

الجمهورية الجزائرية الديمقراطية الشعبية  
République Algérienne démocratique et populaire

وزارة التعليم العالي والبحث العلمي  
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد دحلب البليدة  
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا  
Faculté de Technologie

قسم الإلكترونيك  
Département d'Électronique



## Mémoire de Master

Mention Électronique

Spécialité Signaux en Ingénierie des Système, Informatique Industrielle(SISII)

présenté par

**Moussidene Abdelkader**

---

# Développement d'un système SCADA pour le contrôle d'un réchauffeur gaz bain marie

---

Proposé par : Amoura NABIL & Benselama Zoubir & Bennila Noureddine

Année Universitaire 2015-2016

## Remerciements

---

Je commence par rendre grâce à Dieu « soubhânahou wa ta'âlâ » qui m'a guidé sur le droit chemin tout au long du travail, m'a inspiré les bons pas et les justes réflexes et m'a donné la capacité d'écrire et de réfléchir, la force d'y croire, la patience d'aller jusqu'au bout. Sans sa miséricorde, ce travail n'aurait pas abouti. Merci Allah !

Il me serait impossible de citer nommément toutes les personnes qui m'ont aidé, encouragé et soutenu afin que ce travail puisse voir le jour. Que toutes ces personnes trouvent ici l'expression de ma sincère reconnaissance.

Je tiens tout d'abord à remercier très chaleureusement tous les enseignants qui ont contribué de près ou de loin à ma formation.

Je tiens également, à remercier M. Amoura Nabil pour avoir assuré l'encadrement de ce projet au sein de la centrale électrique de LARBAA. Je le remercie tout particulièrement de m'avoir donné l'opportunité de travailler sur un projet d'une telle envergure.

Je témoigne mes sincères gratitude à M. Bennila Noureddine ainsi que M. Benselama Zoubir mes promoteurs à l'université de Blida 1 pour nos séances de travail agréables et fructueuses, leurs remarques pertinentes, mais aussi pour leurs écoute et leurs discours bienveillants.

Il m'est difficile de ne pas remercier M. Bennila Noureddine ainsi que M. Benselama Zoubir de m'avoir aidé à comprendre et à exposer le plus justement possible ce travail.

Je remercie vivement Mesdames et Messieurs les membres du jury d'avoir accepté d'évaluer ce travail.

Ces remerciements ne seraient pas complets sans une pensée sincère à mes amis proches et ma famille et plus particulièrement mes parents pour leur soutien constant.

*Moussidene*

## Dédicaces

---

Avec tout mon amour éternel et avec l'intensité de mes émotions. Je dédie ce modeste travail à :

Mon très cher père « Boualem » qui m'a beaucoup aidé pour accomplir ce travail. Il est très difficile de choisir les termes adéquats pour t'exprimer mon amour et mon respect que Dieu te garde en bonne santé pour nous.

Mon adorable mère « khaira » voudrais-je des profondeurs de mon amour et de ma reconnaissance te dédier ce mémoire, dont tous les mots que je pourrais utiliser seraient insuffisants pour te témoigner l'amour que je te porte.

Mes chers frères « Mohammed », « Abderrahmene » et « Abdenour » et ma sœur « Salssabil » En témoignage de mon amour éternel que Dieu vous garde, vous protège et vous offre une vie pleine de joie.

Mes amis « Ahmed », « Djawad », « Abderezzak », « Nassim », « Zakaria » et à tous mes collègues.

Mlle « Souna » merci d'avoir parcouru avec moi le chemin menant à ce stade et de m'avoir encouragé à aller toujours plus loin.

Toutes les personnes que j'aime, à ceux qui m'ont soutenue dans les moments difficiles, en témoignage des liens qui nous unissent, je vous dédie ce travail, je vous souhaite beaucoup de succès, de courage et de bonheur tant professionnel que familial.

*Abdelkader*

---

**ملخص :** للإستجابة إلى المتطلبات التي تتزايد بدون توقف، يستعمل أصحاب المصانع عدة طرق، من بينها المراقبة الصناعية والتي تعتبر من أهم الوسائل التي تستعمل من أجل متابعة عملية الإنتاج. مما أدى إلى ظهور وسائل بصرية و بيانية حديثة. يعتبر نظام SCADA من أهم هاته التكنولوجيات. نسعى من خلال هذا العمل إلى إعداد تطبيق يقوم بمراقبة متحكم صناعي قابل للبرمجة من نوع Allen Bradley SLC 5/03 باستعمال تكنولوجيا التفاعل بين الآلة والإنسان.

**كلمات المفاتيح:** المراقبة الصناعية، وسائل بيانية، التحكم ، متابعة، SCADA ، IHM.

---

**Résumé :** Pour répondre à des exigences sans cesse croissantes, les entreprises industrielles ont mis au point plusieurs outils. La supervision industrielle, est l'un des outils exploités. C'est la surveillance et le contrôle de l'état de fonctionnement d'un procédé pour l'amener et le maintenir à son point de fonctionnement optimal. Ce qui a augmenté le besoin d'un moyen de visualisation des processus industriels, les systèmes SCADA sont parmi les nouvelles tendances utilisées. Le présent sujet consiste alors à développer un système SCADA pour superviser l'API Allen Bradley SLC 5/03 de l'unité réchauffeur gaz à bain marie en exploitant la technologie des IHM.

**Mots clés :** Supervision industrielle; visualisation; surveillance ; contrôle; SCADA ; IHM.

---

**Abstract :** To meet the requirements ever increasing, industrial enterprises have developed several tools. Industrial supervision, is one of the operating tools. This is monitoring and controlling the operating state of a process to bring and maintain at its optimum operating point. Which increased the need for a display means of industrial processes, SCADA systems are among the new trends used. The present work aims at developing a SCADA system to monitor the Allen Bradley SLC 5/03 PLC unit heater gas bath exploiting the HMI technology.

**Keywords :** industrial supervision, visualisation, monitoring, controlling, SCADA, HMI.

---

## Listes des acronymes et abréviations

ABB : ASEA Brown Boveri.

API : Automate Programmable industriel.

CAN : Convertisseur Analogique Numérique.

CEI : Commission Electrotechnique Internationale.

CIM : Computer Integrated Manufacturing.

CPU : Central Processing Unit.

DCS : Distributed Control System.

E/S : Entrées/Sorties.

EGA : Electricité et Gaz d'Algérie.

EPIC : Etablissement Publique à caractère Industriel et Commercial.

FBD : Function Block Diagram (Schéma par blocs).

GI : Groupe Industriel.

IDE : Integrated Development Environment (Un environnement de développement libre).

IHM : Interface Homme-Machine.

IL : Instruction List.

LD : Ladder Diagram.

OPC : OLE for Process Control.

PC : Partie Commande.

PC : Personal computer.

PFE : Projet de Fin d'Etude.

PLC : Programmable logic controller.

PO : Partie Opérative.

RTU : Remote Terminal Unit.

SCADA : Supevisory Control And Data Acquisition.

SFC : Sequential Function Chart.

SISII : Signaux en Ingénierie des Système, Informatique Industrielle.

SONELGAZ : Société Nationale de l'Electricité et du GAZ.

SPA : Société Par Action.

SPA : Systèmes de Production Automatisés.

SPE : Société algérienne de Production de l'Electricité.

ST : Structured Text.

TG : Turbine à Gaz.

TOR : Tout Ou Rien.

TRT : Temps de Réponse Total.

USB : Universal Serial Bus (Bus universel en serie).

VB : Visual Basic.

WYSIWYG : What You See Is What You Get.

# Table des matières

Introduction générale .....	1
Chapitre 1 Cadre général du projet .....	4
1.1 Introduction .....	4
1.2 Présentation de l'organisme d'accueil .....	4
1.2.1 Historique.....	5
1.2.2 Organisation du groupe .....	6
1.2.3 Le groupe en chiffres .....	8
1.2.4 Le métier de la production de l'électricité .....	8
1.3 La centrale turbine à gaz de LARBAA .....	9
1.3.1 Dates phares .....	9
1.3.2 Fiche technique de la centrale.....	10
1.3.3 Principes de fonctionnement de la centrale.....	10
1.3.4 Réchauffeur gaz à bain marie .....	14
a Fonctionnement du réchauffeur .....	14
b Les instruments de mesure .....	17
1.4 Etude des besoins et présentation du sujet .....	18
1.5 Conclusion .....	19
Chapitre 2 Les systèmes automatisés .....	20
2.1 Introduction .....	20
2.2 Les systèmes automatisés.....	20
2.3 Objectifs de l'automatisation.....	21
2.4 Automate programmable industriel .....	21
2.5 ARCHITECTURE .....	22
2.6 Constitution d'un API .....	23

2.7	Fonctionnement.....	23
2.8	Caractéristiques des API.....	26
2.9	PROGRAMMATION.....	26
2.9.1	Les langages de programmation :.....	26
2.9.2	Les outils de programmation.....	29
2.10	Sécurité de l'API.....	29
2.11	Critères de choix d'un automate.....	30
2.12	Les automates et la communication.....	30
2.13	Conclusion.....	32
Chapitre 3 La supervision des systèmes automatisés.....		33
3.1	Introduction.....	33
3.2	La supervision.....	33
3.3	Supervision industrielle.....	34
3.4	La supervision dans la hiérarchie d'une entreprise.....	35
3.5	Objectifs et rôles de la supervision.....	36
3.6	Fonction de la supervision.....	37
3.7	Les Techniques de la supervision.....	40
3.8	La supervision dans un environnement SCADA.....	42
3.8.1	Définition du SCADA.....	42
3.8.2	Composants du SCADA.....	42
3.8.3	Les Interfaces Homme-Machine SCADA.....	43
a	Définition d'une Interface Homme-Machine.....	43
b	Evolution des interfaces Homme machine.....	44
3.8.4	Fonctionnalités d'un système SCADA.....	44
3.8.5	Communication des systèmes SCADA.....	45
3.9	La supervision dans un environnement DCS.....	46



3.10	Superviseurs SCADA .....	46
3.11	Choix et discussions.....	47
3.12	Conclusion .....	47
Chapitre 4 Conception et mise en œuvre de la solution.....		48
4.1	Introduction .....	48
4.2	Etude de l'existant.....	48
4.2.1	Présentation de l'API .....	49
4.2.2	L'architecture de l'API.....	49
a	L'unité centrale CPU .....	50
b	Le module d'alimentation .....	51
c	Modules d'entrée/sorties analogique .....	52
d	Modules d'entrée/sorties (TOR).....	52
4.2.3	Communication de l'API .....	52
4.2.4	Identification des Entrées/Sorties : .....	53
a	Les entrées logiques (TOR) .....	53
b	Les sorties logiques (TOR).....	55
c	Les entrées analogiques .....	57
d	Les sorties analogiques.....	57
4.3	Conception de la solution .....	57
4.3.1	Description de l'application .....	58
4.3.2	Fonctionnalités de l'application.....	58
a	Analyse préliminaire .....	58
b	Analyse détaillée.....	59
4.3.3	Architecture générale de la solution .....	60
4.4	Mise en œuvre de la solution .....	61
4.4.1	Périmètre technique et fonctionnel .....	61

a	Matériel .....	61
b	Systèmes d'exploitation .....	61
c	Logiciel .....	62
4.4.2	Processus de développement.....	62
a	Interface Homme Machine.....	62
b	Base de données.....	68
4.5	Conclusion .....	68
Chapitre 5	Simulation et résultats obtenus.....	69
5.1	Introduction .....	69
5.2	Simulation .....	69
5.2.1	Les limites de la simulation.....	70
5.2.2	Avantages de la simulation.....	70
5.2.3	Les raisons du succès de la simulation .....	70
5.3	Principe de fonctionnement .....	71
5.3.1	Sans simulation .....	71
5.3.2	Avec simulation.....	72
5.4	Environnements de simulation .....	73
5.5	Conclusion .....	78
	Conclusion générale.....	79
	Annexe A - visuel studio express .....	1
	Annexe B – RSLinx Classic .....	6
	Annexe C – RSLogix Emulate 500.....	12
	Annexe D – Null-modem Emulator (com0com).....	14
	Annexe E – RSLogix 500 .....	16
	Bibliographie .....	21

## Liste des figures

Figure 1 : Planification et conduite du projet. ....	3
Figure 2: Les sociétés du groupe SONELGAZ [2]. ....	7
Figure 3:Schéma synoptique du parcour de gaz.....	10
Figure 4: Système d'admission d'air turbine. ....	11
Figure 5: a Station fuel, b skid final fuel. ....	12
Figure 6: Cycle de fonctionnement d'une turbine à gaz (cycle simple).....	12
Figure 7: a Transformateur principal, b Station d'évacuation d'énergie électrique. ....	13
Figure 8:Réchauffeur gaz à bain marie (Vue de face) [6]. ....	15
Figure 9:Réchauffeur gaz bain marie (Vue de coté) [6]. ....	16
Figure 10: Dispositifs de l'unité: a PT100, b capteur de pression, c électrovanne, d vanne régulatrice. ....	18
Figure 11 : Structure d'un système automatisé.....	21
Figure 12 : Automate compact. ....	22
Figure 13 : Automate modulaire.....	22
Figure 14 : Fonctionnement d'un API. ....	23
Figure 15 : Cycle de fonctionnement d'un API [11]. ....	25
Figure 16: Temps de scrutation vs Temps de réponse [11]. ....	26
Figure 17 : List d'instruction. ....	27
Figure 18 : Langage littéral structuré.....	27
Figure 19 : Les graphes de fonction séquentielle. ....	28
Figure 20 : Langage à contacts.....	28
Figure 21 : Blocs Fonctionnels. ....	29
Figure 22 : Bus de terrain [11]. ....	31
Figure 23 : pyramide CIM.....	32
Figure 24: supervision industrielle [16]. ....	35
Figure 25 : Le modèle de référence CIM des processus industriels. ....	35
Figure 26 : Exemple d'une interface synoptique [16]. ....	37
Figure 27 : Exemple d'une interface des courbes [16]. ....	38
Figure 28 : exemple d'une interface de gestion des alarmes [16]. ....	38
Figure 29 : exemple d'une interface d'historisation du procédé [16]. ....	39

Figure 30: exemple d'une interface de gestion des gammes de fabrication [16].	39
Figure 31: Architecture générale d'un système de Supervision en ligne [19].	40
Figure 32 : Composantes de la surveillance industrielle [19].	41
Figure 33 : Interaction Homme Machine [27].	44
Figure 34: L'API Rockwel Allen Bradley SLC 5/03.	49
Figure 35: Le processeur SLC 5/03.	50
Figure 36: Ports de processeur SLC 5/03.	51
Figure 37: Les boites d'alimentation de l'API.	51
Figure 38: Modules d'E\S.	52
Figure 39 : a. Contrôle/surveillance sans IHM, b. Contrôle/surveillance avec IHM.	58
Figure 40: fonctionnalité préliminaire du système.	59
Figure 41: Architecture globale de la solution.	61
Figure 42 : Page d'accueil.	63
Figure 43 : La page A-propos.	64
Figure 44: Configuration du port COM.	64
Figure 45 : Vue de côté (Bruleur).	65
Figure 46 : interfaces d'affichage des lampes.	65
Figure 47 : Page de l'historique des alarmes.	66
Figure 48 : Vue de face (réchauffeur).	66
Figure 49 : Interface du mode avancé.	67
Figure 50 : Interface graphe de données.	67
Figure 51 : Table de la Base de données.	68
Figure 52 : Système de supervision sans simulation.	72
Figure 53 : supervision avec simulation.	73
Figure 54 : Création d'un nouveau driver.	74
Figure 55 : Ouverture du projet sous RSLogix Emulate.	75
Figure 56 : Configuration du canal de transmission.	75
Figure 57 : Création d'une paire de ports.	76
Figure 58 : lancement de RSLogix 500.	76
Figure 59 : Choix du dispositif.	77
Figure 60: chargement du programme dans RSLogix Emulate 500.	77
Figure 61 : Passage au mode RUN.	78

## Liste des tableaux

Tableau 1: Le groupe SONELGAZ en chiffres [3].....	8
Tableau 2 : Fiche technique de la centrale [5].....	10
Tableau 3: Les entrées logiques de l'API.....	54
Tableau 4: Les sorties logiques de l'API. ....	57
Tableau 5: Les entrées analogiques de l'API.....	57
Tableau 6: Les sorties analogiques. ....	57

# Introduction générale

---

## Contexte général

C'est dans un environnement fortement complexe et hautement concurrentiel qu'évolue la majeure partie, si ce n'est la totalité, des entreprises. Ce climat de forte concurrence exige aux entreprises industrielles de répondre, le plus rapidement possible, aux attentes du marché, de leur clientèle et de leurs partenaires. Afin de ne pas se laisser distancer par les concurrents, elles doivent adopter une politique qui tient compte de l'évolution spectaculaire des technologies de l'automatisation de la production industrielle.

En effet, cette évolution rapide a permis de contourner la plupart des difficultés rencontrées dans le monde industriel, et a fourni plusieurs possibilités pour satisfaire les exigences et les critères demandés tels que la productivité, la sécurité, l'optimisation des coûts de production et l'amélioration des conditions de travail. Néanmoins, en raison d'une telle modernisation, les systèmes industriels deviennent de plus en plus complexes et sophistiqués.

Ces systèmes complexes contiennent des dispositifs, juger à nos jours indispensables, appeler Automates Programmables Industriels (API). Ces API peuvent être affectés par des défauts provenant de leurs composants internes (défaillance d'un moteur, défaillance ou dérive d'un capteur) sachant qu'un simple défaut sur le fonctionnement du système n'est pas du tout toléré car une défaillance d'une partie d'un processus peut s'aggraver et paralyser ou endommager tout le système de production ce qui peut entraîner des pertes en vies humaines et des dommages sur le plan écologique et économique.

Pour faire face à ce problème, et afin de consolider son rang de leader, le groupe SONELGAZ constate qu'il est alors obligatoire de mettre en œuvre des techniques fiables de surveillance et de contrôle de ces systèmes afin de détecter de façon précise l'apparition de signes d'anomalies.

Aujourd'hui grâce au développement de l'électronique et de l'informatique, le problème de la communication entre les équipements d'automatisme est résolu via des solutions dites SCADA ou DCS.

Le but de ce travail est l'élaboration d'un système de supervision SCADA tel que API/IHM peuvent communiquer des données.

## **Problématiques et objectifs du projet**

L'absence d'une interface de supervision permettant un contrôle visuel par l'opérateur, notamment lorsqu'il s'agit de répondre aux événements anormaux, a rendu les tâches de l'opérateur fastidieuses et gourmandes en temps, et elle a empêchée l'historisation de certaines informations. Afin de pallier aux problèmes précédemment cités, la centrale électrique de LARBAA a initiée le présent projet. Ce projet a pour but d'introduire un système de supervision industrielle SCADA au niveau de l'unité réchauffeur gaz à bain marie de la centrale tout en conférant aux opérateurs un support fiable pour une meilleure prise de décision.

## **Planification et conduite du projet**

L'initiation de tout projet nécessite une phase de planification. Celle-ci permet de définir les tâches à réaliser, maîtriser les risques et rendre compte de l'état d'avancement du projet.

« Planifier optimise ainsi les chances de réussite d'un projet en améliorant la productivité grâce à une meilleure maîtrise de la qualité. » [1].

Pour garantir le bon déroulement du projet, tout en respectant les délais, nous avons élaboré une planification globale de conduite du projet. Le diagramme de la (Figure 1) décrit cette planification ainsi que l'ordonnancement prévu des phases du projet.

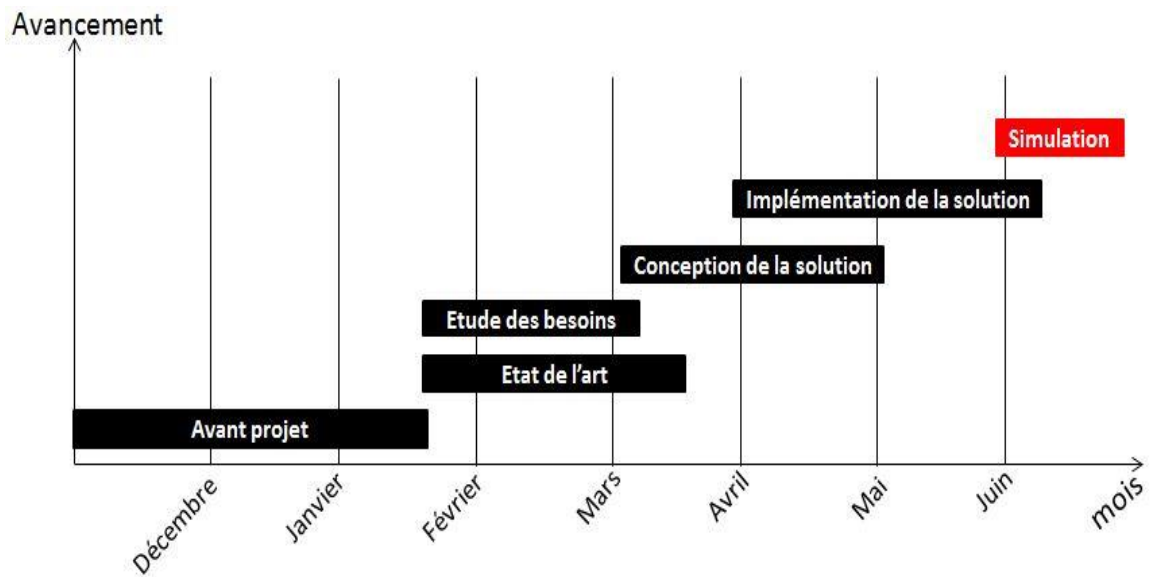


Figure 1 : Planification et conduite du projet.

Outre cette introduction générale, le présent mémoire comporte cinq chapitres et se présente comme suit :

- Le chapitre un sert à dévoiler le contexte général de notre projet. Ceci, en présentant l'organisme d'accueil, en abordant aussi avec plus de détails l'unité réchauffeur gaz à bain marie et le principe de son fonctionnement ainsi qu'une étude de besoin.
- Le chapitre deux consiste à clarifier les notions de base sur les systèmes de production automatisés.
- Le chapitre trois dresse un état de l'art en matière de la supervision et d'environnement SCADA, en fournissant les briques de base des systèmes de supervision dans le domaine industriel.
- Le chapitre quatre couvre l'étude de l'existant, la conception de la solution et sa mise en œuvre.
- Le chapitre cinq décrit la méthodologie de simulation adoptée.

Une conclusion générale est proposée afin de synthétiser le travail réalisé et de citer les perspectives du projet.



# Chapitre 1 Cadre général du projet

---

## 1.1 Introduction

Nous commençons dans ce premier chapitre par une mise en contexte de notre projet, en présentant l'organisme d'accueil qui, comme toute autre organisation, possède ses propres spécificités et se distingue des autres structures qui l'entourent. Il y a donc lieu de le présenter sous ses différents aspects historiques, organisationnels et fonctionnels afin d'avoir une idée précise sur la nature de ses activités, les relations, souvent complexes, qu'il peut entretenir avec son environnement aussi bien interne qu'externe.

Nous introduisons ensuite, la centrale turbine à gaz de LARBAA au sein de laquelle nous avons effectué le stage relatif au présent PFE. Ce dernier est inclus dans la formation master 2 SISII de l'université de Blida 1 afin de pouvoir nous immerger dans le monde professionnel. Nous concluons enfin par définir les problématiques liées à notre projet ainsi que nos objectifs.

## 1.2 Présentation de l'organisme d'accueil

SONELGAZ ou société nationale de l'électricité et du gaz créée en 1969, pour remplacer l'entité précédente Électricité et Gaz d'Algérie (EGA), est l'opérateur historique dans le domaine de la fourniture des énergies électriques et gazières en Algérie. Le groupe SONELGAZ ayant son siège social au « 02, Boulevard Krim Belkacem-Alger » œuvre principalement dans la production, le transport et la distribution de l'électricité ainsi que le transport et la distribution du gaz par canalisations [2].

### **1.2.1 Historique**

Durant son existence le groupe a connu des évolutions majeures [2] qui peuvent être résumées comme suit :

**1969 Création de SONELGAZ** : C'est l'ordonnance N°69-59 du 28 juillet 1969 qui dissout l'établissement public d'Electricité et Gaz d'Algérie (EGA), issu des lois françaises de nationalisation de 1947, et promulgue les statuts de la Société Nationale de l'Electricité et du Gaz (SONELGAZ). Dès sa mise en place, l'entreprise a pris le monopole de la vente d'énergie, l'installation et l'entretien d'appareils domestiques fonctionnant à l'électricité ou au gaz. L'ensemble des biens de l'ex-EGA lui a été légué.

**1977 Plan National d'Electrification** : Dès le milieu des années 70, l'Algérie s'est engagée dans un ambitieux plan national d'électrification qui avait comme objectif l'amélioration des conditions de vie des populations des campagnes tout en assurant un développement harmonieux de l'espace rural. Ainsi, SONELGAZ a largement contribué à la modernisation de l'économie.

**1983 Première restructuration** : naissance des filiales travaux. SONELGAZ s'est restructurée une première fois et a donné naissance à cinq entreprises travaux spécialisées : KAHRIF, KAHRAKIB, KANAGHAZ, INERGA et ETTERKIB. Ainsi qu'une entreprise de fabrication : AMC.

**1991 Un nouveau statut** : SONELGAZ change de nature juridique et devient Etablissement Public à caractère Industriel et Commercial (EPIC) en 1991. Le décret exécutif N° 95-280 du 17 septembre 1995 confirme la nature de SONELGAZ en tant qu'EPIC placé sous tutelle du Ministre chargé de l'énergie et des mines et doté de la personnalité morale tout en jouissant de l'autonomie financière.

**1998, création de filiales périphériques** : Le 1<sup>er</sup> janvier 1998, neuf filiales périphériques ont vu le jour.

