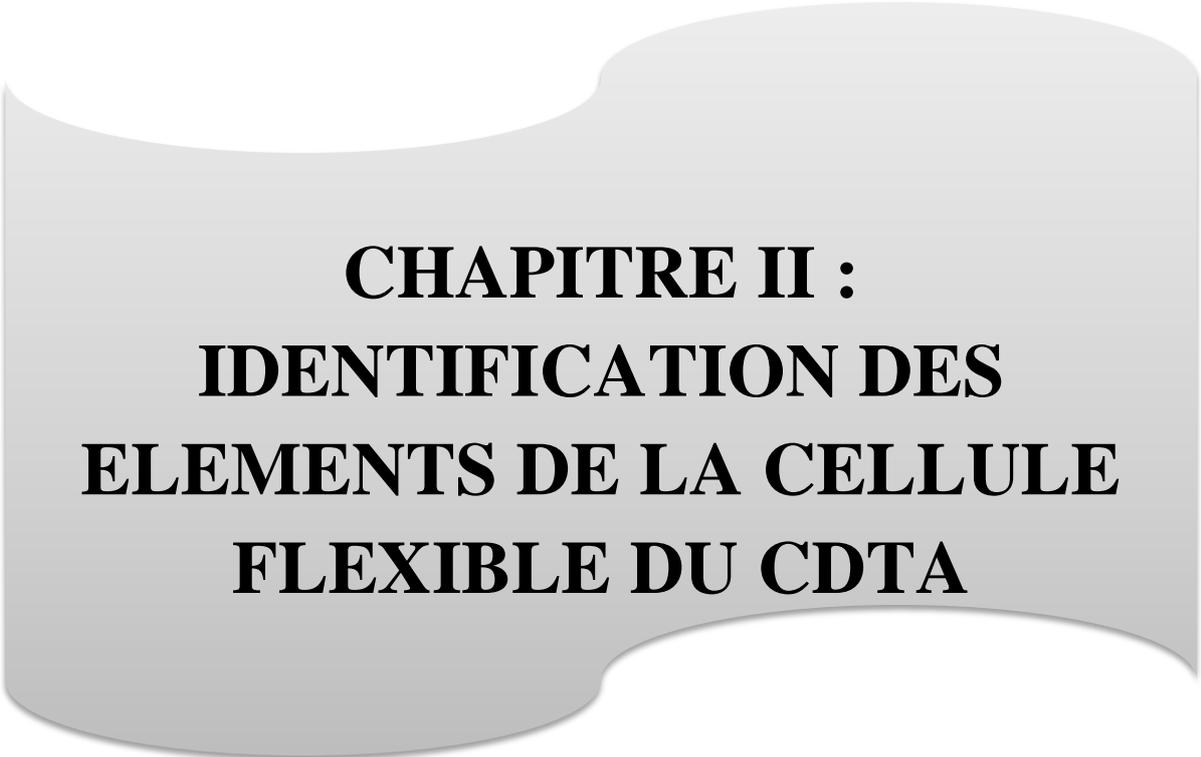
Cette technologie permet de connaître certaines caractéristiques qui sont propres à une pièce ou à un lot de pièces permettant alors de les identifier. Dans ce cas la pièce ne porte plus à proprement parler d'informations sur elle.

I.3.5 La Technologie RFID :

La RFID (Radio Frequency IDentification) est une nouvelle technologie d'identification qui se répand de plus en plus dans le monde industriel et permet entre autre la traçabilité des produits. Comme son nom l'indique, l'identification se fait par radio fréquence. C'est-à-dire que chaque produit sera équipé d'un tag, une puce électronique sans alimentation électrique. Un transpondeur permet de détecter, d'écrire et de lire dans la puce [6].

I.4 Conclusion :

Dans ce chapitre nous avons donné un aspect général sur les systèmes automatisés et l'intégration des robots dans ces derniers et ainsi que l'utilisation des différents systèmes d'identification dans l'industrie. Dans le chapitre qui suit on donnera un exemple plus détaillé sur ces systèmes dans le fait que ce dernier est compatible avec le système d'allocation dynamique d'adresse IP. Ceci nous permet de brancher directement le lecteur RFID sur un PC pour que son adresse soit reconnue et que l'on puisse communiquer avec lui.



**CHAPITRE II :
IDENTIFICATION DES
ELEMENTS DE LA CELLULE
FLEXIBLE DU CDTA**

II.1 INTRODUCTION :

De plus en plus les systèmes industriels deviennent compliqués avec toujours des fonctions de plus en plus complexes. Il est évident qu'un système construit autour d'une logique câblée est aujourd'hui trop limité, car difficile à créer et à maintenir.

L'utilisation d'un automate dans un système permet de simplifier son câblage électrique et de plus son fonctionnement est assuré non pas par le câblage mais le programme contenu dans l'automate. Ainsi, modifier ou améliorer le fonctionnement d'un système est obtenu uniquement en intervenant sur le programme.

Enfin la majorité des automates sont communicants, ainsi toutes interventions sur son programme ou pour commander le système peuvent se faire à distance.

Dans ce qui en suit, on présentera l'ancienne cellule du CDTA en précisant son fonctionnement.

II.2 DESCRIPTION DE LA CELLULE FLEXIBLE (état des lieux) :

La cellule flexible de production est un exemple typique d'une cellule industrielle destinée à des applications d'assemblage et de manutention. Actuellement l'application représente le tri de pièces mécanique basé sur un contrôle de qualité via une vision industrielle. Toute pièce non-conforme est rejetée par un convoyeur, par contre les pièces de bonne qualité subissent d'autres transformations via un robot.

La cellule est constituée de :

- ⌘ Un automate GE FANUC 90-30 IC 693CPU323.
- ⌘ Un robot 6 Axes GT6A.
- ⌘ Un convoyeur.
- ⌘ Deux capteurs.
- ⌘ Interface de communication (PC)
- ⌘ Un variateur de vitesse SSD-582.



Figure 2.1 : la cellule flexible de production

II.2.1 Présentation de l'automate GE FANUC 90-30 IC 693CPU323 :

La cellule flexible du CDTA été doté d'un automate GE FANUC 90-30 (VERSA-PRO est le logiciel de programmation de cet automate). L'automate GE FANUC série 90-30 IC693CPU323 est destiné à la commande et à la supervision en temps réel de processus industriels. L'IC693CPU323 est un automate modulaire fabriqué par la famille GE FANUC Automation. Ces modules sont assemblés sur un profilé support et vissés pour former un ensemble robuste. [5]

II.2.2 variateur de vitesse (SSD-582):

Un **variateur de vitesse** est un dispositif électronique destiné à commander la vitesse d'un moteur électrique. Il est constitué principalement d'un convertisseur statique et d'une électronique de commande. Les variateurs récents contiennent aussi un étage de correction du facteur de puissance afin de respecter les normes de compatibilité électromagnétique. En général, le **convertisseur** statique est un hacheur ou un onduleur. [5]



Figure 2.2 : variateur de vitesse SSD-582

II.2.3 Présentation du robot GT 6A :

Durant la dernière décennie, la recherche dans le domaine de la commande des robots manipulateurs a connu une évolution significative. Les motivations de cette avancée étaient d'une part l'exigence de plus en plus accrue au niveau des performances dans la réalisation des tâches.

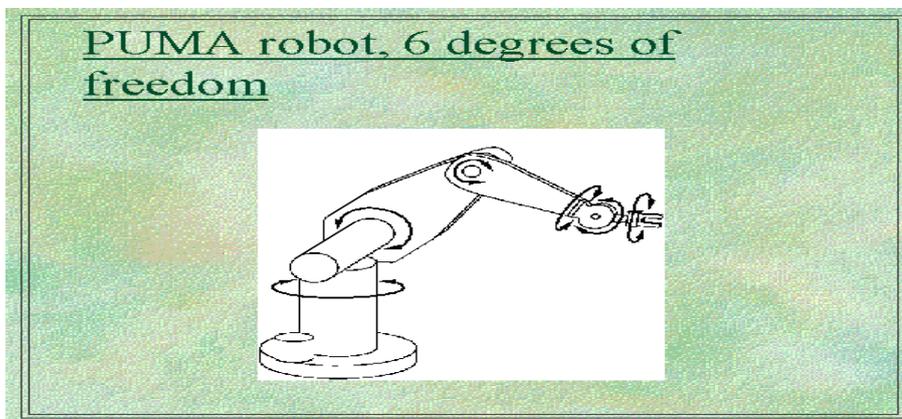


Figure 2.3 : robot PUMA

Un robot est un système mécanique poly-articulé mû par des actionneurs et commandé par un ordinateur qui est destiné à effectuer une grande variété de tâches. [7]

Dans notre chaîne de production on trouve un robot fixe GT à 6Axes qui déplace les produits du convoyeur vers la table tournante tel qu'illustré par la figure suivante :



Figure 2.4 : le robot GT 6A

II.2.4 Capteurs :

Ce sont des éléments capable de détecter une information physique dans leur environnement (présence d'objet, chaleur, lumière bruit...ect) et de la retransmettre sous forme de signal.

Un capteur transforme une grandeur physique en une grandeur normée, généralement électrique, qui peut être interprétée par un dispositif de contrôle commande. [8]

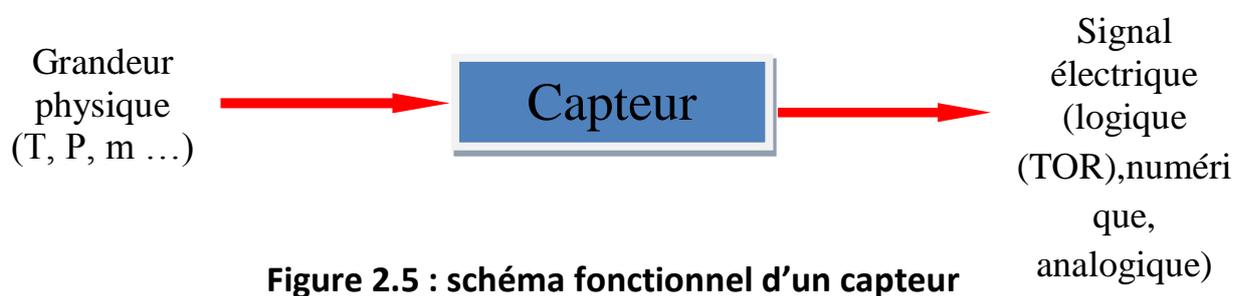


Figure 2.5 : schéma fonctionnel d'un capteur

Dans notre chaîne de production nous intéressons aux capteurs présentés ci-dessous.

Les deux capteurs de notre cellule sont :

- **1^{er} capteur** : capteur infrarouge (ACEL 9405).



Figure 2.6 : capteur infrarouge (ACEL, 9405)

est un capteur infrarouge formé à partir de deux composants autonomes, un émetteur et un récepteur placés l'un en face de l'autre, le signal sera interrompu quand la pièce coupe le faisceau lumineux. Ce système est appelé : système barrage [8]

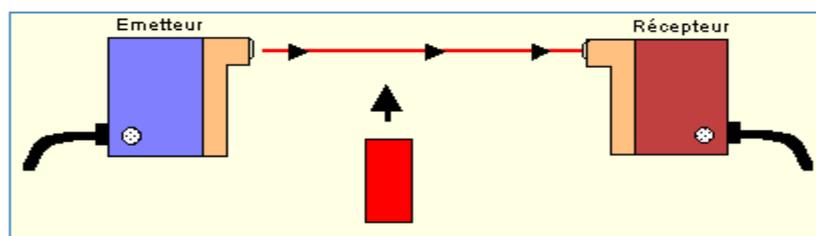


Figure 2.7 : système barrage

- **2^{ème} capteur** : capteur photoélectrique.

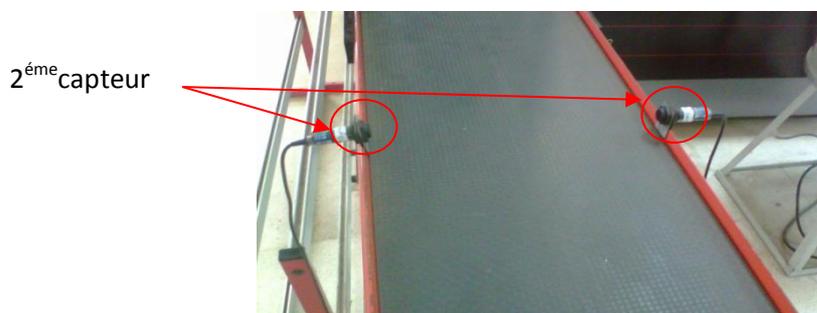


Figure 2.8 : capteur photoélectrique

Il est composé d'un émetteur de lumière (diode électroluminescente) et d'un récepteur sensible à la quantité de lumière reçue (phototransistor), placé l'un en face de l'autre. Le phénomène de détection est effectué lorsque la pièce transperce le faisceau lumineux émis par le détecteur. Cette opération altère passablement la quantité de lumière reçue par le récepteur ce qui entraîne un enchaînement d'état de la sortie.

II.2.5 Convoyeur :

Un **convoyeur** est un mécanisme ou machine qui permet le transport d'une charge isolée (cartons, bacs, sacs, ...) ou de produit en vrac (terre, poudre, aliments...) d'un point A (où se trouve le capteur infrarouge) à un point B (où se situe le capteur photoélectrique).



Figure2.9 : convoyeur

II.3 FONCTIONNEMENT DE LA CELLULE:

L'objectif de cette section est de donner un aspect général sur le pilotage et la supervision d'un système automatisé. Un code à barre est intégré sur chaque pièce, la lecture du code se fait par une caméra (webcam), dans le cas où le code à barre est non identifié ou qu'il n'est pas pris en charge par le système, la pièce doit être redirigé vers un système d'évacuation.

II.3.1 Le cahier de charge de cette application :

- Initialisation du système.
- Mise en marche du système.
- Détection de pièces par le capteur 1(infrarouge).
 - Arrêt du convoyeur.
 - Actualisation d'une temporisation de 5 secondes (5s).
 - Acquisition de l'image.
 - Lecture du code à barre.
- sélection de la tache adéquate.
- fin de temporisation : redémarrage du convoyeur

II.3.2 Diagramme de fonctionnement du système :

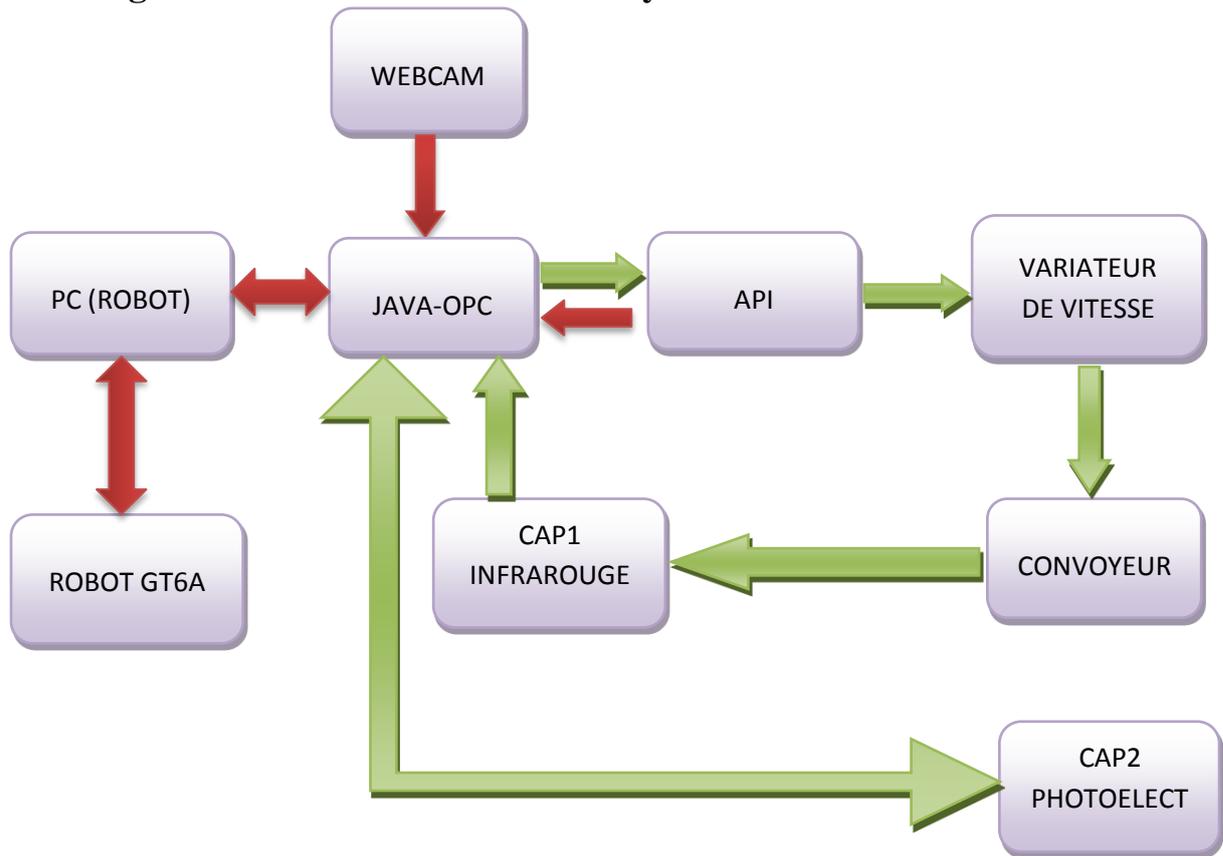


Figure 2.10 : Diagramme de fonctionnement du système

II.3.2.1 Interprétation du schéma

En exécutant le programme sous VERSA_PRO, ce dernier va donner l'ordre au convoyeur de démarrer, une fois qu'un produit arrive au niveau du capteur1 (INFRAROUGE) qui va le détecter et envoyer un message à JAVA-OPC puis vers l'API qui donne l'ordre au convoyeur de s'arrêter.

A cet instant la WEBCAM prendra une photo du code à barre puis l'envoie à JAVA qui la traitera. Une fois cette tâche effectuée l'API qui va redémarrer le convoyeur.

A l'arrivée du produit au capteur2 (PHOTOELECTRIQUE) qui va informer JAVA-OPC et ce dernier communiquera avec le PC du robot qui lui ordonnera d'effectuer la tâche adéquate (déplacement du produit) .

Afin d'assurer la communication entre les différents constituants de la cellule (robot, automate, convoyeur, capteur) comme indiqué précédemment, on a utilisé une plateforme qu'on appelle « JAVA-OPC ».

II.4 Le protocole de communication OPC

II.4.1 HISTORIQUE DE L'OPC :

L'utilisation des ordinateurs dans le domaine de l'automatisation à augmenter d'une façon significatif au début des années 90, en particulier, l'utilisation du système d'exploitation Windows. Ce dernier était utilisé dans le domaine de la visualisation et du contrôle. Auparavant il était presque impossible de combiner efficacement les produits de différents fournisseurs, en fournissant des serveurs dits OPC avec leurs appareils, les bus de données et les logiciels, donc les fournisseurs ouvrent leurs produits à tout client compatible, OPC capable de se connecter au serveur pour l'échange de données aussi gérer plusieurs clients à la fois.

L'effort de développement de la première spécification de communication s'est fait dans un groupe nommé Win SEM (Windows pour la science, l'ingénierie et l'industrie) en avril 1992 à Roadmount (siège de Microsoft).

En 1995, Cinq entreprise (Fisher-Rosemount, Rockwell Software, Opto 22, Intellution, and intuitive Technology) ont décidé de prendre une initiative d'élaboration d'un projet de standard ouvert (Open Connectivity) appelé OPC TASK FORCE.

Ce projet de norme fut présenté à WINSEM en janvier 1996 et adopté comme une feuille de route pour un standard de normalisation.

Des travaux furent lancer pour compléter et élaborer ce standard en organisant des séminaires JumpStart (Dallas-Mars 1996, Londre-Juillet 1996, Japon-Août 1996) La version finale de la spécification fut annoncée le 29 août 1996 OPC spécification V1.0) et elle fut révisée en mois de septembre 1997 (OPC V1.0A). [9]

II.4.2 Définition d'OPC:

OPC (ole for process control) est une norme de communication crée par la fondation opc (410 membres fabricants de logiciels/matériels et utilisateurs, dans (40pays) la norme opc définie un standard de communication entre les applications Windows et les matériels /logiciels utilisé régulièrement dans le domaine du contrôle de processus (ex automates).

Opc décrit une méthode cohérente et universelle permettant d'accéder aux données de terrain des dispositifs d'usine.

OLE (Object linking et Embedding ou en Français Objet lié et Embarqué de Microsoft) est un protocole et un système de gestion d'objets distribués, permettant à des applications utilisant des formats différents de dialoguer.

Opc à l'origine l'abréviation de « OLE for process control », est un processus standardisé et aussi une spécification de l'interface de communication.

Au fils des ans, la fondation OPC à ajouter des spécifications supplémentaires. [10]

II.4.3 SPECIFICATIONS DE LA NORME OPC :

AE (Alarmes et Evénements) : Décrit l'échange basé sur les événements non cyclique des alarmes.

Data exchange : est une spécification de 2002 qui régit la communication directe entre deux serveurs OPC.

OPC batch : pour les traitements par lot.

OPC Common :

OPC Common et OPC Security (spécification commune à tous les serveurs)

Trois fonctionnalités de base pour tout serveur OPC:

- Enregistrement des serveurs dans la base de registre du système.
- Interface IOPC Server List qui permet aux applications clientes de connaître l'existence des serveurs installés sur une machine (mais pas obligatoirement actifs)
- Interface IOPC Shut Down qui permet aux serveurs de signaler à leurs clients qu'ils s'arrêtent.

HDA : l'accès aux données déjà stockées, les archives peuvent être récupérées. [10]

OPC Data Access :

DA (DATA Access) est la norme originale la plus largement utilisée de OPC son but est l'interrogation cyclique des données en temps réel, par exemple la visualisation des applications ; Un serveur OPC Data Access a pour fonction de :

- Collecter les données issues des périphériques matériels (lecture) ou leurs faire parvenir les mises à jour de données (écriture) Ces opérations sont périodiques (lecture) ou aperiodiques (lecture, écriture), en mode synchrone (attente de la réponse) ou asynchrone (réponse)
- Assurer l'intégrité des données (qualité et fraîcheur)
- Répondre aux requêtes de clients (fournir les données, écrire vers le matériel...)
- Avertir les clients abonnés des changements d'état des variables par un événement
- Informer le client des problèmes de sécurité (dysfonctionnement logiciel ou matériel) [10]

II.4.4 Structure de données d'un serveur OPC Data Access :

Objet OPC Item

- Objet interne au serveur, contenant l'information utile (valeur numérique, état d'une variable...). Si le serveur physique est un automate, les items correspondent bits et registres.
- Pour le client, l'Item OPC ne représente pas la valeur de la donnée source mais la référence à cette donnée.

↪ Plusieurs Items peuvent exister pour la même donnée source (méthodes d'accès différentes, redondance...)

Objet OPC Branche

- Les Item OPC sont regroupés par branche dans le serveur. Une branche correspond à une spécificité fonctionnelle (type de bus, automate, esclave...)