**2002, La transformation en SPA** : promulgation de la loi 02 / 01 du 5 février 2002, la nouvelle loi relative à l'électricité et à la distribution du gaz par canalisations est venue supprimer le monopole exercé jusque-là par SONELGAZ, en ouvrant le secteur de

l'électricité et du gaz à la concurrence, sauf pour les activités de Transport qui ont un caractère de monopole naturel. Ensuite, En vertu du décret présidentiel n° 02-195 du 1<sup>er</sup> juin 2002 SONELGAZ est passé d'EPIC à une Société Par Actions (SPA) dont le capital est détenu par l'Etat. Ce statut lui donne la possibilité d'élargir ses activités à d'autres domaines relevant du secteur de l'énergie et aussi d'intervenir hors des frontières de l'Algérie.

**2004 - 2006 L'expansion du Groupe SONELGAZ :** En 2004, SONELGAZ devient une holding de sociétés. Une partie de ses entités en charge de ses métiers de base sont érigées en filiales assurant ces activités.

**En 2006, cinq (05) autres sociétés sont créées :** Il s'agit de l'Opérateur du Système Electrique (OS), de la SDA, SDC, SDE et SDO. Durant cette même année, les cinq (05) entreprises travaux ont réintégré le Groupe. Au-delà de cette évolution, assurer le service public reste la mission essentielle de SONELGAZ et constitue le fondement de sa culture d'entreprise.

**2007 - 2009 Parachèvement de la restructuration :** Réorganiser pour mieux progresser, l'enjeu étant la qualité du service rendu à la clientèle; un projet mûri au sein de l'entreprise, pour aboutir à la finalisation de son organisation en Groupe Industriel (maison mère/ filiales) constitué de trente- trois (33) filiales et de six (06) sociétés en participation directe. La dynamique d'investissement a concerné tous les métiers et toutes les zones géographiques.

**2011 - Amendement des statuts de SONELGAZ :** les statuts de SONELGAZ, adoptés en 2002, ont été révisés et approuvés par le conseil des Ministres, le lundi 2 mai 2011 et deviennent, de ce fait, en conformité avec le dispositif de la loi N°02 - 01 du 5 février 2002.

## **1.2.2 Organisation du groupe**

La SONELGAZ, et afin de se mettre en conformité avec la loi de février 2002, qui lui confère le statut de SPA, s'est organisée aujourd'hui en Groupe Industriel (Figure 2) constitué de 40 filiales et 7 sociétés en participation [3]. Chacune des sociétés ayant des missions et objectifs différents, il en ressort les principes d'organisation suivants :

- **Maison Mère** : chargée essentiellement de l'élaboration de la stratégie et de pilotage du Groupe, le contrôle des filiales, l'élaboration et la mise en œuvre de la politique financière et la définition de la politique de développement de la Ressource Humaine.
- **Filiales du Pôle des Industries Energétiques**: assurent la production, le transport et la distribution de l'électricité ainsi que le transport et la distribution du gaz par canalisations.
- **Filiales du Pôle Travaux et Services** : spécialisées dans le domaine de la réalisation des infrastructures énergétiques (engineering, montage industriel, réalisation de réseaux...).
- **Filiales métiers périphériques** : sont en charge d'activités annexes, telles que la maintenance d'équipements énergétiques, la distribution de matériel électrique et gazier, le transport et la manutention exceptionnels, ... etc.
- **De plus, SONELGAZ détient également des participations dans des sociétés,** dont le métier est en rapport avec le domaine de l'électricité et du gaz.

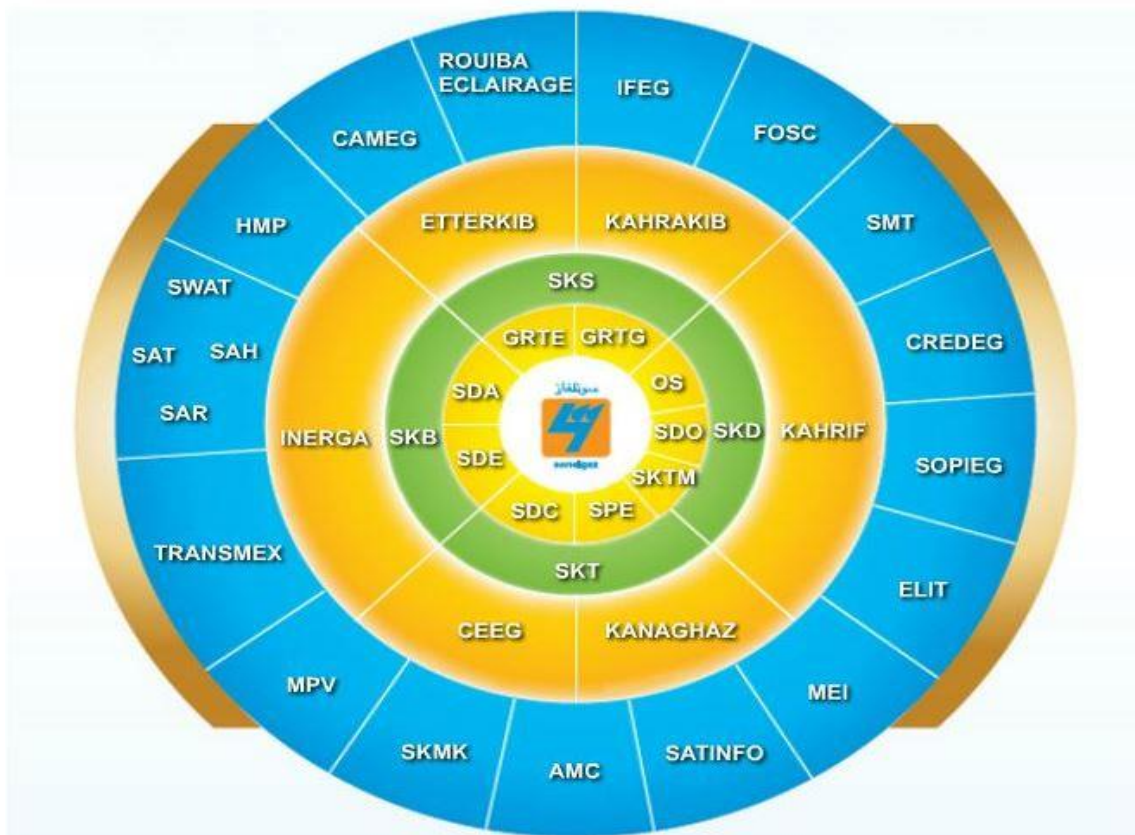


Figure 2: Les sociétés du groupe SONELGAZ [2].

### 1.2.3 Le groupe en chiffres

Le tableau suivant donne les chiffres relatifs à l'activité du groupe pour l'année 2014:

Critères		Chiffres
Chiffre d'affaires		351 539 MDA
Investissements		598 582 MDA
ventes	Electricité	183 674 MDA
	Gaz	32 241 MDA
	TPR & autres	8 456 MDA
Nombre de clients	Electricité	8 092 341
	Gaz	4 249 857
Production d'électricité	SPE	25 379 Gwh
	SKTM	683 Gwh
	Tiers	34 443 Gwh
Longueur réseau	Electricité	317 097 kms
	Gaz	91 079 kms
Ressources humaines	Permanents	60 691 agents
	Temporaires	23 719 agents
Couverture géographique	Electricité	99%
	Gaz	52%

Tableau 1: Le groupe SONELGAZ en chiffres [3].

### 1.2.4 Le métier de la production de l'électricité

L'un des métiers les plus importants du groupe, et dans lequel s'inscrit notre projet, est la production de l'énergie électrique. Ce métier, vu l'organisation du groupe, est assuré principalement par la société algérienne de production de l'électricité (SPE) qui dispose du plus grand parc de production avec 8400 MW installés, ce qui lui confère une position du premier opérateur sur le réseau interconnecté.

Afin de répondre à la demande de ces clients, le groupe SONELGAZ via sa filiale SPE et dans le cadre de la réalisation du plan d'urgence 2000 MW pour le renforcement des capacités de production a mis en service sept centrales électriques de type turbines à

gaz pour un coût d'investissement de près de 2 milliards de dollars US [2]. Parmi eux, la centrale turbine à gaz (TG) de LARBAA.

### 1.3 La centrale turbine à gaz de LARBAA

La centrale électrique de LARBAA d'une puissance de 560 MW bornes usine (4 x 140 MW conditions site) est constituée de quatre groupes turbines à gaz type V94.2 ANSALDO. Chaque turbine est du type mono arbre à cycle simple tournant à 3000tr/min, fonctionnant au gaz naturel, provenant de HASSI R'MEL, comme combustible principal et au gasoil comme combustible de secours [4].

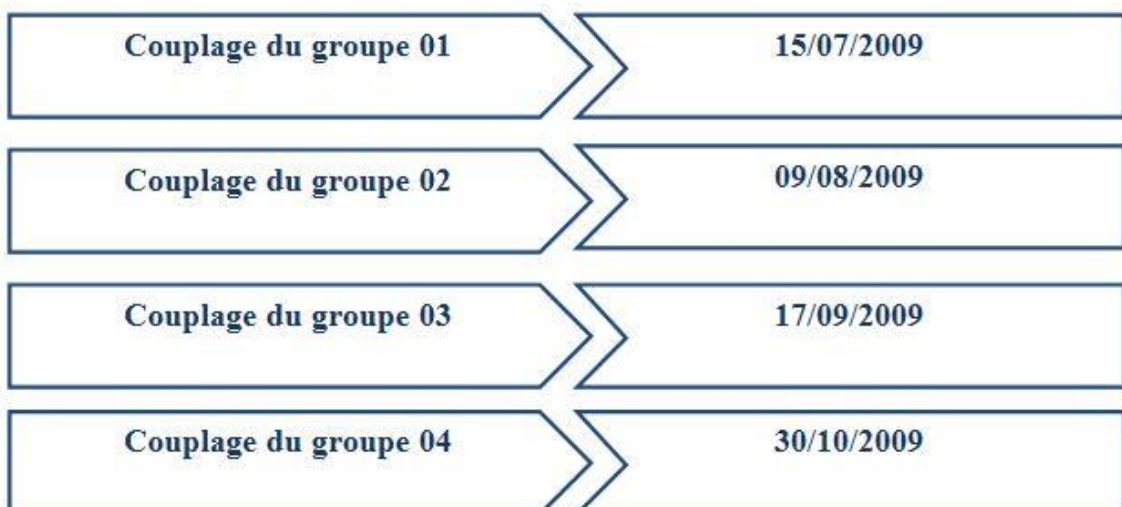
La centrale est dotée de moyens technologiques lui permettant de fonctionner dans le respect des normes en matière de préservation de l'environnement.

La centrale de LARBAA est destinée à répondre aux impératifs suivants :

- Faire face à la demande grandissante en énergie électrique dans la région.
- Assurer un appoint au réseau général interconnecté du pays.

#### 1.3.1 Dates phares

- **Marché:** Contrat N°2007/242/KDM/007.
- **Date de mise en vigueur :** 30 mai 2007.
- **Délai de réalisation global:** 30.5 mois.
- **Dates des premiers couplages :**



### 1.3.2 Fiche technique de la centrale

Le tableau ci-dessous résume les informations notables de la centrale électrique de LARBAA:

<b>Situation géographique</b>		<b>Belouadi, LARBAA wilaya de Blida</b>
<b>Superficie du site</b>		<b>5.4 hectares</b>
<b>Type de centrale</b>		<b>Cycle ouvert-turbine à gaz</b>
<b>Constructeur</b>		<b>Ansaldo Energia (Italie)</b>
<b>Montant total du projet (centrale électrique et évacuation)</b>		<b>32 110 millions de dinars</b>
<b>Investissement</b>	Partie devise	<b>254 millions euro</b>
	Partie dinars	<b>6614 millions dinars</b>
<b>Ressources humaines</b>	Personnel cadre	<b>40</b>
	Personnel maîtrise	<b>103</b>
	Personnel exécution	<b>29</b>

Tableau 2 : Fiche technique de la centrale [5].

### 1.3.3 Principes de fonctionnement de la centrale

**Parcours du gaz :** le gaz venant de HASSI R'MEL passe par plusieurs réseaux dans la centrale afin de répondre aux exigences de fonctionnement des turbines pour tous les régimes [4]. Ce qui nous amène à résumer le parcours du gaz dans la centrale par le schéma synoptique suivant :

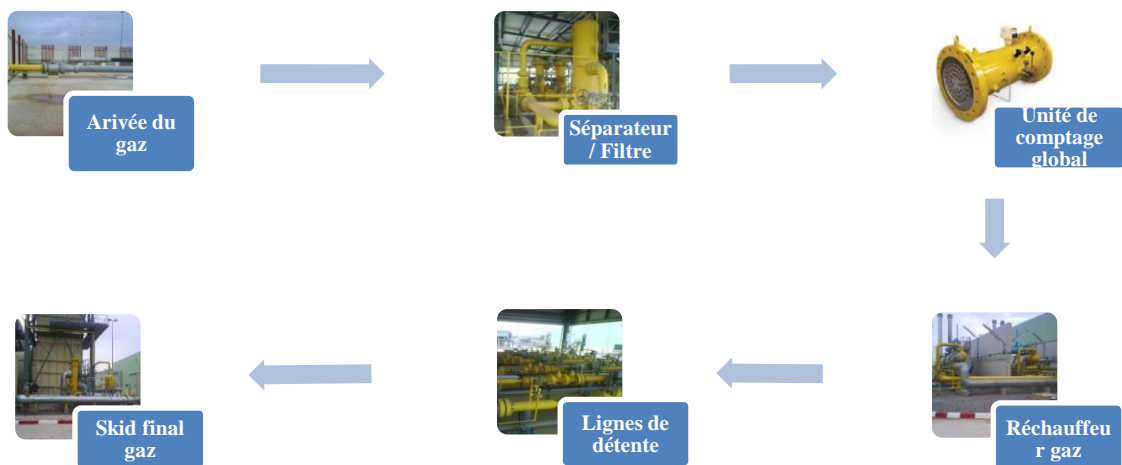


Figure 3:Schéma synoptique du parcours de gaz.

Le système du gaz naturel est composé des éléments principaux suivants :

- Une électrovanne pneumatique de sécurité avec by-pass.
- Un séparateur de liquides au 100% de la capacité totale avec by-pass, destiné à débarrasser le gaz des liquides en suspension qu'il est susceptible d'entraîner (eau, huiles légères ou lourdes,... etc).
- Deux filtres, de 100% chacun de la capacité totale, destinés à débarrasser le gaz des particules solides en suspension qu'il est susceptible d'entraîner.
- Une unité de comptage global, servant à mesurer la quantité de gaz qui va circulée dans la centrale.
- 04 Réchauffeurs gaz type " Bain marie "destinés à amener la température du gaz supérieure à 15 °C.
- 08 lignes de détente gaz pour réduire la pression à 23 bars. dont 04 lignes sont destinées pour GR1&2 et 04 autres lignes pour GR3&4.
- En aval des lignes de détente il y aura un départ indépendant pour chaque turbine à gaz qui arrive au **skid final d'unité**.

**Parcours de l'air :** L'air nécessaire à la combustion est soutiré d'un ensemble de modules qui permettent une filtration efficace avant l'entrée du compresseur [6]. Le système présent dans la (Figure 4) est du type autonettoyant par jet intermittents d'air comprimé à 7 bars. Les poussières sont rejetées à l'aide des ventilateurs extracteurs afin d'éviter le risque de pollution.



Figure 4: Système d'admission d'air turbine.



**Parcours de fuel :** le fuel léger est utilisé comme combustible de secours, de ce fait il est conditionné dans une station spécialisée (Figure 5.a) puis il y aura un départ indépendant pour chaque turbine à gaz au skid final de l'unité (Figure 5.b).

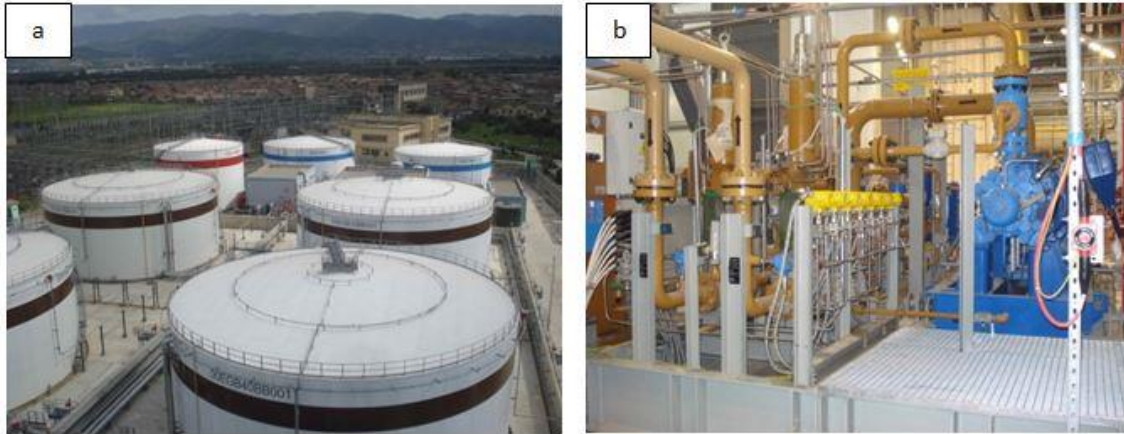


Figure 5: a Station fuel, b skid final fuel.

**Génération du courant électrique :** Dans un cycle simple de turbine à gaz (Figure 6), une base pression d'air est insufflée dans un compresseur où l'air sera comprimé à forte pression. Le combustible (gaze ou fuel), est injecté dans les chambres de combustion où il se mélange à l'air comprimé pour entretenir une combustion continue. Les gaz chauds se détendent en traversant la turbine, où l'énergie thermique et cinétique des gaz chauds fait tourner la turbine qui entraîne à son tour un alternateur. Les gaz de combustion s'échappent par la cheminée à travers un diffuseur. Grâce à l'énergie fournie par la turbine, l'alternateur produit un courant électrique alternatif.

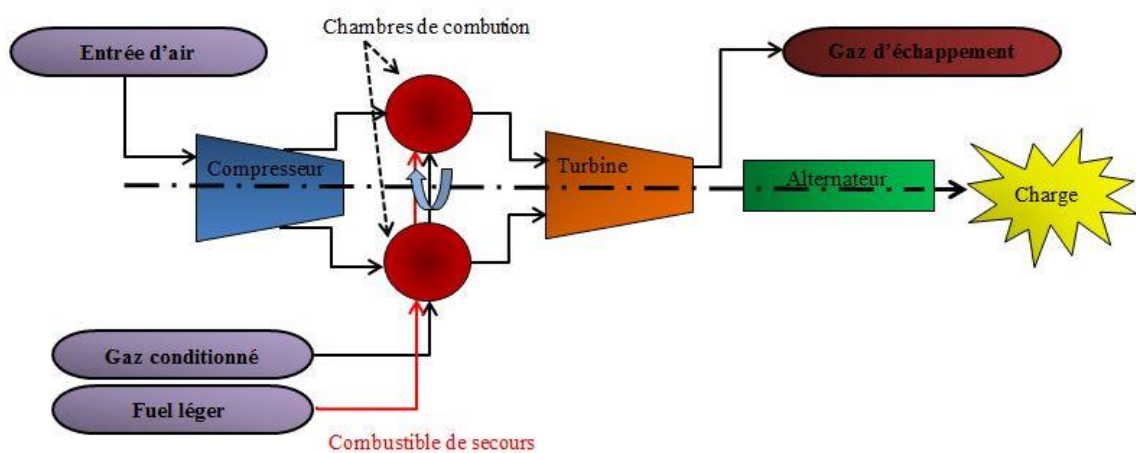


Figure 6: Cycle de fonctionnement d'une turbine à gaz (cycle simple).

**Evacuation d'énergie électrique générée :** Un transformateur (Figure 7.a) élève la tension du courant électrique produit par l'alternateur à 225 KV pour alimenter le poste électrique d'évacuation d'énergie. Le poste d'évacuation (Figure 7.b) est de type extérieur, il est constitué de quatre (04) travées groupes aboutissant sur 02 jeux de barres 220 KV. Le poste 220 KV est équipé de (quatre arrivées groupes, cinq Départs 220 KV « Alger Est1, Alger Est2, Métal Sider, Béni Merad, Hamr el Aïn » et un Coupleur de barres).



Figure 7: a Transformateur principal, b Station d'évacuation d'énergie électrique.

**Salle de commande :** Le démarrage des quatre groupes s'effectue à partir de la salle de commande commune, avec la possibilité de lancer chaque groupe à partir de sa propre salle de commande locale [5]. La supervision de l'ensemble des équipements est exécutée depuis cette salle.

**Système de contrôle commande :** Le Système de contrôle commande installé (GTCMPS) est basé sur le contrôle Harmonie symphonie d'ABB, il est conçu pour exécuter les fonctions de contrôle, d'analyse, de protection et de supervision de l'ensemble turbine- générateur. Les quatre groupes et la partie commune échangent les données nécessaires pour le contrôle et la surveillance à travers une boucle C-Net type redondant. L'exploitation est prévue soit au niveau de la station locale, soit au niveau de la salle de commande centrale.

### 1.3.4 Réchauffeur gaz à bain marie

Notre stage se déroule au sein de l'unité Réchauffeur gaz à bain marie. Ce département a pour mission de faire :

- Réchauffer le gaz en évitant le givrage.
- Augmenter sa température afin de la faire diminuer dans le compartiment du réseau de détente car à l'entrée de ce dernier une pression de [60-70] bar est enregistrée alors que celle-ci doit être ramenée à 23.5 bar. Nous pouvons justifier la relation proportionnelle entre la pression et la température par la formule suivante :

$$PV = nRT$$

**P**:pression. **V**:volume. **n**:nombre de mole. **R**: constante des gaz parfait. **T**: température.

#### *a Fonctionnement du réchauffeur*

Dans les conditions normales de marche, un seul réchauffeur fonctionnera et l'autre sera en attente automatique [5].

Le gaz combustible passe dans un serpentin (Figure 8), immergé dans l'eau potable d'un réservoir sous pression atmosphérique horizontal.

La chaleur est transférée du tube de chaudière à l'eau puis de l'eau au gaz combustible. Le gaz de combustion pour les réchauffeurs est pris en amont des réchauffeurs, réchauffé et ensuite envoyé aux brûleurs (Figure 9) à travers le robinet de contrôle de la pression automatique. L'alimentation du gaz combustible à chaque brûleur se compose d'une ligne pour alimenter le guide avec un bloc double et robinet de purge de sécurité, et quatre lignes pour alimenter le brûleur principal protégé par un bloc et des robinets de purge.

Tout le système de gaz combustible est équipé d'un interrupteur de pression en cas de déclenchement d'alarme très haute concernant les deux brûleurs du réchauffeur en question. La fumée de la combustion, du brûleur, passe dans le tube de chaudière jusqu'à une colonne d'évent.

Pour protéger tout le système, le réchauffeur est équipé d'une série de contrôles en cas de déclenchement d'alarmes des deux brûleurs du réchauffeur pertinente telle que :

- Température très haute de l'eau du bain marie.
- Niveau très bas de l'eau dans le bain marie.
- Brûleur flamme #1 et #2 non détectés.

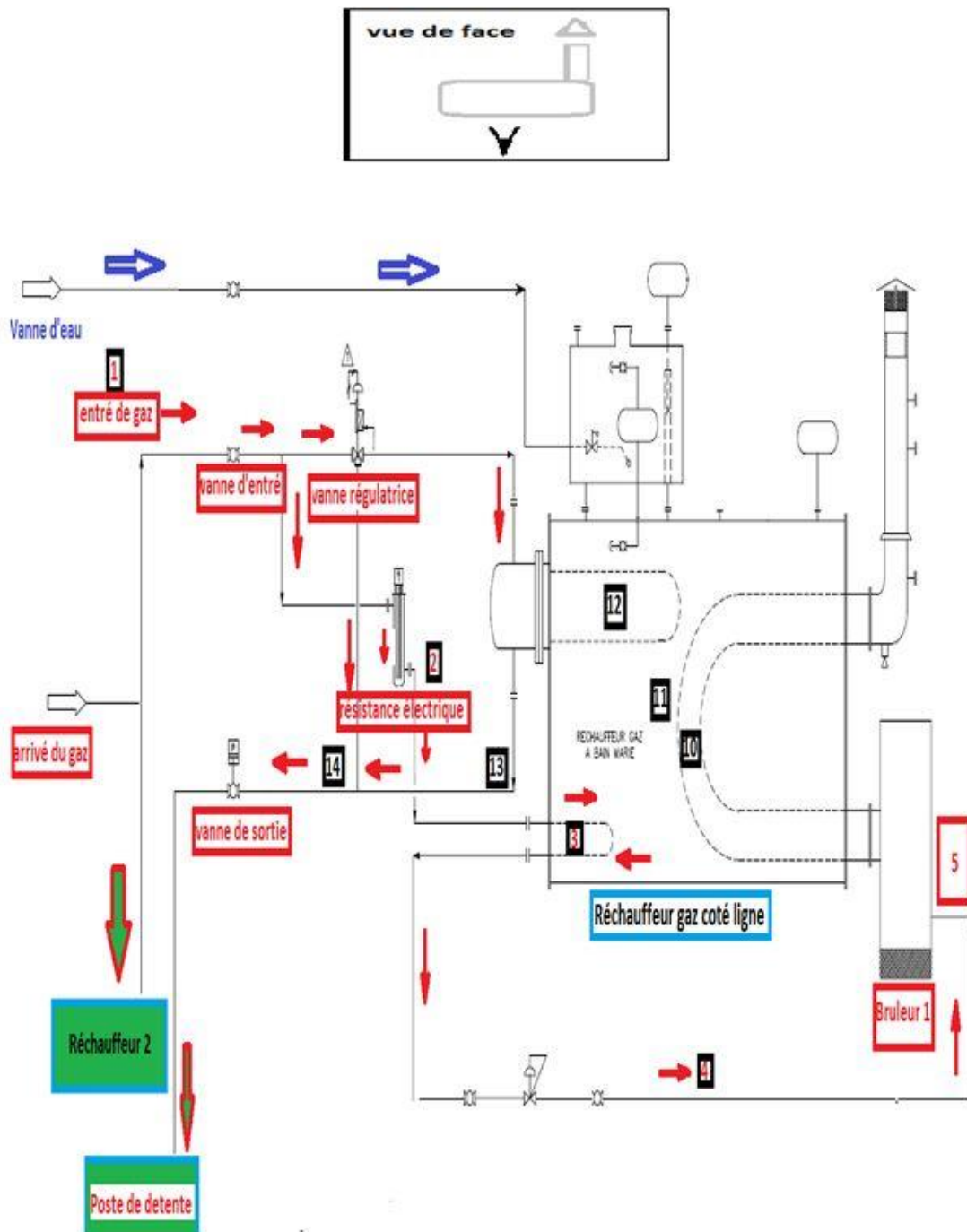


Figure 8: Réchauffeur gaz à bain marie (Vue de face) [6].