↪ Une branche peut comprendre des sous-branches [10]

II.4.5 Côté serveur :

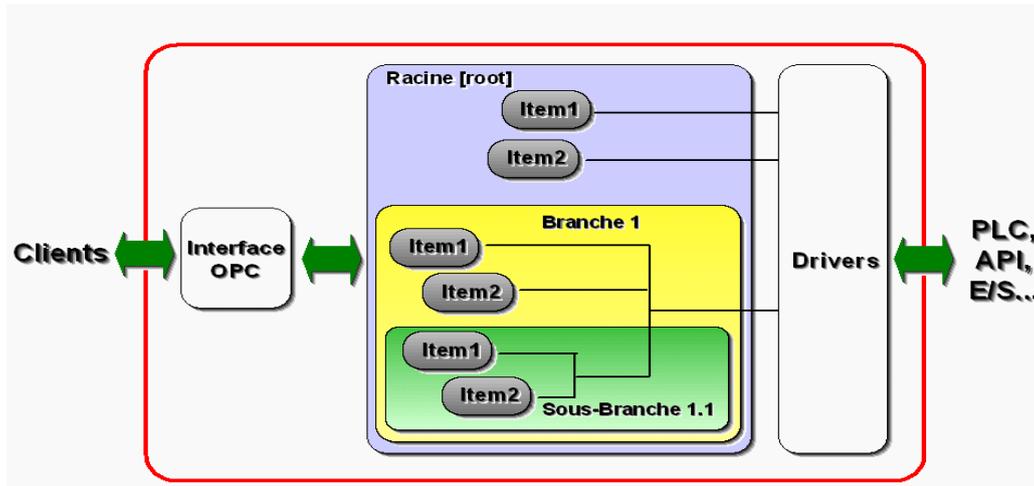


Figure 2.11 : structure OPC côté serveur

II.4.6 Structure de données vues du côté client SERVEURS OPC :

Objet OPC Server

Pour chaque client, le serveur OPC établit un objet de type OPC Server et crée un canal de communication séparé avec le client.

Objet OPC Group

Les transmissions se font sur la base du Groupe.

L'objet Groupe contient les informations sur lui-même et décrit l'organisation logique des items côté client.

- ☞ Le nom de Groupe doit être unique parmi les Groupes du client.
- ☞ Les Groupes peuvent être renommés ou créés dynamiquement à la demande du client. Leur contenu est modifiable dynamiquement par le client.
- ☞ Il existe des Groupes public et local (private)

Le regroupement des items a pour objectif de minimiser le nombre d'échanges entre le serveur et le client. [10]

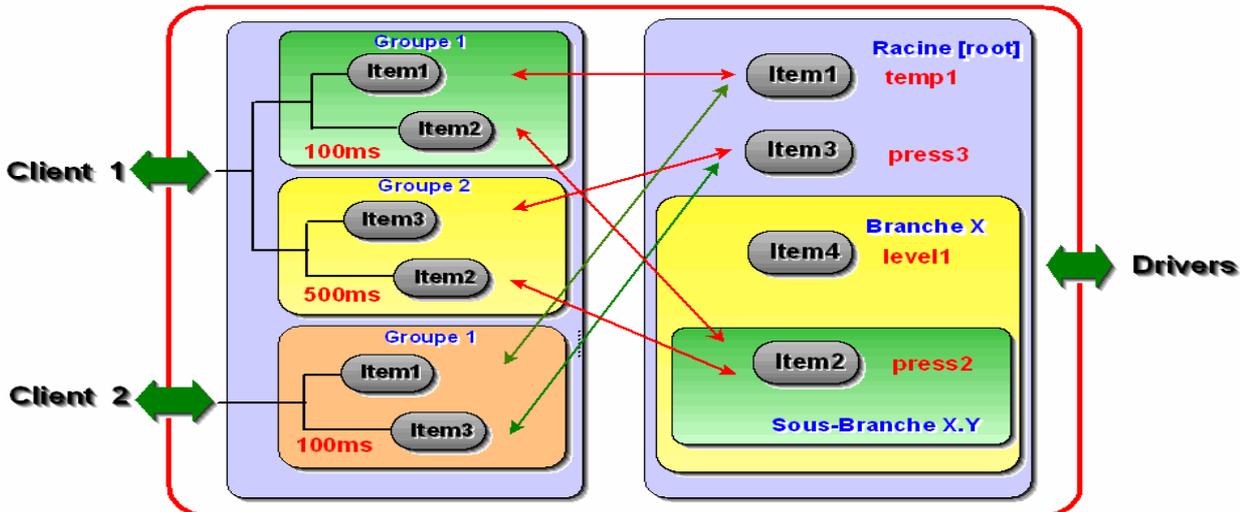


Figure 2.12 : structure OPC côté client

II.4.7 OPC Data Access côté client :

II.4.8 Composant Object COM :

Le modèle COM ou ActiveX est conçu pour permettre à des applications d'accéder à des composants pré-écrits et enregistrés sur une machine Windows. Le composants ActiveX est généralement un fichier de type exe , dll ou ocx . C'est une norme propriétaire (Microsoft) et non ouverte, sa durée de vie est imposée par le propriétaire. Microsoft souhaite actuellement faire disparaître DCOM au profit de .NET

- Tous les composants ActiveX suivent un modèle de programmation unique défini par Microsoft. Les outils de développement usuels (famille Visual Studio) permettent de gérer le modèle de façon transparente pour le programmeur.
- Pour le client (appelant), tous les composants ActiveX sont accessibles par l'intermédiaire de pointeurs d'interface. Si le composant est sur la même machine que l'appelant (composant in-process), l'appel du client est direct pour le composant. [10]

II.4.9 Le modèle COM distribué ou DCOM :

Lorsque le composant ActiveX est hors de la machine du client (out-of-process) , l'appel atteint d'abord un objet de type proxy fourni par le modèle COM. Cet appel est reçu par un stub (squelette de classe) sur la machine distante, qui appelle lui-même le composant COM. Le processus est totalement transparent pour l'utilisateur.

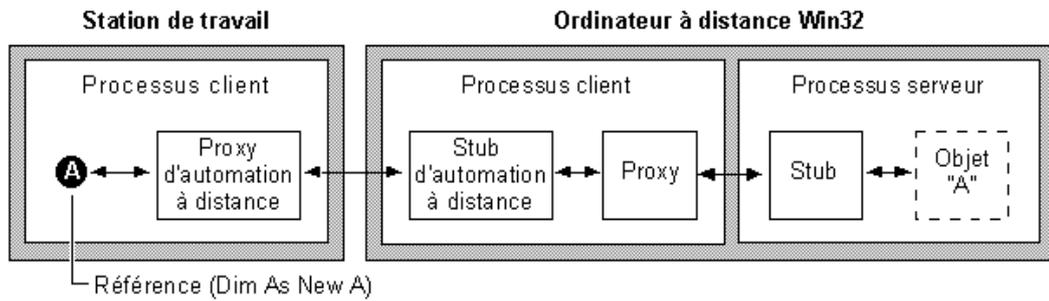


Figure 2.13 : structure d'un modèle COM distribué

II.4.10 Avant OPC :

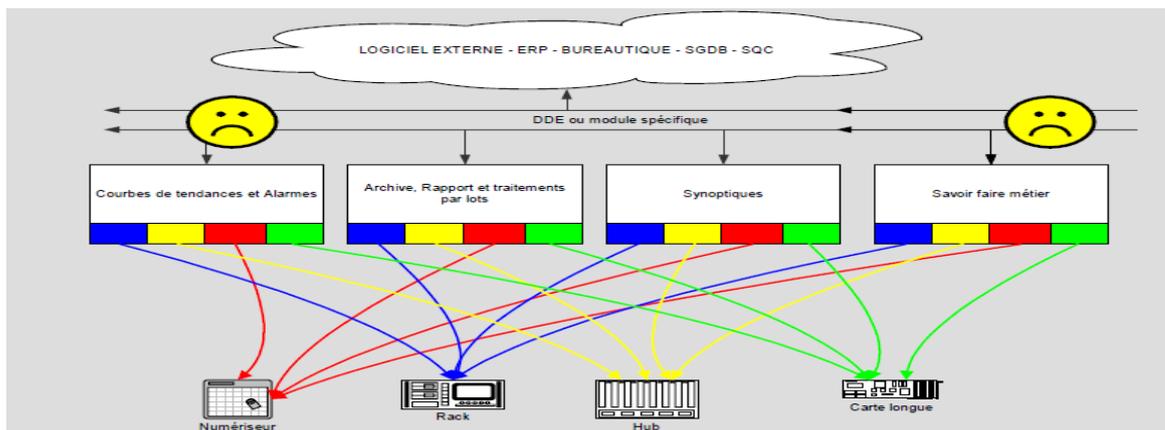


Figure 2.14 : Avant OPC

Dynamic Data Exchange (DDE) ne tenaient pas comptes des spécificités des environnements, DDE est la propriété de Microsoft qui n'a pas souhaité maintenir cette fonctionnalité depuis de nombreuses années (abandon total sur la plate-forme .net puis sur Win Seven) au profit de COM. Certains éditeurs (Wonderware) poursuivent la maintenance des liaisons établies sous DDE (FastDDE).

II.4.11 Depuis OPC :

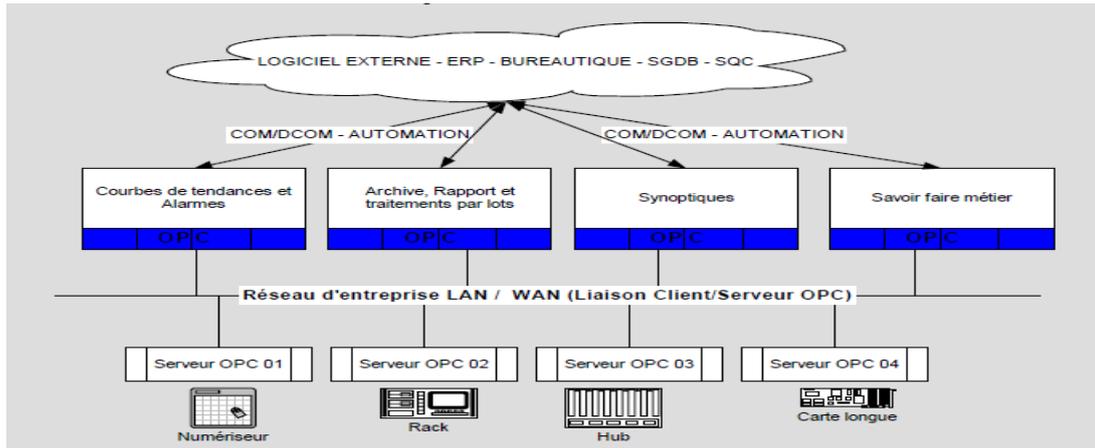


Figure 2.15 : Depuis OPC

Notez bien que COM/DCOM est une spécification indépendante du matériel et du système d'exploitation. [10]

II.5 Introduction au langage Java

Le langage Java est un langage généraliste de programmation synthétisant les principaux langages existants lors de sa création en 1995 par Sun Microsystems. Il permet une programmation orientée-objet (à l'instar de SmallTalk et, dans une moindre mesure, C++), et reprend une syntaxe très proche de celle du langage C. Outre son orientation objet, le langage Java a l'avantage d'être **modulaire** (on peut écrire des portions de code génériques, ç-à-d utilisables par plusieurs applications), **rigoureux** (la plupart des erreurs se produisent à la compilation et non à l'exécution) et **portable** (un même programme compilé peut s'exécuter sur différents environnements). En contrepartie, les applications Java ont le défaut d'être plus lentes à l'exécution que des applications programmées en C par exemple. [11]

II.5.1 Environnement Java

Java est un langage interprété, ce qui signifie qu'un programme compilé n'est pas directement exécutable par le système d'exploitation mais il doit être interprété par un autre programme, qu'on appelle interpréteur. La figure illustre ce fonctionnement.

Exemple :

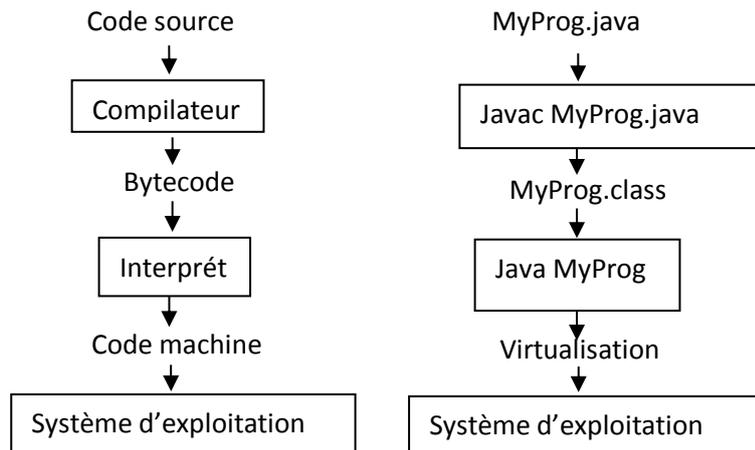


Figure2.16 : Environnement JAVA

Un programmeur Java écrit son code source, sous la forme de classes, dans des fichiers dont l'extension est .java. Ce code source est alors compilé par le compilateur javac en un langage appelé bytecode et enregistré le résultat dans un fichier dont l'extension est .class. Le bytecode ainsi obtenu n'est pas directement utilisable. Il doit être interprété par la machine virtuelle de Java qui transforme alors le code compilé en code machine compréhensible par le système d'exploitation.

C'est la raison pour laquelle Java est un langage portable : le bytecode reste le même quel que soit l'environnement d'exécution.

En 2009, Sun Microsystems est racheté par Oracle Corporation qui fournit dorénavant les outils de développement Java SE (Standard Edition) contenus dans le Java Development Kit (JDK). [11]

Compilation

La compilation s'effectue par la commande javac suivie d'un ou plusieurs noms de fichiers contenant le code source de classes Java. Par exemple, javac MyProg.java compile la classe My-Prog dont le code source est situé dans le fichier MyProg.java. La compilation nécessite souvent la précision de certains paramètres pour s'effectuer correctement.

II.5.2 Programmation orientée-objet :

Chaque langage de programmation appartient à une "famille" de langages définissant une approche ou une méthodologie générale de programmation. Par exemple, le langage C est un langage de programmation procédurale car il suppose que le programmeur s'intéresse en priorité aux traitements que son programme devra effectuer. Un programmeur C commencera par identifier ces traitements pour écrire les fonctions qui les réalisent sur des données prises comme paramètres d'entrée. La programmation orientée-objet (introduite par le langage SmallTalk) propose une méthodologie centrée sur les données. Le programmeur Java va d'abord identifier un ensemble d'**objets**, tel que chaque objet représente un élément qui doit être utilisé ou manipulé par le programme, sous la forme d'ensembles de données. Ce n'est

que dans un deuxième temps, que le programmeur va écrire les traitements, **en associant chaque traitement à un objet donné**. Un objet peut être vu comme une entité regroupant un ensemble de données et de **méthodes** (l'équivalent d'une fonction en C) de traitement.

1) *Classe* :

Un objet est une variable (presque) comme les autres. Il faut notamment qu'il soit déclaré avec son type. Le type d'un objet est un type complexe (par opposition aux types primitifs entier, caractère, ...) qu'on appelle une **classe**.

Une classe regroupe un ensemble de données (qui peuvent être des variables primitives ou des objets) et un ensemble de méthodes de traitement de ces données et/ou de données extérieures à la classe. On parle d'**encapsulation** pour désigner le regroupement de données dans une classe.

2) *Objet* :

Un objet est une instance (anglicisme signifiant « cas » ou « exemple ») d'une classe est référencé par une variable ayant un état (ou valeur). Pour créer un objet, il est nécessaire de déclarer une variable dont le type est la classe à instancier, puis de faire appel à un constructeur de cette classe. [11]

II.6 Travail demandé

Pour effectuer un changement à l'ancienne cellule, on a remplacé l'armoire par une nouvelle qui est doté d'un automate SCHNEIDER M-340 (qu'on expliquera dans l'annex2) à la place de GE-FANUC et aussi identifier le câblage de l'ancienne armoire pour pouvoir câbler la nouvelle. Ainsi de remplacer le système d'identification code à barre par la nouvelle technologie RFID qui sera bien détailler dans le 3^{ème} chapitre.

I.6.1 Identification du câblage de l'ancienne armoire:

Pour pouvoir câbler la nouvelle armoire, on a fait une identification pour tous les éléments de l'ancienne, pour le faire d'abord on a essayé de comprendre le fonctionnement de chacun de ces derniers et savoir comment se fait leur câblage et enfin suivre le câblage vers l'automate. Voir figure (2-7).



Figure 2.17 : L'ancienne armoire

II.7 Conclusion :

Dans ce chapitre, on a pu identifier l'ancienne cellule et comprendre son fonctionnement tout en précisant l'interface de communication utilisé entre les éléments de la cellule (JAVA-OPC) et aussi d'énoncer le travail qu'on doit effectuer ainsi de présenter l'automate

M-340.



**CHAPITRE III :
PRESENTATION DE LA
TECHNOLOGIE RFID**

III.1 Introduction :

Insérer une clé pour démarrer un véhicule, badgé pour accéder à un bâtiment ou une salle, valider un titre de transport dans le bus ou le métro sont des gestes entrés dans le quotidien de bon nombre d'entre nous. On utilise, sans en être toujours conscient, des technologies de capture automatique de données basées sur les ondes et rayonnements radiofréquence.

En effet la Radio-Identification ou la RFID est l'annonce d'une mutation radicale dans l'organisation du commerce, du transport, de la sécurité et de la surveillance.

L'objectif de ce chapitre est de présenter la technologie RFID. Dans la première section, une présentation du fonctionnement des systèmes RFID est réalisée. Elle aborde les différents composants existants des dispositifs RFID. La seconde section aborde les différents types d'identifiants ou tags. La troisième section développe les applications de la RFID.

III.2 Bref historique :

Le principe de la RFID est utilisé pour la première fois lors de la Seconde Guerre Mondiale en 1940 pour identifier/authentifier des appareils en vol (IFF: Identifie Friendly Foie). Il s'agissait de compléter la signature RADAR des avions en lisant un identifiant fixe permettant l'authentification des avions alliés. Mais, elle ne connaît un réel intérêt que depuis les années 90.

En 2003, la RFID s'inscrit dans le standard Electronic Product Code (EPC global). Ce dernier, souvent considéré comme la nouvelle génération de codes à barres, le numéro EPC est encodé dans une étiquette RFID, et d'un réseau de partage d'informations, l'EPC Network. La RFID reste la plus intéressante parmi toutes les techniques existantes de traçabilité, d'identification et de sécurité. [12]

III.3 Présentation de la RFID :

Pour lire un code barre, il faut s'équiper d'un lecteur ou d'un scanner à infrarouge. Celui-ci lit la série de barres blanches et noires, la traduit et l'envoie à l'ordinateur sous la forme d'une chaîne de caractère. Dans notre application, le lecteur de code barre ne sera pas connecté à l'ordinateur mais à l'automate programmable. Bien que le code barre soit populaire dans la traçabilité, cette technologie présente quelques défauts.

Le premier est que le code barre n'est pas fiable. En effet, il peut être détérioré facilement, par exemple une tâche, ce qui rendrait son utilisation impossible par un scanner automatique. Le second est que le code barre n'est pas évolutif. Il est impossible de modifier un code barre après une transformation subite par le produit, la solution est d'en rajouter un autre ou de le remplacer pour affecter une nouvelle identité au produit. Le dernier défaut est que le code barre ne contient pas d'autres informations que l'identité du produit. Il aurait été intéressant pour les professionnels qu'il puisse contenir le poids, le volume, la date de péremption du produit ou encore sa date de livraison. Mais on comprend bien que l'on ne peut pas stocker toutes ces informations dans un simple code alphanumérique de treize caractères. C'est pour pallier ces défauts qu'une nouvelle technologie émerge dans le monde de la traçabilité: le

RFID (Radio Frequency IDentifica) qui est une nouvelle technologie d'identification qui va être de plus en plus répandue dans le monde industriel pour assurer la traçabilité de produits. Comme son nom l'indique, elle fait de l'identification par radio fréquence. C'est-à-dire que chaque produit sera équipé d'un tag, une puce électronique sans alimentation électrique, qui peut être détecté, écrit et lu par un transpondeur.

III.4 Composition d'un système RFID

Un système RFID se compose de deux éléments : l'étiquette (tag) et le lecteur illustrer par la figure suivante :

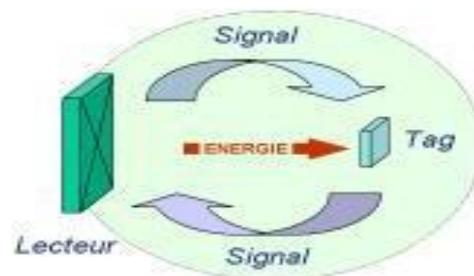


Figure 3.1 : la communication entre le lecteur et le tag

III.4.1 Le tag :

Le tag (l'étiquette) appelé aussi transpondeur, pour transmettre-répondre, comprend une puce, dotée d'une mémoire, reliée à une antenne bobinée. Le plus souvent, le tag est collé sur un film en plastique ou moulé dans une carte au format carte de crédit.

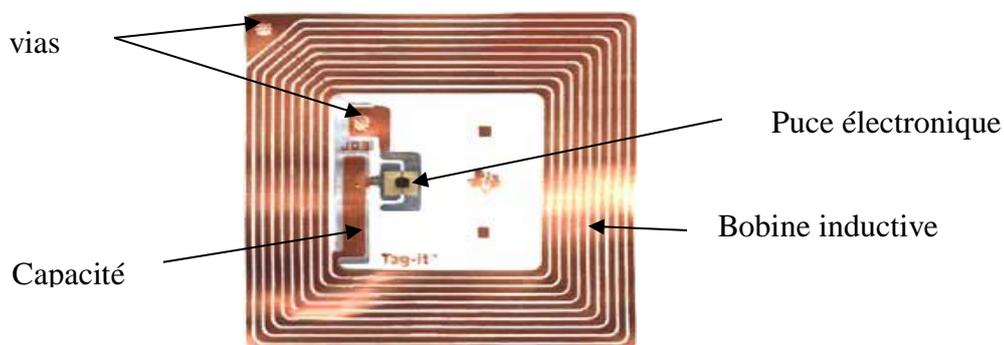


Figure 3.2 : la structure du tag

Il existe trois types d'étiquette : des étiquettes passives, des étiquettes actives et des étiquettes semi-actives.

III.4.1.1 Type des tags :

La plupart des étiquettes n'exigent aucune source d'énergie si ce n'est quand elles sont interrogées par le capteur. Celui-ci les alimente alors par son signal radio. D'autres étiquettes, dites actives, sont équipées d'une mini batterie qui permet de communiquer à plus grande distance ou d'enregistrer des informations.

La technologie RFID inclut l'usage de tags (étiquettes), ces tags peuvent être actifs ou passifs afin de transmettre les données souhaitées en termes de traçabilité des objets/personnes. [12]

Tags Passifs :

Les tags passifs ne disposent pas de batterie; toute l'énergie nécessaire à leur fonctionnement leur est fournie par le lecteur. En conséquence, les étiquettes passives transmettent des données seulement quand « s'est réveillé » par un lecteur.

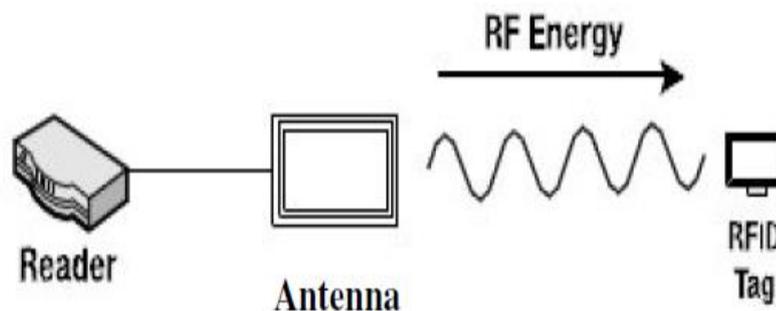


Figure 3.3 : Alimentation du tag passif

Les tags passifs utilisent différentes bandes de fréquences radio selon :

- leur capacité à transmettre les données à des distances plus ou moins grandes.
- les substances différentes que les données doivent traverser (air, eau, métal etc.).

Les applications où la technologie de RFID passive offre un avantage certain sont nombreuses et diverses, prenons l'exemple de l'identification d'animaux, veiller à la sécurité des nouveaux nés ainsi que la traçabilité des déchets et la gestion des stocks.

L'étiquette passive repose davantage sur le coût des tags moins onéreux que les tags actifs.

Ce système s'avère très utile pour les marchandises en volume important lorsque les marchandises peuvent être lues à courte distance (passage à la caisse des supermarchés).

Aussi, la distance de lecture est cependant un réel frein à ce système puisque le lecteur doit se situer dans le champ du tag afin d'en récupérer les données. [12]

Tags Semi-actifs :

Les tags semi-actifs fonctionnent comme les tags passifs, sauf qu'ils comportent une batterie. Ces tags semi-actifs, également appelés semi-passifs, ressemblent au fonctionnement des étiquettes actives puisqu'ils sont également alimentés par une source d'énergie embarquée. Seulement, la différence entre ces 2 types d'étiquettes repose sur l'alimentation de la batterie

c'est-à-dire elle alimente la puce RFID non pas en continue mais à des intervalles de temps réguliers et programmables et n'envoie aucun signal. [12]

➤ **Domaine d'utilisation :**

La batterie des étiquettes qui constituent la RFID semi-active n'envoie pas de signal comme c'est le cas pour la RFID passive. Cependant elles permettent d'enregistrer des données par exemple lors du transport de marchandises.

La technologie RFID semi-actif s'avère notamment très utile dans le domaine de la traçabilité alimentaire et la traçabilité logistique pour :

- enregistrer les changements de température lors du transport,
- la surveillance de parcs de machines, etc.

La radio-identification présente en elle-même de multiples avantages. Utiliser un support d'identification semi-actif présente, à son tour, des avantages certains par rapport aux tags actifs ou passifs. Elle combine des avantages que l'on ne retrouve pas dans d'autres technologies d'identification :

Tags Actifs :

La radio identification active est une forme de technologie d'identification caractérisée par l'usage de tags actifs également appelés étiquettes actives. Les tags sont de petits objets qui peuvent être collés sur des objets ou insérés dans ces mêmes objets, ils sont composés :

- d'une puce électronique.
- d'une antenne.

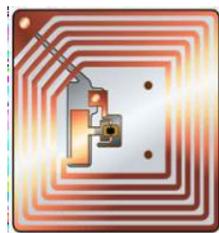


Figure 3.4 : Tag RFID actif

Ainsi, on parle de radio-identification active lorsque les étiquettes ou les tags sont actifs c'est-à-dire qu'ils sont alimentés par une source d'énergie embarquée: batterie, pile... Cette source d'énergie permet à la puce de diffuser un signal vers le lecteur RFID.

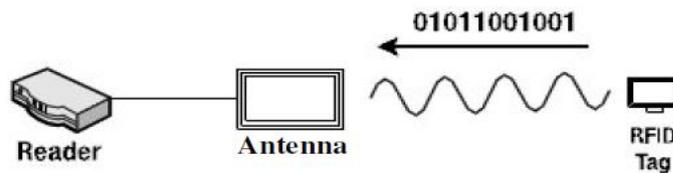


Figure 3.5 : Transmission des données vers le lecteur

Tout comme le système RFID passive, la RFID active connaît des domaines d'application très divers, prenons ces quelques exemples :

- Contrôle d'accès des véhicules et piétons.
- Protection de personnes dans le milieu industriel.
- Suivi de containers.

Contrairement au système d'étiquettes passives, les étiquettes actives sont équipées d'une énergie propre qui leur permet d'émettre un signal de manière autonome.

De ce fait le principal avantage repose sur la longue distance à laquelle elles peuvent communiquer les données sans qu'un lecteur RFID se situe à proximité du tag. Sauf que la multiplication des échos radios engendre à leur tour des difficultés de localisations des puces par les lecteurs ce qu'on appelle le phénomène d'interférence. En plus de cet inconvénient, cette technologie repose aussi sur le coût des étiquettes, la confidentialité des informations transmises ainsi que l'impact sur la santé très controversé dû à l'émission d'ondes magnétiques et la durée de fonctionnement limité des étiquettes.

Le tableau suivant dresse un comparatif entre les systèmes actifs, semi-actifs et passifs. On voit tout l'intérêt que présentent les systèmes passifs en termes de coût et de durée de vie.

Type	Distance	Données	Durée de vie	coût
Actif	importante	Importantes	faible	important
Semi actif	moyenne	Moyennes	moyenne	moyenne
Passif	faible	Faible	importante	faible

Tableau1 : Comparaison de différents systèmes RFID

Chacune de ces technologies de radio-identification présente des avantages qui leurs garantissent un marché. Cependant la forte demande dans le domaine de la traçabilité suscite un réel intérêt pour le développement des tags RFID faible coût et performant en terme de débit, de distance de lecture et de rapidité de lecture. La technologie RFID passive est celle qui répond le mieux à ce besoin.

II.4.1.2 Types de lecture d'étiquette :

Aussi il existe trois autres types pour la lecture d'étiquettes : les étiquettes en lecture seule, celles en écriture une fois et lecture plusieurs fois, et enfin celles en lecture / écriture multiples.

On peut stocker de l'information de deux façons différentes :

- soit on mémorise un "pointeur" (une clé d'accès) dans l'étiquette, en indiquant ainsi un chemin vers une base de données où trouver cette information nécessaire au traitement. L'application est dite centralisée.

- soit on mémorise l'information elle-même dans l'étiquette. L'application est alors décentralisée.

Les étiquettes à lecture seule permettent la réalisation de systèmes appartenant à la première catégorie : systèmes centralisés.

Les étiquettes à écriture unique et lecture multiple permettent la réalisation de systèmes appartenant aux deux catégories : systèmes centralisés ou décentralisés.

Les étiquettes à écriture et lecture multiple permettent la réalisation de systèmes appartenant plus avantagement à la deuxième catégorie : systèmes décentralisés. [12]

Lecture seul :

Les données sont inscrites dans l'étiquette par le fabricant, et ne peuvent pas être modifiées ni complétées par la suite. Les utilisateurs ne peuvent que lire les données contenues.

Ces étiquettes, dont les fonctions peuvent sembler réduites, sont néanmoins sans doute promises à un grand avenir. Leur prix est plus faible que celui de leurs consœurs offrant des fonctions plus complexes. Et, dans de nombreuses applications un simple numéro d'identification, si son unicité est garantie, peut suffire à tracer un objet. Les données complémentaires sur l'objet n'ont pas besoin d'être stockées dans l'étiquette, mais peuvent être mise à disposition, échangée ou retrouvée dans les bases de données des systèmes d'information. [13]

Ecriture une fois, lecture plusieurs fois (WORM) :

L'étiquette est fournie à l'utilisateur, vierge. En fait, dans la majorité des cas le fondeur ou le fournisseur l'a déjà munie d'une identification. Lors de sa pose sur l'objet à tracer, l'utilisateur va écrire les données qui lui seront utiles par la suite. Lors de la vie ultérieure de l'étiquette, cette information pourra être lue, mais ne pourra être ni modifiée ni complétée. [13]

L'ecture /écriture multiples :

L'étiquette est fournie, vierge, dans les mêmes conditions que les WORM, mais elle pourra être écrite plusieurs fois, effacée, modifiée, complétée, et lue. Le nombre de répétitions de ces opérations peut dépasser 500 000. [13]

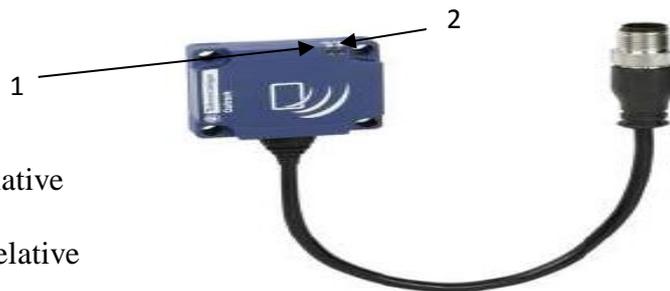
Les étiquètes qu'on utilisera dans notre projets sont comme suit :



Figure 3.6 : l'étiquette RFID

II.4.2 le lecteur

Le lecteur, selon la technologie utilisée, peut lire mais aussi écrire des données sur le tag. Il émet des ondes radio et des champs magnétiques, puis écoute les réponses des tags qui se trouvent dans son champ de lecture. Le lecteur contient typiquement un module radio (émetteur et récepteur) et une interface de contrôle. La plupart des lecteurs fournissent une interface supplémentaire de type RS-232 pour transférer les données reçues à d'autres systèmes (PC, robots, etc.). [12]



- 1 → TAG : DEL relative aux étiquettes
- 2 → COM : DEL relative à la communication

Figure 3.7 : le lecteur RFID

La communication entre le lecteur et l'étiquette s'effectue en quatre temps :

- 1) Le lecteur transmet par radio l'énergie nécessaire à l'activation du tag.
- 2) Il lance alors une requête interrogeant les étiquettes à proximité.
- 3) Il écoute les réponses et élimine les doublons ou les collisions entre réponses.
- 4) Enfin, il transmet les résultats obtenus aux applications concernées.

➤ Description de l'interface RS-232 :

Le port série des PC sont conforme à la norme de télécommunications RS-232 (Recommended Standard 232). Sur les systèmes d'exploitation MS-DOS et Windows, les ports RS-232 sont désignés par les noms COM1, COM2, etc. Cela leur a valu le surnom de «ports COM », encore utilisé de nos jours. Cependant, ils sont de plus en plus remplacés par les « ports USB ». Certains composants FTDI (La société FTDI a bien mis sur le marché des circuits intégrés qui permettent de traiter un port USB comme n'importe quel port série). Permettent d'émuler un flot de données à la norme RS232 sur une transmission du type USB. La plupart des ports de transmission utilisent seulement +5 V et -5V. Ces niveaux sont compatibles avec la plupart des périphériques et des ordinateurs hôtes.

III.5 Principe de fonctionnement :

Une application d'identification automatique RFID se compose d'un lecteur ou interrogateur qui transmet un signal selon une fréquence déterminée vers une ou plusieurs étiquettes radio situées dans son champ de lecture. Celles-ci transmettent en retour un signal. Lorsque les étiquettes sont « éveillées » par le lecteur, un dialogue s'établit selon un protocole de communication prédéfinie, et les données sont échangées. Les étiquettes sont aussi appelées « transpondeur », c'est à dire un équipement destiné à recevoir un signal radio et à renvoyer immédiatement en réponse un signal radio différent et contenant une information pertinente.

Les applications RFID fonctionnant à basse ou moyenne fréquence (fréquences de 9 Khz à quelques Mhz), utilisent le champ électromagnétique créé par l'antenne du lecteur et l'antenne / bobine de l'étiquette pour communiquer. Le champ électromagnétique alimente l'étiquette et active la puce.

Cette dernière va exécuter les programmes pour lesquels elle a été conçue. Pour transmettre les informations qu'elle contient, elle va créer une modulation d'amplitude ou de phase sur la fréquence porteuse. Le lecteur reçoit ces informations et les transforme en binaire (0 ou 1). Dans le sens lecteur vers étiquette, l'opération est symétrique, le lecteur émet des informations par modulation sur la porteuse. Les modulations sont analysées par la puce et numérisées. Une des particularités de ce principe est que plus la fréquence porteuse est basse plus le nombre de tours de l'antenne de la puce doit être important pour créer un voltage suffisant pour alimenter la puce et par voie de conséquence augmente la complexité du processus de fabrication en grande quantité. L'étiquette peut être apposée, portée, insérée dans un objet. Le mot « objet » est entendu au sens large, ce peut être un colis, une carte intelligente (téléphone, banque), un véhicule, Plus les différents éléments composants ces applications sont standardisés, plus est grand le nombre d'utilisateurs potentiels et de processus industriels ou marchands pouvant bénéficier de cette technologie. [13]

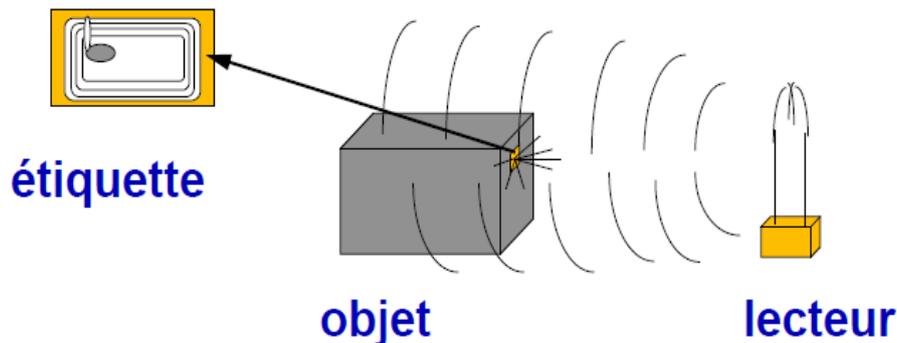


Figure 3.8 : Principe de fonctionnement de RFID

III.6 La communication :

Quand le transpondeur, qui ne possède généralement pas d'alimentation propre, n'est pas dans le champ d'action d'un lecteur, il est totalement passif. L'énergie, les données et les pulsations d'horloge nécessaires à l'activation et au fonctionnement du transpondeur lui sont fournies par le lecteur. On distingue deux cas, qui peuvent se recouvrir : la communication par champs électromagnétiques et la communication par ondes radio.

III.6.1 La communication par champs électromagnétiques :

Dans le cas des basses fréquences moins de quelques MHz un courant alternatif dans l'antenne du lecteur induit du courant dans l'antenne bobinée du tag, ce qui éveille et alimente la puce. La puce effectue les opérations pour lesquelles elle a été conçue, puis crée une modulation d'amplitude ou de phase sur la fréquence porteuse. Le lecteur reçoit ces informations, qu'il transforme en code binaire. Dans l'autre sens, du lecteur vers la puce, les informations circulent selon le même principe, par modulation sur la porteuse. Plus la fréquence est basse, plus le nombre de tours de l'antenne bobinée nécessaires à la création d'un voltage suffisant est important. Cela augmente la complexité et les coûts de fabrication.

III.6.2 La communication par ondes radio :

Sur d'autres systèmes RFID, notamment si la fréquence utilisée dépasse quelques MHz ou que le tag se trouve au-delà d'une certaine distance du lecteur, les données ne peuvent plus être transmises par modulation ; on utilise alors la réflexion des ondes radio. L'électronique du tag modifie l'impédance de l'antenne, renvoyant une partie des ondes radio au lecteur. Le lecteur, doté d'un capteur très sensible, décode les données du tag d'après le type de réflexion reçu.

On distingue également les communications full duplex / half duplex et les communications séquentielles. En mode full et half duplex, le tag diffuse ses informations d'es qu'ils trouvent dans le champ du lecteur. À l'inverse, dans les procédures séquentielles, le tag qui se trouve dans le champ du lecteur est activé brièvement, à intervalles réguliers.

Les nouvelles technologies RFID, par leur plus grande souplesse, rendent l'échange d'information nettement plus rapide et efficace. Elle devrait remettre en question les structures de télécommunications existantes en ouvrant de nouvelles perspectives à des modèles de gestion inédits. En effet, leurs développements devraient bientôt permettre l'identification individuelle et unique des objets et mener à la création d'un « Internet des objets » : une prolongation de l'internet que nous connaissons au monde réel.

III.7 Distance de communication en technologie RFID :

Au stade où nous en sommes, nous allons introduire en quelques mots la notion de distances de communication pour la technologie RFID et par conséquent les applications sous entendues par les utilisateurs du système.

III.7.1 Très courte distance :

Il existe de nombreuses applications dont la distance de fonctionnement entre la base station (lecteur) et l'identifiant doit/peut être quasi nulle, l'essentiel étant principalement de réaliser un isolement électrique galvanique pour satisfaire l'application.

III.7.2 Courte distance :

Les lecteurs Mifare permettent une lecture sans contact de faible distance en utilisant la technologie RFID à 13,56 MHz (en accord avec la norme ISO10536), fonctionnant en général sur des distances de l'ordre de quelques millimètres voir une dizaine de millimètres.

III.7.3 Proximité :

Il en est de même pour la notion de sans contact dit de proximité (Proximity) par exemple ISO 14 443 – qui représente sous-entend des distances de l'ordre de la dizaine de centimètres. Sont incluses dans cette famille d'applications les cartes à puces sans contact « à geste volontaire » de présentation pour les applications de types bancaires, paiement, transports, contrôle d'accès, etc.

III.7.4 Voisinage :

Même motif, même punition pour le sans contact dit de voisinage (vicinity) –par exemple ISO 15 693/ISO 18 000-x. Les distances sous-entendues sont de l'ordre de 50 cm à 1 m et supportent les applications « mains libres » dans lesquelles rentrent par exemple les applications de contrôle d'accès, reconnaissance et suivi de bagages dans les aéroports, passage de chariots, etc.

III.7.5 Longue distance :

Ce terme est généralement utilisé pour des applications fonctionnant sur des distances de l'ordre de 1 à 10 m. Citons par exemple les applications de sorties (gates) des super/hypermarchés ou de lectures de palettes. Au-delà de ces distances, on parle de systèmes très longues distances (supérieures à la dizaine voire centaine de mètres) qui offrent des solutions de types de liaisons radiofréquences dans lesquelles les identifiants possèdent leurs

propres alimentations embarquées (type actifs) contrairement aux applications précédentes (de type passifs).

III.8 la fréquence :

Plus la fréquence est élevée, plus la vitesse d'horloge permettra des calculs rapides, comme des applications cryptographiques. De même, une bande passante plus large permettra des débits plus élevés. Par contre, la portée se trouve réduite et la transmission est plus sensible aux interférences. Donc les différentes fréquences de communication peuvent être BF (125 - 134,2 kHz), HF (13,56 MHz), UHF (869 - 950 MHz) et SHF (autour de 2,45 GHz). [14]

III.9 Application de la technologie RFID :

Si on regarde de plus près le principe de fonctionnement de la technologie RFID, on comprendra qu'elle peut générer toutes sortes d'applications impliquant l'identification, la traçabilité, l'analyse des données, le contrôle des accès etc. Les applications RFID s'appuient sur différents standards dépendant des fonctionnalités exigées par les processus métier et par certaines contraintes locales. La RFID permet de répondre à un grand nombre de besoins. Elle se développe bien en intra entreprise et dans la logistique. [12]

Dans les entreprises :

Les applications des étiquettes RFID sont déjà très nombreuses, il s'agit simplement ici de donner quelques exemples des possibilités offertes par cette technologie.

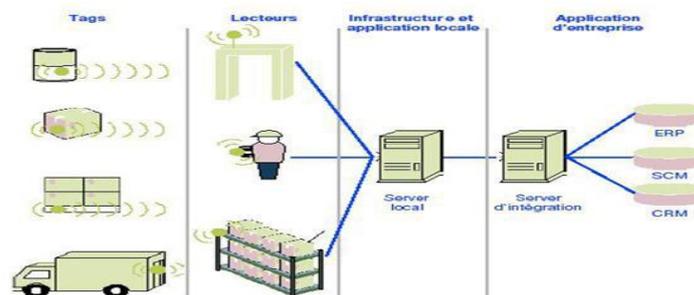


Figure 3.9 : La gestion de la chaîne d'approvisionnement

En matière logistique, quatre niveaux d'applications peuvent être distingués:

1) Expédition :

L'étiquette peut faciliter le choix des produits, la constitution des palettes, leurs tris par destination, le contrôle du chargement. L'étiquette peut contenir, outre l'identification du produit, ou du contenu d'une palette, celle du numéro de lot de fabrication, l'identification du destinataire, le numéro de commande, des particularités de manutention, etc.

Ces informations collectées au moment du chargement peuvent être stockées dans l'identifiant RFID du conteneur ou du moyen de transport afin de faciliter les contrôles en cours de transit, douane.

Des informations peuvent également être accessibles, celles-ci relatives aux identifiants, via des serveurs si les entités partagent un réseau ou certaines informations.

2) Réception :

Les données pourront automatiquement être collectées pour effectuer les contrôles et mettre à jour les stocks et effectuer les rapprochements avec documents commerciaux.

3) En transit :

L'étiquette permet de tracer le produit à chaque point de chargement et de déchargement ou simplement de passage. Ainsi l'expéditeur peut être à tout moment informé du déroulement du transport.

4) Local :

Les étiquettes permettent les inventaires de produit, mais aussi la gestion des supports de manutention et des équipements [12]

En logistique :

Non seulement, on transformant les animaux en machine communicante mais en implantant dans chaque chose chaque être un mini mouchard électronique.

Cette technologie permet d'éviter les attentes au péage, facilite les trajets en transport en commun, elle est implantée à même la peau pour tracer les chevaux de course ou dans un bracelet pour veiller à la sécurité des nouveaux nés.

- ✓ Traçabilité d'objets tels que les livres (bibliothèque, librairie...).
- ✓ Traçabilité d'objets en zone de douanes.
- ✓ Traçabilité des bagages en zone aéroportuaire.
- ✓ Marquage des produits dans les grandes surfaces (lutte contre le vole).
- ✓ Contrôle d'accès à partir du badge.
- ✓ Gestion de parc de location de véhicule.
- ✓ Contrôle des performances des athlètes dans des compétitions de masses (marathon) ou il est impossible du fait de la densité de faire un contrôle individuel.
- ✓ Des puces sous-cutanées sont implantées aussi chez l'être humain : accès à des sites hautement sécurisés. [12]

III.10 Avantages et inconvénients :

Avantages

La capacité de mise à jour du contenu par les intervenants A la différence du code à barres pour lequel les données sont figées une fois imprimée ou marquée, le contenu des données stockées dans une étiquette radio fréquence va pouvoir être modifié, augmenté ou diminué par les intervenants autorisés (étiquettes en lecture et écriture multiple).