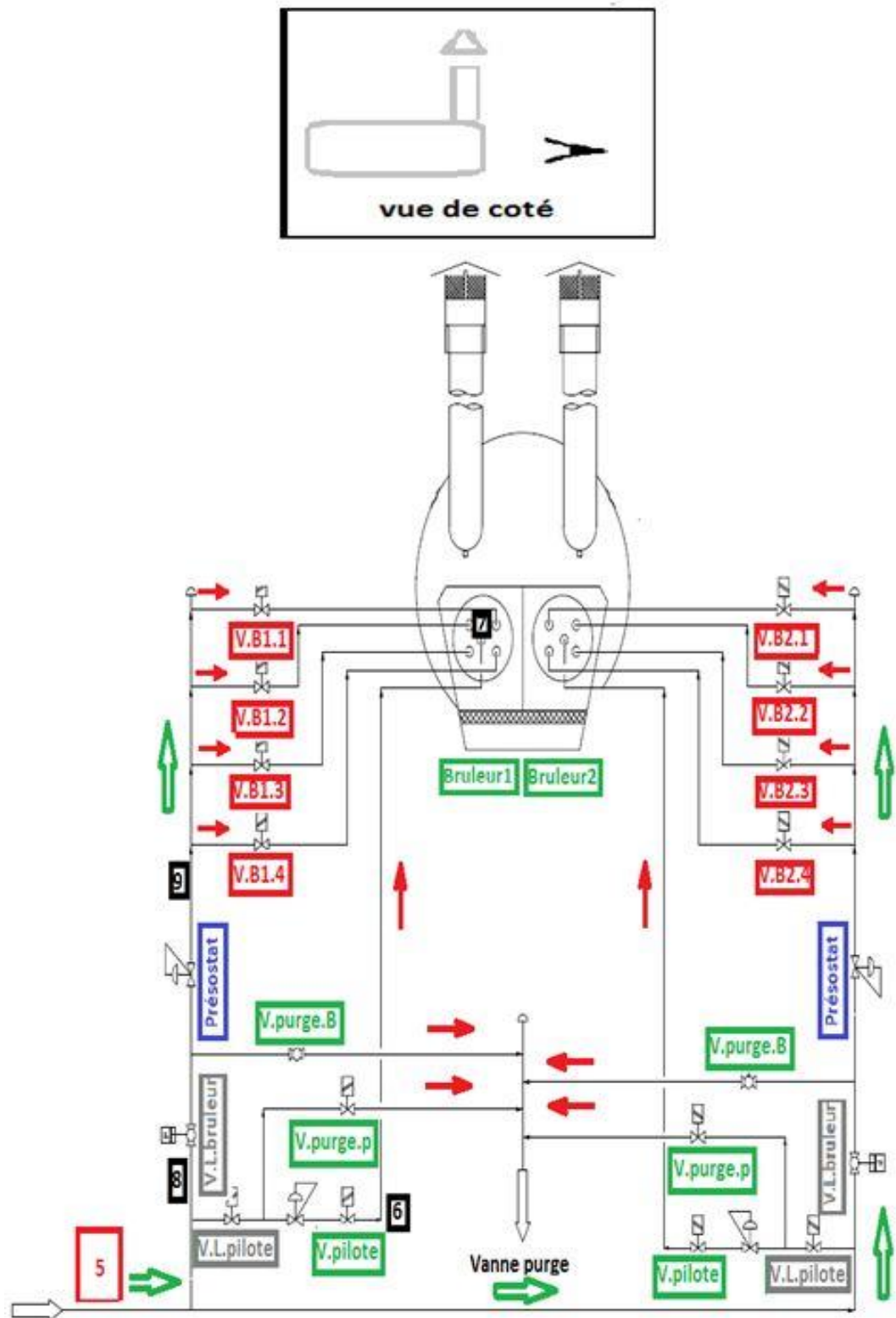


Figure 9: Réchauffeur gaz bain marie (Vue de coté) [6].

## ***b Les instruments de mesure***

Outre les équipements permettant le bon fonctionnement du réchauffeur gaz à bain marie, il existe d'autres instruments permettant de relier tout le système, de collecter des informations sur le fonctionnement du réchauffeur et d'assurer une meilleure gestion du processus entier. Nous allons citer les dispositifs principaux suivants :

- **Les capteurs** : ce sont des composants qui permettent de prélever une information sur une grandeur physique de la partie opérative et de la convertir en une information exploitable par la partie commande.
  - **Capteur de température PT100** : est une sonde de température régulièrement utilisée dans l'industrie pour la mesure de la température. Le PT100 (Figure 10.a) est constitué d'une résistance en Platine. La valeur initiale du Pt100 est de 100 ohms correspondant à une température de 0°C. La résistance de la sonde s'élève en fonction de l'augmentation de la température ambiante [4].
  - **Capteur de pression** : La prise de pression est enregistrée à travers d'une interface spécifique compatible avec les capteurs 4 à 20 mA. Ce module s'adapte à toute variété de capteurs délivrant une boucle de courant (max 25 mA) et alimentant cette dernière en énergie (Figure 10.b).
- **Actionneurs** : Ce sont des organes mécaniques ou électriques, de la partie opérative, qui agissent sur le fonctionnement d'une machine ou d'un système en utilisant un ordre provenant de la partie commande.
  - **Électrovanne** : C'est un dispositif commandé électriquement permettant le passage d'un fluide dans une canalisation (Figure 10.c).
  - **Vanne régulatrice** : La vanne principale de régulation gaz (Figure 10.d) est de type 3 voies utilisées comme une vanne de dérivé gaz froid et chaud. Elle est dotée d'un positionneur qui sert à convertir le signal de 4 à 20 mA fournie par l'automate a un signal pneumatique de 11 à 23 Psi [4].



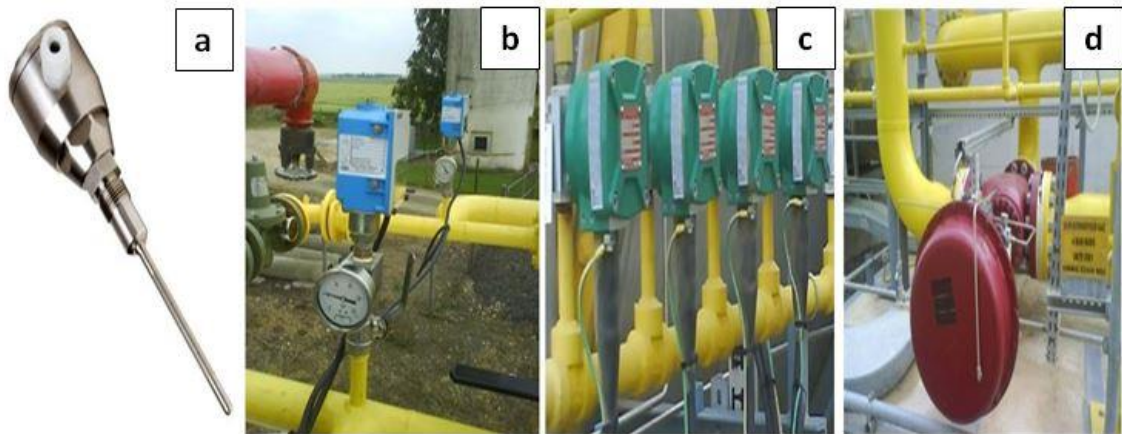


Figure 10: Dispositifs de l'unité: a PT100, b capteur de pression, c électrovanne, d vanne régulatrice.

## 1.4 Etude des besoins et présentation du sujet

Lors de notre étude de l'existant, nous avons pu constater que les opérateurs, de l'unité réchauffeur gaz à bain marie, font la supervision du système contrôle commande installé dans l'unité d'une manière manuelle.

Cette étude nous a permis d'avoir une vision générale de la procédure de contrôle. Elle fait aussi ressortir les insuffisances de la procédure actuelle en soulignant les points faibles suivants :

- Manque d'une vue d'ensemble de l'évolution du processus pour le suivi des taux de température et de pression.
- Les opérateurs trouvent des difficultés pour détecter le type d'alarmes s'il y aura lieu.
- Eventuellement, les ingénieurs ont besoin d'avoir certaines informations anciennes.
- Les opérateurs devront être en mesure de suivre l'évolution des sorties dans le temps et selon différents critères.
- La procédure manuelle est fastidieuse et demande beaucoup de réflexe.

Afin de pallier aux insuffisances citées précédemment, le présent sujet nous a été communiqué par le responsable de la centrale turbine à gaz de LARBAA. Il consiste donc en l'étude, la conception, le développement d'une Interface Homme-Machine (IHM) sur ordinateur basée sur les principes des systèmes SCADA pour la supervision

des différents paramètres traités par l'API Allen Bradley SLC 5/03 existant dans l'unité réchauffeur gaz à bain marie.

Comme toute autre application, la nôtre doit être en mesure de répondre aux attentes des utilisateurs. Cela a été, évidemment, déduit lors d'une étude approfondie de leurs besoins et a abouti aux fonctionnalités suivantes :

- Représentation synoptique de l'unité réchauffeur gaz à bain marie en temps réel.
- Assure l'accès aux Entrées/Sorties de l'API (Temps réel/Historique).
- Détermination du mode de fonctionnement de l'API (Automatique/Manuel).
- Affichage et gestion des alarmes (Temps réel/Historique).
- Affichage de l'état du système (Marche/Arrêt).
- Enregistrement des données.
- Modifier certaines informations pour rétablir un fonctionnement normal.
- Protection du mode avancé par mot de passe.

## **1.5 Conclusion**

Avec ses décennies d'activité dans le domaine de l'énergie et une réputation qui dépasse les frontières du pays, le groupe SONELGAZ représente un acteur majeur et incontournable de l'économie nationale. Cette brève présentation nous a permis de connaître un peu plus le groupe SONELGAZ, notamment dans sa nouvelle configuration de holding industriel.

Par ailleurs, cette présentation nous a fait comprendre le processus de la production d'électricité au niveau de la centrale de LARBAA.

Ce PFE porte donc essentiellement sur la conception et le développement d'une IHM pour la supervision d'un API Allen-Bradley Rockwell installé au niveau de l'unité réchauffeur gaz à bain marie. Nous s'intéresserons alors à l'API en tant que matériel, et à son programme qui nous permet de le manipuler, mais essentiellement dans un but de trouver la solution désirée. Des fondements théoriques préalables s'imposent afin d'étudier le domaine de plus près et de repérer la procédure optimale, et c'est ce que nous avons envisagé pour le prochain chapitre.



# Chapitre 2 Les systèmes automatisés

---

## 2.1 Introduction

Après avoir introduit le contexte de notre projet de fin d'étude, nous entamons ce deuxième chapitre dont l'objectif est de présenter un certain nombre de concepts relatifs aux systèmes automatisés.

Simple ou complexes, ces systèmes font de plus en plus partie du milieu industriel. Comprendre leur fonctionnement va nous permettre de réaliser notre projet de la manière la plus juste possible.

## 2.2 Les systèmes automatisés

Un système de production est dit automatisé lorsqu'il peut gérer de manière autonome un cycle de travail préétabli après avoir reçu les consignes d'un opérateur [7]. Les systèmes automatisés utilisés dans le secteur industriel ont une structure de base identique (Figure 11). Ils sont constitués de deux parties reliées entre elles :

- **Partie opérative** : C'est la partie visible du système qui effectue les opérations [8], elle comporte les éléments du procédé c'est-à-dire :
  - Les actionneurs : sont le plus souvent des composants électroniques capables de produire un phénomène physique à partir de l'énergie qu'ils reçoivent.
  - Les capteurs : sont des éléments capables de détecter (avec ou sans contact) un phénomène physique dans leur environnement et de rendre compte de ce phénomène à la partie commande.
- **Partie Commande** : Cette partie gère selon une suite logique le déroulement ordonné des opérations à réaliser. Elle reçoit des informations en provenance

des capteurs de la partie opérative et les restitue en direction des pré-actionneurs et actionneurs. Elle est constituée essentiellement d'un API et éventuellement d'un système de supervision [9].

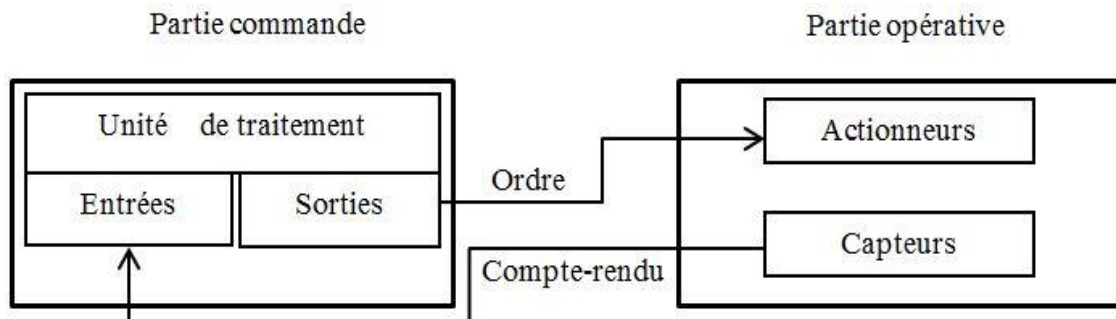


Figure 11 : Structure d'un système automatisé.

## 2.3 Objectifs de l'automatisation

L'automatisation consiste à « rendre automatique » les opérations qui exigeaient auparavant l'intervention humaine, elle permet de :

- Accroître la productivité du système (rentabilité, compétitivité).
- Améliorer la qualité du produit.
- Améliorer la flexibilité de production.
- Adaptation à des contextes particuliers :
  - Environnements hostiles pour l'homme (milieux toxiques, dangereux...etc.).
  - Tâches physiques ou intellectuelles pénibles pour l'homme (manipulation de lourdes charges, tâches répétitives parallélisées...etc.).
- Augmenter la sécurité.

## 2.4 Automate programmable industriel

Qui dit automatique, ou bien Systèmes de Production Automatisés (SPA), dit forcément API. Un automate programmable industriel (API) en anglais Programmable Logic Controller (PLC), est un dispositif électronique programmable destiné à la commande de processus industriels par un traitement séquentiel. Il envoie des ordres vers les pré-actionneurs « partie opérative (PO) côté actionneur » à partir de données

d'entrées capteurs « partie commande (PC) côté capteur », de consignes et d'un programme informatique [10].

## 2.5 ARCHITECTURE

Les automates programmables peuvent être de type compact ou modulaire [11]:

- **De type compact :** Il intègre le processeur, l'alimentation, les entrées et les sorties (Figure 12). Selon les modèles et les fabricants, les API compacts pourront réaliser certaines fonctions supplémentaires (comptage rapide, E/S analogiques ...etc.) et recevoir des extensions en nombre limité. Ces automates, de fonctionnement simple sont généralement destinés à la commande de petits automatismes.



Figure 12 : Automate compact.

- **De type modulaire :** le processeur, l'alimentation, et les interfaces d'entrées/sorties résident dans des unités (modules) et sont fixés sur un ou plusieurs racks contenant le fond de panier (Figure 13).



Figure 13 : Automate modulaire.

## 2.6 Constitution d'un API

L'API est structuré autour d'une unité de calcul ou processeur (CPU), d'une alimentation par des sources de tension alternative (AC) ou continue (DC), et de modules dépendant des besoins de l'application [12], tels que:

- Des cartes d'entrées - sorties numériques « tout ou rien (TOR)» ou analogiques pour des signaux à évolution continue.
  - Cartes d'entrées pour brancher des capteurs, boutons poussoirs, etc.
  - Cartes de sorties pour brancher des actionneurs, voyants, vannes, etc.
- Des modules de communication obéissant à divers protocoles pour dialoguer avec d'autres automates, des E/S déportées, des supervisions ou autres interfaces homme-machine (IHM), etc.
- Des modules spécifiques aux métiers, tels que comptage rapide, pesage, etc.
- Des modules d'interface pour la commande de mouvement, dits modules Motion, tels que démarreurs progressifs, variateurs de vitesse, commande d'axes.

## 2.7 Fonctionnement

L'automate programmable reçoit les informations relatives à l'état du système et puis commande les pré-actionneurs suivant le programme inscrit dans sa mémoire (Figure 14).

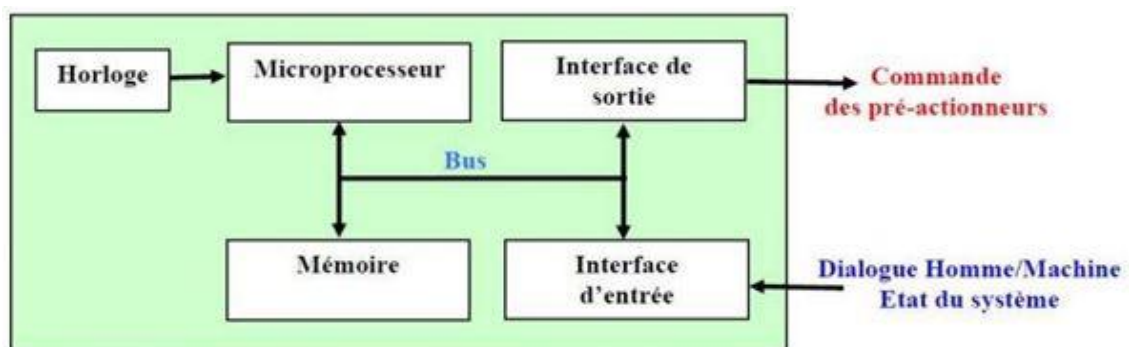


Figure 14 : Fonctionnement d'un API.

**Le microprocesseur** : réalise toutes les fonctions logiques et, ou, les fonctions de temporisation, de comptage, de calcul...etc. à partir d'un programme contenu dans sa

mémoire. Il est connecté aux autres éléments (mémoire et interface E/S) par des liaisons parallèles appelées BUS qui véhiculent les informations sous forme binaire.

**La zone mémoires** : permet de recevoir les informations issues des capteurs d'entrées, d'acquérir les informations générées par le processeur et destinées à la commande des sorties (valeur des compteurs, des temporisations, ...etc.) et de conserver le programme du processus.

**Les interfaces d'entrées/sorties :**

- Les interfaces d'entrées sont destinées à recevoir l'information en provenance des éléments de détection « capteurs, pupitre opérateur » et à traiter le signal en le mettant en forme « L'électronique de traitement de l'API fonctionne sous une tension de 5V. Le capteur fournissant une tension de 24V, la carte des entrées doit adapter le signal. »
- Les interfaces de sorties sont destinées à commander les actionneurs (relais, électrovannes, ...etc.) et les éléments de signalisation du système et à adapter les niveaux de tensions de la PC à celle de la PO du système en garantissant une isolation galvanique entre ces dernières.

**Cartes d'entrées / sorties analogiques** : permettent de réaliser l'acquisition d'un signal analogique et sa conversion numérique (CAN) indispensable pour assurer un traitement par le microprocesseur. La fonction inverse, pour obtenir une sortie analogique, est également réalisée.

**Les BUS de communication interne** : permet la communication de l'ensemble des blocs de l'automate et les éventuelles extensions.

Le cycle de fonctionnement de l'API (Figure 15) est décrit ci-dessous :

- **Traitement interne** : L'automate effectue des opérations de contrôle et met à jour certains paramètres systèmes (détection des passages en RUN / STOP, mises à jour des valeurs de l'horodateur, ...).
- **Lecture des entrées** : L'automate lit les entrées (de façon synchrone) et les recopie dans la mémoire image des entrées.

- **Exécution du programme** : L'automate exécute le programme instruction par instruction et écrit les sorties dans la mémoire image des sorties.
- **Ecriture des sorties** : L'automate bascule les différentes sorties (de façon synchrone) aux positions définies dans la mémoire image des sorties.

Ces quatre opérations sont effectuées continuellement par l'automate, ce dernier a un fonctionnement cyclique (Figure 15).

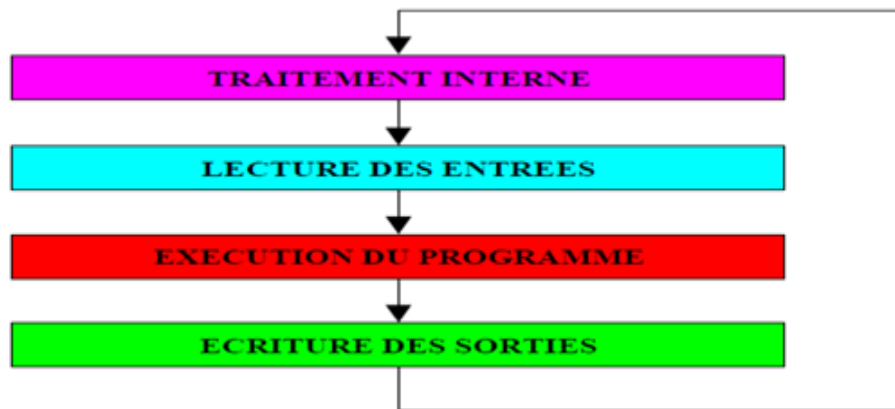


Figure 15 : Cycle de fonctionnement d'un API [11].

L'ensemble de ces quatre opérations est appelé scrutation. Ainsi, le temps de scrutation est le temps mis par l'automate pour traiter la même partie de programme (Figure 16). Ce temps est de l'ordre de la dizaine de millisecondes pour les applications standards. Il varie selon la taille du programme, la complexité des calculs, le nombre d'entrées/sorties, la puissance de l'API et les besoins du procédé piloté. Il est protégé par un chien de garde, au cas par exemple où l'algorithme exécuterait indéfiniment une même boucle de programme.

Le temps de réponse total (TRT) est le temps qui s'écoule entre le changement d'état d'une entrée et le changement d'état de la sortie correspondante (Figure 16). Ce dernier est au plus égal à deux fois le temps de scrutation (sans traitement particulier).

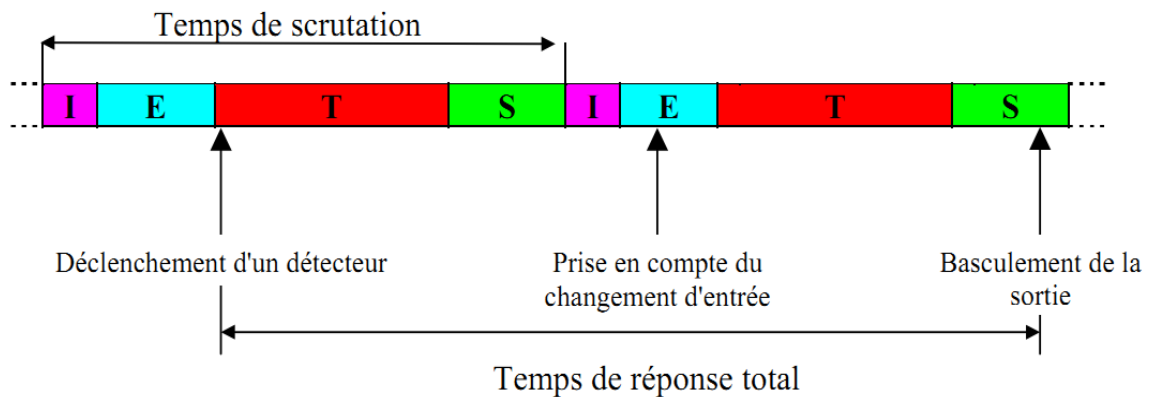


Figure 16: Temps de scrutation vs Temps de réponse [11].

## 2.8 Caractéristiques des API

Les API se caractérisent par rapport aux ordinateurs par :

1. leur robustesse : conçus pour pouvoir travailler en milieu hostile, ils utilisent des circuits durcis et sont prévus pour résister aux vibrations, aux températures des ateliers etc.
2. leur réactivité aux indications fournies par les capteurs (dispositifs anticollisions, alarmes diverses);
3. leur facilité de maintenance (bien que les ordinateurs industriels atteignent également un très bon degré de fiabilité). Les modules des API peuvent être changés très facilement et leur le redémarrage est très rapide.

## 2.9 PROGRAMMATION

### 2.9.1 Les langages de programmation :

Nous programmons chaque automate soit via une console de programmation propriétaire ou par un ordinateur équipé d'un logiciel de constructeur spécifique. Toutefois, puisque chaque constructeur a eu tendance à développer ses propres versions, une norme internationale a été établie pour toutes les méthodes de programmation des API. Cette norme, publiée en 1993 par la Commission électrotechnique internationale, est désignée sous la référence CEI 61131-3 a normalisée cinq langages [13]:

- **Liste d'instructions « Instruction List (IL) »** : C'est un langage textuel de bas niveau, à une instruction par ligne, de même nature que l'assembleur (Figure 17). Très peu utilisé par les automaticiens.

```

! %L0 : LD          %I1.0
        ANDN       %M12
        OR (      %TM4.Q
        AND       %M17
        )
        AND       %I1.7
        ST        %Q2.5
! %L5 : LD          %I1.10
        ANDN       %Q2.9
        ANDN       %M27
        IN        %TM0
        LD        %TM0.Q
        AND       %M25
        AND       %M000:XS
        [%M0015 := %M0018+S00]

```

Figure 17 : List d'instruction.