- Une plus grande capacité de contenu
- La vitesse de marquage
- Une sécurité d'accès au contenu
- Une plus grande durée de vie
- Une plus grande souplesse de positionnement
- Une moindre sensibilité aux conditions environnementales [15]

Inconvénients

- Le coût Les prix restent nettement supérieurs à ceux des étiquettes code à barres pour des unités consommateurs.
- La perturbation par l'environnement physique
- Les perturbations induites par les étiquettes entre elles
- La sensibilité aux ondes électromagnétiques parasites
- Les interrogations sur l'impact de la radio fréquence sur la santé [15]

III.11 Boite de raccordement réseaux :

La communication peut donc se faire soit par port série, soit par un port Ethernet permettant la mise en réseau du boîtier RFID. La connexion par RS232 est assez rapide, après un simple branchement, il suffit de configurer le port série de la manière suivante : 115200 bits par seconde, 8 bits de données, sans parité mais avec un bit de stop. Nous ne nous étendons pas sur la connexion série car nous avons décidé d'utiliser la connexion TCP/IP. Pour cela, il suffit de brancher le lecteur sur un PC ou sur un routeur pour pouvoir accéder au contrôle du boîtier.



Figure 3.10 : boîtier ositrack

III.12 Câblage d'un réseau Modbus Ethernet TCP/IP :

III.12.1 Schéma du montage :

Exemple de montage d'un réseau Modbus Ethernet TCP/IP avec la boîte de raccordement XGS Z33ETH :

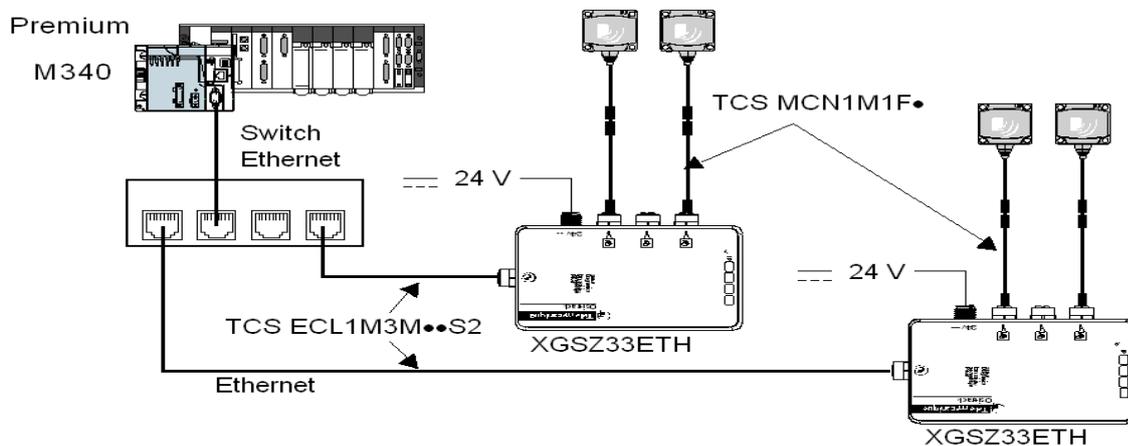


Figure 3.11 : Schéma de raccordement réseau

➤ Présentation du Modbus :

Le protocole Modbus est un protocole Maître-Esclave qui permet à un seul et unique maître de demander des réponses à des esclaves ou d'agir en fonction de la requête. Le maître peut s'adresser aux esclaves individuellement ou envoyer un message de diffusion générale à tous les esclaves. Les esclaves renvoient un message (réponse) aux requêtes qui leur sont adressées individuellement. Les requêtes de diffusion générale du maître n'attendent pas de réponses en retour. Jusqu'à 32 nœuds peuvent résider sur un réseau RS232 (1 maître et jusqu'à 31 esclaves). [16]

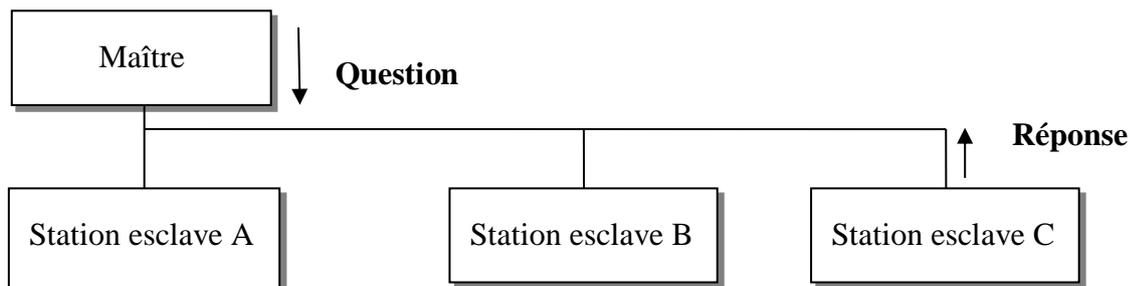


Figure 3.12 : présentation du Modbus

III.13 Configuration du transpondeur RFID :

Pour que la maquette soit prête d'utilisation, il ne reste plus qu'à configurer le transpondeur RFID. En effet, celui-ci communiquera avec l'automate grâce à un réseau Ethernet, il est donc indispensable de donner sa propre adresse au transpondeur même si dans notre application il est le seul membre de ce réseau. Avec le système Ositrack est fournie une carte de configuration : [17]

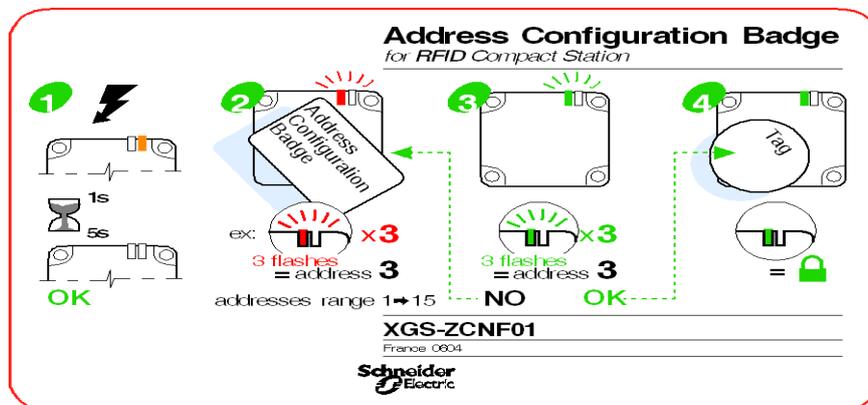


Figure3.13 : badge de configuration du RFID

III.13.1 Démarche à suivre :

- alimentez le système Ositrack et attendez 5 secondes.
- Approchez la carte de configuration du transpondeur, le nombre de clignotement **ROUGE** que son voyant 'tag' fera sera l'adresse que vous lui affecterez. Retirez la carte de configuration.
- Le voyant 'tag' clignote en **VERT** le nombre de fois correspondant à la valeur de l'adresse affectée.
- Approchez un tag quelconque du transpondeur pour valider l'opération, le voyant 'tag' doit s'allumer en **VERT**.

Les numéros d'adresse sont compris entre 1 et 15.

Le voyant 'tag' du transpondeur clignotera régulièrement en vert le nombre de fois correspondant à la valeur de l'adresse affectée. [17]

III.13.2 La programmation de lecture /écriture sous Unity Pro :

La lecture/écriture de RFID se fait en programmation sous Unity Pro et cela en suivant ces instructions :

☞ Créer une section :

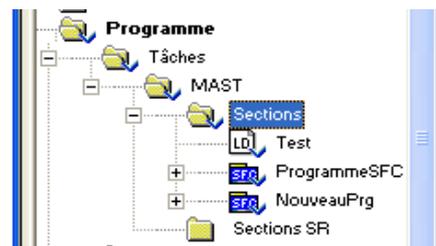


Figure 3.14 : création d'une section sous UNITY-PRO

Utiliser des blocs fonctions :

Nous devons écrire des données sur les cartes RFID. Pour réaliser ceci nous avons le choix entre des blocs WRITE/READ VAR et des blocs REQUEST. Dans un souci de facilité d'utilisation nous avons choisi les blocs READ_VAR et WRITE_VAR :

READ_VAR :

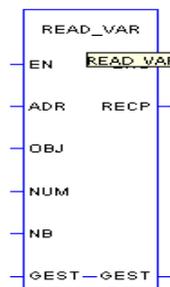


Figure 3.15 : Bloc fonction READ-VAR

La fonction « READ_VAR » est utilisée pour envoyer une requête de lecture d'objets (mots ou bits) vers un module de communication à distance. Description du bloc :

- L'adresse du module à distance est : « ADDR('0.0.0.3') ». Elle se décompose comme Pour le bloc INPUT_CHAR : le premier terme (0) pour le numéro du rack, le deuxième terme (0) pour le numéro du module, le troisième terme (0) pour le numéro de la voie et le dernier (3) correspond à l'adresse de l'objet à distance. L'adresse de l'objet, dans notre cas, est l'adresse du transpondeur RFID de lecture/écriture configurable. On configure l'adresse 3 par la méthode décrite précédemment dans la partie « Réalisation de la maquette ».
- L'entrée OBJ configure le type des variables à lire et donc le format des valeurs reçues. Pour notre application on lit des mots donc OBJ='%MW'.
- L'entrée NUM correspond au numéro du premier objet lu. NUM=0 pour la lecture des temps de traitement. NUM=32768 pour savoir si une étiquette est présente devant le transpondeur RFID.
- NB permet d'informer du nombre d'objets à lire à partir de NUM. NB=2 pour la lecture des deux mots de 16 bits contenant les poids faibles et forts du temps de traitement (UDINT). NB=1 pour la lecture du mot %MW32768.
- GEST est un tableau de gestion : même utilité que pour le bloc INPUT_CHAR.

- En sortie, RECP est un tableau de NB objets de type OBJ. [17]

WRITE_VAR :

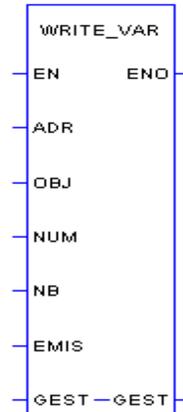


Figure 3.16 : Bloc fonction WRITE-VAR

La fonction « WRITE_VAR » est utilisée pour envoyer une requête d'écriture d'objets (mots ou bits) vers un module de communication à distance. Description du bloc :

- L'adresse du module à distance est : « ADDR ('0.0.0.3') ». Elle se décompose de la même manière que pour la lecture de données à distance.
- L'entrée OBJ configure le type des variables à écrire et donc leur format. Pour notre application on écrit des mots donc OBJ='%MW'.
- L'entrée NUM correspond au numéro du premier objet écrit. NUM=0 car on écrit à partir du premier mot de la zone mémoire de la carte.
- NB permet d'informer du nombre d'objets de type OBJ à écrire à partir de NUM. NB=2 pour écrire les 16 bit de poids faible puis les 16 bit de poids forts de l'heure courante (plus précisément, au moment de l'arrivée du produit sur le tapis, la durée écoulé en secondes depuis 1980).
- En entrée, EMIS est un tableau contenant les données à écrire. Il doit être créé en concordance avec les valeurs du nombre NB et du type OBJ.
- GEST est un tableau de gestion : même utilité que pour le bloc précédent. [17]

III.13.3 La zone mémoire :

Pour accéder aux données d'une carte RFID. La zone mémoire adressage est découpée en deux zones : la zone mémoire étiquette et la zone mémoire station. Définition des zones d'adresses de mots utilisées : [17]

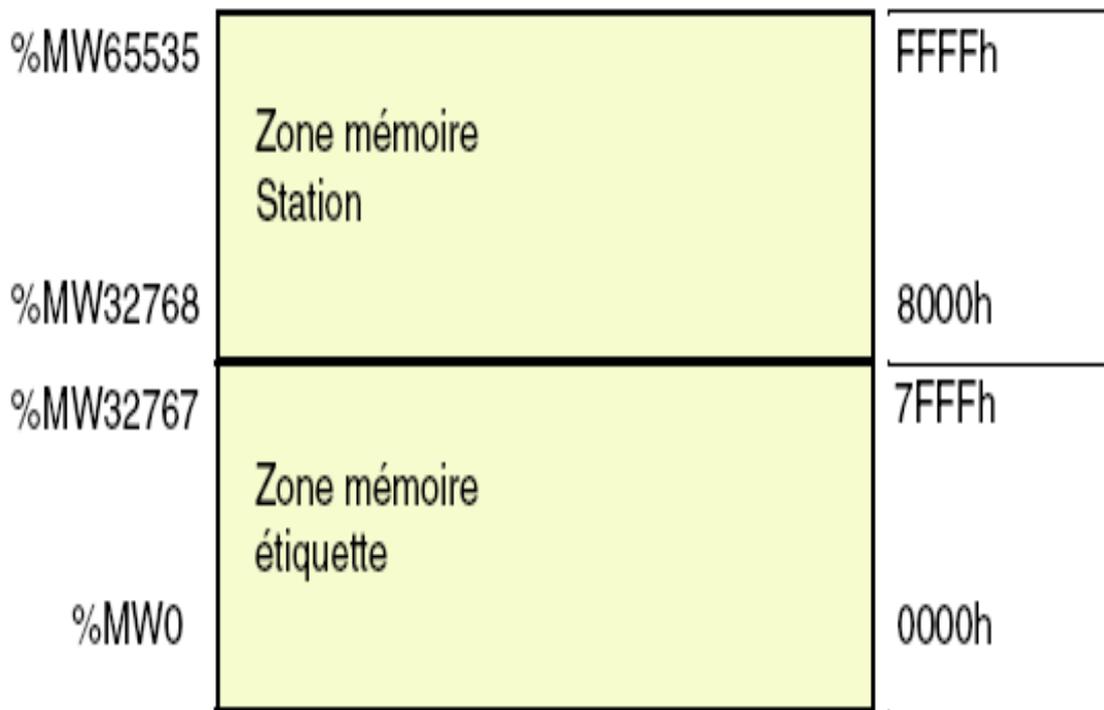


Figure 3.17 : Zone d'adresses des mots

Plus précisément chaque zone est définie ainsi :

1) La zone mémoire Station :

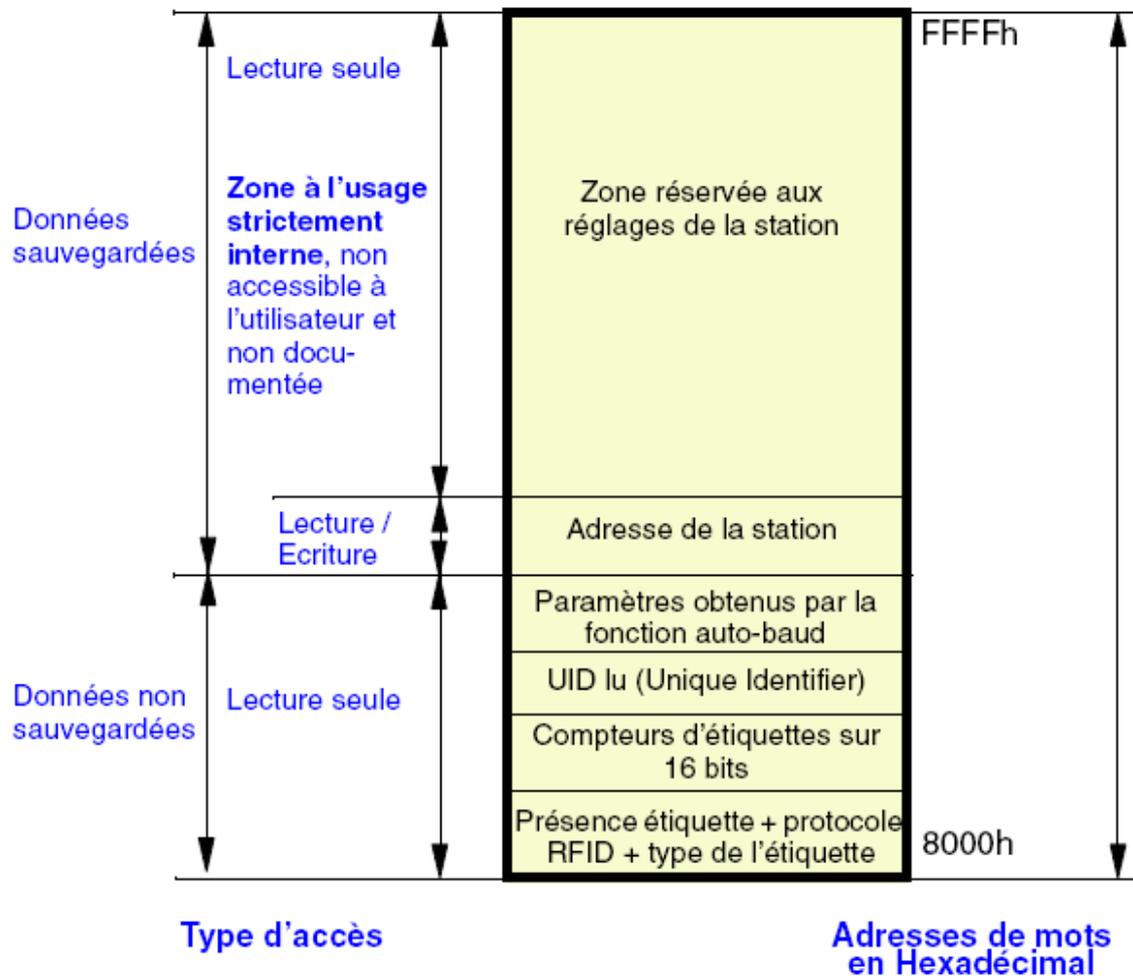


Figure 3.18 : La zone mémoire station

A l'aide d'un tableau descriptif, on en déduit que pour savoir si l'étiquette est présente devant le transpondeur RFID, nous devons lire le premier bit de la zone mémoire Station. Ce bit présence étiquette est donc le premier bit du mot %MW32768.

1) La zone mémoire Etiquette

Types d'étiquettes	XGH B90E340		XGH B221346		XGH B320345		XGH B211345		XGH B444345		XGH B445345	
	256 octets		256 octets		112 octets		256 octets		3408 octets		13632 octets	
Adresses	Dec.	Hex.	Dec.	Hex.	Dec.	Hex.	Dec.	Hex.	Dec.	Hex.	Dec.	Hex.
	0	0	0	0	0	0	0	0	0	0	0	0
	à	à	à	à	à	à	à	à	à	à	à	à
	127	7F	127	7F	55	37	127	7F	à	à	à	à
(1)		(1)		(1)		(1)		1703 6A7		à à		
								(2)		6815 1A9F		
										(2)		

Zone mémoire de l'étiquette

Tableau2 :les plages mémoire des étiquettes

Ce tableau nous montre que chaque type d'étiquette possède une plage de mémoire différente. Ces plages allant de 112 octets à 13632 octets sont largement suffisantes dans notre application. Nous devons donc écrire ou lire les données de la carte à partir du mot %MW0.

Nous devons écrire des données sur les cartes RFID. Pour réaliser ceci nous avons le choix entre des blocs WRITE/READ VAR et des blocs REQUEST. Dans un souci de facilité d'utilisation nous choisissons les blocs READ_VAR et WRITE_VAR. [17]

III.14 Avantages :

- Pour une communication de données rapide, flexible, sans contact physique et sans contact visuel.
- Apte au service en milieu industriel, dans tout environnement.
- Performant et économique.

III.15 Conclusion :

La RFID comme toute nouvelle invention a soulevé un énorme enthousiasme, la nature humaine est ainsi faite toujours prête à s'émerveiller dans une nouvelle naissance.

L'identification par radiofréquence est aujourd'hui utilisée dans de nombreux domaines, notre projet s'est intéressé à l'étude et l'intégration d'un dispositif d'identification basé sur la technologie RFID, donc nous avons essayé de lever le voile sur différents concepts théoriques touchant le domaine d'identification, précisément la technologie d'identification par radiofréquence (Radio Frequency Identification RFID).

**CHAPITRE IV : CONCEPTION
ET MISE EN PLACE DE LA
NOUVELLE CELLULE**

IV.1 INTRODUCTION :

Dans ce chapitre on présentera notre application ainsi que la mise en place de toute la cellule. Le câblage des différents éléments de la cellule sera présenté. On commencera par l'armoire puis du capteur et du convoyeur et enfin du robot.

Pour pouvoir lancer l'application c.-à-d., démarrer le convoyeur, détecter le produit et lire l'étiquette du produit, ensuite ordonner au robot de déplacer le produit, on utilisera l'outil de programmation UNITY-PRO qui est doté d'une plateforme de communication JAVA-OPC.

La nouvelle cellule est présentée sur la figure suivante :

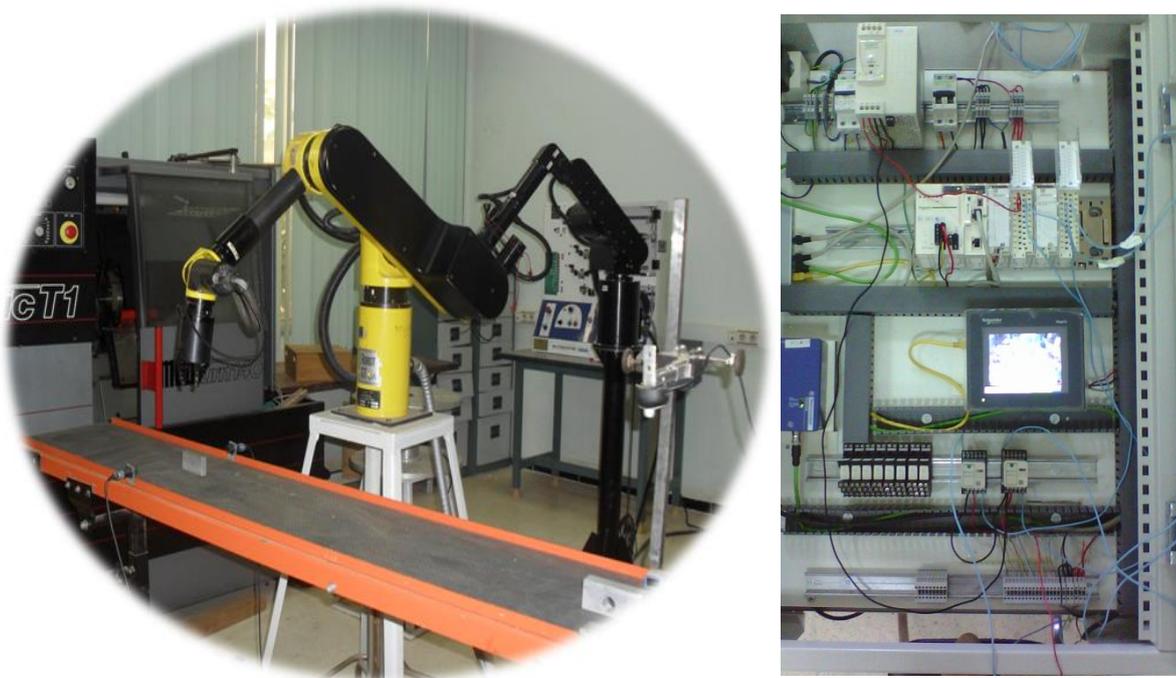


Figure 4.1 : la nouvelle cellule flexible

IV.2 Câblage des éléments :

En commencera notre application par le câblage de tous les éléments dans l'armoire et communiquer l'ordinateur avec l'automate ainsi que la boîte de raccordement réseau Ositrack avec un réseau local Ethernet TCP/IP (qu'on présentera dans ce qui suit).