- **Langage littéral structuré « Structured Text (ST) »** : C'est un langage textuel de haut niveau (Figure 18). Il est de même nature que le langage informatique Pascal, il utilise les fonctions comme (if, then, else ...). Peu utilisé par les automaticiens.

```

IF %M0 THEN
  FOR %MW99 := 0 TO 31 DO
    IF %M000 [%MW99] > 0 THEN
      %MW10 := %M000 [%MW99];
      %MW11 := %MW99;
      %M1 := TRUE;
      EXIT;          (*Sortie de la boucle FOR*)
    ELSE
      %M1 := FALSE;
    END_IF;
  END_FOR;
ELSE
  %M1 := FALSE;
END_IF;

```

Figure 18 : Langage littéral structuré.

- **Les graphes de fonction séquentielle « Sequential function chart (SFC)»** : C'est un langage graphique de programmation des API. Ce langage est une interprétation assez libre et plus permissive du grafcet dont il est inspiré (Figure 19). le grafcet est dédié à la spécification, alors que SFC est plus appliqué à la programmation.



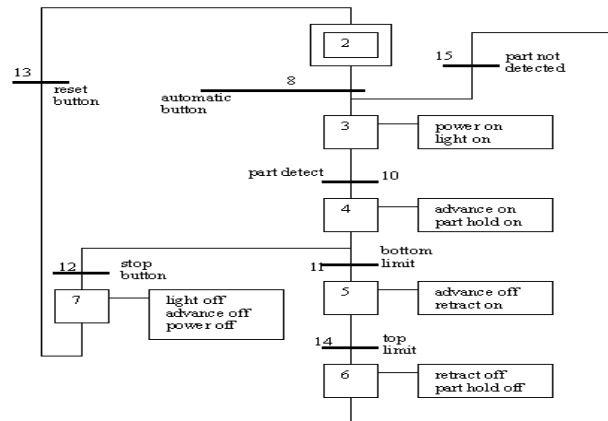


Figure 19 : Les graphes de fonction séquentielle.

- **Langage à contacts (LD : Ladder diagram)** : langage graphique développé pour les électriciens ou ayant une connaissance de la schématique électrique sans pour autant apprendre un langage spécifique (Figure 20). Il permet d'écrire un programme de contrôle en utilisant des symboles tels que : contacts, relais et blocs fonctionnels et s'organise en réseaux (labels). C'est le langage le plus utilisée pour les PLC.

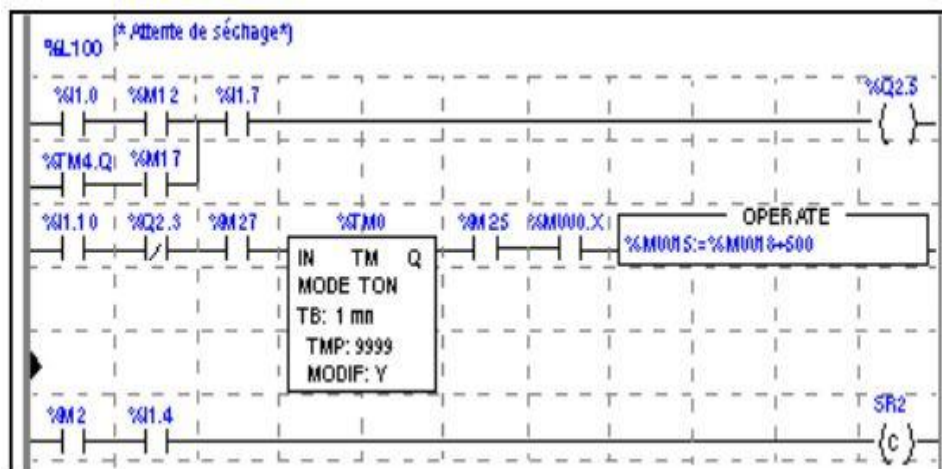


Figure 20 : Langage à contacts.

- **Blocs Fonctionnels (FBD : Function Bloc Diagram)** : C'est un langage graphique où les fonctions sont représentées par des rectangles avec les entrées à gauche et les sorties à droites (Figure 21). Les blocs sont programmés (bibliothèque) ou programmables. Il est utilisé par les automaticiens.

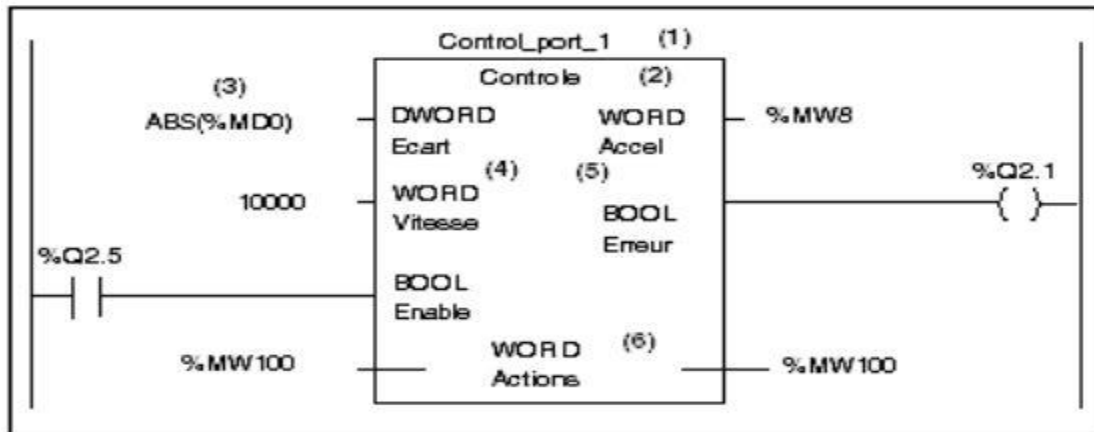


Figure 21 : Blocs Fonctionnels.

## 2.9.2 Les outils de programmation

Après avoir été établie, le programme doit être écrit dans la mémoire de l'API et visualisé pour le vérifier. Le moyen utilisé est appelé "outil de programmation" et se présente sous forme d'un élément connectable ou incorporé à l'automate.

## 2.10 Sécurité de l'API

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

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

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

Aussi, l'automate fait l'objet de nombreuses dispositions pour assurer la sécurité [11]:

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

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

**Mode RUN/STOP** : Seul un technicien peut mettre en marche ou arrêter un automate et la remise en marche se fait par une procédure d'initialisation (programmée). En mode RUN, l'API exécute le programme. En mode STOP, l'API ne peut pas exécuter le programme.

**Contrôles cycliques** : Procédures d'autocontrôle des mémoires, de l'horloge, de la batterie, des tensions d'alimentation et des entrées / sorties. Enclenchement d'une procédure d'alarme en cas de dépassement du temps de scrutation (réglé par l'utilisateur).

**Visualisation** : nous pouvons surveiller l'évolution des entrées / sorties via une IHM.

## 2.11 Critères de choix d'un automate

Le choix d'un automate programmable est en premier lieu le choix d'une société ou d'un groupe, il est basé généralement sur :

- Disponibilité des outils de développement et facilité d'accès à ces outils.
- Prix relativement bas par rapport aux autres fournisseurs.
- La qualité du service après-vente.
- Les capacités de traitement du processeur (vitesse, données, opérations...).
- Le type et le nombre des entrées/sorties nécessaires.
- La communication avec d'autres systèmes. L'automate doit pouvoir communiquer avec les autres systèmes de commande si nécessaire pour l'application (API, supervision...) et offrir des possibilités de communication avec des standards normalisés (Ethernet, Modbus...).

## 2.12 Les automates et la communication

L'évolution rapide de la technologie électronique et le développement des systèmes automatisés avec des contraintes fonctionnelles de plus en plus complexes et critiques (recevoir les données liées à une application le plus rapidement possible, consulter ou

contrôler les paramètres d'une application à distance) et non fonctionnelles (la recherche de la baisse des coûts, temps de mise sur le marché, . . .) a conduit à l'apparition de nouvelles technologies de câblage et de communications entre les divers constituants des automatismes [8].

## Les bus de terrain

Le bus de terrain est un standard de communication qui permet la connexion des équipements à partir du point de connexion intégré au processeur (Figure 22). Il utilise des blocs d'E / S déportés pour diminuer les coûts de câblage.

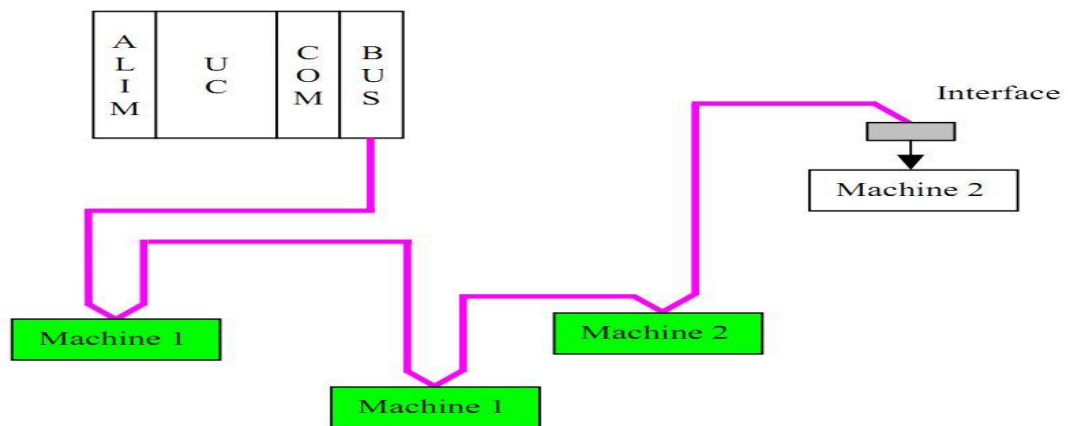


Figure 22 : Bus de terrain [11].

Pour assurer le "multiplexage" de toutes les informations en provenance des capteurs /pré-actionneurs plusieurs protocoles de communication ont été développés et des standards sont apparus (normalisés ou standards de fait).

L'émergence de ces nouvelles technologies a conduit à la définition de plusieurs catégories de réseaux locaux industriels (pyramide CIM ci-dessous) :

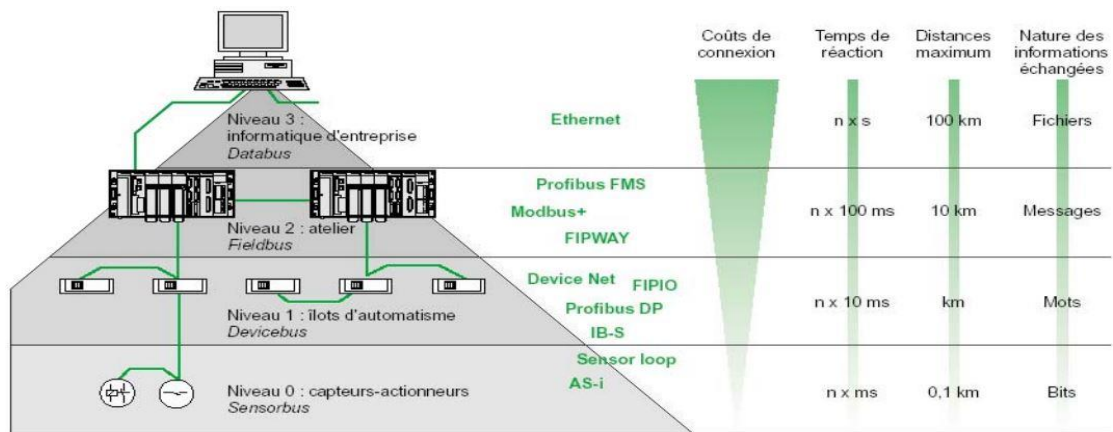


Figure 23 : pyramide CIM.

La nécessité de communication entre cellules (communication entre automates) a permis de voir apparaître de nombreuses normes de communication (Profibus, Fip ...). Le déterminisme nécessaire pour certaines applications conduit à l'utilisation de réseaux Maître / Esclave. Au niveau de l'entreprise, le temps n'est plus critique et la norme Ethernet a pu se développer rapidement, permettant ainsi la visualisation et la commande des processus via le réseau Internet.

## 2.13 Conclusion

Les automates sont souvent reliés à un pupitre opérateur, une interface graphique (écran d'affichage ou écran tactile) ou un PC. Dans ce dernier cas, on parle de supervision.

En général, si API et PC coexistent dans un atelier, les API fonctionnent au plus près des processus physiques et prennent en charge les questions de sécurité, les PC s'occupant plutôt de supervision et des rapports extérieurs. Les PC peuvent ainsi fixer au mieux les consignes aux API, qui donnent les ordres détaillés, traitent les urgences, et rendent compte de l'état des processus.

L'absence d'Interface Homme-machine (IHM) permanente, pour visualiser l'action du programme sur la partie opérative et son fonctionnement, dans l'unité réchauffeur gaz à bain marie de LARBAA nous conduit à réaliser un système de supervision pour accomplir notre projet, pour cela il faut passer par une étude sur les systèmes de supervision ainsi que les IHM.

# Chapitre 3 La supervision des systèmes

## automatisés

---

### 3.1 Introduction

Malgré les progrès technologiques bouleversants dans le domaine industriel, qui ont permis la réalisation des API aux fonctionnalités plus nombreuses et plus complexes. Aujourd'hui, l'inconvénient majeur de certains API, que nous avons présenté dans le chapitre précédent, est l'absence d'une interface de supervision permettant un contrôle visuel par l'opérateur. Alors que, la supervision de processus industriel est une tâche très importante, notamment lorsqu'il s'agit de répondre aux événements anormaux.

Par conséquent, une bonne maîtrise des techniques de supervision s'impose pour répondre aux besoins de la qualité, de la quantité de la production et de la concurrence. Ceux-ci en diminuant les risques de pannes qui peuvent survenir sur le fonctionnement de l'installation, et en augmentant la sécurité du personnel et de l'environnement.

C'est pourquoi, nous allons évoquer dans un premier temps les principes de la supervision des procédés industriels et pourquoi une telle technique s'avère nécessaire dans ces circonstances, Ensuite, nous verrons les fonctions fondamentales et les techniques utilisées dans le domaine de la supervision ainsi que les aspects fondamentaux de la supervision dans un environnement SCADA. Finalement, nous allons présenter nos choix et les discutés.

### 3.2 La supervision

Le mot "supervision" se décompose en deux mots latins "supra" et "videre", ce qui veut dire "regarder d'en haut". Il s'en dégage deux notions fondamentales : la notion

de surveillance pour s'assurer que les tâches qui ont été assignées au collaborateur (supervisé) sont exécutées conformément aux normes et instructions mise en place et la position même de celui qui est chargé de cette surveillance [14].

Aujourd'hui, elle est utilisée dans tous les domaines pour le suivi et le contrôle d'un ensemble d'équipements depuis un poste informatique. La supervision est alors une forme évoluée de dialogue homme-machine, elle consiste à collecter des informations permettant aux utilisateurs de suivre, réagir et décider rapidement et d'apporter une vision temps réel sur les états de système depuis un site distant.

### 3.3 Supervision industrielle

En informatique industrielle, Les systèmes de supervision des procédés de fabrication (machines, ateliers, usines,...) offrent un ensemble de moyens de surveillance, de contrôle-commande évolué et de diagnostic qui marchent aussi bien en situation normale qu'anormale et répond aux préoccupations suivantes:

- **Technique** : pilotage de l'infrastructure et des machines, surveillance du réseau et mesure des performances.
- **Applicative** : surveillance des applications et des processus métiers [15].

Un logiciel de supervision industrielle fonctionne généralement sous un ordinateur en communication, via un réseau local ou distant industriel, avec un ou plusieurs équipements (API, ordinateur, carte spécialisée). Il peut être monoposte ou multiposte. Dans le cas d'une supervision multiposte, l'architecture peut être de type redondant, de type client/serveur voire les deux en même temps.

Un logiciel de supervision industrielle est composé d'un ensemble de pages (d'écrans), dont l'interface opérateur est présentée très souvent sous la forme d'un synoptique. Il permet l'acquisition de données (mesures, alarmes, retours d'état de fonctionnement et de paramètres de commande des processus) et assure aussi le rôle de gestionnaire d'alarmes d'événements déclenchés par des dépassements de seuils, d'enregistrement d'historique de défauts, de temps de fonctionnement et de recettes et de fabrication (Figure 24).

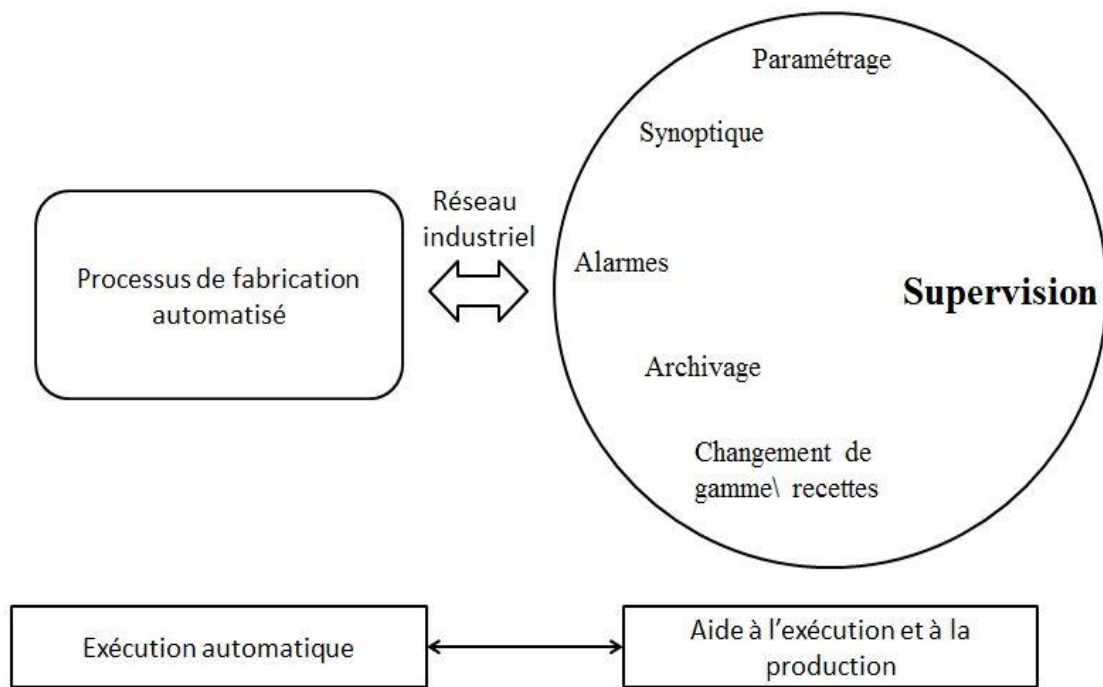


Figure 24: supervision industrielle [16].

### 3.4 La supervision dans la hiérarchie d'une entreprise

Dans la Figure ci-dessous, les différents niveaux de la structure de l'entreprise d'après le modèle CIM sont représentés sous forme d'une pyramide. Les activités des systèmes de supervision se situent au niveau 2 du modèle CIM. Ces activités assurent la communication entre les systèmes de contrôle\ commande API et les MES.

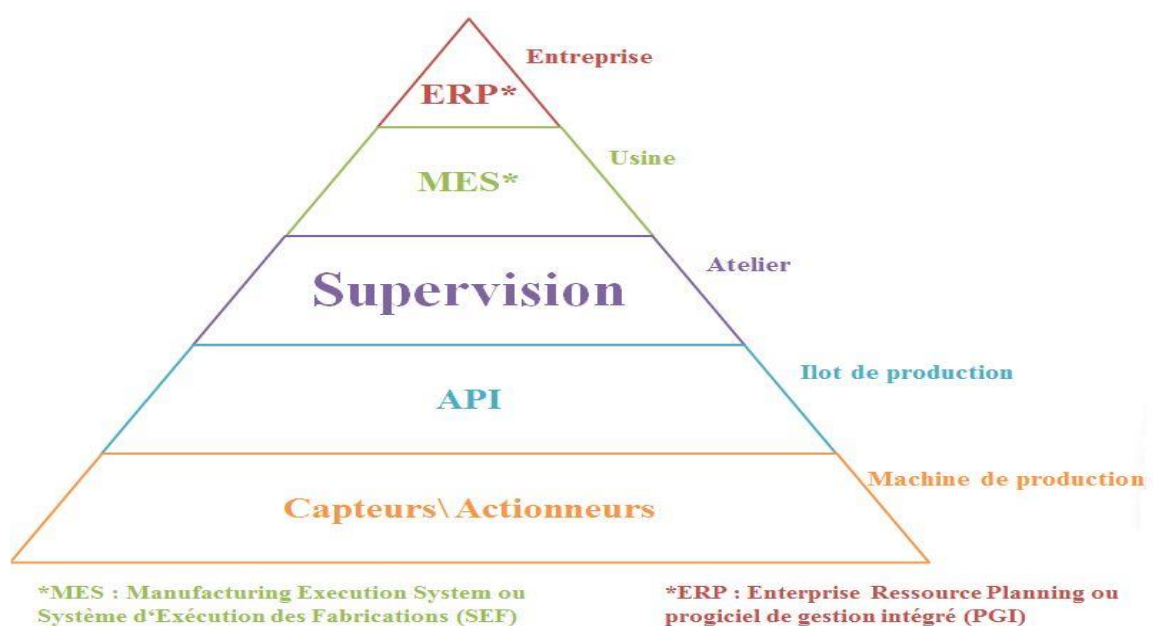


Figure 25 : Le modèle de référence CIM des processus industriels.



### 3.5 Objectifs et rôles de la supervision

La supervision a pour but de surveiller et contrôler l'exécution d'une opération effectuée à distance sans rentrer dans les détails de cette exécution. La supervision recouvre l'aspect fonctionnement normal et anormal :

- **en fonctionnement normal** : son rôle est surtout de prendre en temps réel les dernières décisions correspondant aux degrés de liberté exigés par la flexibilité décisionnelle.
- **en présence de défaillance** : la supervision va prendre toutes les décisions nécessaires pour le retour vers un fonctionnement normal. Après avoir déterminé un nouveau fonctionnement, il peut s'agir de choisir une solution curative, d'effectuer des ré-ordonnements "locaux", de prendre en compte la stratégie de surveillance de l'entreprise, de déclencher des procédures d'urgence [17].

De manière générale les objectifs de supervision sont :

- En ce qui concerne le superviseur, il s'agit de :
  - Se rendre compte de la manière dont les tâches sont exécutées.
  - Avoir une meilleure connaissance des conditions de travail du supervisé.
  - Identifier les difficultés et aider les maintenanciers à les résoudre.
  - Permettre de mieux apprécier le personnel et la qualité du travail qu'il exécute.
  - S'assurer de l'exécution correcte des tâches.
  - Motiver le personnel [14].
- En ce qui concerne le supervisé, il s'agit de :
  - Se sentir responsabilisé, valorisé, assisté et appartenant à une équipe.
  - Avoir la possibilité de discuter de ses problèmes avec le superviseur.
  - Etre formé pour accomplir correctement ses tâches [14].
- Au niveau du service, la supervision vise à :
  - Améliorer la qualité des services.
  - Accroître l'efficacité.
  - Corriger les erreurs à temps.

- Permettre une meilleure organisation.
- Améliorer l'accessibilité des services.
- Créer un climat de bonne entente, de franche collaboration et de bonne communication [14].

### 3.6 Fonction de la supervision

Nous pouvons tirer, de ce qu'il a été évoqué précédemment, les fonctions suivantes :

**Synoptique** : fonction essentielle de la supervision, fournit une représentation synthétique, dynamique et instantanée de l'ensemble des moyens de production de l'unité (Figure 26). Elle permet à l'opérateur d'interagir avec le processus et de visualiser le comportement du système.

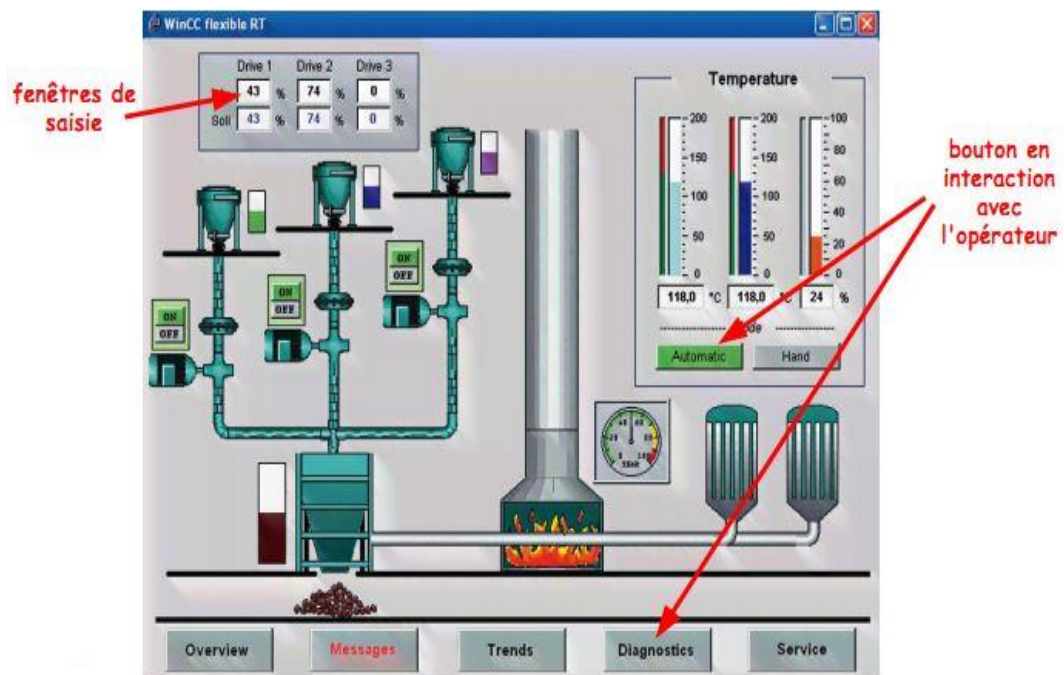


Figure 26 : Exemple d'une interface synoptique [16].