➤ Ethernet :

Ethernet est une technologie universelle qui dominait déjà les réseaux locaux bien avant le développement de l'Internet. La clé de la longévité de cette technologie, c'est sa méthode

d'accès. Souvent critiquée, elle a toujours été plus facile à utiliser et à mettre en œuvre que ses concurrentes. Et aussi un réseau à diffusion développé à l'origine par les sociétés Xerox, Intel et Digital Equipment. L'IEEE a ensuite normalisé ce réseau par la norme IEEE 802.3, reprise ensuite par l'ISO sous la norme 8802-3. Les messages transmis par Ethernet sont appelés des trames.

➤ **TCP/IP :**

Le terme TCP/IP englobe un ensemble de protocoles qui assurent le mécanisme d'échange d'informations sur internet.

Le protocole IP : a été conçu pour s'adapter à des types de réseaux différents par un mécanisme de fragmentation de paquets (découpe le paquet de données en plus petit) et veille à ce que les messages soient correctement traités et acheminés.

Le protocole TCP : ce protocole est chargé de gérer la connexion entre un client et un serveur, en s'assurant que les données qui doivent être transmises seront divisées en plusieurs paquets et finissent par atteindre leurs destination mais s'ils ne le font pas ou si l'un d'eux est corrompu, TCP définit que ces données manquantes seront retransmises.

On notera que la combinaison TCP/IP est simplement un protocole de transport et ne définit pas le contenu des données ou bien comment les données doivent être interprété(c'est le travail du protocole MODBUS dans ce cas).

IV.3 Réalisation d'un système de pilotage automatisé :

IV.3.1 Description de l'application :

L'objectif de cette application est de commander un système de production par un automate et une interface de communication tout en introduisant un système d'identification qu'on appelle la technologie RFID.

Dans cette application on aura besoin d'une lecture/écriture d'information d'un produit dans l'industrie, et pour le faire on utilisera un convoyeur qui fonctionnera au démarrage d'un moteur à courant continu, lorsque le produit est détecté par le capteur qui causera l'arrêt du convoyeur ensuite le lecteur lira le produit à partir de l'étiquette, à la fin de cette tâche le robot interviendra pour le stockage du produit.

IV.3.2 Cahier de charge :

Le fonctionnement de notre système est comme suit :

- Initialisation du système.
- Mise en marche du système après exécution du programme.
- Détection du produit par le capteur.
 - Arrêt du convoyeur.
 - Lecture du produit par la RFID.

- L'envoi du message de JAVA vers OPC.
- L'envoi du message d'OPC vers l'automate M340.
- L'envoi du message de l'automate vers OPC.
- OPC vers JAVA
- D'OPC vers le PC du robot.
- Exécution de la tâche du robot.
- Temporisation.
- Fin de la tâche du robot
- Redémarrage du convoyeur pour d'autres cycles de travail.

IV.3.3 Diagramme de fonctionnement du système :

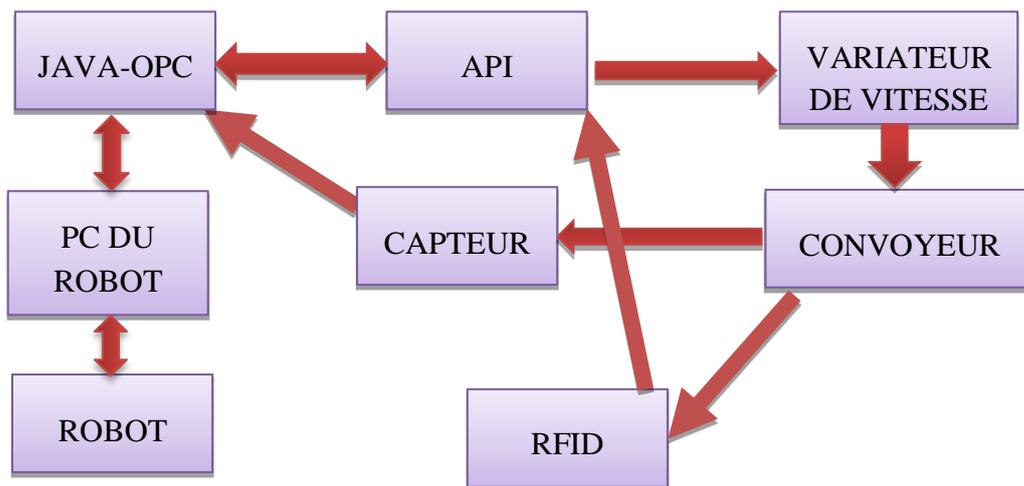


Figure 4.2 : schéma de fonctionnement du système

IV.3.4 Câblage de l'armoire électrique :

Pour notre application on a pu câbler tous les éléments de l'armoire électrique, tout en commençant par le capteur sur le module des entrées ensuite on a alimenté le module des sorties (output Relay) par une tension de 24V via l'alimentation de l'armoire puis à partir de ce module on a câblé deux sorties vers les deux relais qui sont aussi reliés vers le variateur de vitesse et le transformateur qui délivre une tension de 24v pour faire un circuit fermer, et aussi on a câblé le module des sorties analogique vers le variateur de vitesse .

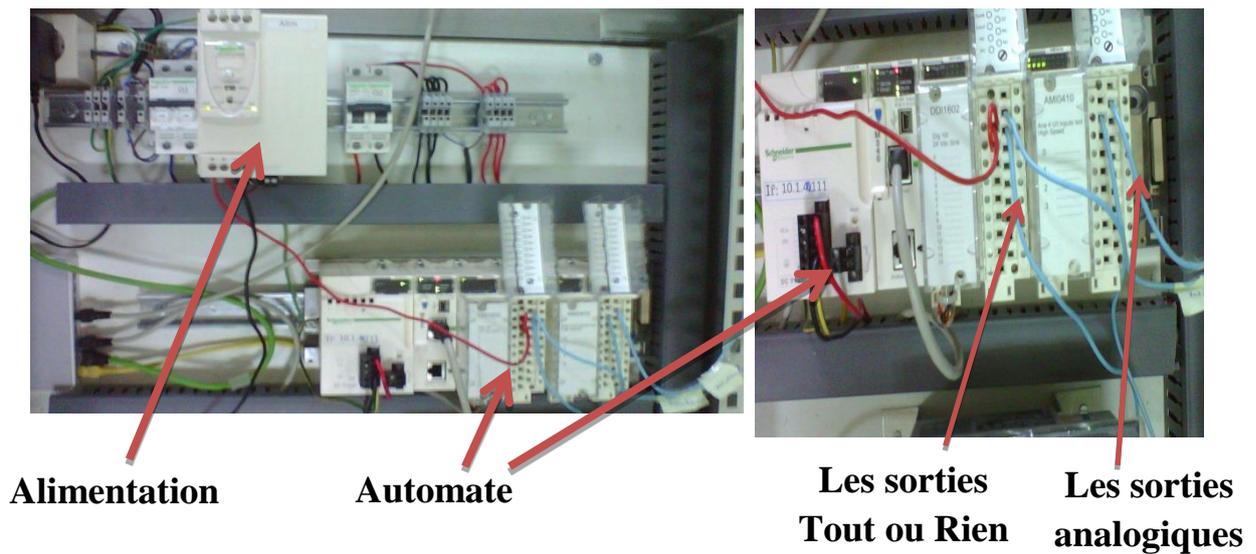


Figure 4.3 : câblage de l'automate

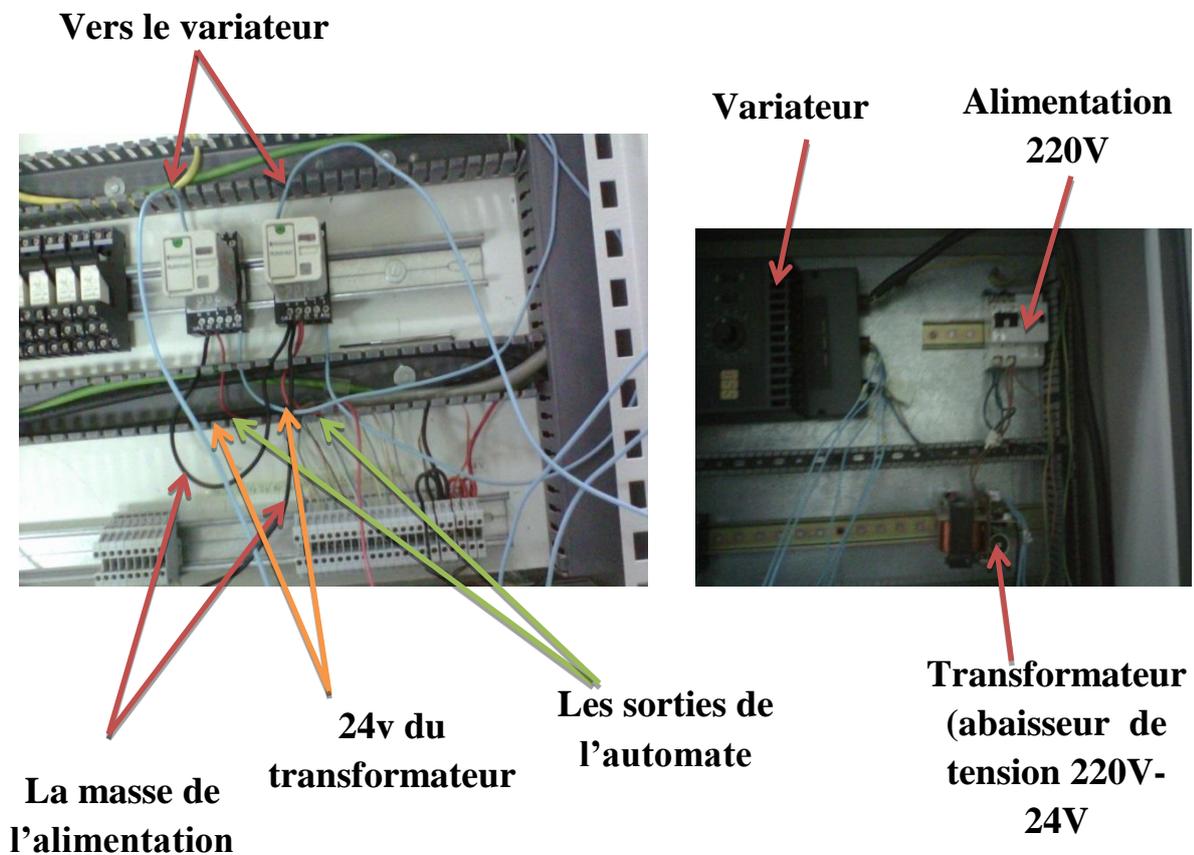


Figure 4.4 : câblage du variateur et du transformateur

IV.3.5 La configuration matérielle sous UNITY-PRO :

Avant de commencer la programmation, il est nécessaire de choisir une CPU, pour notre cas nous avons choisi la CPU BMX P34 2020 et un module d'entrée BMX DDI 1602, un module de sortie BMX DRA 0805 et un module de sortie analogique BMX AMO 0210 (voir la figure suivante) :

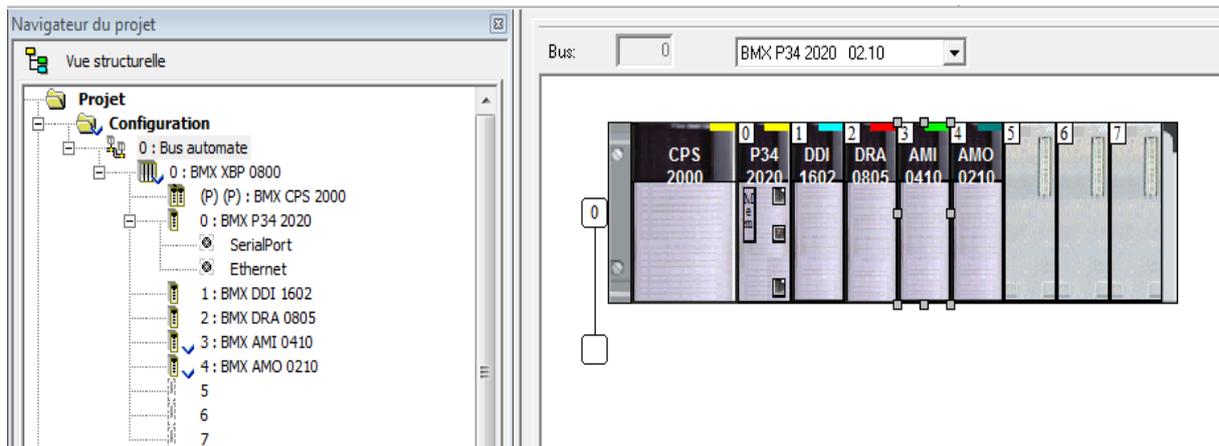


Figure 4 .5 : configuration matérielle

IV.3.6 Jeux d'instruction sous unity-pro :

Instruction de transfert de données :

Ces instructions offrent des fonctions de base permettant de transférer des données. L'instruction de transfert utiliser est MOVE, cette dernière est utilisée dans le but de copier les données d'une position à une autre, elle détient deux paramètres d'entrées et deux de sorties, lorsque l'instruction reçoit le flux validant elle copie les données du paramétré d'entrée IN vers le paramétré de sortie OUT sous forme d'entier par exemple transférer la valeur 30000 vers la mémoire %QW0.3.0

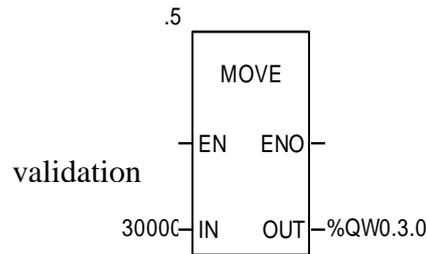


Figure 4.6 : bloc pour le variateur de vitesse

Instruction de temporisation :

Le temporisateur utilisé lors de la réalisation du programme est un temporisateur TON (délai à l'activation) qui s'incrémente pendant qu'il reçoit le flux validant et conserve sa valeur lorsque le flux cesse. Ce bloc fonction est utilisé pour le retard de mise en marche et l'état initial de ET lors du premier appel du bloc fonction doit être à "0".

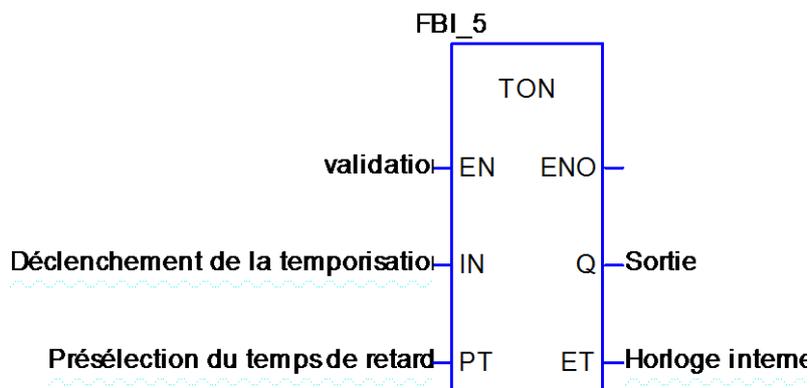


Figure 4.7 : Bloc de temporisation

Instruction de conversion d'adresse :

La fonction ADDM permet de convertir une chaîne de caractères en une adresse pouvant être utilisée directement par les fonctions de communication suivantes : READ_VAR, WRITE_VAR.

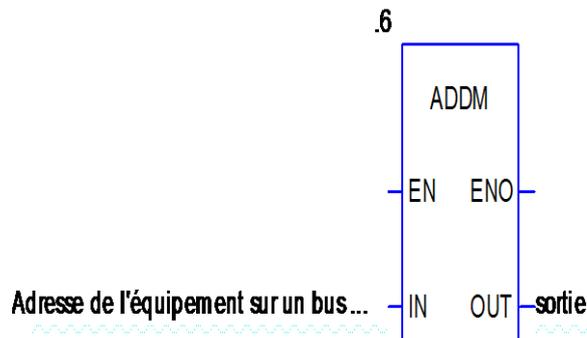


Figure 4.8 : Bloc de construction d'adresse

IV.3.7 Editeur de données :

La déclaration de toutes les variable utilisés :

Nom	Type	Adresse	Valeur	Commentaire
5S	TIME			
BP	BOOL			
BP_1	BOOL			
BT	BYTE			
cap1_c...	EBOOL	%I0.1.14		capteur1
cap2_c...	EBOOL	%I0.1.15		capteur2
capt...	BOOL			
capt...	BOOL			
Gestion	ARRAY[1...	%MW1		
init	EBOOL	%I0.1.3		initialisation
Initi	BOOL			
Recepti...	ARRAY[1...			
VARI	EBOOL	%Q0.2.1		Moteur
VARI2	EBOOL	%Q0.2.2		
tem...	BOOL			
ecrir	ARRAY[1...			
VTS	INT	%QW0.4.0		
T	BOOL			

Figure 4.9 : Editeur de données

IV.4 Phase d'implémentation :

IV.4.1 programme sous unity-pro:

IV.4.1.1 Programme automate :

Afin de pouvoir implémenter notre application sur l'automate SCHNEIDER M-340, nous procédons en premier à la modélisation du fonctionnement de la cellule par le langage LADDER en fin l'outil du GRAFCET.

a) Langage LADDER :

Ladder Diagram (LD) ou **Langage Ladder** ou **schéma à contacts** est un langage graphique très populaire auprès des automaticiens pour programmer les Automates Programmables Industriels. Il ressemble un peu aux schémas électriques, et est facilement compréhensible.

➤ Pour la configuration de l'étiquette :

Avant de placer l'étiquette sur le produit, il faut d'abord écrire dessus c'est-à-dire transmettre les informations du produit dans la puce en utilisant le lecteur.

➤ Table d'animation :

Après l'exécution du programme, on a pu récupérer les données écrites dans l'étiquette

Nom	Valeur	Type	Commentaire
ecrire		ARRAY[1..5] OF...	
ecrire[1]	10	INT	
ecrire[2]	20	INT	
ecrire[3]	30	INT	
ecrire[4]	40	INT	
ecrire[5]	50	INT	
Reception		ARRAY[1..5] OF...	transmission
Reception[1]	10	INT	transmission[1]
Reception[2]	20	INT	transmission[2]
Reception[3]	30	INT	transmission[3]
Reception[4]	40	INT	transmission[4]
Reception[5]	50	INT	transmission[5]

Figure 4.10 : Table d'animation

Modélisation de l'application par l'outil GRAFCET :

1) Définition :

Le GRAFCET est un modèle de représentation graphique des comportements dynamiques de la partie commande.

Le GRAFCET décrit les interactions entre la partie commande et la partie opérative. Il établit une relation entre :

- les **entrées**, correspondant aux transferts d'informations de la partie opérative vers la partie commande ;
- les **sorties**, correspondant aux ordres transmis de la partie commande vers la partie opérative. [18]

Le GRAFCET est défini par :

Un ensemble d'éléments graphiques de base :

- les **étapes**,
- les **transitions**,

- les **liaisons orientées** reliant les étapes aux transitions et les Transitions aux étapes ;

Une interprétation, traduisant le comportement de la partie commande vis-à-vis de ses entrées et de ses sorties, caractérisée par :

- les **actions** associées aux étapes,
- les **réceptivités** associées aux transitions ;

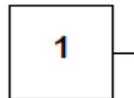
Des **règles d'évolution** définissant formellement le comportement dynamique de la partie commande ainsi décrite. [18]

☞ **Étape**

Une étape caractérise un comportement invariant d'une partie ou de la totalité de la partie commande à un instant donné ; suivant l'évolution du système :

- une étape est soit **active**, soit **inactive** ;
- l'ensemble des étapes actives définit la situation de la partie commande.

Une ou plusieurs **actions** élémentaires ou complexes peuvent être associées à une étape afin de traduire « ce qui doit être fait » chaque fois que cette étape est active. [18]



☞ **Transition :**

Une transition indique la possibilité d'évolution entre étapes. Chaque transition représente une, et une seule, possibilité d'évolution. Une transition est dite **validée** lorsque toutes les étapes immédiatement précédentes reliées à cette transition sont actives. [18]



☞ **Réceptivité :**

À chaque transition est associée une proposition logique appelée réceptivité qui regroupe, parmi toutes les informations disponibles, uniquement celles qui sont susceptibles, à un instant donné, de faire évoluer la situation de la partie commande. [18]



☞ **Liaison :**

Les liaisons orientées relient les étapes aux transitions et les transitions aux étapes : elles indiquent les voies d'évolution du GRAFCET. [18]



☞ **Action :**

L'action est associée à une étape, elle est active lorsque le cycle est arrivé sur l'étape. Il est possible de définir les actions conditionnelles, temporisé . . . (électrovanne, enclenchement d'un contacteur. . .) [18]



☞ **Etape active :**

Le point indique que l'étape est **active**. [18]



Règles du Grafcet :

1) Situation initiale :

- Un grafcet commence par une étape initiale qui représente la situation initiale avant évolution du cycle

2) Franchissement d'une transition

- Une transition est soit validée ou non validée ; elle est valide lorsque toutes les étapes immédiatement précédentes sont actives.

Lorsque la transition est valide et que la réceptivité associée est vraie elle est alors obligatoirement franchie.

3) Évolution des étapes actives

- Le franchissement d'une transition entraîne l'activation des étapes immédiatement suivante et la désactivation des étapes immédiatement précédentes.

4) Transitions simultanées

- Plusieurs transitions simultanément franchissables sont simultanément franchies.

5) Activation et désactivation simultanées

- Si au cours du fonctionnement, une même étape doit être désactivée et activée simultanément, elle reste active.