**Courbes**: donnent une représentation graphique de différentes données du processus sous forme d'un graphe (Figure 27). C'est un outil puissant d'analyse des variables historisées.

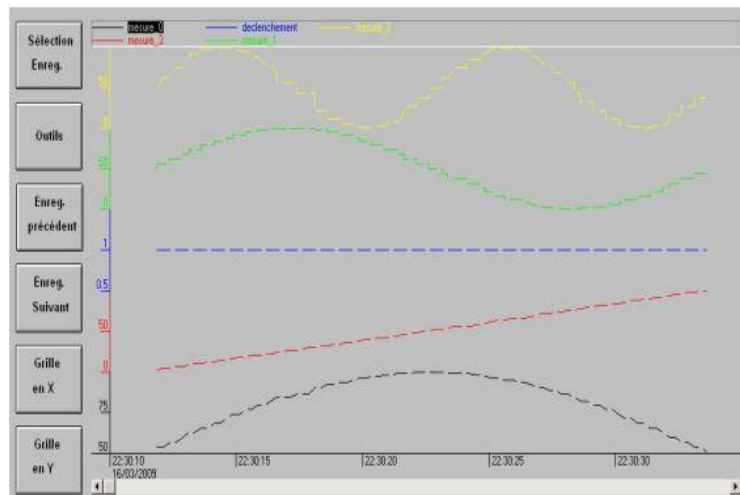


Figure 27 : Exemple d'une interface des courbes [16].

**Alarmes:** le système calcule en temps réel les conditions de déclenchement des alarmes, affiche l'ensemble des alarmes selon des règles de priorité (Figure 28), donne les outils de gestion depuis la prise en compte jusqu'à la résolution complète et assure l'enregistrement de toutes les étapes de traitement de l'alarme.

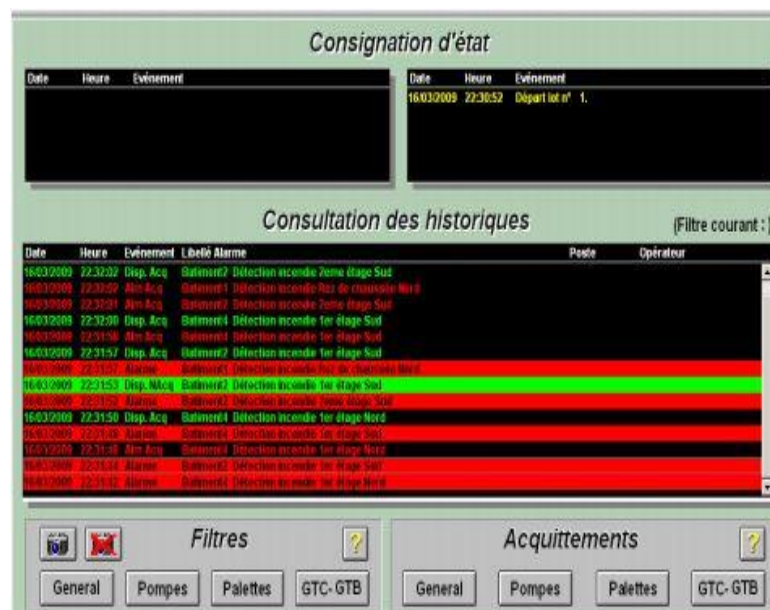


Figure 28 : exemple d'une interface de gestion des alarmes [16].

**Historisation du procédé:** fonction permettant la sauvegarde périodique de grandeurs et la sauvegarde d'événements horodatés (archivage sélectif), de plus, elle fournit les outils de recherche dans les données archivées et permet de garder une trace validée de données critiques (traçabilité de données de production) (Figure 29).

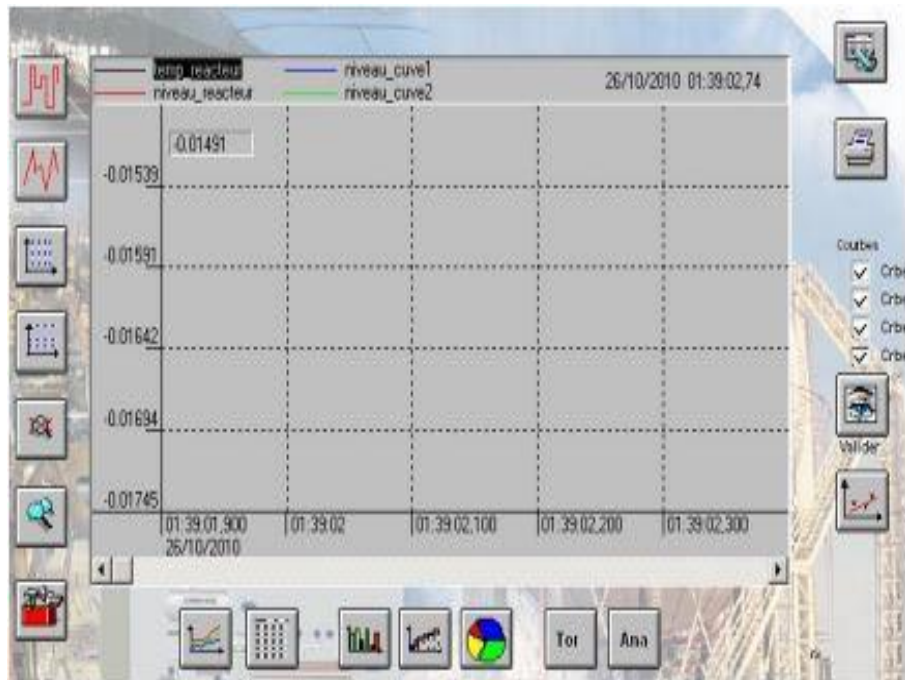


Figure 29 : exemple d'une interface d'historisation du procédé [16].

**Gestion des gammes de fabrication et recettes:** donne un outil de gestion des lots de fabrication (batches), gère les paramètres de réglage des machines pour chacun des lots (recettes) (Figure 30).

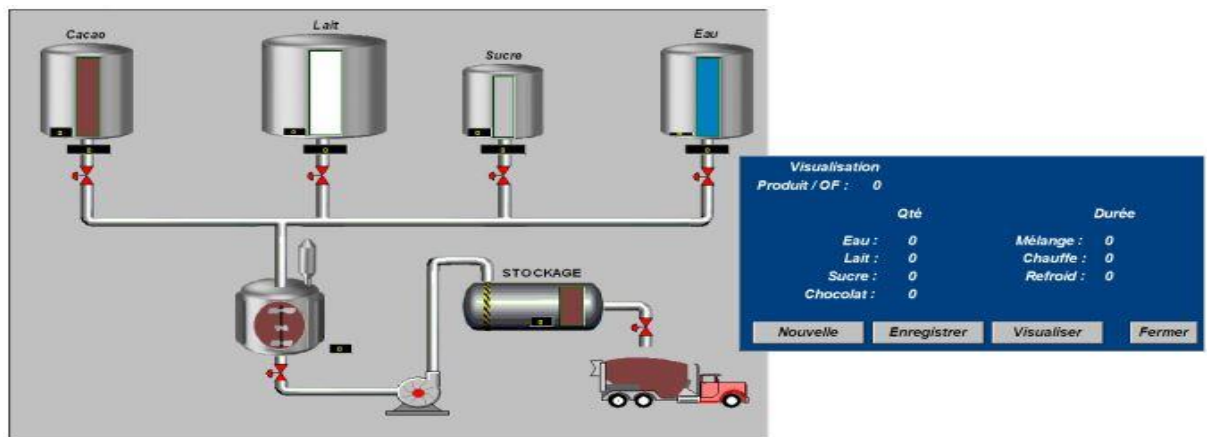


Figure 30: exemple d'une interface de gestion des gammes de fabrication [16].

La supervision est alors d'un niveau supérieur et qui superpose à la boucle de commande, elle assure les conditions d'opérations pour lesquelles les algorithmes de commande ont été conçus [18]. L'architecture d'un système de supervision est illustrée sur la (Figure 31):

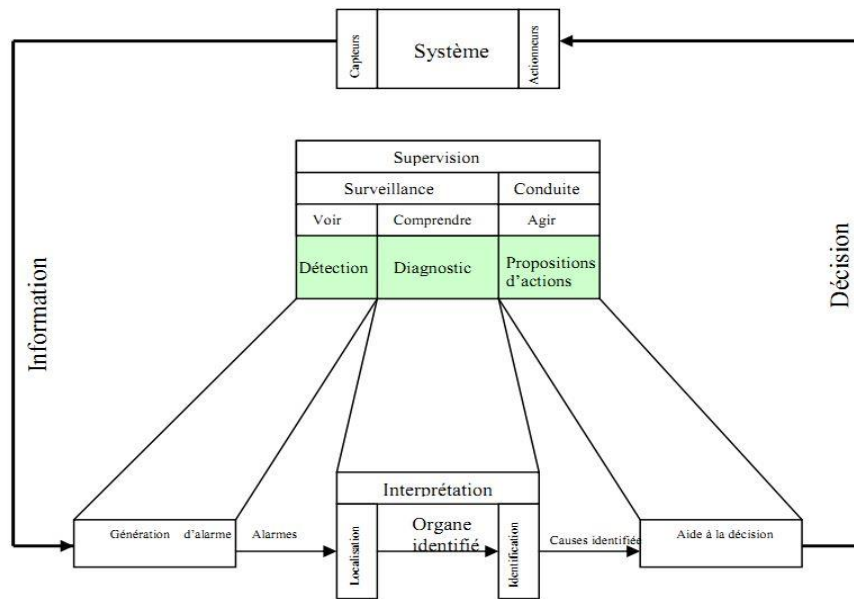


Figure 31: Architecture générale d'un système de Supervision en ligne [19].

### 3.7 Les Techniques de la supervision

Pour concevoir un système de supervision nous avons besoin de maîtriser les techniques suivantes :

- **Acquisition de données** : l'acquisition de données est la première étape de la supervision, telle que, elle consiste à recueillir, à valider, et à assurer l'acheminement des informations sur l'état de système jusqu'au poste de pilotage, cette tâche est exécuter sans interruption [18].
- **Surveillance** : toute prise de décision est basée sur les informations acquises des ressources et de l'historique des activités exécutées. La surveillance est responsable de l'acquisition des signaux en provenance des ressources et de la commande (Figure 32). Ces informations sont utilisées pour la reconstitution de l'état réel du système commandé, et pour faire les inférences nécessaires afin de produire les données utilisées pour reconstituer l'état réel du système ou pour dresser des historiques de fonctionnement [17].
- **La détection** : La procédure de détection vise à déterminer l'apparition et l'instant d'occurrence d'une défaillance (Figure 32). La détection d'une défaillance se résumera alors à déclencher une alarme lorsqu'on observe au moins une panne [20].

- **Le Diagnostic** : établit un lien de cause à effet entre un symptôme observé et la défaillance qui est survenue, ses causes et ses conséquences. On distingue classiquement trois étapes :
  - **La Localisation** : elle détermine le sous-système fonctionnel à l'origine de l'anomalie et progressivement affine cette détermination (Figure 32) pour désigner l'organe ou dispositif élémentaire défectueux [21].
  - **L'identification** : détermine les causes qui ont engendré la défaillance constatée (Figure 32).
  - **Le Pronostic** : Le pronostic s'intéresse à la propagation des défaillances. Il indique les conséquences inéluctable et mesure celles qui peuvent être anticipées de façon à ne pas solliciter les sous-systèmes de la ressource défaillante .le pronostic est une étape sans laquelle la prise de décision n'est pas faisable [22].

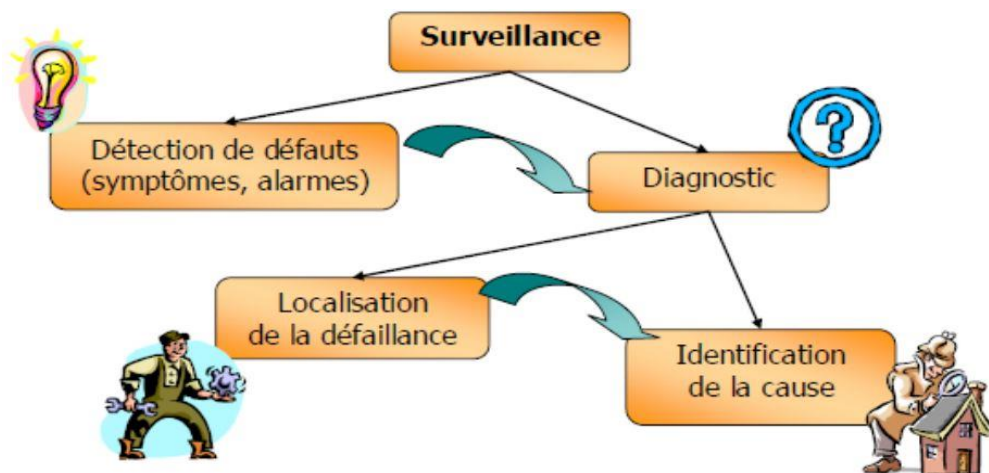


Figure 32 : Composantes de la surveillance industrielle [19].

- **L'aide à la décision** : les systèmes d'aide à la décision pour la supervision sont conçus pour gérer la conduite et maintenir un système avec un degré de sûreté de fonctionnement élevé. Afin de maintenir ces indices aux niveaux souhaitables, ces systèmes doivent d'une part, aider les opérateurs à prendre des décisions rapidement face aux situations anormales, comme la prise de décision du démarrage ou d'arrêt du système ou des sous-systèmes ou la reconfiguration du système .D'autre part, ces systèmes doivent aider l'opérateur à prendre les décisions sur les politiques de maintenance des différents équipements [23].

- **La maintenance** : la maintenance des systèmes industriels est devenue un point essentiel lors de leur conception et de leur exploitation, tant pour des questions de sécurité et de sûreté de fonctionnement, que pour des questions de rentabilité. Une maintenance mal adaptée à un système peut également conduire à une situation critique, dangereuse aussi bien pour les personnes que pour le matériel ou l'environnement [24].

Pour les mettre en œuvre, les industriels ont divisé les techniques de la supervision en deux grandes catégories selon son étendue, en effet, si le superviseur surveille une machine ou un nombre réduit de machines on parle dans ce cas d'un système de supervision SCADA, si le nombre de machine a contrôlé augmente, le flux d'informations devient vite très important et insupportable pour les SCADA. Pour pouvoir traiter ce flux en temps réel et préserver la sécurité des biens et des personnes, les constructeurs des APIs se sont vite penchés vers une solution plus adaptée, dénommée DCS.

## **3.8 La supervision dans un environnement SCADA**

### **3.8.1 Définition du SCADA**

SCADA est un acronyme qui signifie le contrôle et la supervision par acquisition de données (en anglais : Supervisory Control And Data Acquisition). L'environnement SCADA collecte des données de divers appareils d'une quelconque installation, puis transmet ces données à un ordinateur central, que ce soit proche ou éloigné, qui alors contrôle et supervise l'installation, ce dernier pourra être subordonné par d'autres postes d'opérateurs [25].

La supervision SCADA est une solution très performante pour la commande des systèmes industriels complexes.

### **3.8.2 Composants du SCADA**

Un dispositif SCADA est généralement composé des sous-systèmes suivants :

- un système de supervision et contrôle informatique, faisant l'acquisition des données de l'API et envoyant des commandes (consignes) à l'API, composé lui-même :

- d'une IHM permettant à l'opérateur de superviser la machine à partir d'un tableau de bord virtuel comportant des boutons, des voyants, des alertes.
- d'une base de données pour stocker toutes les données dont il a besoin pour la prise de décision.
- des APIs utilisés sur le terrain pour leur versatilité et flexibilité due à leur capacité d'être configurables.
- une unité terminale distante (RTU) reliant les capteurs convertissant les signaux en flux de données numériques et envoyant les données numériques à l'API.
- une infrastructure de communication reliant le système de supervision et contrôle aux éléments terminaux.
- divers instruments d'analyse.

L'ensemble des couples API/IHM forme ce qu'on appelle le SCADA. Bien entendu, le SCADA peut comporter plusieurs API qui sont extensibles en plusieurs modules d'E/S. Il présente une souplesse et une adaptabilité dans son installation puisque les fonctions logiques sont toutes rassemblées en un seul programme qui peut être aisément modifié [26].

### **3.8.3 Les Interfaces Homme-Machine SCADA**

L'un des éléments les plus importants d'un système de supervision SCADA est l'Interface Homme-Machine. Elle représente le seul point d'interaction entre l'opérateur et le système supervisé.

#### ***a Définition d'une Interface Homme-Machine***

L'interaction Homme-Machine représente l'ensemble des mécanismes d'échange d'information entre un humain et une machine (système interactif) pour accomplir une tâche ou atteindre un but particulier pour l'humain. Elle est caractérisée par le triplet (Figure 33): opérateur humain (ou utilisateur), machine et environnement de l'interaction (interface).



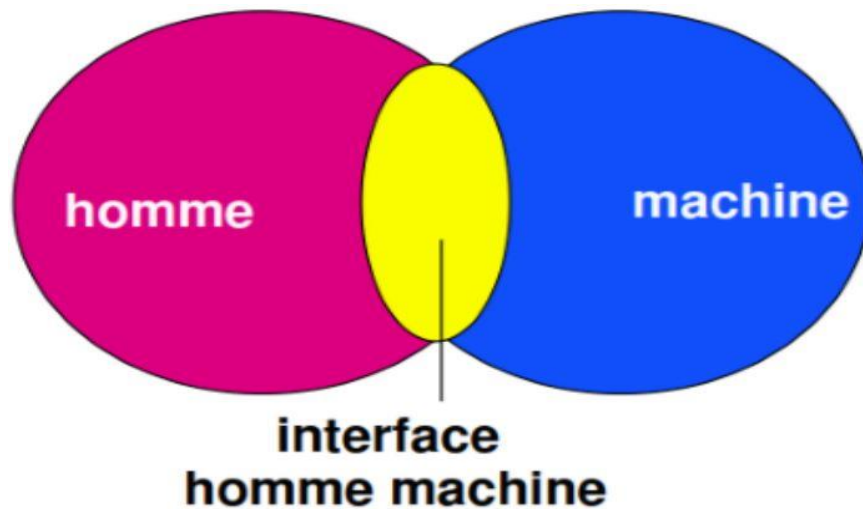


Figure 33 : Interaction Homme Machine [27].

Une IHM sert à afficher de manière claire et concise une vue d'ensemble du système et l'état des services surveillés, de générer des rapports et de visualiser l'historique [28]. Depuis quelques années, les IHM s'imposent comme un facteur déterminant dans le succès des applications de supervision. Cet intérêt accordé aux IHM est à l'origine de différents travaux qui se sont intéressés à la définition, la caractérisation, la conception et l'implémentation des IHM. Elles sont un outil très important pour le bon déroulement du système industriel de même pour la réception d'informations de son environnement SCADA [27].

#### ***b Evolution des interfaces Homme machine***

Les premières interfaces apparues, dès la fin des années 60, sont les interfaces en ligne de commandes basées sur une interaction textuelle entre l'utilisateur et la machine. Elles étaient révolues lors de l'avènement des interfaces WIMP en début des années 80 (Windows, Icones, Menus and Pointing device) pour exprimer qu'elle utilise des métaphores de type : fenêtres, icônes, menus et dispositif de pointage. Les interfaces WIMP sont basées sur le principe WYSIWYG (What You See Is What You Get) pour ce que vous voyez est ce que vous obtenez. Ces interfaces offrent une interaction plus intuitive en réduisant les efforts d'apprentissage, elles ont grandement contribué à l'utilisation des ordinateurs par le grand public.

### **3.8.4 Fonctionnalités d'un système SCADA**

Un système SCADA comprend deux sous-ensembles fonctionnels:

**Un sous-système commande** : son rôle est de faire exécuter un ensemble d'opérations au procédé en fixant des consignes de fonctionnement en réponse à des ordres d'exécution. La commande regroupe toutes les fonctions qui agissent directement sur le processus du procédé:

- le fonctionnement en l'absence de défaillance.
- la reprise ou gestion des modes.
- les traitements d'urgence.
- une partie de la maintenance corrective.

**Un sous-système surveillance** : la partie surveillance d'un superviseur a pour objectifs :

- La détection d'un fonctionnement ne correspondant plus à ce qui est attendu.
- la recherche des causes et conséquences d'un fonctionnement non prévu.
- La collaboration avec les opérateurs humains pour les prises de décision critiques, pour le recueil d'informations non accessibles directement.
- reconstitue l'état réel du système commandé.
- fait toutes les inférences nécessaires pour produire les données utilisées pour dresser des historiques de fonctionnement.

### **3.8.5 Communication des systèmes SCADA**

La notion temps réel est devenue très importantes et indispensable dans la procédure de surveillance et de supervision en générale, elle permet de suivre l'évolution de l'état du système d'une façon continue [18].

Dans un système de supervision SCADA afin d'accéder en temps réel aux informations, en lecture et écriture, qui se trouvent au niveau des API nous avons recours a :

- Des moyens de communication physiques :
  - liaison série standard RS 232 ou RS485.
  - liaison dédiée ASI, CAN, Profibus.
  - réseau Ethernet.
- Des protocoles différents :
  - Modbus ASCII, RTU, TelWay.
  - CanOpen, DeviceNet.
  - UDP, TCP/IP, ModbusTCP.

### 3.9 La supervision dans un environnement DCS

Le système de contrôle commande DCS (Distributed Control System), développé au début des années 70, ressemble en grande partie au SCADA. Sauf que ce dernier est destiné pour gérer des processus plus étendus et plus complexes.

La principale différence entre un DCS et un SCADA réside essentiellement dans la nature de l'architecture et la criticité du procès supervisé. En effet le DCS gère beaucoup plus rapidement le transfert de données et se distingue par un temps de réponse remarquable.

Le DCS présente une architecture très organisée qui empêche toutes sortes de conflits et de collisions de données. Ses contrôleurs sont reliés entre eux via un réseau Profinet pour échanger des données partagées. Chaque contrôleur est doté de modules d'E\S qui lui sont propres via lesquels il communique avec le système [26].

Dans un DCS, les fonctions d'acquisition et de contrôle des données sont effectuées par un certain nombre d'unités à microprocesseur, qui sont distribuées et situées à proximité des appareils qui lui sont connectés, ou l'instrument à partir duquel les données sont recueillies.

### 3.10 Superviseurs SCADA

Il existe une multitude de logiciel SCADA, leurs nombre augmentent avec l'évolution des technologies et la demande du marché, ces logiciels peuvent être divisés en deux types :

- **Propriétaire** : Les constructeurs des API et les développeurs des logiciels SCADA développent parfois des logiciels propriétaires pour communiquer avec leur propre matériel. Ces systèmes sont vendus comme (clés en main). Le principal problème de ce système est la dépendance écrasante avec le fournisseur du système.
- **Ouverts** : les logiciels ouverts ont beaucoup gagné en popularité en raison de l'interopérabilité qu'ils apportent aux systèmes. L'interopérabilité est la capacité de mélanger les équipements des différents fabricants sur le même système.

### **3.11 Choix et discussions**

Nous avons vu précédemment que les techniques de la supervision sont divisées en deux grandes catégories. Or, après avoir réalisé notre étude nous avons décidé d'opter pour une solution SCADA vu que :

- Elle présente une souplesse et une adaptabilité dans son installation (le flux d'informations est supportable).
- Elle est une solution très performante pour la commande des systèmes industriels.
- Nous avons un seul API a supervisé donc une seule source de données.
- Elle dispose de la capacité de contrôle d'un système industriel complexe.
- Elle permet de visualiser même la position où se situent la faute et l'élément défectueux.
- Elle aide l'opérateur à prendre la bonne décision, et ne pas se tromper dans son intervention.

### **3.12 Conclusion**

Ce chapitre a été consacré à la présentation de la supervision et les différentes fonctions couramment employées dans ce domaine. Bien évidemment, ce chapitre n'a pas prétendu à l'exhaustivité mais il a espéré aborder et présenter les principales notions.

Nous avons donc présenté l'état de l'art des techniques de la supervision. Ensuite, nous avons passé en revue les principales catégories utilisés à nos jours.

Un intérêt particulier a été porté à la supervision dans un environnement SCADA car, comme précédemment dit, il nous servira de support au long de ce mémoire pour présenter le travail que nous avons réalisé.

Après avoir détaillé tous les fondements théoriques proposés dans la littérature ainsi que dégagé les notions de bases, nous allons aborder la partie suivante qui représente la partie pratique.

# Chapitre 4 Conception et mise en œuvre de la solution

---

## 4.1 Introduction

Dans les chapitres précédents, nous avons vu que l'environnement SCADA permet une supervision en temps réel, un bon contrôle sur les API en offrant des interfaces graphiques sur les quelles est représenté le système automatisé avec possibilités d'intervention en modifiant les valeurs d'un paramètres, et surtout il nous offre un suivi de tous les événements ayant lieu au sein du système.

Dans ce chapitre nous allons mettre en œuvre toutes les notions acquises durant notre étude bibliographique. Il sera subdivisé en trois sections. Dans la première section, étude de l'existant, nous allons commencer par présenter notre API, son architecture ainsi que ses options de communication pour aboutir à l'identification de ses E/S. La deuxième section, conception de la solution, portera sur une description de notre application, son organisation et son architecture générale adoptée pour la réaliser. Dans la troisième partie, mise en œuvre de la solution, il a été nécessaire de recourir à un certain nombre d'outils, ainsi qu'à la mise en place des environnements d'exécution, pour présenter par la suite la démarche globale choisie, l'architecture technique de la solution et le processus de développement de notre système.

## 4.2 Etude de l'existant

Nous allons essayer, à travers cette section, de présenter une analyse aussi complète que possible de l'existant au sein de l'unité réchauffeur gaz à bain marie. Cette section a pour but de faire connaître l'API, que nous allons travailler dessus, de plus près, ainsi

que sa configuration matérielle et ses modes de communication et elle va nous permettre d'identifier les E/S de l'API. Cette section constitue ainsi le point de départ pour pouvoir entamé la partie conception.

#### 4.2.1 Présentation de l'API

L'unité réchauffeur gaz est équipée d'un API **Allen Bradley SLC 5/03** fabriqué par **Rockwell Automation** et illustré dans la (Figure 34). Il est un automate modulaire, destiné à des taches d'automatisation moyenne et haut de gamme.

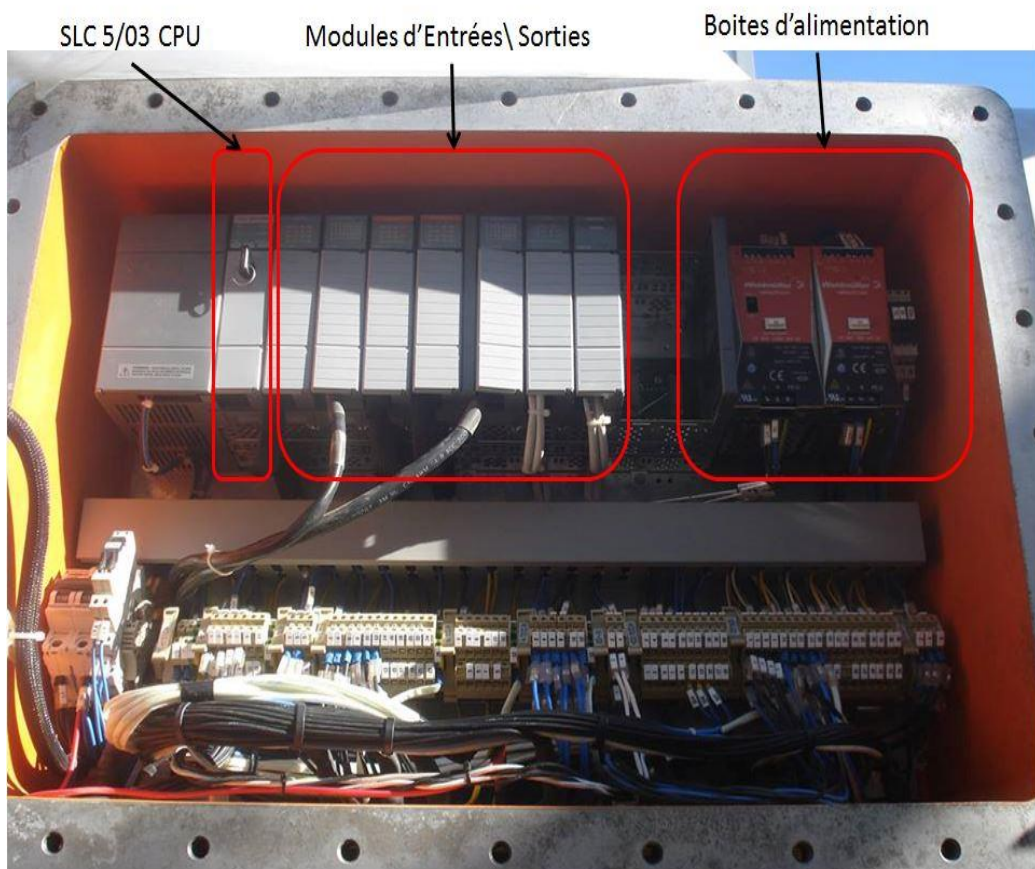


Figure 34: L'API Rockwel Allen Bradley SLC 5/03.

#### 4.2.2 L'architecture de l'API

L'API Allen Bradley SLC 5/03 est un équipement très performant constitué d'un ensemble de châssis distincts contenant les composants principaux suivants :

### **a L'unité centrale CPU**

Allen Bradley propose 5 processeurs appartenant à la famille SLC 500, qui se différencient par leurs capacités mémoire, vitesse de traitement, nombre et type de ports de communication [29]. Notre processeur (Figure 35) a les caractéristiques suivantes :

- Modèle : SLC 5/03 CPU processor
- MPN : 1747-L531
- Série : E
- Capacité mémoire : 8Ko
- Vitesse de traitement d'un programme : 1ms/kword.
- Vitesse de traitement des E/S : 0.225 ms



Figure 35: Le processeur SLC 5/03.

Le processeur Allen Bradley SLC 5/03 (Figure 36) intègre :

- Un port DH485 [29].
- Un port série RS232 [29].



Figure 36: Ports de processeur SLC 5/03.

### ***b Le module d'alimentation***

Comme la boîte d'alimentation originale a été isolée pour des raisons techniques, Les deux boîtes d'alimentation désignées seront ceux illustrées dans la figure ci-dessous. Une seule boîte fonctionnera et l'autre sera en attente en cas où la première tombe en panne [30].



Figure 37: Les boîtes d'alimentation de l'API.



### ***c Modules d'entrée/sorties analogique***

Notre API a deux modules d'E/S analogiques : un module d'entrées analogiques 5 voies et un module de sorties analogiques 2 voies [30].

### ***d Modules d'entrée/sorties (TOR)***

Notre API possède 2 modules d'entrées logiques et 3 modules de sorties logiques [30]. Ces modules d'E/S (TOR) offerts par Allen Bradley sont des modules standards occupant un seul emplacement. Les entrées reçoivent les signaux en provenance des capteurs et réalisent plusieurs fonctions (acquisition, adaptation, filtrage, protection contre les signaux parasites...etc.).

Les sorties réalisent les fonctions de mémorisation des ordres donnés par le processeur, pour permettre la commande des pré-actionneurs. La figure ci-dessous montre les modules d'E/S.



Figure 38: Modules d'E/S.

### **4.2.3 Communication de l'API**

Les options de communication des API SLC 5/03 sont [29]:

- DH485.
- Les protocoles RS-232 (DF1 Full-duplex, DF1 Half-duplex Maître/Esclave, DF1 Radio Modem, DH-485, or ASCII).

- Data Highway Plus (une passerelle ControlLogix est requise).
- Ethernet TCP/IP (une interface 1761-NET-ENI est requise).
- ControlNet (via le module 1747-KFC15).
- DeviceNet (via l'interface 1761-NET-DNI).

#### 4.2.4 Identification des Entrées/Sorties :

D'après l'étude profonde faite sur les différentes installations de réchauffeur gaz à bain marie et le programme de ce dernier, nous avons identifié les E/S associées. Les tableaux suivants contiennent la liste de ces derniers en indiquant leurs adresses et leurs descriptions.

##### a Les entrées logiques (TOR)

Adresses	Descriptions
I:1/0	ARRET D'URGENCE
I:1/1	ESSAI DE LAMPES
I:1/2	RESET
I:1/3	DEPART BRULEUR 1
I:1/4	ARRET BRULEUR 1
I:1/5	DEPART BRULEUR 2
I:1/6	ARRET BRULEUR 2
I:1/7	OUVERT/FERMEE
I:1/8	RECHAUFFEUR PERMISSIF
I:1/9	RECHAUFFEUR SHUT-DOWN
I:1/10	FONCTIONNER/NON-FONCTIONNER
I:1/11	CONTACT FLAME BRULEUR1 ALLUMEE
I:1/12	CONTACT FLAME BRULEUR2

	ALLUMEE
I:1/14	RUNNING RECH
I:1/15	TRIPPED RECH
I:2/0	SORTIE DE GAZ (VANNE OUVERTE)
I:2/1	SORTIE DE GAZ (VANNE FERMEE)
I:2/2	LEVEL SWITCH ALARM
I:2/3	PRESSION SWITCH ALARM BRULEUR1
I:2/4	PRESSION SWITCH ALARM BRULEUR2
I:2/5	TEMPERATURE SWITCH ALARM
I:2/6	VANNE ALIMENTATION BRULEUR1 OUVERTE
I:2/7	VANNE ALIMENTATION BRULEUR1 FERMEE
I:2/8	VANNE TORCHE BRULEUR1 OUVERTE
I:2/9	VANNE TORCHE BRULEUR1 FERMEE
I:2/10	VANNE ALIMENTATION BRULEUR2 OUVERTE
I:2/11	VANNE ALIMENTATION BRULEUR2 FERMEE
I:2/12	VANNE TORCHE BRULEUR2 OUVERTE
I:2/13	VANNE TORCHE BRULEUR2 FERMEE
I:2/14	RESISTANCE RECHAUFFEUR ELECTRIQUE
I:2/31	BY-PASS BAIN MARIE

Tableau 3: Les entrées logiques de l'API.

**b Les sorties logiques (TOR)**

<b>Adresses</b>	<b>Descriptions</b>	<b>Variables internes</b>
O:4/0	LAMPE RECHAUFFEUR PRÊT	
O:4/1	LAMPE BALAYAGE EN COURS	
O:4/2	LAMPE BRULEUR1 EN SERVICE	
O:4/3	LAMPE BRULEUR1 EN SERVICE	
O:4/4	LAMPE RECHAUFFEUR EN BLOC	
O:4/5	LAMPE FLAME1 DEFAULT	
O:4/6	LAMPE FLAME2 DEFAULT	
O:4/7	LAMPE TRES BAS NIVEAU H2O	
O:4/8	LAMPE TRES HAUTE TEMPERATURE H2O	B3:3/8
O:4/12	LE CONTACT DE LA COMBUSTION BRUELUR1 230VAC 50HZ	B17:2/4
O:4/13	LE CONTACT DE LA COMBUSTION BRUELUR2 230VAC 50HZ	B17:3/4
O:4/14	LAMPE RECHAUFFEUR ELECTRIC EN ARRET	
O:3/0	RECHAUFFEUR MARCHE/ARRET	B12:0/15 ou B12:1/15
O:3/1	RECHAUFFEUR HORS SERVICE	B12:0/15=0 et B12:1/15=0
O:3/2	CUMULATIF ALARME	B3:2/0
O:3/3	BAIN MARIE TEMPERATURE. PRÊT	
O:3/4	VANNE OUVERTE	
O:3/5	VANNE FERMEE	
O:3/8	RUNNING RECH	
O:3/9	TRIPPED RECH	
O:5/0	00EKA26AA004 FLAMME	B12:0/15 et B12:0/13

	BRULEUR1 ET TORCHE	
O:5/1	00EKA26AA005 BRULEUR1 ET TORCHE	
O:5/2	00EKA26AA006 TORCHE	B17:0/15
O:5/3	00EKA26AA007 TORCHE	
O:5/4	00EKA26AA008 FLAMME BRULEUR1	B3:7/0
O:5/5	00EKA26AA009 1 <sup>er</sup> ETAGE BRULEUR1	B3:6/0
O:5/6	00EKA26AA010 2 <sup>eme</sup> ETAGE BRULEUR1	B3:6/1
O:5/7	00EKA26AA011 3 <sup>eme</sup> ETAGE BRULEUR1	B3:6/2
O:5/8	00EKA26AA012 4 <sup>eme</sup> ETAGE BRULEUR1	B3:6/3
O:5/9	00EKA26AA013 FLAMME BRULEUR2 ET TORCHE	B12:1/15 et B12:1/13
O:5/10	00EKA26AA014 FLAMME BRULEUR2 ET TORCHE	
O:5/11	00EKA26AA015 TORCHE	
O:5/12	00EKA26AA016 TORCHE	
O:5/13	00EKA26AA017 FLAMME BRULEUR2	B3:7/1
O:5/14	00EKA26AA018 1 <sup>er</sup> ETAGE BRULEUR2	B3:5/0
O:5/15	00EKA26AA019 2 <sup>eme</sup> ETAGE BRULEUR2	B3:5/1
O:5/16	00EKA26AA020 3 <sup>eme</sup> ETAGE BRULEUR2	B3:5/2
O:5/17	00EKA26AA021 4 <sup>eme</sup> ETAGE BRULEUR2	B3:5/3
O:5/18	10EKA26AA002 VANNE VERS	

	DETENTE	
--	---------	--

Tableau 4: Les sorties logiques de l'API.

**c Les entrées analogiques**

Adresses	Descriptions	Variable internes
I:6.0	TEMPERATURE D'EAU	N13:20
I:6.1	TEMPERATURE GAZ RECHAUFFEUR "BRULEUR"	N13:27
I:6.2	TEMPERATURE SORTIE GAZ	N13:37
I:6.3	TEMPERATURE ARRIVEE GAZ	F14:14
I:6.4	PRESSION D'ARRIVEE GAZ	F14:15

Tableau 5: Les entrées analogiques de l'API.

**d Les sorties analogiques**

Adresses	Descriptions	Variabes internes
O:6.0	TEMPERATURE D'EAU	N13:20
O:6.1	TEMPERATURE GAZ RECHAUFFEUR "BRULEUR"	N13:27
O:6.2	TEMPERATURE SORTIE GAZ	N13:37
O:6.3	TEMPERATURE ARRIVEE GAZ	F14:14
O:6.4	PRESSION ARRIVEE GAZ	F14:15
O:7.0	POSITION DE VANNE A 3 VOIS	N9:85
O:7.3	RETRANSMISSION PRESSION ARRIVEE GAZ	F14:15

Tableau 6: Les sorties analogiques.

### 4.3 Conception de la solution

Tout projet englobant une application informatique, que ce soit dans le domaine de la recherche ou dans le domaine professionnel, doit être en mesure de répondre aux attentes de ses utilisateurs. Ainsi, les développeurs des applications, avant d'arriver au produit final, adoptent une méthode de conception afin de faciliter la compréhension

de leurs applications, d'éviter les anomalies et de simplifier une éventuelle mise à jour. Dans cette section nous allons apporter plus de détails à la solution et présenter son architecture.

### 4.3.1 Description de l'application

Bien que les opérateurs puissent contrôler et surveiller l'API sans une interface graphique comme mentionner dans la (Figure 6.a). L'implémentation de cette dernière (Figure 6.b) permet une supervision visuelle, facilite la tâche des opérateurs et la rend moins couteuse en terme de temps.

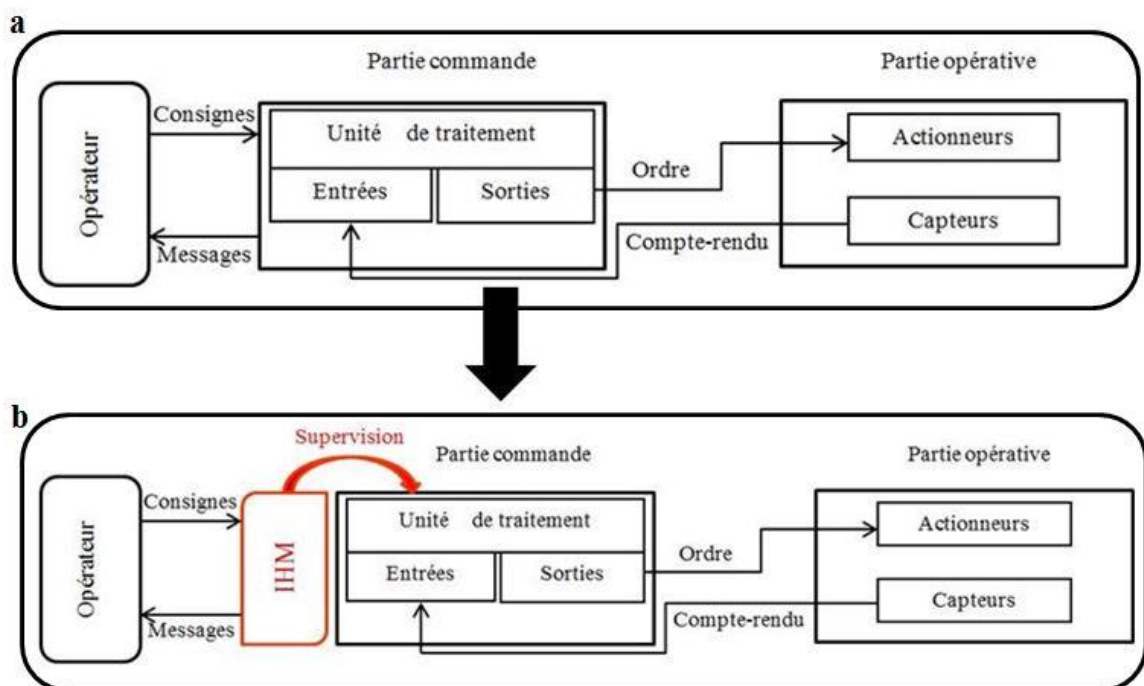


Figure 39 : a. Contrôle/surveillance sans IHM, b. Contrôle/surveillance avec IHM.

### 4.3.2 Fonctionnalités de l'application

#### a Analyse préliminaire

L'utilisateur du système peut suivre et piloter l'API. Le suivi consiste alors à vérifier le bon fonctionnement de chaque équipement du réchauffeur gaz à bain marie par l'intermédiaire d'une IHM. Lorsque le fonctionnement devient anormal, le rôle de l'utilisateur est de détecter les erreurs et les pannes et les corrige. Il gère aussi le démarrage et l'arrêt de l'API.

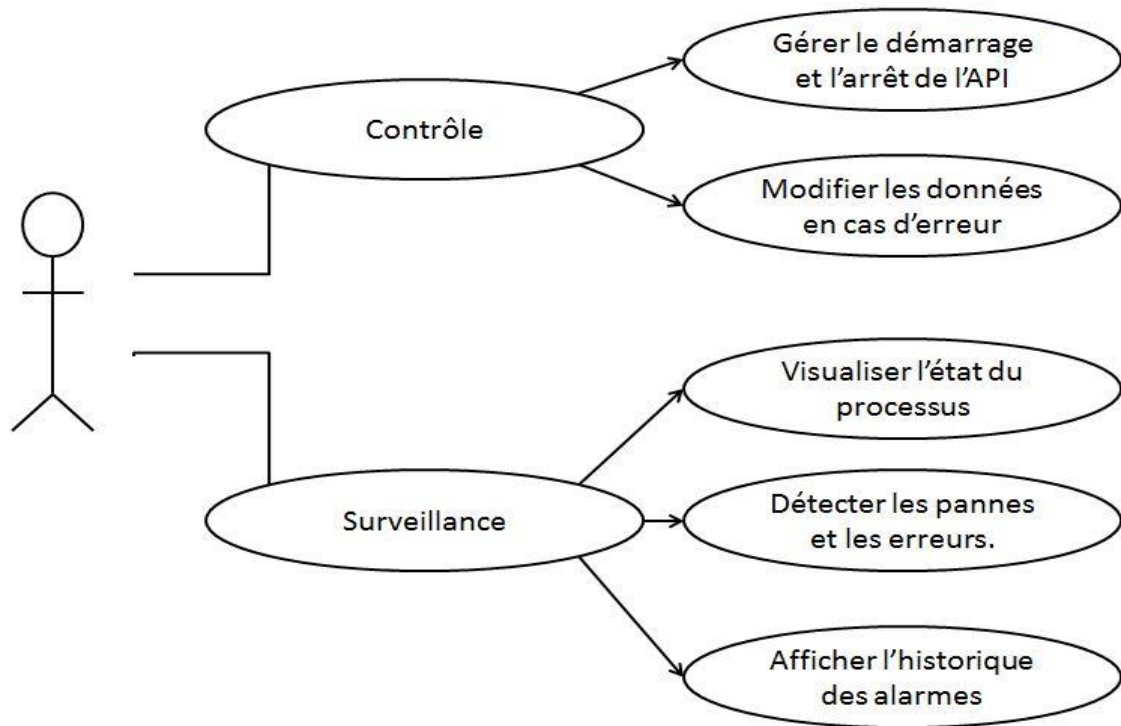


Figure 40: fonctionnalité préliminaire du système.

### ***b* Analyse détaillée**

Après une analyse plus profonde nous avons tiré les fonctionnalités suivantes :

#### ➤ **Etablir la connexion avec l'API**

Nous allons utiliser le port RS-232 disponible sur l'API pour communiquer avec l'IHM à l'aide d'un pilote propre aux API Allen Bradley dénommé DF1.

Un pilote est un logiciel qui permet à l'ordinateur de communiquer avec du matériel ou des périphériques. Sans pilote, le matériel que vous connectez à l'ordinateur (par exemple, une carte vidéo ou une imprimante) ne fonctionnera pas correctement.

Comme ce pilote n'est pas commercialisé par Rockwell Software Automation, il faut programmer l'ordinateur de telle sorte qu'il comprend les données échangées avec l'API.

Nous avons trouvé un programme de ce protocole, programmé sous Visual Basic que nous allons utiliser pour l'implémentation de notre solution.



➤ **Modifier les données en cas d'erreur**

Lors d'un fonctionnement anormal un signal d'erreur se génère en temps réel, l'IHM récupère ce dernier et affiche un message d'erreur à l'utilisateur. L'utilisateur à son rôle essaye de détecté l'erreur qui s'est produite, si c'est une erreur qui a une relation avec les paramétrages l'utilisateur modifie les paramètres via le mode avancé pour que le fonctionnement soit normal sinon si l'erreur détectée est une erreur matérielle un arrêt d'urgence est imposé.

➤ **Gérer le démarrage et l'arrêt de l'API**

Pour que l'API soit en marche l'utilisateur doit appuyer sur le bouton « ALLUMER » dans l'IHM, un message est livré à l'utilisateur disant que l'API est en marche. Pour que l'API s'arrête l'utilisateur doit appuyer sur le bouton « ARRET URGANCE » dans l'IHM, un message est livré à l'utilisateur disant que l'API est en arrêt.

➤ **Visualiser l'état du processus**

L'utilisateur peut vérifier et visualiser le changement d'états du processus, en plus savoir toutes les informations qui proviennent du processus avec un simple accès depuis l'IHM (vues synoptiques).

➤ **Afficher l'historique**

L'IHM enregistre les messages d'erreur puis ces messages seront affichés doivent être vérifié par un expert pour qu'il puisse détecter des anomalies s'ils existent, en cas d'erreur il procède à modifier les paramètres qui ont causé l'erreur et enfin le bon fonctionnement est vérifié.

➤ **Afficher la courbe de données**

L'IHM donnent une représentation graphique de différentes données du processus sous forme d'un graphe. C'est un outil puissant d'analyse des variables historisées.

### **4.3.3 Architecture générale de la solution**

La figure ci-dessous illustre la structure et l'architecture de la solution proposée :

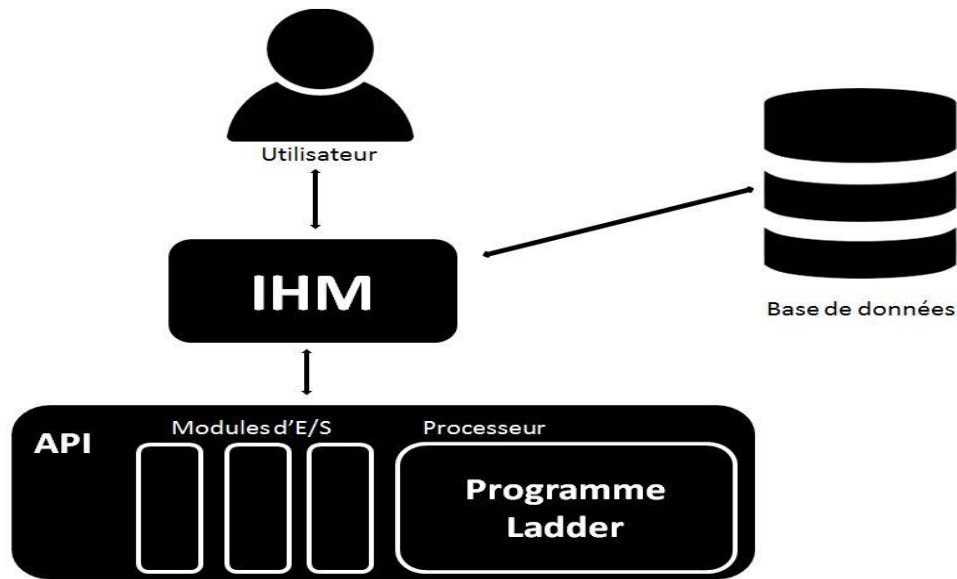


Figure 41: Architecture globale de la solution.

## 4.4 Mise en œuvre de la solution

Cette section a pour objectifs de décrire le matériel et le système d'exploitation existants ainsi que les outils et technologies mis en place formant l'environnement dans lequel évoluera notre outil. Enfin nous allons exposer le travail achevé et quelques captures d'écran traduisant le déroulement de l'application.

### 4.4.1 Périmètre technique et fonctionnel

Dans cette partie nous allons décrire les infrastructures mises en place. En effet, cette étape est à ne pas négliger, car la diversité des plateformes techniques pourra engendrer des problèmes de compatibilité.

#### *a Matériel*

Les programmes sources sont développés sur la plateforme suivante:

- Machines HP :
  - Processeur : Intel core™ i3-3110m CPU 2.40GHz.
  - Mémoire : 4.00 Go.

#### *b Systèmes d'exploitation*

Lors de notre étude, nous avons utilisé le système d'exploitation suivant :

- Windows XP : Service Pack 3.

### ***c Logiciel***

Pour choisir l'outil permettant la programmation de notre interface graphique, la difficulté que nous rencontrons en la matière est multiforme, la plus importante est la diversité des logiciels de programmation. Cependant, nous avons choisis d'utiliser le langage Visual Basic (VB) sous le logiciel Visual Studio Express car :

- Pour communiquer avec l'API disponible au niveau de l'unité réchauffeur gaz, nous avons besoin du driver DF1, qui est disponible sous le langage VB.
- Il permet l'élaboration des interfaces graphiques, la conception des formulaires, la gestion du temps, l'accès à une base de données et l'écriture dans les fichiers.
- Le programme du réchauffeur de la centrale électrique de LARBAA a été réalisé sous le langage Ladder via le logiciel propriétaire Rockwell RSLogix qui est compatible avec le langage VB.
- Les autres solutions propriétaires sont payantes et ne possède pas la capacité de mélanger les équipements des différents fabricants sur le même système (interopérabilité).