La durée de franchissement d'une transition ne peut jamais être rigoureusement nulle, même si elle peut être rendue aussi petite que l'on veut. Il en est de même pour la durée d'activation d'une étape. [18]

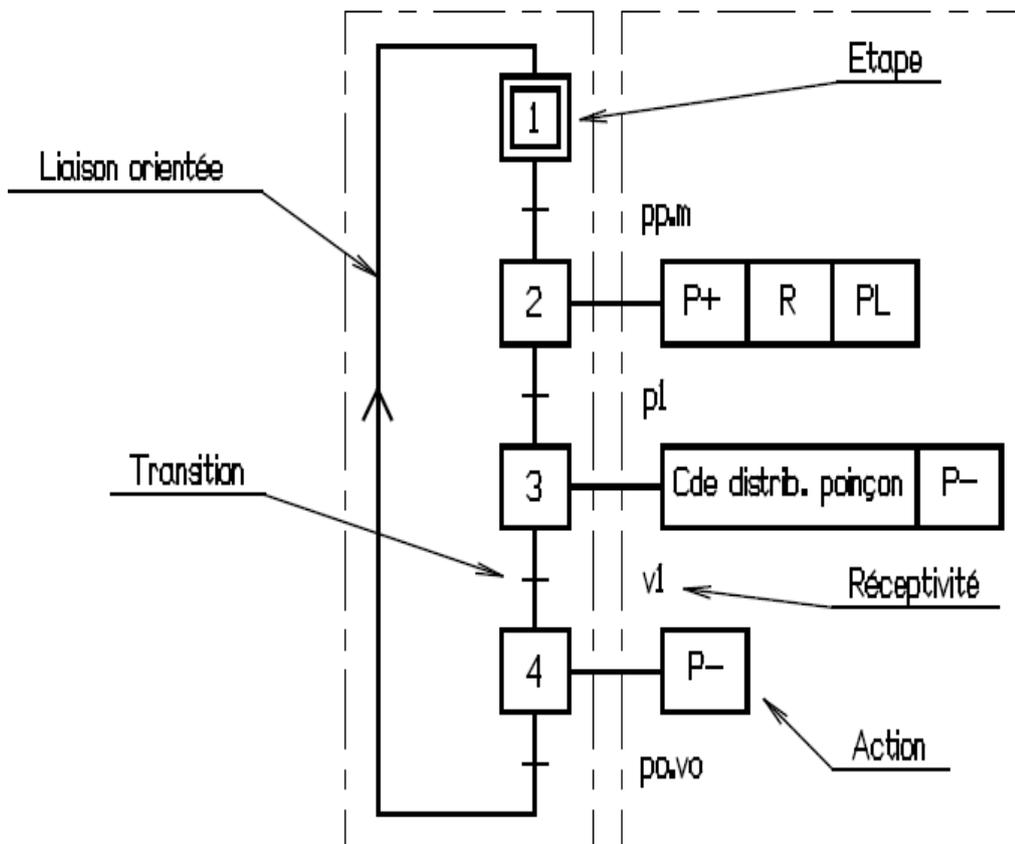
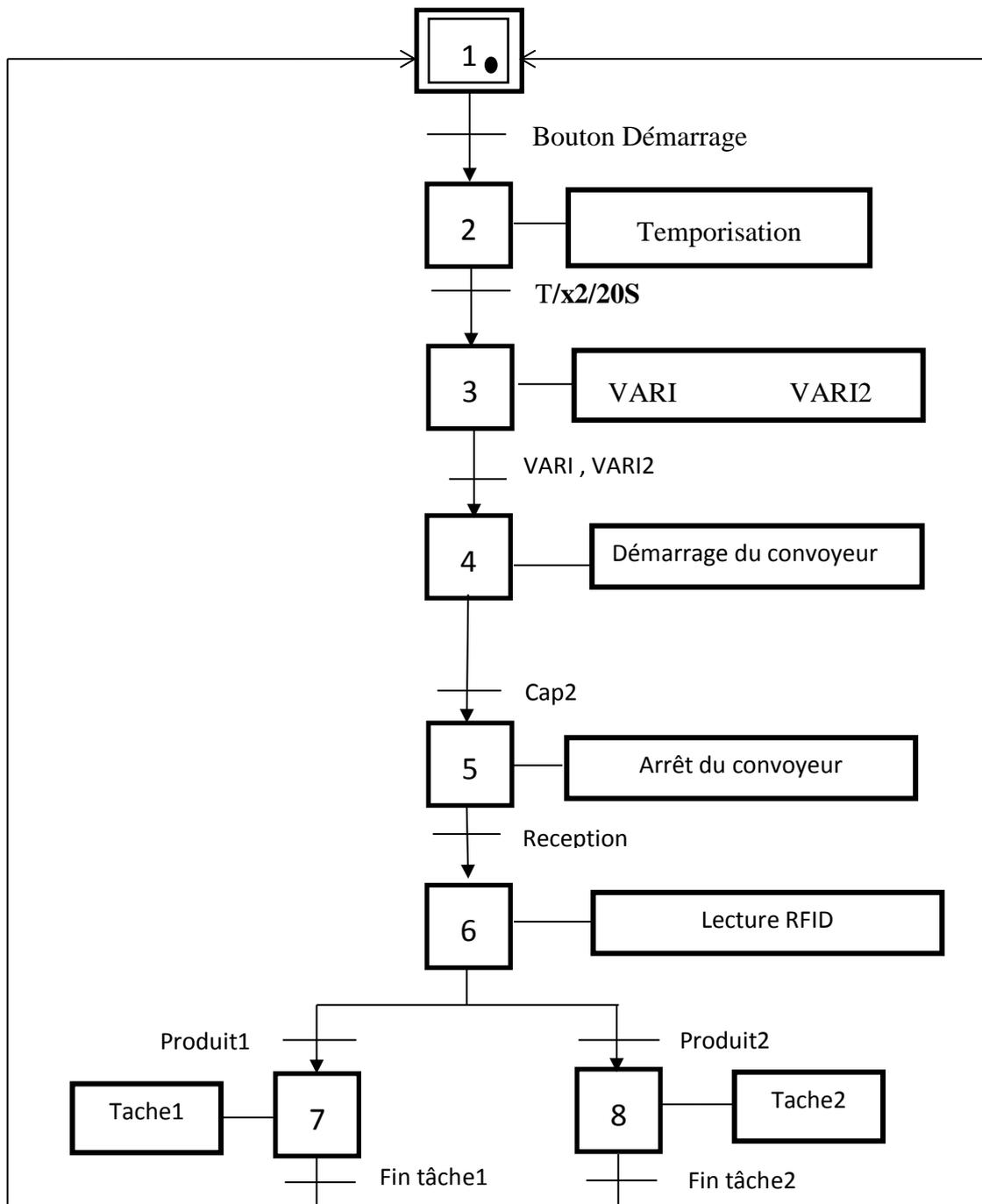


Figure 4.11 : illustration du GRAFCET

b) Le GRAFCET de notre application :



Pour pouvoir réaliser une communication entre l'automate SCHNEIDER M-340, le capteur infrarouge, convoyeur, java et le PC du robot on utilise le protocole de communication OPC.

Pour le faire on va utiliser un logiciel qu'on appelle KEPServerEX V5.0, dans le quelle on configurera toute les variable qu'on utilisera avec les adresse qui leurs convienne.

IV.4.2 Logicielle KEPServer :

Ce serveur basé sur le logiciel est conçu pour des communications fiables, installation rapide et l'interopérabilité inégalée entre les applications clientes, les dispositifs et les systèmes industriels. Le serveur fournit un large éventail de conducteurs et de composants périphériques plug-in (brancher) qui conviennent à la plupart des besoins de communication. La conception de plug-in et une interface utilisateur unique permet un accès cohérent à partir des applications basées sur les standards (tels que OPC) et les non-normes avec des applications basées sur les interfaces natives.

Serveur kepware :

Le serveur KEPServerEx V5.0 est un serveur OPC Data Access qui a pour fonction de :

- Collecter les données issues des périphériques matériels (le capteur infrarouge, le bras robotisé...) ou leurs faire parvenir les mises à jour de données.
- Assurer l'intégrité des données (qualité et fraîcheur).
- Répondre aux requêtes de clients (fournir les données, écrire vers le matériel...).
- Avertir le client des changements d'état des variables par un événement.

Paramétrage du serveur KEPServerEx V5.0 :

Les étapes à suivre pour le paramétrage du serveur KEPware sont comme suit :

☞ Sélection du port :

Dans cette étape nous choisissons le port à travers lequel le serveur va communiquer avec les périphériques.

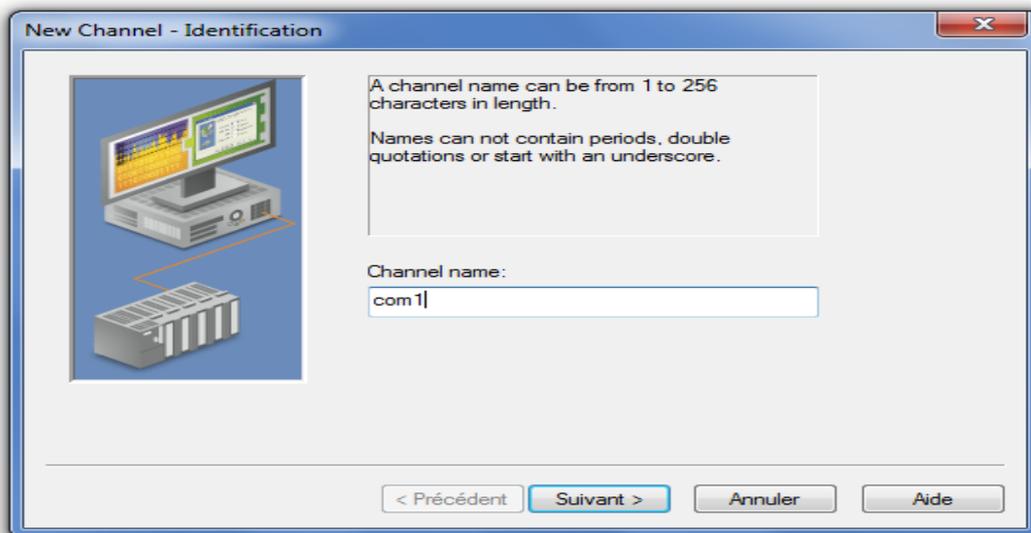


Figure 4.12 : configuration du port

☞ **Choix de l'automate :**

Cette figure illustre la sélection de l'automate dont le serveur va nous procurer les données en temps réel.

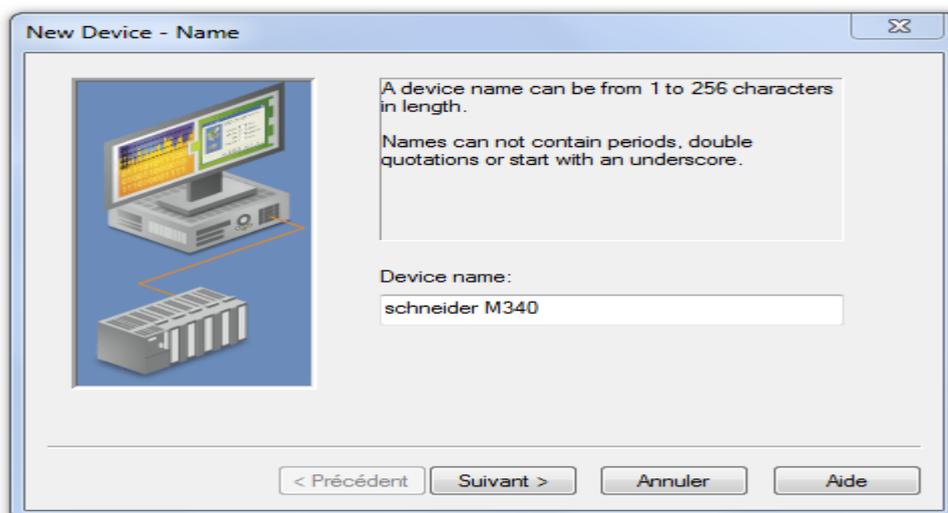


Figure 4.13 : choix de l'automate

☞ **Création d'un groupe :**

Cette figure illustre comment créer un groupe sur le serveur KEPware.

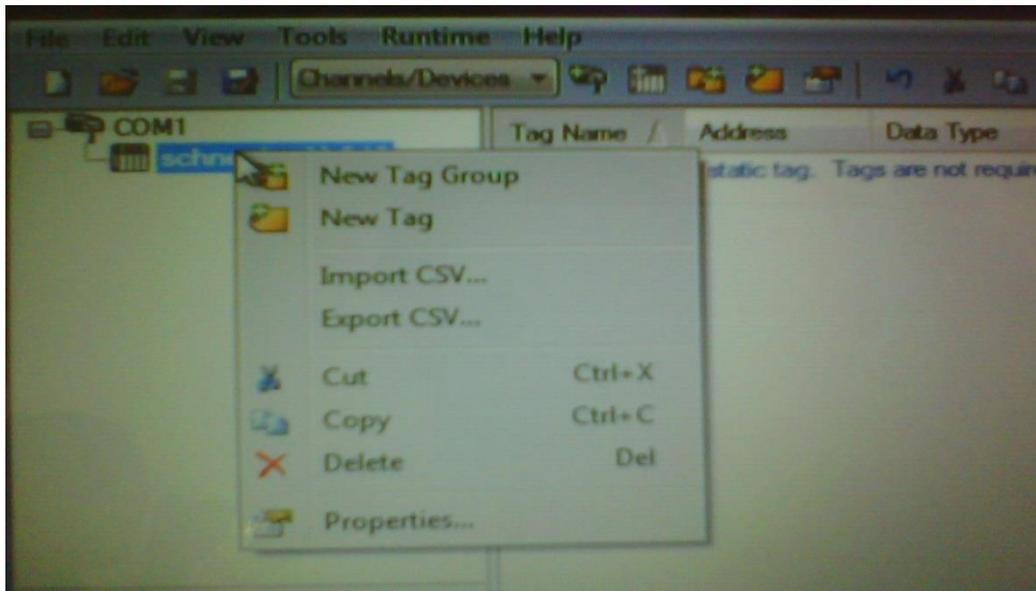


Figure 4.14 : création d'un group

☞ **Création d'un Item :** la figure suivante illustre comment créer un item. Une fois tout le paramétrage du serveur KEPware est effectué, on obtient la figure suivante

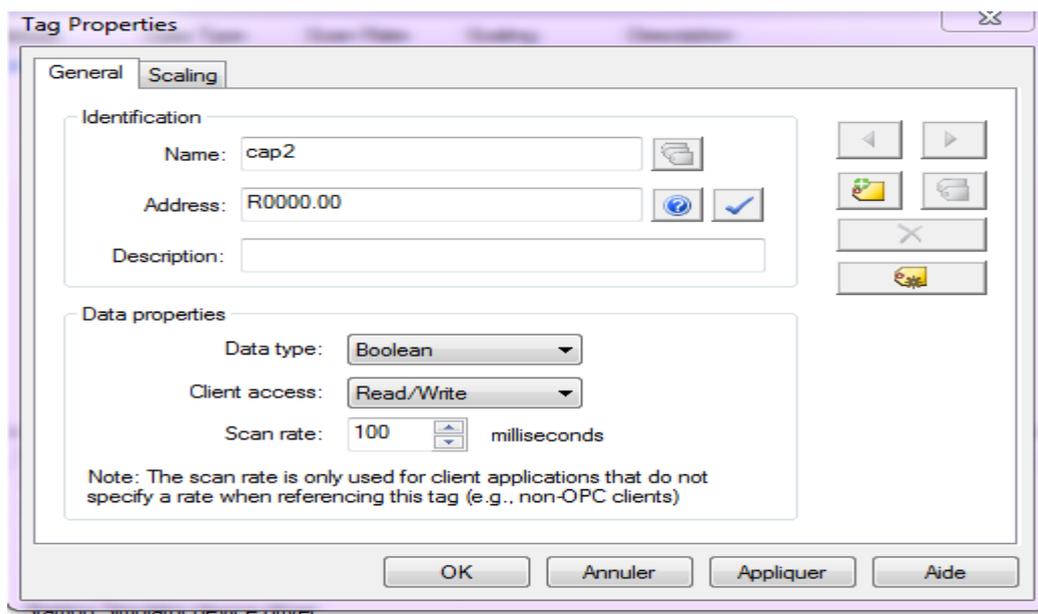
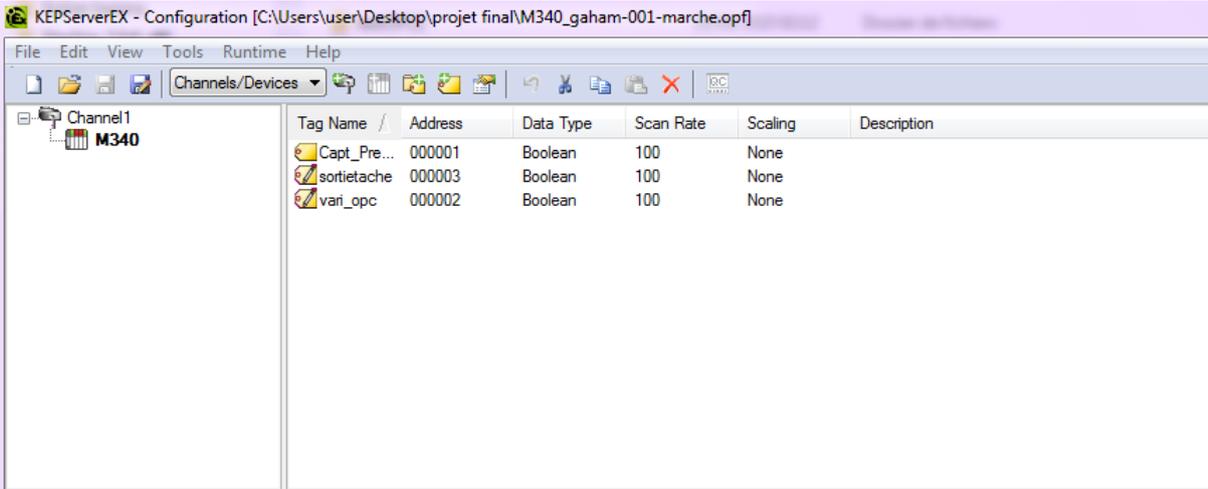


Figure 4.15 : création d'Item



Tag Name	Address	Data Type	Scan Rate	Scaling	Description
Capt_Pre...	000001	Boolean	100	None	
sortietache	000003	Boolean	100	None	
vari_opc	000002	Boolean	100	None	

Figure 4.16 : Table des Items

IV.4.3 Programmation des tâches du robot :

Afin d'introduire le robot dans notre application, il faut d'abord lui spécifier les tâches qu'il doit exécuter. Pour ce faire, on utilisera un poste de pilotage avec une centrale de commande utilisant le MS-DOS comme système d'exploitation. On utilisera un programme qui va permettre l'apprentissage, l'enregistrement et l'exécution des tâches

IV.4.3.1 Les différentes tâches sous MS-DOS :

MS-DOS (abréviation de **M**icrosoft **D**isk **O**perating **S**ystem) est le système d'exploitation de type DOS développé par Microsoft pour l'IBM PC d'abord, puis pour les compatibles PC. Il s'agit d'un système fonctionnant en mode réel, monotâche et mono-utilisateur, et équipé par défaut d'une interface en ligne de commande. Des années 1980 au début des années 1990, il a été le système le plus utilisé sur compatible PC, avant d'être progressivement remplacé par des systèmes d'exploitation plus évolués, notamment Windows. Son développement est maintenant arrêté.

Les étapes à suivre pour la programmation des tâches du robot :

1^{ère} étape : l'allumage de la centrale de commande et l'ordinateur.



Figure 4.17 : station du robot

2^{ème} étape : initialisation du robot.

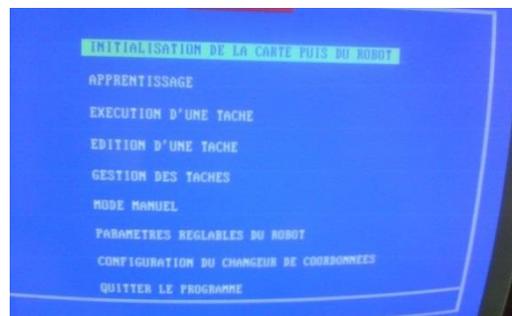


Figure 4.18 : Initialisation du robot

3^{ème} étape : Apprentissage :

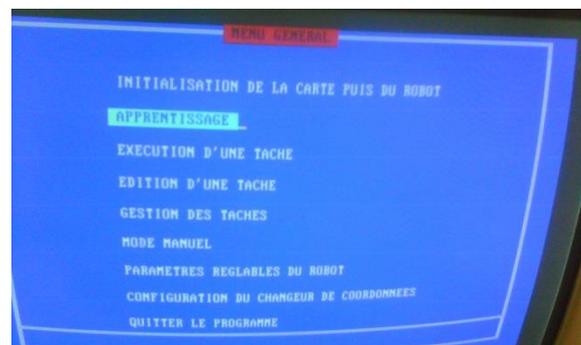


Figure 4.19 : Sélection de l'étape apprentissage

4^{ème} étape : création d'une tâche : dans cette étape on va créer des points qui nous permettront de valider les actions et les mouvements que le robot doit suivre.

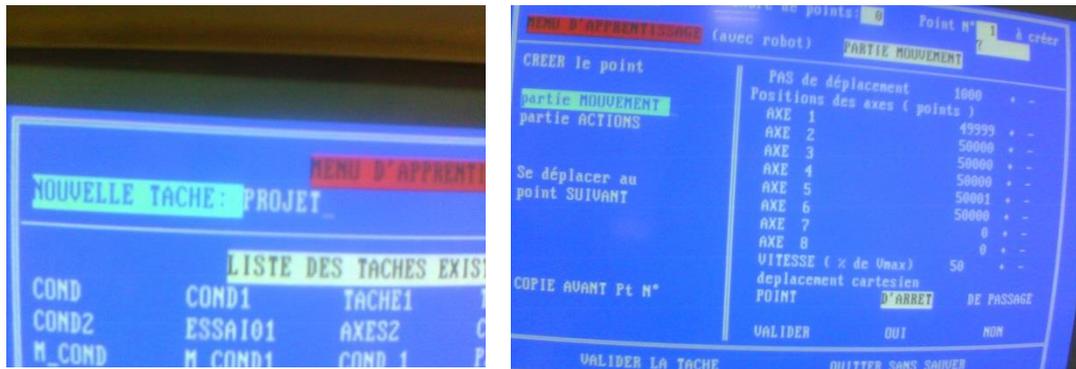


Figure 4.20 : création des tâches

5^{ème} étape : validation des tâches créées.

6^{ème} étape : exécution des tâches.



Figure 4.21 : exécution d'une tâche

Conclusion :

Dans ce chapitre on a présenté notre application, d'abord on a programmé sous unity- pro ainsi que sur JAVA-OPC et aussi on a pu programmer les tâches du robot. Enfin on a abouti au résultat attendu.

Conclusion générale

Au préalable, nous tenons à souligner l'expérience très enrichissante acquise au cours du stage au sein du centre de Développement des Technologies Avancées, pour l'élaboration de ce projet de fin d'étude. Le travail qui nous a été demandé consiste à réaliser une commande par automate Schneider M-340 et IHM des systèmes de production tout en introduisant la technologie RFID.

Nous avons réalisé une commande pour un système de production qui consiste à se connecter à l'automate via le réseau Ethernet TCP/IP afin de lire les entrées /sorties qui sont en relation avec le produit capté par le lecteur RFID ainsi que la modification des données du produit. Pour le démarrage du système et la récupération des données du tag RFID on a utilisé un programme ladder sous le logiciel Unity-Pro. Pour faire introduire le robot dans la cellule flexible on a utilisé une Interface Homme Machine JAVA-OPC qui a servi à réaliser la communication entre le robot et l'automate. On a utilisé JAVA pour la programmation des tâches et OPC pour récupérer l'état des entrées/sorties de l'automate et les données du produit à partir du lecteur RFID.

Pour les tâches du robot on a utilisé une plateforme de commande utilisant un PC doté d'un système d'exploitation MS-DOS, et la communication entre les deux PC (le PC où on a programmé JAVA-OPC et celui du Robot) se fait avec le port RS 232.

Au cours de ce projet nous avons acquis beaucoup de connaissance que ce soit sur le plan matériel (hardware) ou sur le plan programmation (software). En premier lieu nous avons eu la chance d'interagir directement avec des équipements industriels c.-à-d. le câblage et de comprendre le rôle de chaque élément de notre cellule et même de les manipuler par leurs logiciels de base.

En deuxième lieu nous avons appris le langage JAVA suivi du protocole OPC et aussi les principes de base de l'Ethernet industriel.

Les objectifs attendus de ce projet ont été atteints puisque nous avons réussi à implémenter le système avec les spécifications attendues par le cahier des charges.

Comme perspectives à ce travail, on peut améliorer la lecture/écriture des données de la RFID avec des fonctions sous le logiciel NODE-JS avec JavaScript comme langage de programmation ainsi que l'utilisation de l'interface graphique pour la communication HTML. D'un autre côté, on peut envisager de changer le robot fixe par un robot mobile.



ANNEXE

ANNEXE 1 :

Caractéristiques de l'automate GE FANUC séries 90-30 :

L'automate GE FANUC séries 90-30 offre les caractéristiques suivantes :

- Gamme diversifiés de CPU.
- Gamme complète de module :
 - Entrées/sorties TOR.
 - Entrées/sorties Analogiques.
 - Emplacements max.10 modules.
- Possibilité de mise en réseau avec modbus-RTU ou Genius.
- Liberté de montage aux différents emplacements.
- Configuration et paramétrage à l'aide de l'outil configuration matérielle.

Constitution de l'API GE FANUC 90-30 (IC693CPU323) :

a) L'unité centrale CPU 323 :

L'unité centrale est constituée d'un module d'alimentation, d'un processeur ainsi que d'une mémoire, nous analyserons successivement chacun des ces composants. [5]

b) Alimentation

Intégré à la CPU, il assure la distribution d'énergie électrique aux différents modules. [5]

c) Processeur

Le processeur de l'API IC693CPU323 assure la réalisation de toutes les fonctions logiques, arithmétiques et de traitement numérique (transfert, comptage, temporisation...), il s'occupe de l'exécution du programme chargé dans l'automate (à partir du logiciel VERSA PRO), de la première instruction jusqu'à la dernière et cela se fait en boucle.

Le processeur est connecté aux autres éléments (mémoires et interface E/S) grâce au bus interne qui véhicule les informations sous forme binaire. [5]

d) Mémoire

La zone mémoire va permettre de :

- ✓ Recevoir les données issues des capteurs infrarouge et photo-électrique.
- ✓ Recevoir les informations générées par le processeur et destinées à la commande du convoyeur.
- ✓ Recevoir et stocker le programme du processus. [5]

Nous avons trois types de mémoires [5]:

- ❖ **RAM** : mémoire vive c'est l'espace où est exécuté le programme. **ROM** : mémoire morte accessible uniquement en lecture.
- ❖ **EPROM** : mémoires mortes reprogrammables.

Les Modules d'entrées sorties (TOR)

Les modules d'entrées /sorties sont des interfaces de communication entre l'unité centrale et les différents capteurs et actionneurs, ils assurent le filtrage et l'adaptation des signaux électriques. [5]

1) Les modules d'entrées

Ils permettent à l'automate de recevoir des informations prévenantes de la part des capteurs.

2) Les modules de sorties

Les modules de sorties TOR assurent le raccordement de l'automate aux différents actionneurs et pré-actionneurs tels que (moteur, relais).

Les modules d'entrées/ sorties analogiques :

Ils sont les interfaces entre l'automate et le processus commandé.

1) Les modules d'entrées

Ils convertissent les signaux analogiques mesurés (vitesse) en signaux numériques à l'aide des convertisseurs CAN (Convertisseur Analogique/ Numérique) dont dispose l'automate pour être traité.

2) Les modules de sorties

Ils convertissent les signaux numériques en signaux analogiques à l'aide des convertisseurs CNA (Convertisseur Numérique/ Analogique) dont dispose l'automate GE FANUC.

Interface de communication

Intégré à la CPU, elle assure la connexion entre l'ordinateur et l'automate via un port RS232. (Port COM, ce dernier permet d'interconnecter deux matériels). [5]

ANNEXE 2 :

L'automate programmable industriel Schneider M340 :

L'armoire de la cellule est équipée d'un API (Automate Programmable Industriel) ou bien PLC (Programmable Logic Controller) Schneider M340 qui pilote et contrôle les différents composants de la chaîne. Voir figure



Figure A.1 : L'automate Schneider M-340

Définition d'un automate programmable industriel :

Un automate programmable industriel 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 PO côté actionneur) à partir des données d'entrées (capteurs) (partie commande ou PC côté capteur), l'API est un dispositif similaire à un ordinateur utilisé pour automatiser des processus comme la commande des machines sur une chaîne de montage dans une usine, là où les systèmes automatisés plus anciens emploieraient des centaines ou des milliers de relais et de cames, un simple automate suffit.

L'API peut traiter aussi :

- ❖ Des fonctions de calcul arithmétique.
- ❖ Temporisation.
- ❖ Comptage.
- ❖ Des liaisons avec d'autres appareils.

Structure d'un API :

Quel que soit l'automate on peut dire qu'il est composé de 4 parties :

- **L'alimentation** : Elle doit fournir l'énergie nécessaire au fonctionnement correct de l'automate et à l'ensemble de ses cartes.
- **Le processeur** : C'est le cerveau de l'automate qui traite les données. Il contient en mémoire le programme et élabore les ordres de commande. Il sert également de plus en plus à la communication sur les réseaux Modbus, Ethernet, Canopen, etc.
- **Les modules d'entrées** : Ce sont des cartes spécialisées capables de recevoir en toute sécurité pour l'automate les signaux issus des capteurs, des boutons poussoirs, etc.

Elles peuvent être en tout ou rien (TOR) ou analogiques.

- **Les modules de sorties** : Ce sont des cartes spécialisés capables de commander en toute sécurité pour l'automate les circuits extérieurs. Elles peuvent être en tout ou rien (TOR), ou analogiques.

Remarque :

Une entrée TOR reçoit un signal qui ne peut avoir que 2 valeurs (ex : 0 V et 24 V DC). Ce signal peut correspondre à l'état un BP (actionné, non actionné).

Une entrée analogique reçoit un signal qui peut varier entre 2 valeurs (ex : 0 – 10 V DC, 4 – 20mA). Ce signal peut être l'image d'une température mesurée par un capteur.

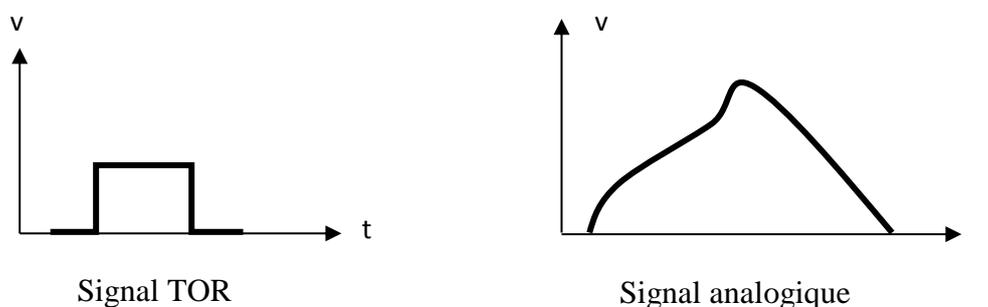


Figure A.2 : Le signal TOR et Analogique

Présentation de l'automate Schneider M340 :

Schneider M340 qui peut être utilisé seul mais se révèle être aussi le compagnon idéal des Modicon premium et quantum. Il améliore la performance, la qualité et la rentabilité des procédés industriels, sites de production ou machines.

Les processeurs Modicon M340 sont fournis de base avec une carte mémoire Flash de type SD card (Secure Digital Card). Cette carte mémoire est destinée à la sauvegarde de la zone programme, symboles, commentaires et de la zone des constantes.

M340 de Télémécanique est l'automate le plus intégré jamais conçu. Il réunit deux propriétés essentielles pour les fabricants de machines complexes, à savoir une capacité de traitement maximale et un volume le plus compact possible. Le traitement des données s'effectue en quelques millisecondes. Le Modicon M340 est rapide en opérations booléennes et en virgule fixe ou flottante. Sa mémoire surdimensionnée de 4 Mbytes lui permet de gérer des applications comportant jusqu'à 70k d'instructions et 256 kbytes de données. L'environnement multitâche puissant permet à l'utilisateur de grouper les tâches de commande en fonction du temps de réponse souhaité: contrôle d'événements, actions réflexes, gestion de tâches rapides (FAST) en quelques ms, exécution de tâches via la tâche maître (MAST) de 10 à 20 ms. Le processeur M340 comporte en standard un port USB-V2 et un ou deux ports de communications supplémentaires. Il y a le choix entre des combinaisons avec Modbus TCP/IP, CANopen ou Modbus série:

Le Modibo M340 puise le maximum dans l'expertise de Télémécanique en comptage, positionnement et régulation. L'utilisateur peut de manière simple et efficace transposer son propre savoir-faire et ses spécificités en un ou plusieurs des 5 langages de programmation de la norme IEC (List, Ladder, Diagramme blocs fonctions, Liste structurée, Grafcet).

Caractéristique de l'automate Schneider M340 :

Caractéristiques mécanique :

- Dimensions : hauteur : 210 mm, largeur : 400 mm, longueur : 410 mm
- Poids : 5, 7 kg environ
- Caractéristiques électriques
- Tension : 230V monophasé $\pm 10\%$
- Puissance absorbée : 50/60 VA
- Fréquence : 50/60 Hz $\pm 5\%$
- Le courant de court-circuit conventionnel est fixé à 10 kA.
- Tension de tenue assignée de tenue aux chocs : 2,5 kV
- Classe de protection : I
- Cet équipement doit être protégé en amont par un disjoncteur DDR (Dispositif Différentiel Résiduel) de sensibilité ≤ 30 mA de classe AC (non fourni).

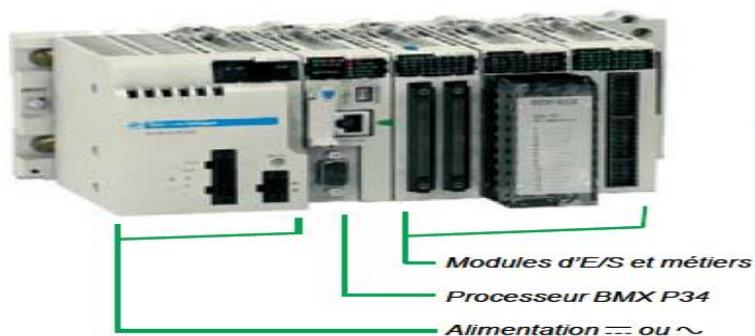
Caractéristiques techniques des principaux constituants L'automate :

Il s'agit d'un automate M340 comprenant :

- Un module d'alimentation.
- Un processeur en programmation comprenant une liaison Modbus ou RS232/RS485
- Un module 16 entrées TOR 24V DC.
- Un module 16 sorties TOR relais 50VA.
- Un module 4 entrées et 2 sorties analogique multigammes.
 - Une alimentation 24 V 4,2 A maximum.
 - 16 douilles sécurisées jaunes repérées : %I1.XX correspondants aux 16 entrées TOR.
 - 17 douilles vertes repérées %Q2.XX correspondants aux 16 sorties TOR et un point commun.
 - 8 douilles jaunes repérées %IW3.X et COM correspondants aux 4 entrées analogiques.
 - 4 douilles vertes repérées %QW3.X et COM correspondants aux 2 sorties analogiques.
 - 2 douilles rouges 24V DC et 2 douilles bleues 0V DC câblées sur l'alimentation 24VDC 4.2A.
 - Un voyant Led blanc "SOUS TENSION".

Description de l'automate Modicon M340 :

Les processeurs Standard et Performance de la plate-forme d'automatisme Modicon M340 gèrent l'ensemble d'une station monorack ou multirack automate dont les emplacements peuvent être équipés de : modules d'entrées/sorties « Tout ou Rien », modules d'entrées/sorties analogiques, modules métiers (comptage, commande d'axe, communication Ethernet Modbus/TCP).



FigureA.3 : Configuration monorack

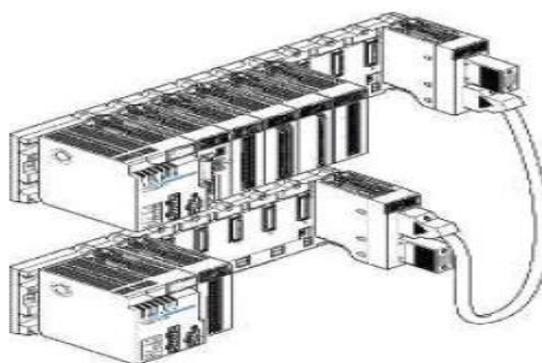


Figure A.4 : configuration multirack



Figure A.5 : la CPU M-340

1. Vis de sécurité pour le verrouillage du module dans l'emplacement du rack.
2. Un bloc de visualisation comprenant, selon modèle 8 ou 10 voyants.
3. Un connecteur type USB mini B pour le raccordement d'un terminal de programmation (ou d'un terminal de dialogue opérateur Magelis XBT GT/GK/GTW).
4. Un emplacement équipé de sa carte mémoire Flash pour la sauvegarde de l'application. Un voyant, situé au-dessus de cet emplacement indique la reconnaissance ou l'accès à la carte mémoire.

5. Un connecteur type RJ45 pour liaison série Modbus ou liaison Mode Caractères (RS 232C/RS 485, 2 fils, non isolée)

6. Un connecteur type RJ45 pour le raccordement au réseau Ethernet Modbus/TCP 10BASE-T/100BASE-TX.

Structure mémoire de l'automate M340 :

RAM interne :

La mémoire application se décompose en zones mémoire, répartie physiquement dans la mémoire RAM interne du processeur Modicon M340, et cette mémoire est de capacité totale 4 MB.

Carte mémoire :

Les processeurs Modicon M340 fournis de base avec une carte mémoire flash de type

SD card (secure digital). Les données sont automatiquement stockées dans cette dernière de capacité de 8 MB. Elle est disponible pour la sauvegarde de 16 MB de fichiers. Comme il n'y a pas de batterie, l'utilisateur fait des économies de maintenance. Le programme application se trouve dans la carte SD et, en cas de disparition de la tension, les données sont stockées dans une mémoire flash intégrée au processeur. L'utilisateur peut mettre, de manière simple, son application à jour ou la transférer vers d'autres machines, sans interrompre leur fonctionnement. Le contenu de la carte peut être chargé sur d'autres pour copier l'application dans d'autres machines du parc.



Figure A.6 : CPU M340 avec une carte mémoire

Alimentation :

Intégré à la CPU, il assure la distribution d'énergie électrique aux différents modules, et dans notre cas l'alimentation est de 24V/10A DC.

Communication :

La programmation des applications dans le Modicon M340 peut s'effectuer via le port USB ou un réseau Ethernet standard. Où que l'on se trouve, on a toujours accès à la machine via un modem ou une liaison ADSL standard. En fonction de la technologie choisie, il est possible de programmer en ligne, de transférer des programmes et d'accéder aux fichiers de données.

logiciel UNITY PRO :

Le logiciel de mise en œuvre UNITY PRO assiste le concepteur de l'application dans la gestion de la structure et de l'occupation de l'espace mémoire de la plate-forme d'automatisme Modicon M340.

1) Présentation :

Unity Pro est le logiciel commun de programmation, mise au point et exploitation des gammes d'automates Modicon M340, M580, Premium, Momentum et Quantum.

Unity Pro est un logiciel multitâche qui offre les fonctionnalités suivantes :

- Logiciel "tout en un",
- 5 langages de programmation IEC 61131-3,
- Langage de programmation LL 984,
- Librairie de blocs fonctions (DFB) intégrée et personnalisable,
- Simulateur automate sur PC pour valider votre programme avant installation,
- Tests intégrés (Built-in test) et diagnostic,
- Large gamme de services en ligne,
- Cyber sécurité.

Cyber sécurité : Des conseils sur la sécurité sont mis à disposition des clients de Schneider afin d'assurer que leurs systèmes soient protégés des attaques.

- Une protection contre les modifications de programmation à distance grâce à un mot de passe,
- Une option permettant d'activer ou de désactiver les services HTTP ou FTP [19]

Pour créer un projet sous Unity Pro il faut suivre ces marches :

1. Lancez le logiciel Unity Pro.
2. Cliquez sur **Fichier**, puis sur **Nouveau** et sélectionnez un automate.

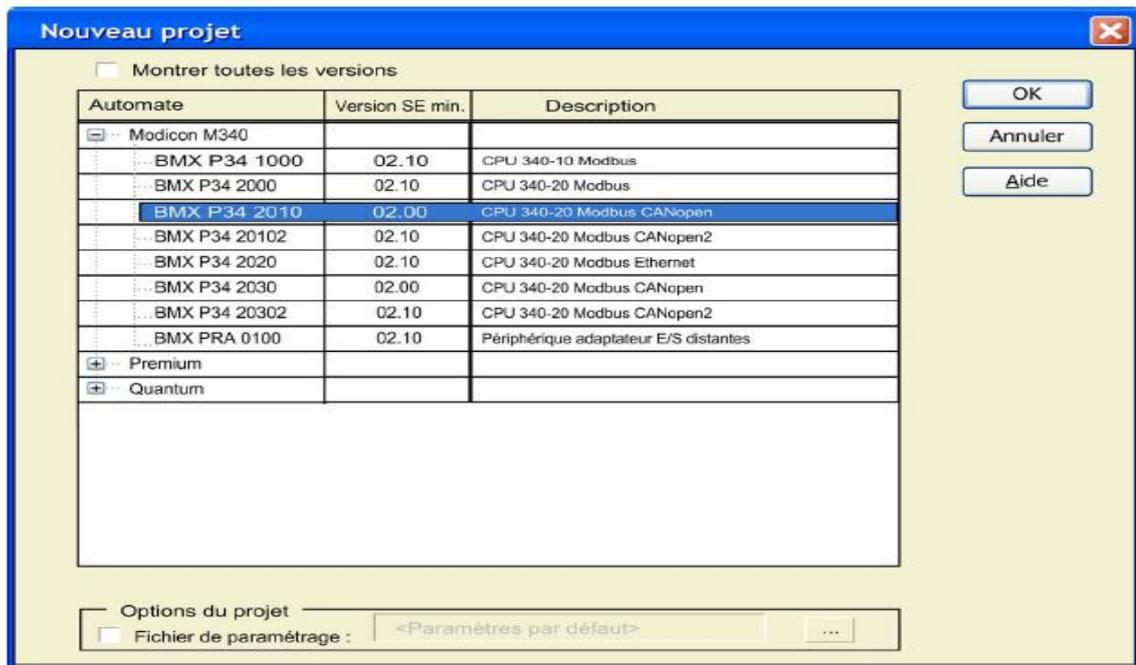


Figure A.7 : Création d'un nouveau projet

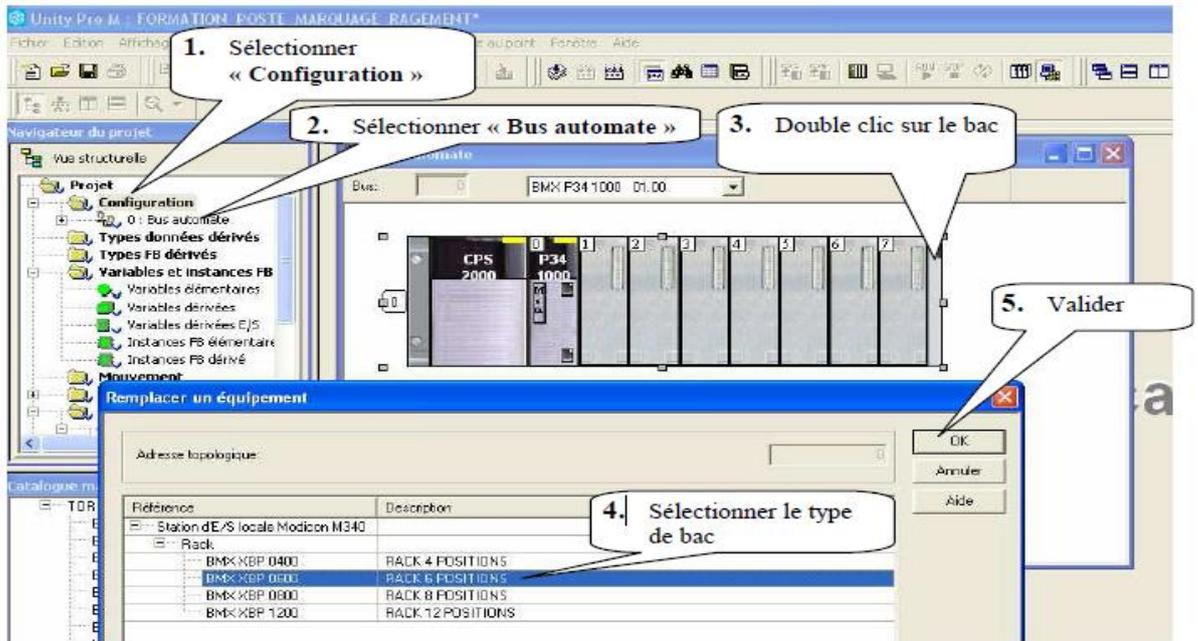
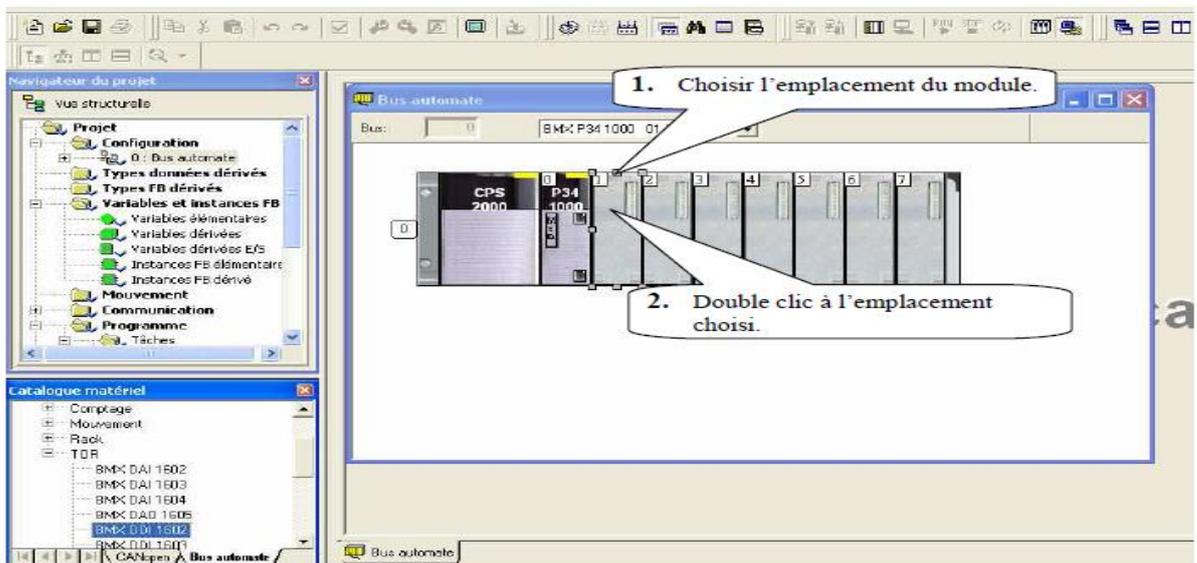
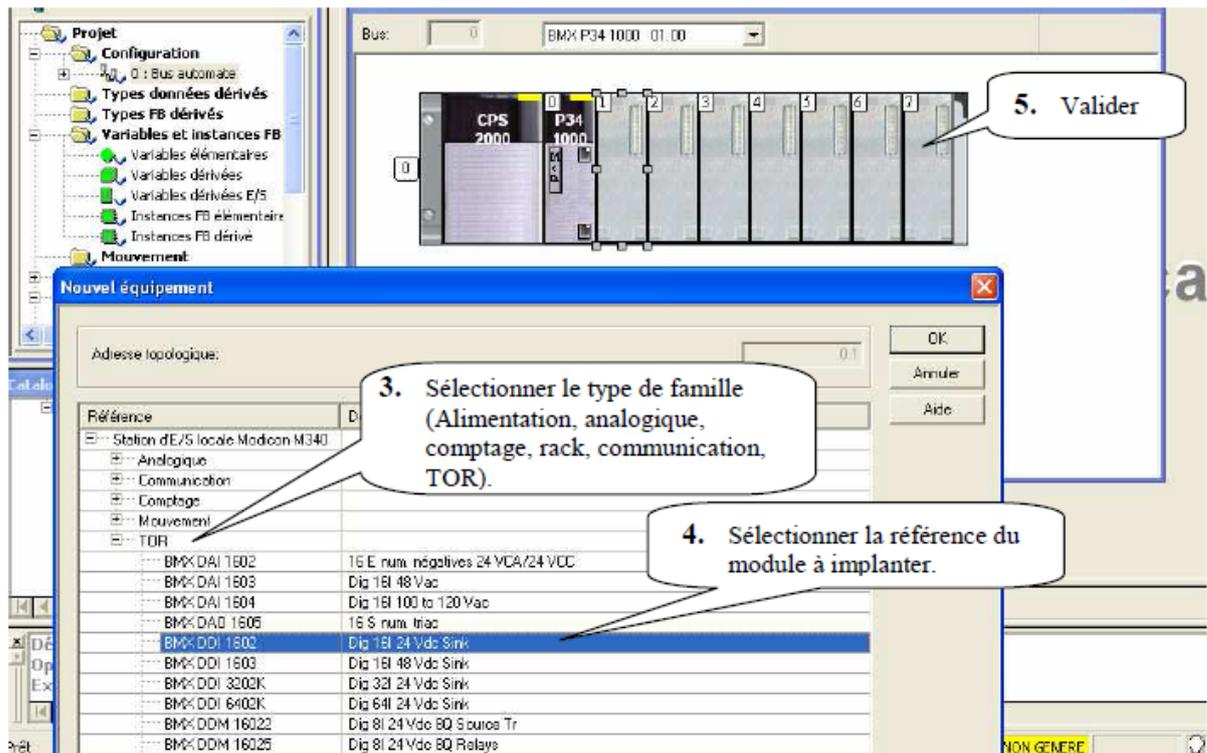