Demeurent toutefois des questions relatives à l'utilité d'une technique particulière ou d'un logiciel spécifique pour élaborer une application de supervision. Nous pouvons dire qu'aucune technique et qu'aucun outil n'ont plus adaptés que les autres pour résoudre tous les types de problèmes. Néanmoins, cela n'empêche pas certains techniques et outils d'être mieux adaptés que d'autres sur des classes particulières de problèmes.

## **4.4.2 Processus de développement**

### ***a Interface Homme Machine***

Pour construire une IHM efficace et professionnelle et en réussir la conception, nous avons suivi les étapes décrites comme suit :

1. Définir l'architecture : nous avons identifié les différentes fenêtres de notre IHM, puis nous avons organisé ces fenêtres et les dénommées.
2. Structurer le contenu : nous avons fait par la suite les différents menus.
3. Fixer les principes de navigation : nous avons organisé les chemins d'accès pour y aller d'une fenêtre à l'autre.
4. Structurer la mise en page : nous avons défini les typologies des pages.
5. Créer un contenu : Construire les synoptiques du réchauffeur (vue de côté et vue de face).
6. Ajouter la dynamique : nous avons choisi les boutons et les zones actives.
7. Mettre en relief : nous avons utilisé les icônes, images et couleurs.
8. Créer textes et outils de dialogue : Rendre lisible et visible.
9. Créer des fenêtres de dialogue composées des messages clairs.

Après le suivi de ces étapes nous avons obtenus les fenêtres suivantes :

- **Page d'accueil** : Nous présentons tout d'abord notre page d'accueil (ci-dessous).

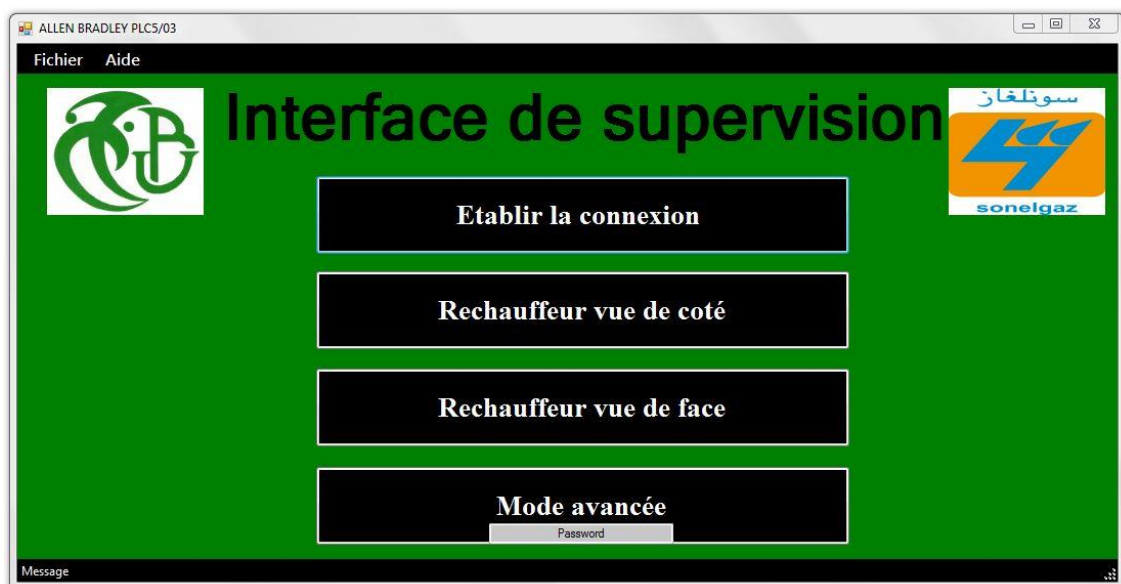


Figure 42 : Page d'accueil.

- **A-propos** : cette page nous donne des informations générales sur le réalisateur et le rôle de l'application (Figure 43), nous pouvons accéder à cette page via le menu de la page principale Aide > a-propos.

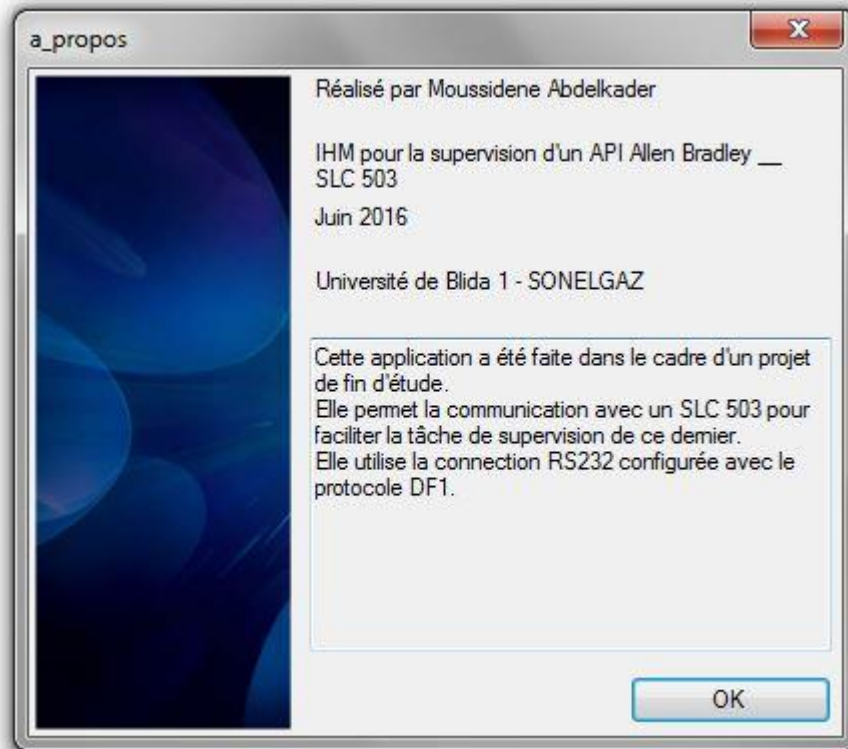


Figure 43 : La page A-propos.

- **Ouvrir le port de communication** : cette seconde page est réservée à la configuration du port COM (Figure 44), nous pouvons accéder à cette page en appuyant sur le bouton de la page d'accueil « Etablir la connexion ».

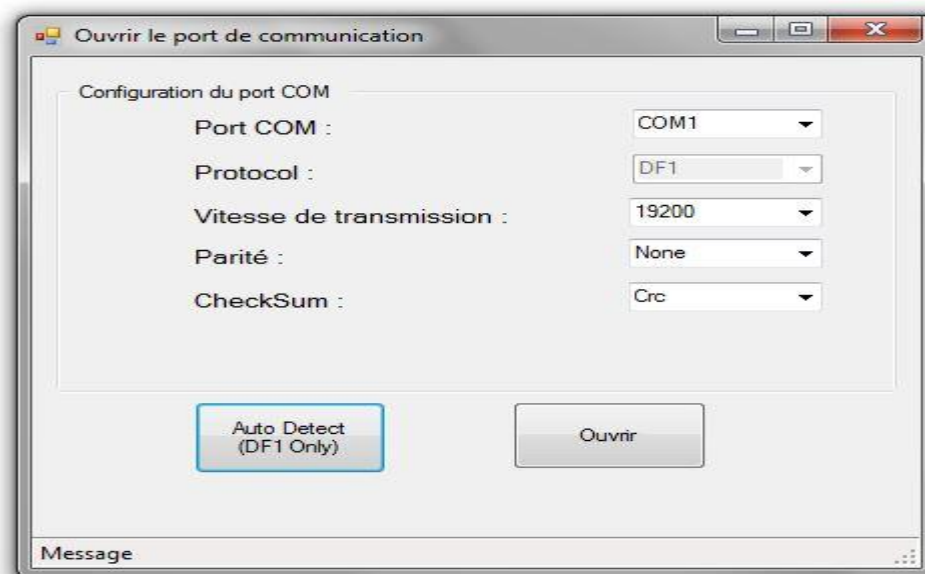


Figure 44: Configuration du port COM.

- **Bruleur** : cette page nous donne une vue synoptique du réchauffeur (de coté) (Figure 45), nous pouvons accéder à cette fenêtre en appuyant sur le bouton de la page d'accueil « Réchauffeur vue de côté ».

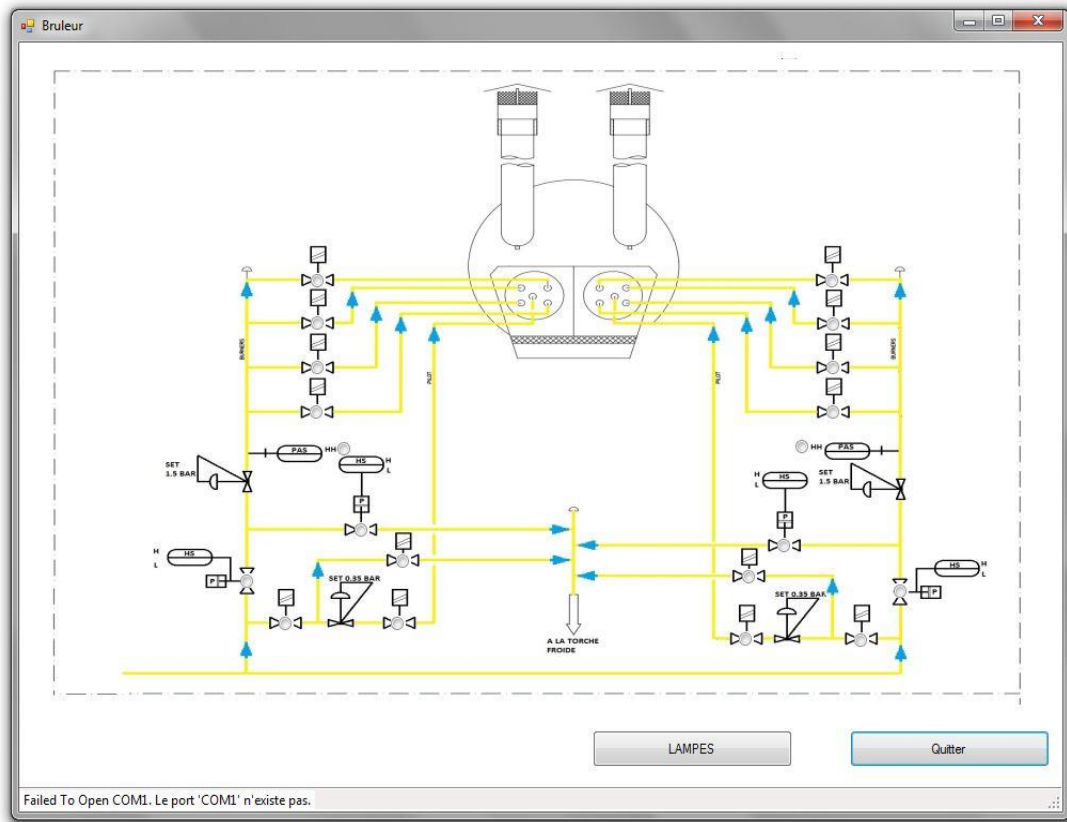


Figure 45 : Vue de côté (Bruleur).

A partir de cette fenêtre nous pouvons accéder à la page (Figure 46) qui affiche les différentes lampes.

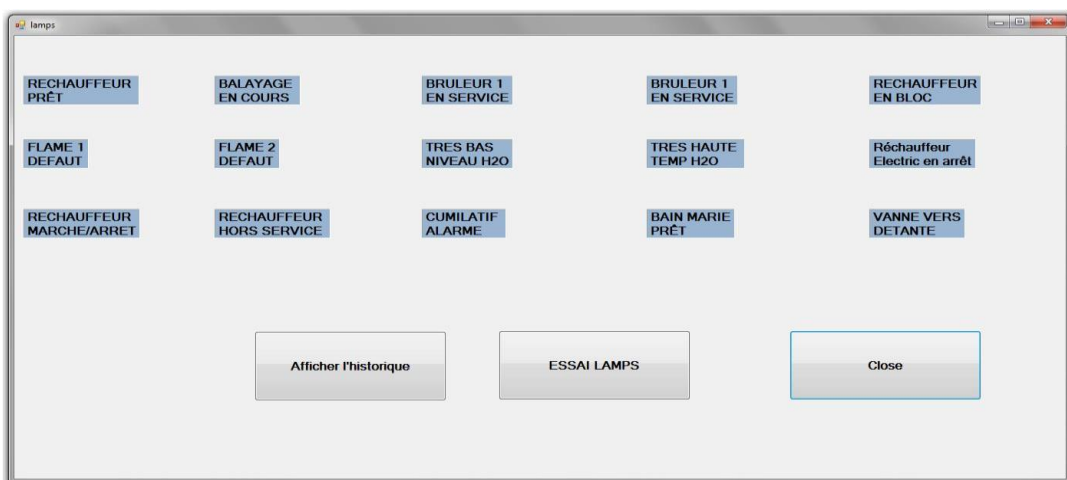


Figure 46 : interfaces d'affichage des lampes.

Ainsi, de même à partir de cette fenêtre (Figure 46) nous pouvons afficher l'historique des alarmes (Figure 47).

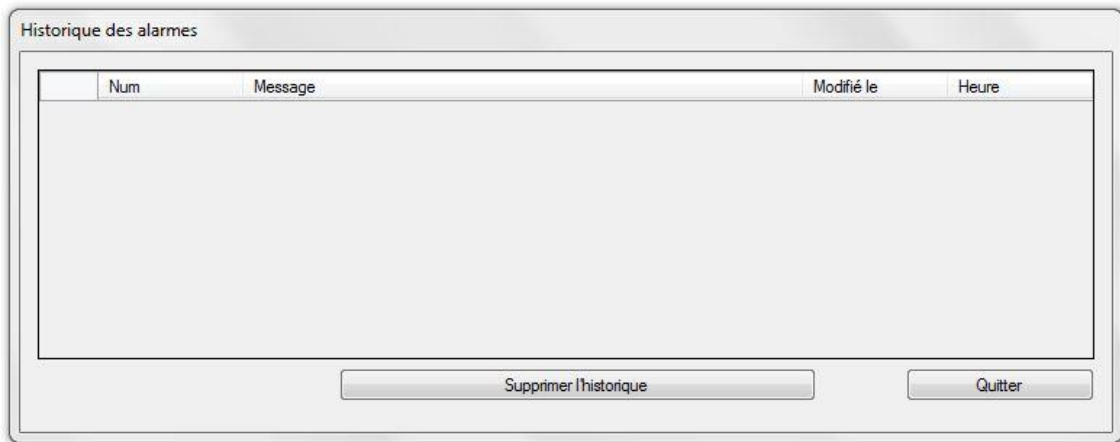


Figure 47 : Page de l'historique des alarmes.

- **Réchauffeur** : cette page nous donne une vue synoptique du réchauffeur (de face) (Figure 48), nous pouvons accéder à cette fenêtre en appuyant sur le bouton de la page d'accueil « Réchauffeur vue de face ».

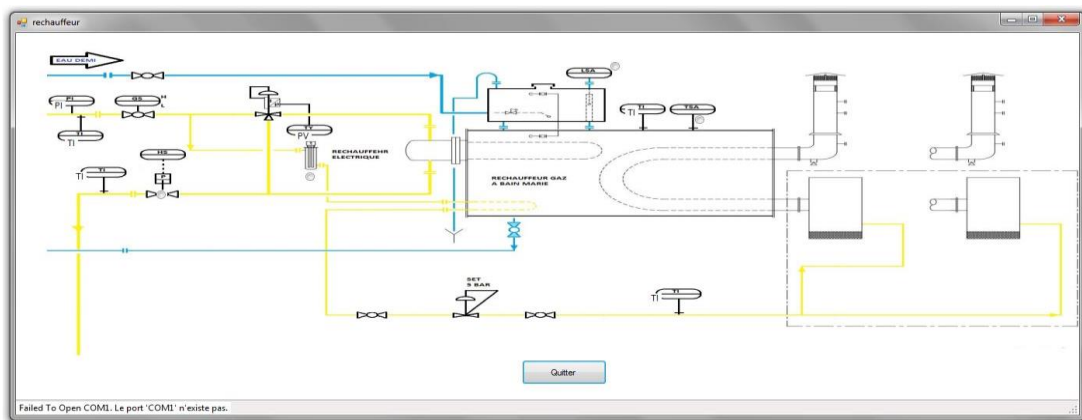


Figure 48 : Vue de face (réchauffeur).

- **Mode avancé** : nous ne pouvons pas accéder à cette page (Figure 49) sauf si nous disposons d'un mot de passe, elle permet de modifier les données en cas d'erreur, de gérer le démarrage et l'arrêt de l'API et d'afficher la courbe de données (Figure 50).

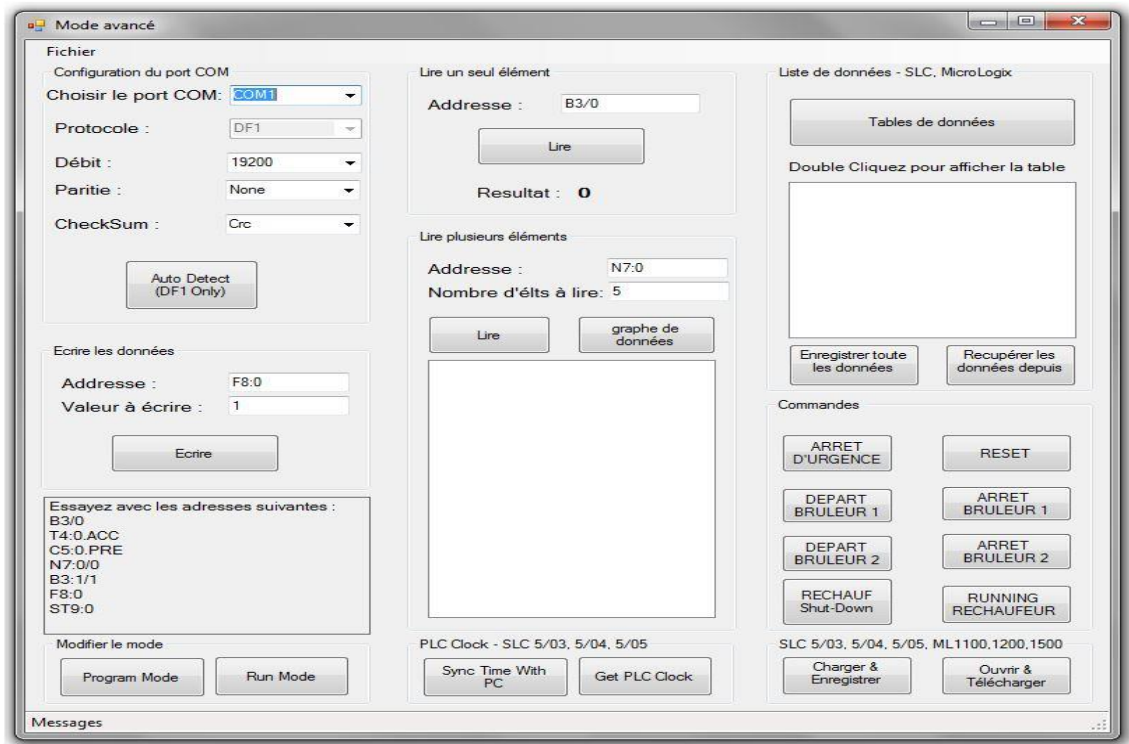


Figure 49 : Interface du mode avancé.

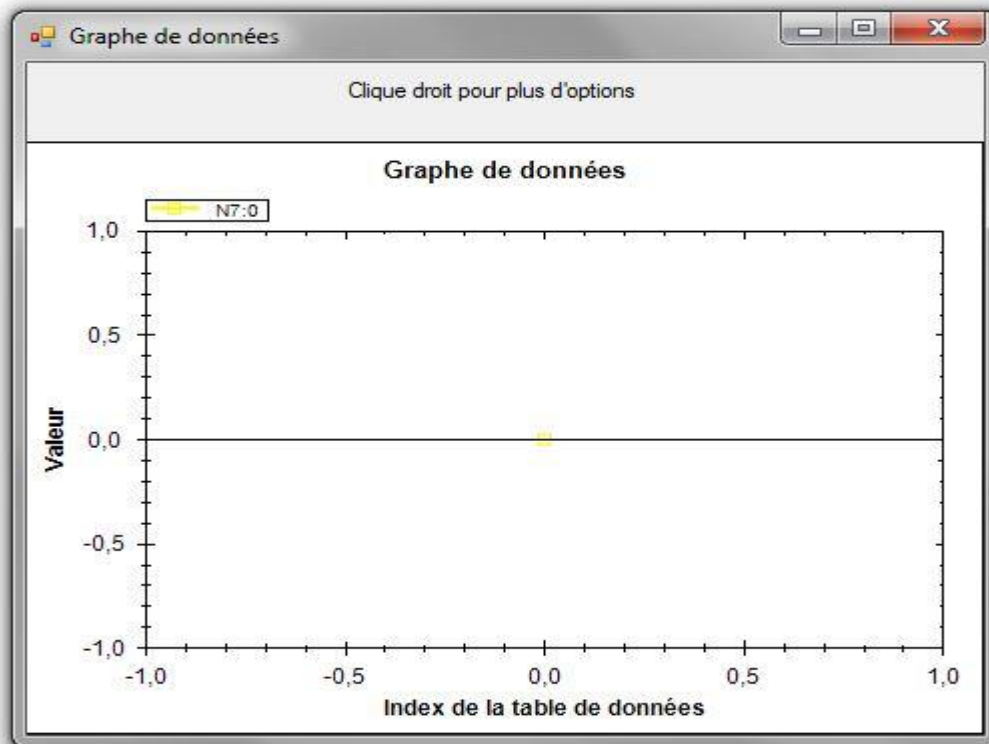
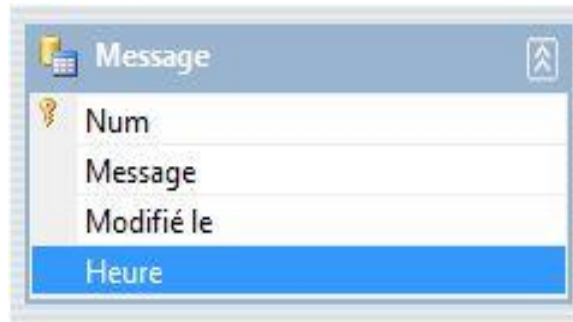


Figure 50 : Interface graphe de données.



### ***b* Base de données**

Elle sert à enregistrer l'historique des alarmes et les récupérer quand nous avons besoin de cela. Elle contient une seule table (Figure 51). Les informations concernant les alarmes forment les différents champs de cette table.



Message	
Num	
Message	
Modifié le	
Heure	

Figure 51 : Table de la Base de données.

## **4.5 Conclusion**

La première section de ce chapitre nous a permis d'avoir une vision générale sur la démarche à suivre.

La phase de conception est très importante dans n'importe quel projet informatique, dans la mesure où elle permet la maintenance de ce dernier, garantit son intégrité et facilite son expansion.

Dans ce chapitre nous avons essayé de montrer notre système qui répond aux exigences exprimées dans la section conception. Ainsi il est plus que nécessaire de respecter la conception lors du développement de l'application.

La technique de simulation que nous avons adoptée sera expliquée dans le prochain chapitre.

# Chapitre 5 Simulation et résultats obtenus

---

## 5.1 Introduction

Après avoir expliqué notre Projet et la procédure suivie pour le mettre en œuvre dans les précédents chapitres, ce chapitre a pour principale vocation d'exposer et de décrire la méthode que nous avons utilisée lors de la simulation.

Nous commencerons par présenter le principe de la simulation ainsi que ses limites et ses avantages. Puis, nous présentons le principe de fonctionnement d'une application de supervision sans et avec simulation ainsi que l'environnement que nous avons mis au place pour effectuer cette simulation.

## 5.2 Simulation

Les chercheurs, les ingénieurs et bien d'autres professionnels se posent souvent la question : quel est le résultat que j'obtiens si j'exerce telle action sur un élément ?

Le moyen le plus simple serait de tenter l'expérience, c'est-à-dire d'exercer l'action souhaitée sur l'élément en cause pour pouvoir observer ou mesurer le résultat. Dans de nombreux cas l'expérience est irréalisable, trop chère ou contraire à l'éthique.

C'est à cause des difficultés liées à l'expérimentation directe que le recours à la simulation est devenu nécessaire voire indispensable, ainsi la simulation consiste à concevoir un modèle du système (réel) étudié et de mener des expérimentations sur ce dernier afin d'interpréter les résultats fournis par le déroulement du modèle puis formuler des décisions relatives au système, selon les besoins.

Aujourd'hui, la simulation est utilisée dans de nombreux domaines de recherche et de développement : mécanique, science des matériaux, aéronautique, météorologie, ...etc. Elle intervient aussi dans des secteurs comme celui de la banque et des finances.

### **5.2.1 Les limites de la simulation**

Les objectifs premiers de la simulation étaient très ambitieux, en revanche, après son utilisation les professionnels et les chercheurs utilisant cette technique trouvent que le pouvoir de la simulation heurte à certaines limites, nous pouvons citer entre autres :

- La simulation est basée sur une connaissance des phénomènes qui ne peut être obtenue que par l'expérimentation. Une simulation ne peut donc être réalisée que si nous disposons d'un acquis de connaissances suffisant obtenu par des expérimentations sur des phénomènes antérieurs et analogues.
- Quelle que soit la qualité de la simulation, elle ne remplace pas totalement l'expérimentation.
- Certaines simulations ont un coût très élevé (même s'il reste faible devant celui de l'expérimentation).