Figure A.8 : Configuration type du bac



FigureA.9 : choix des modules d'entrées /sorties



FigureA.10 : Configuration des cartes d'entrées /sorties

En fin de saisie de la configuration matérielle, sauvegarder.

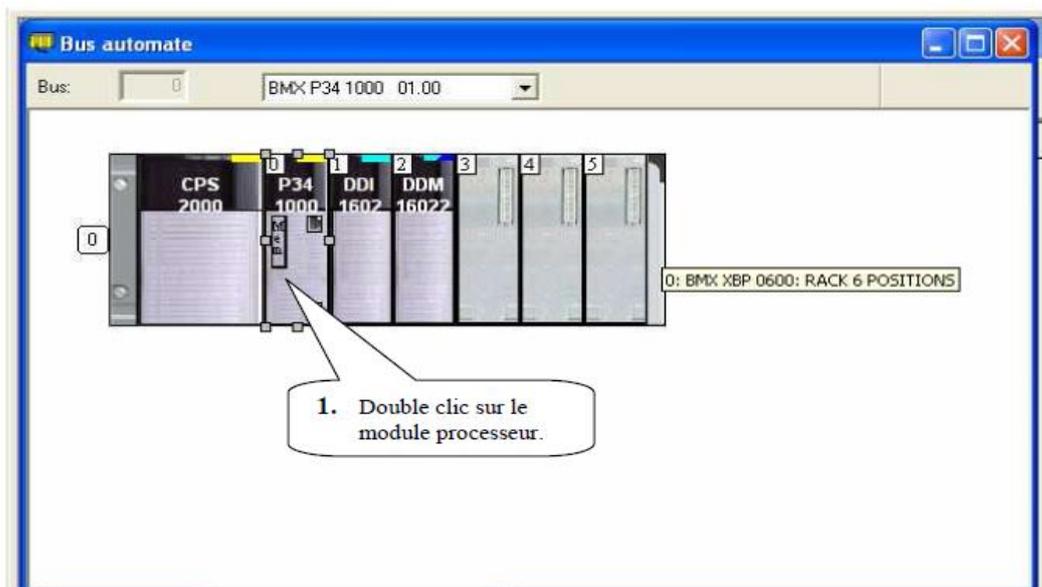
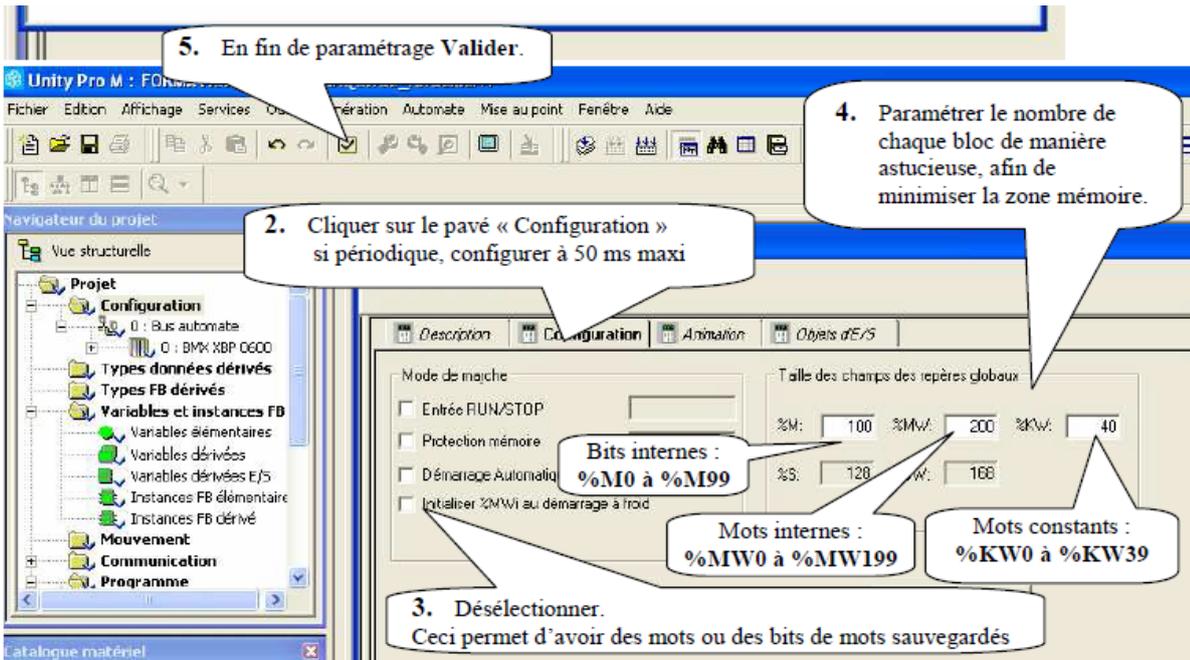


Figure A.11 : configuration du processeur



FigureA.12 : configuration des modules

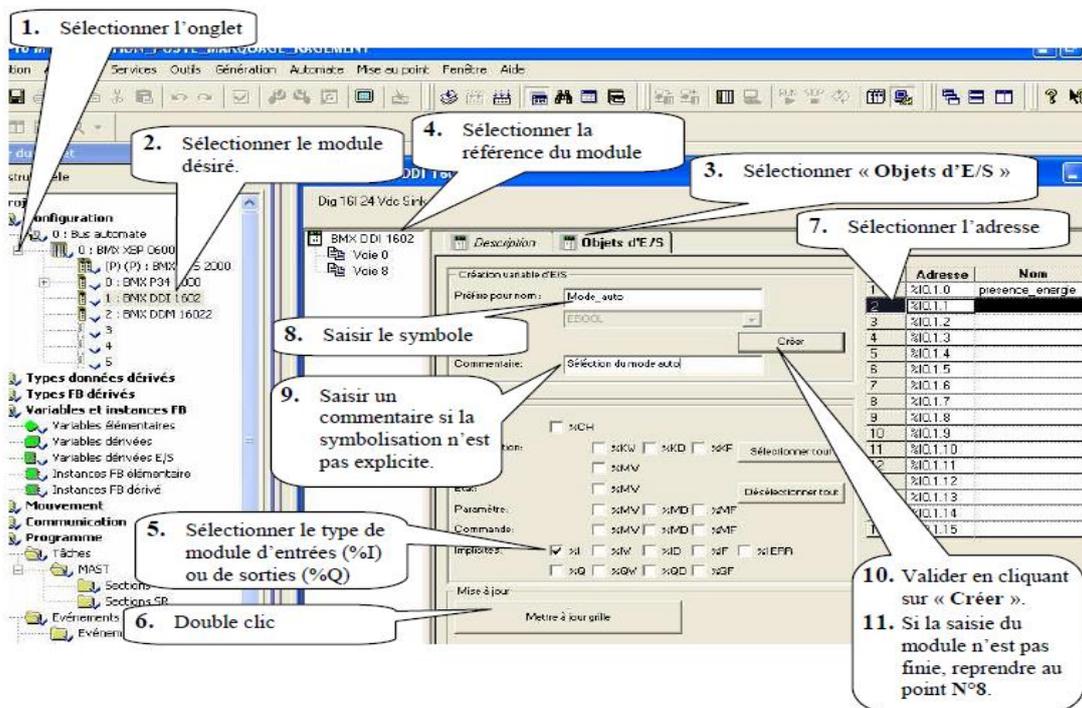


Figure A.13 : Edition des adresses symboliques des entrées/sorties

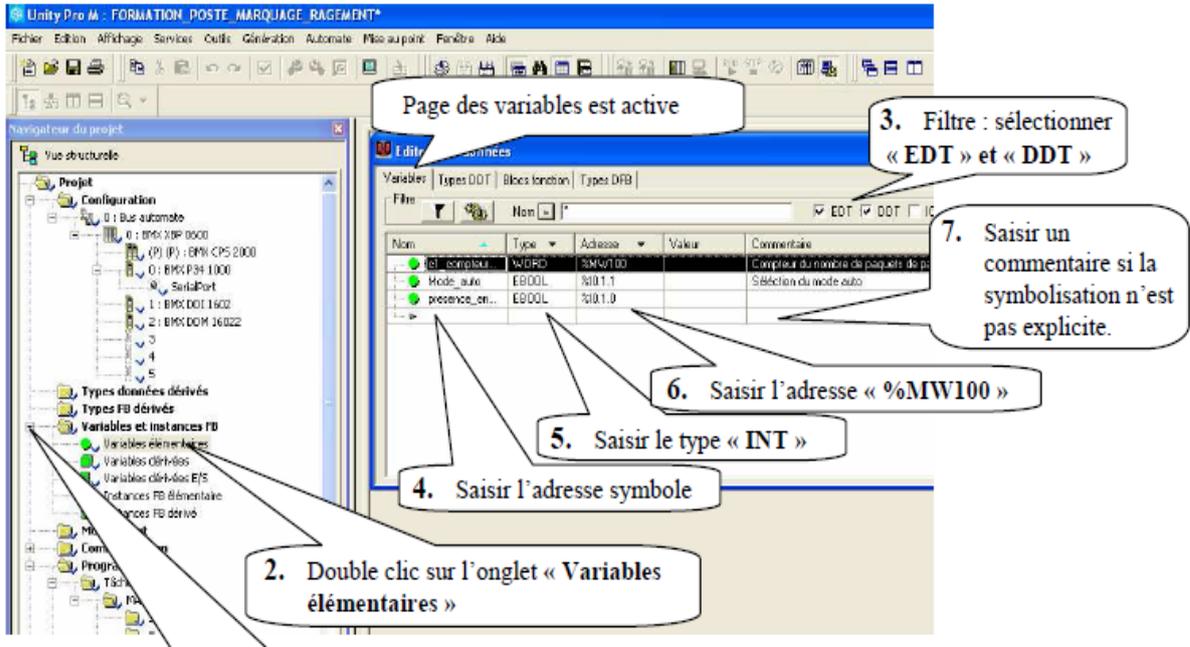


Figure A.14 : Edition des adresses symboliques des mots internes « Editeur de données » non actif

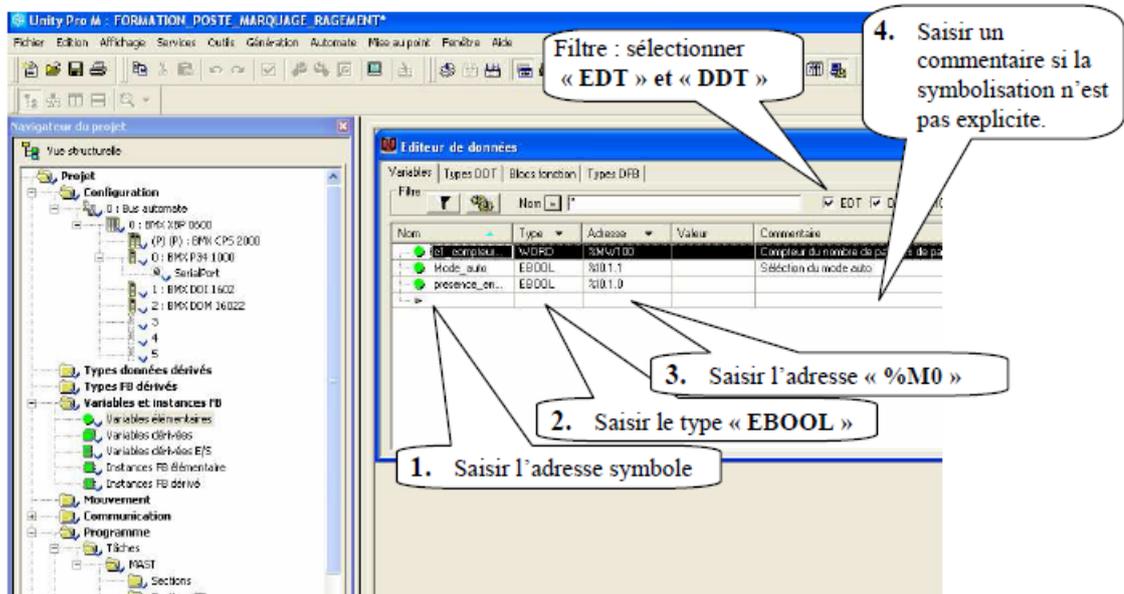


Figure A.15 : Edition des adresses symboliques d'un bit interne

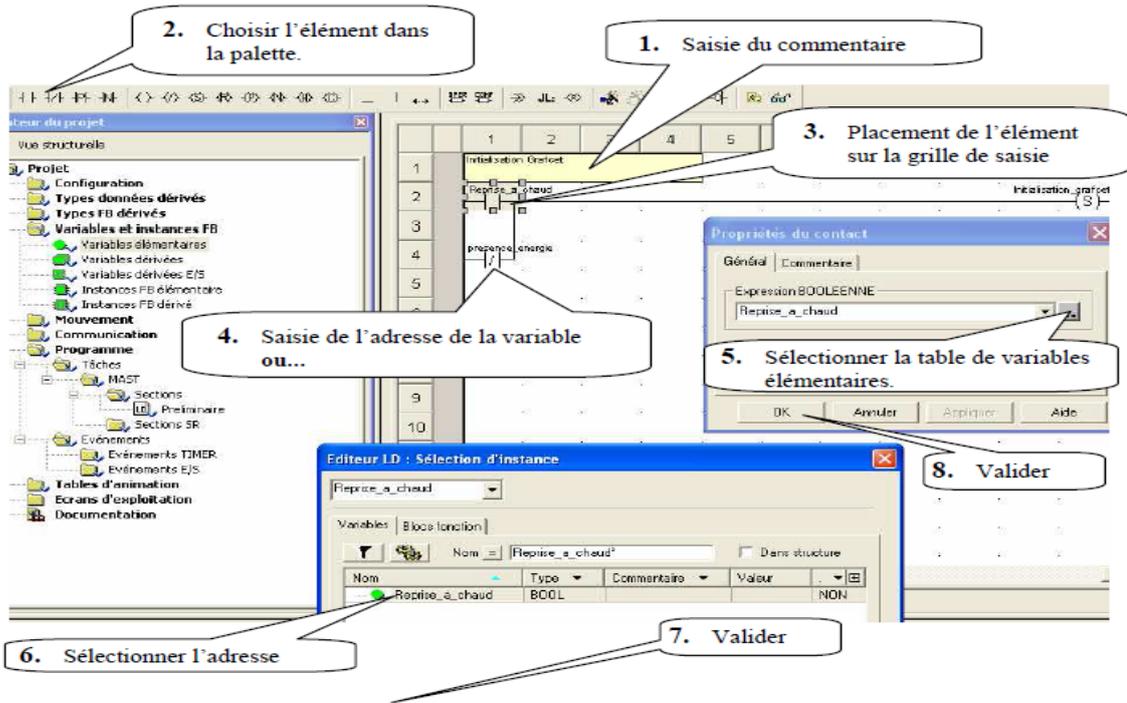


Figure A.16 : Méthodologie de saisie du ladder

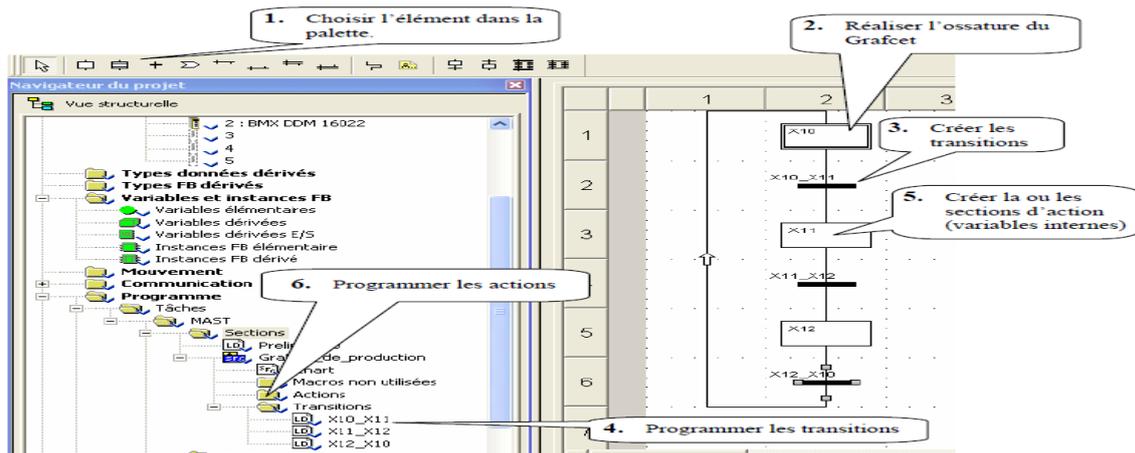


Figure A.17 : Méthodologie de saisie du GRAFCET

BIBLIOGRAPHIE

Bibliographie

- [1] A. REILLER, «Analyse et maintenance des automatismes industriels,» ellipses, Paris, 1999.
- [2] P. Jamet, «LES ROBOTS DANS L' INDUSTRIE,» Le lablogatoire, Quebec, 2009.
- [3] A. GHI, «Conception d'antennes RFID,» thèse, Grenoble, 2008.
- [4] P. T. A. T. B. B. M. NOYEL, «RETOUR D'EXPÉRIENCE INDUSTRIELLE SUR LE CHOIX D'UNE TECHNOLOGIE D'INFORMATION PORTÉES PAR LES PRODUITS,» Bordeaux, 2012.
- [5] H. M. ADNANE MOLOUD, «Mémoire de Master2 automatique,Automatisatin et pilotage des cellules de production robotisées,» Université Mouloud MAMMERI de Tizi_Ouzou, Tizi_Ouzou, 2013.
- [6] A. SARAC, «Modélisation et aide à la décision pour l'introduction des technologies RFID dans les chaînes logistiques,» Gardanne, 2010.
- [7] Z. L. e. S. S. R. Murray, «A mathematical introduction to automatisaton,» CRC Press., Université de Toronto, 1994.
- [8] A. B. Bernard SCHNEIDER, «Automatisation Industrielle,» Yverdon-les-Bains, 26 avril 2006.
- [9] J. Virta, «Application integration for production operations,» Suisse, 2010.
- [10] p. bonnet, «Cours Supervision – Serveur OPC,» Otaniemi, lille1, 2010.
- [11] L. V. Gauthier Picard, «initiation à la programmation orientée -objet avec le langage java,» Saint-Etienne, 2013-2014.
- [12] D. W. Nemmiche Adila, «Etude et intégration d'un dispositif d'identification basé sur la technologie RFID,» Université Abou Bakr Belkaid-Tlemcen, 2013.
- [13] P. GEORGET, «L'identification par radio fréquence principe et application,» EPC GLOBAL, France, 2004.
- [14] N. Seriot, «Les systèmes d'identification radio (RFID),» Yverdon-les-Bains, 2005.
- [15] B. S. B. R. O. J. G. BACHOTI YOUSSEF, «PROJET RFID,» TELECOM Sud paris, Sud Paris, 2011.

-
- [16] B.-H. BAUMER, «Notice d'instalation Modbus,» France, 2010.
- [17] Q. D. T. N. RINEAU Yoann, «Automatisation d'un poste de tri,» Ecole d'ingénieur de l'université d'ANGERS TSTIA, France, 2007/2008.
- [18] L. BERGOUGNOUX, «cours Automates Programmables Industriels,» D'épartement de Mécanique Énergétique, Marseille, 2004/2005.
- [19] L. G. S. F. Tahar, «programmer-sous unity-pro,» France, 2013.