### **5.2.2 Avantages de la simulation**

Dans la logique de faire plus vite, mieux et, surtout, moins cher, la simulation possède énormément d'atouts et cela est dû aux nombreuses contraintes économiques : réactivité, anticipation et compétitivité (gain de productivité), mais aussi des enjeux de sûreté et de sécurité, par une meilleure compréhension des situations accidentelles.

La simulation permet ainsi d'effectuer des recherches sur un système isolé, en faisant varier les paramètres un à un et en recommençant avec les mêmes conditions initiales.

### **5.2.3 Les raisons du succès de la simulation**

Les raisons du succès de la simulation sont variées :

- La disponibilité d'une puissance de calcul toujours croissante.
- La disponibilité d'outils, logiciels, conçus exprès pour la simulation.
- La nécessité et la volonté de manipuler des modèles de plus en plus détaillés et complexes.

## 5.3 Principe de fonctionnement

### 5.3.1 Sans simulation

Avant de commencer à construire un système de supervision sans simulation (Figure 52), il faut d'abord savoir qu'il y a deux dispositifs indispensables un PC et un API. Premièrement, afin qu'il puisse interagir avec son environnement et plus précisément avec un PC un API Allen Bradley SLC 5/03 possède deux ports de communication : le port « DH-485 » et le port « RS-232 ». Dans le cas d'une supervision par PC le port RS-232 est le plus utilisé.

Inopportunément, ce port n'est plus disponible dans les nouvelles générations de PC c'est pour cette raison que pour relier ces deux dispositifs entre eux il faut recourir à un outil intermédiaire appelé « adaptateur USB-RS232 », et un câble rs232 croisé male femelle.

Dans le but d'exploiter cette infrastructure matérielle et afin que les données échangées soient dans le bon format un driver doit être installé sur le PC. Cependant, pour réaliser la configuration du canal de communication entre un PC et un API Allen Bradley SLC 5/03 avec le driver DF1 un appel au gestionnaire de pilotes RSLinx doit être fait.

Dès que le lien de communication est établi, l'application de supervision pourra communiquer avec l'API et récupérer des données via leurs adresses.

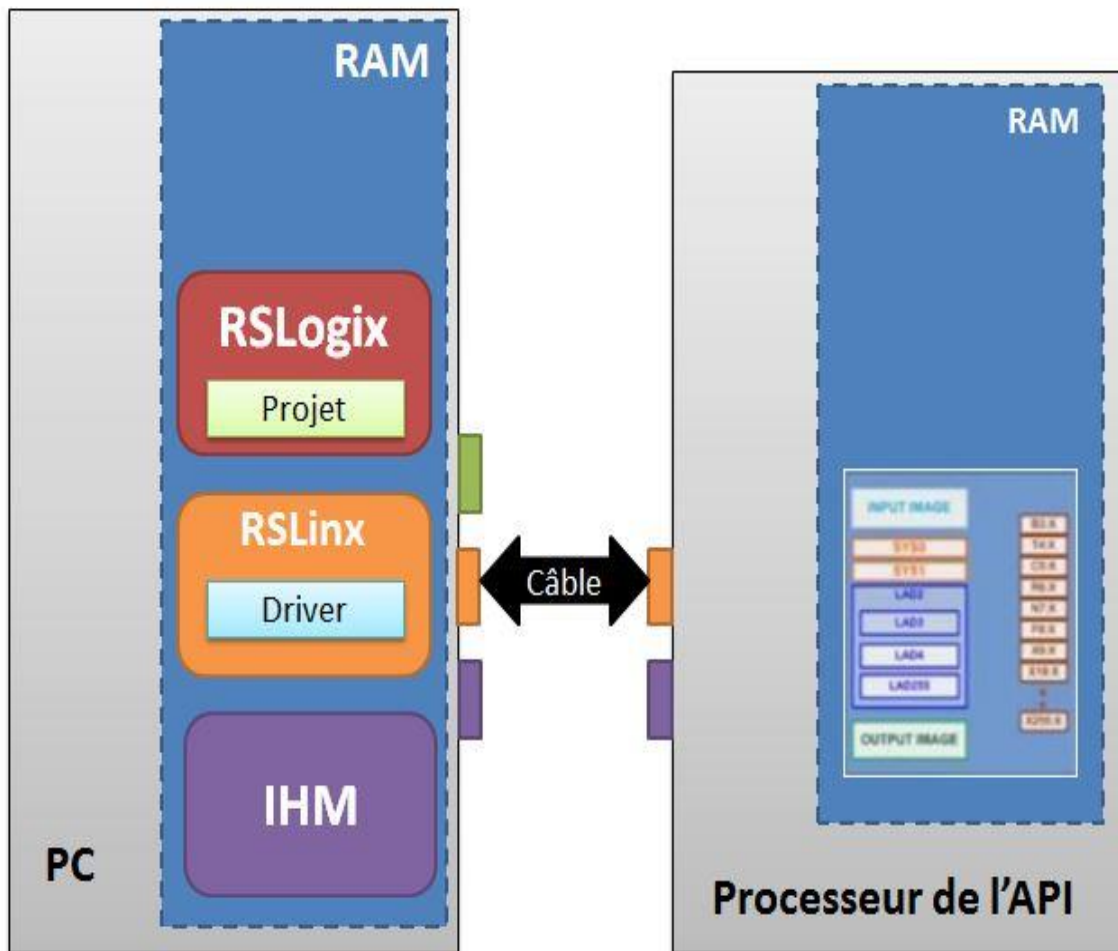


Figure 52 : Système de supervision sans simulation.

### 5.3.2 Avec simulation

Dans le cas d'une supervision avec recours aux outils de simulation (Figure 53). L'infrastructure matérielle est constituée d'un seul dispositif, le PC. Pour pallier ce manque l'infrastructure logicielle sera renforcée avec les logiciels RSLogix Emulate 500 et com0com.

Au début, l'émulateur remplace l'API partiellement car le logiciel n'est pas fabriqué avec des adresses d'E/S comme un API réel Allen Bradley SLC 5/03, donc il faut le définir. Le moyen le plus simple pour le définir (le rendre similaire à l'API réel) est de charger le programme réchauffeur gaz dans RSLogix Emulate 500 lorsque cela est fait, le programme sera interprété par l'émulateur et à ce moment-là nous pouvons dire que l'émulateur remplacera l'API intégralement. Pour que l'API simulé et le PC

puissent communiquer entre eux une paire de ports RS-232 est créé par le logiciel com0com.

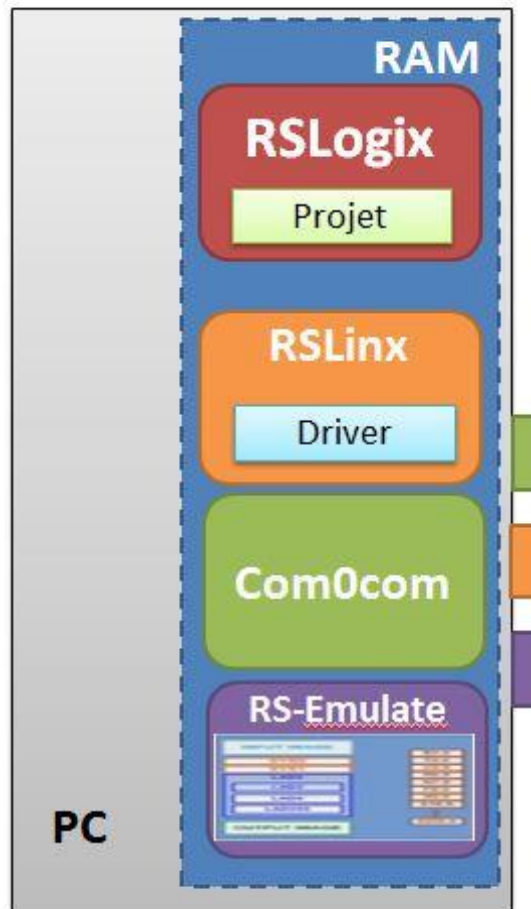


Figure 53 : supervision avec simulation.

## 5.4 Environnements de simulation

Il s'avère que pour réaliser la partie simulation de notre projet, l'utilisation de quatre logiciels est obligatoire, ce sont : RSLogix 500 (Annexe E), RS Linx Classic (Annexe B), RSLogix Emulate 500 (Annexe C) et Null-modem emulator (Annexe D) l'ensemble de ces quatre logiciels forme ce qu'on appelle simulateur.

La première étape dans le processus de simulation mène à créer notre simulateur, pour se faire nous avons besoin de télécharger les 4 logiciels et les installer (Annexes B, C, D et E).

Dans la deuxième étape il faudra faire en sorte que l'API soit intégré au niveau de notre PC. Pour cela il faut lancer nos logiciels et faire les manipulations nécessaires suivantes:

- Commenant par RSLinx, une fois lancé nous utilisons la barre d'outils pour créer un nouveau driver (Figure 54).

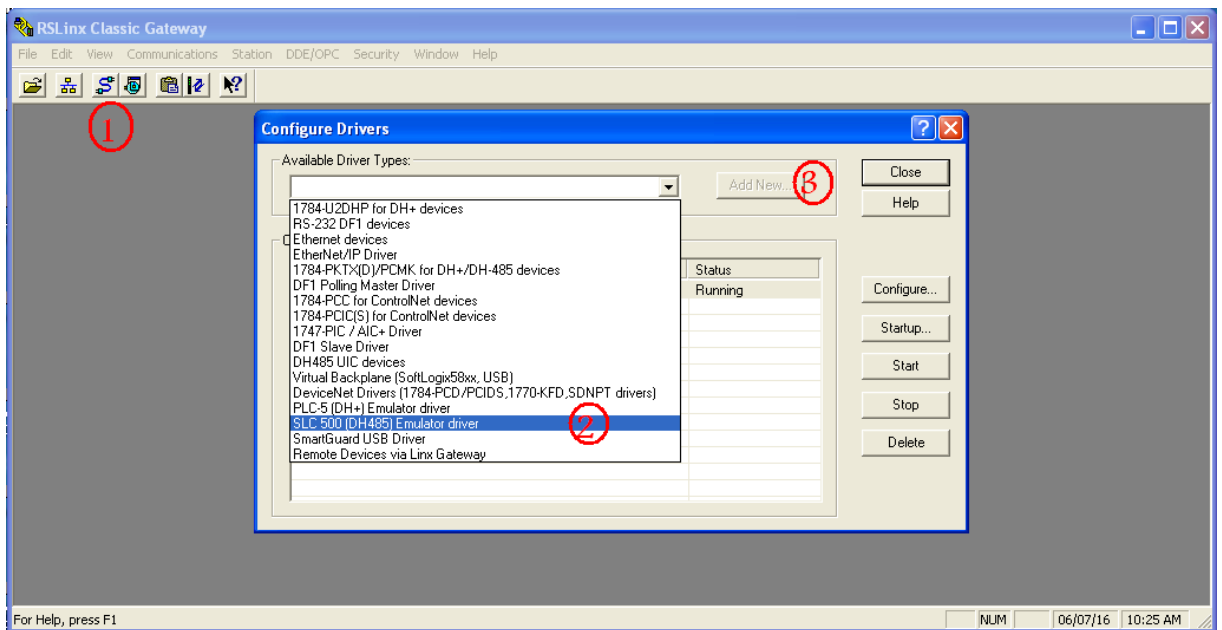


Figure 54 : Création d'un nouveau driver.

**Note :** nous avons choisi le type (SLC 500 (DH485) Emulator driver) car nous sommes en train d'utiliser RSLogix Emulate 500, si par contre nous travaillons avec un vrai API nous devons choisir RS-232 DF1 devices.

- Ensuite, nous lançons RSLogix Emulate 500 et à partir de la barre de menu, nous ouvrons le projet « réchauffeur gaz de LARBAA » comme indiqué à la (Figure 55).

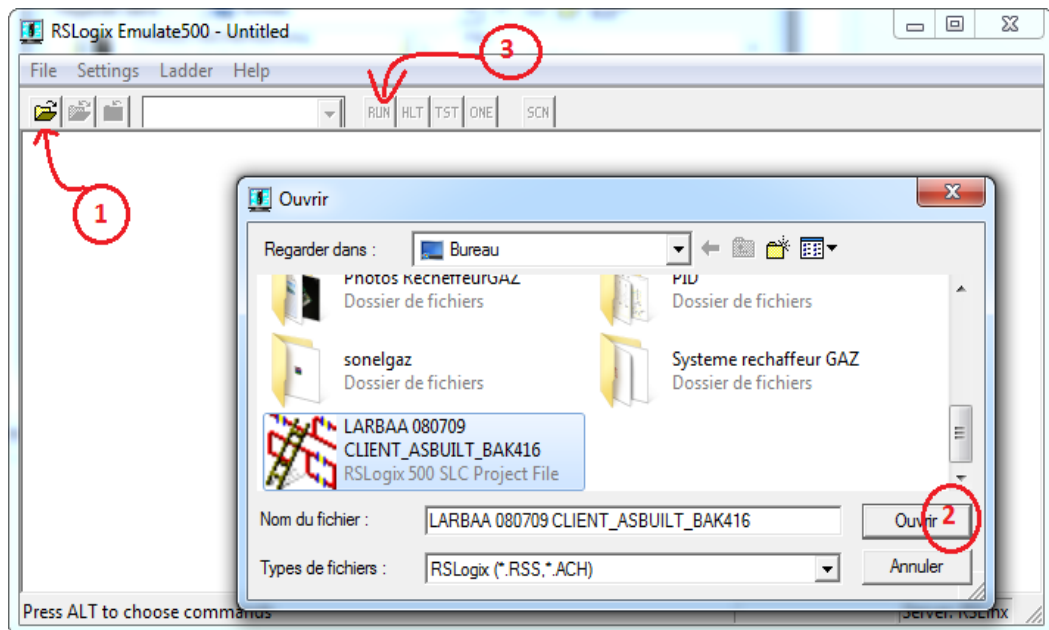


Figure 55 : Ouverture du projet sous RSLogix Emulate.

Une fois le projet ouvert nous allons procéder à la configuration du canal de transmission comme le montre la (Figure 56). Le port COM2 représente le port de l'API.

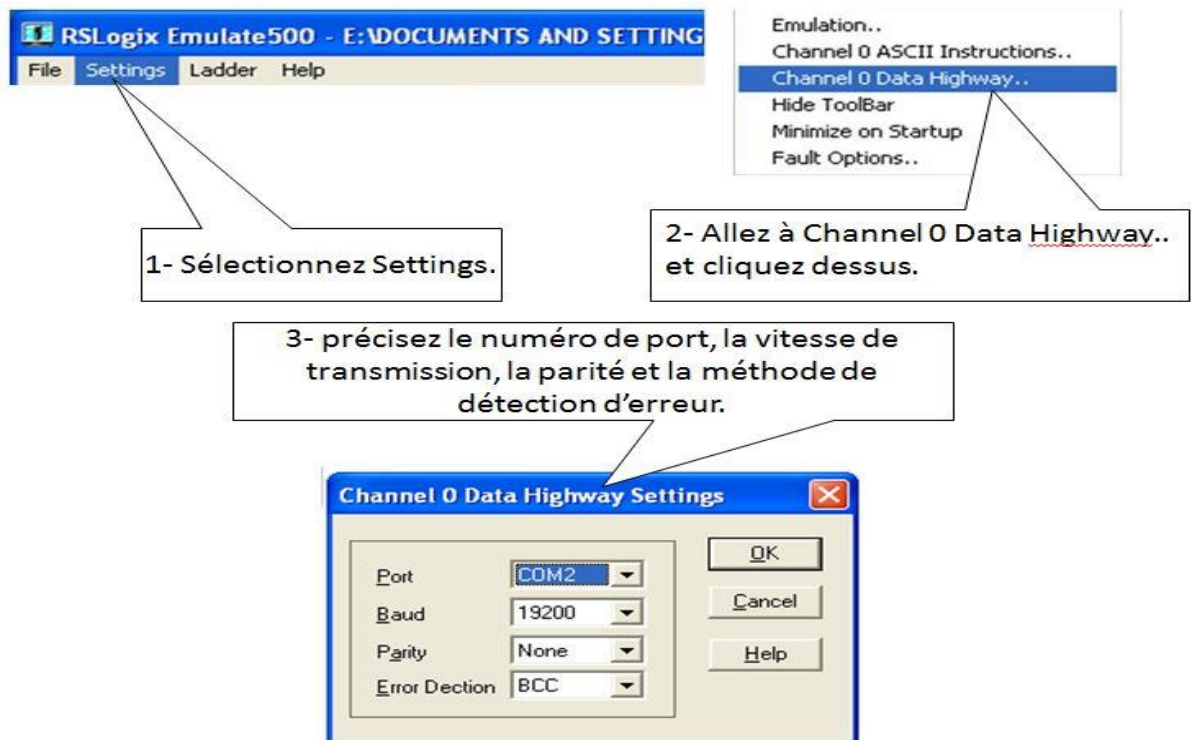


Figure 56 : Configuration du canal de transmission.



- Comme notre PC ne possède pas un port COM réel, alors que pour utiliser l'IHM un port série RS-232 est requis. nous lançons le simulateur de port COM « Null-modem Emulator ». et nous créons une paire de ports (Figure 57).

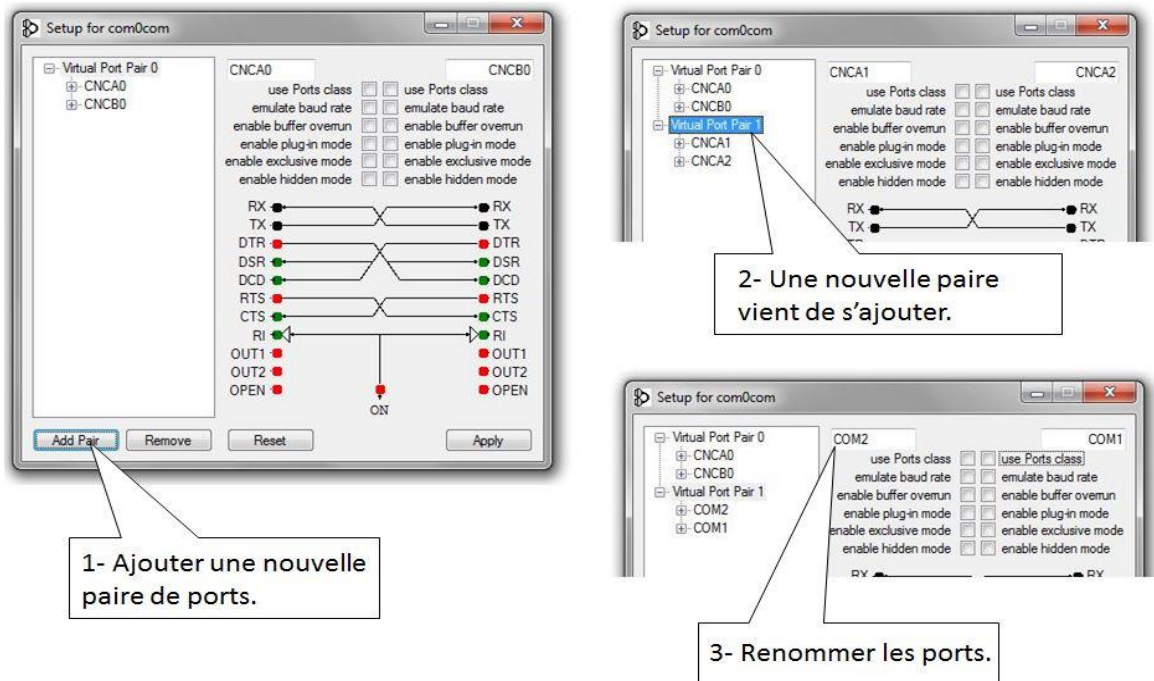


Figure 57 : Création d'une paire de ports.

- Par la suite, nous lançons « RSLogix 500 » avec le programme Ladder « réchauffeur gaz » ().

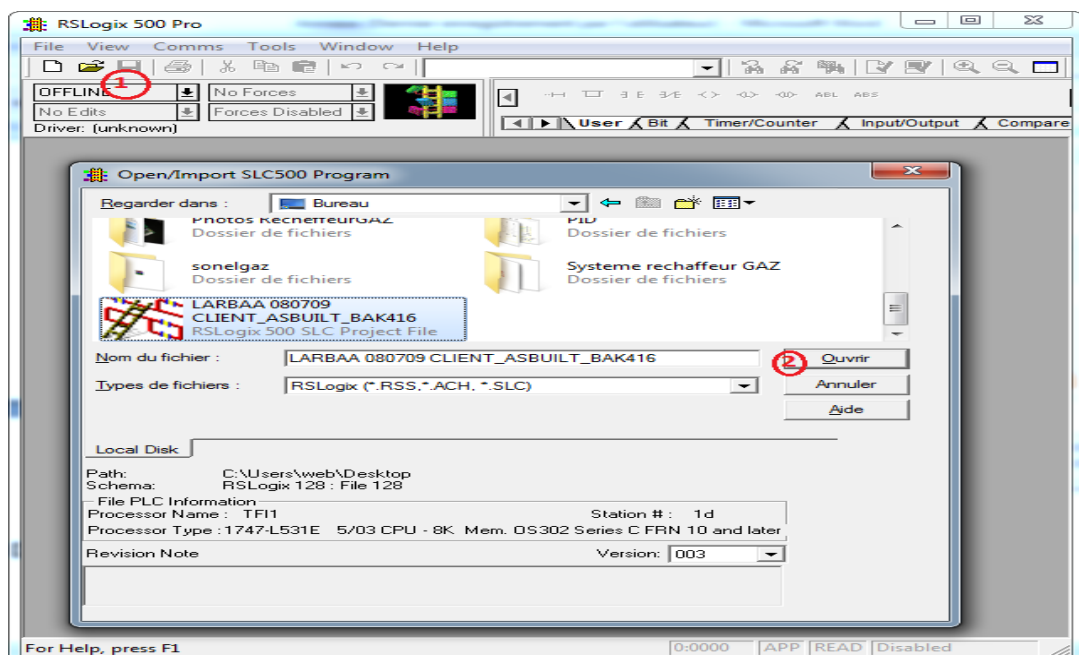


Figure 58 : lancement de RSLogix 500.

Puis nous choisissons le dispositif SLC 5/03 pour cela nous avons suivi les étapes indiqués à la figure suivante.

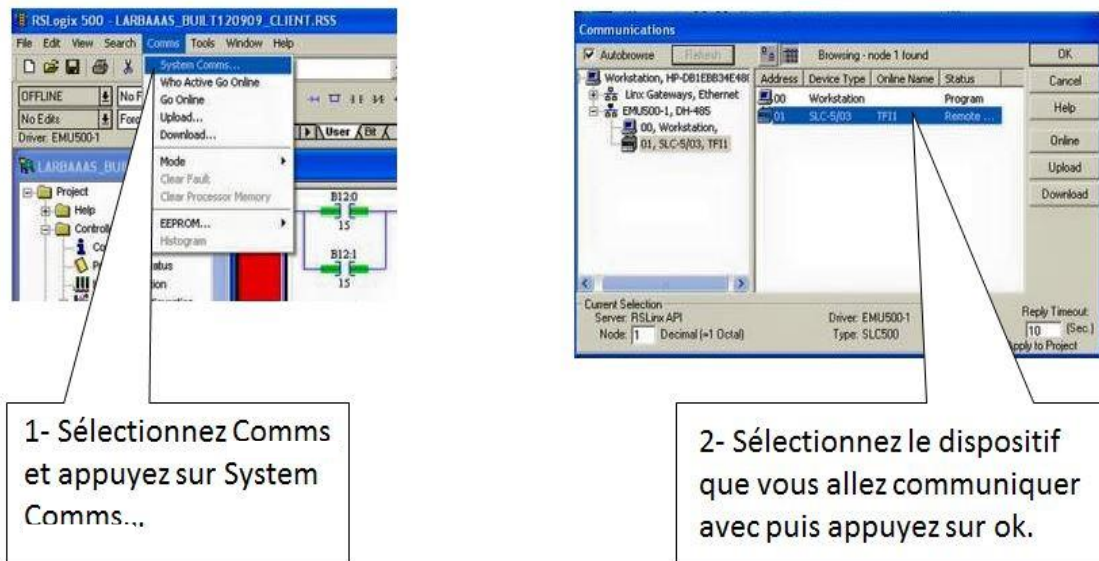


Figure 59 : Choix du dispositif.

A ce stade nous allons charger le programme « réchauffeur gaz » dans l’émulateur, pour cela nous allons suivre les étapes indiqués dans la Figure 60.

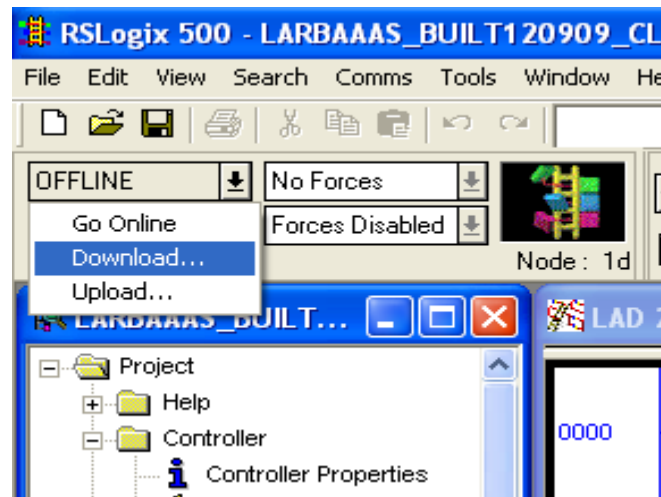


Figure 60: chargement du programme dans RSLogix Emulate 500.

Puis nous mettons le processeur en mode RUN (), si nous sommes en train de travailler avec un API réel nous devons mettre le mode de ce dernier en mode RUN en utilisant le bouton correspondant.

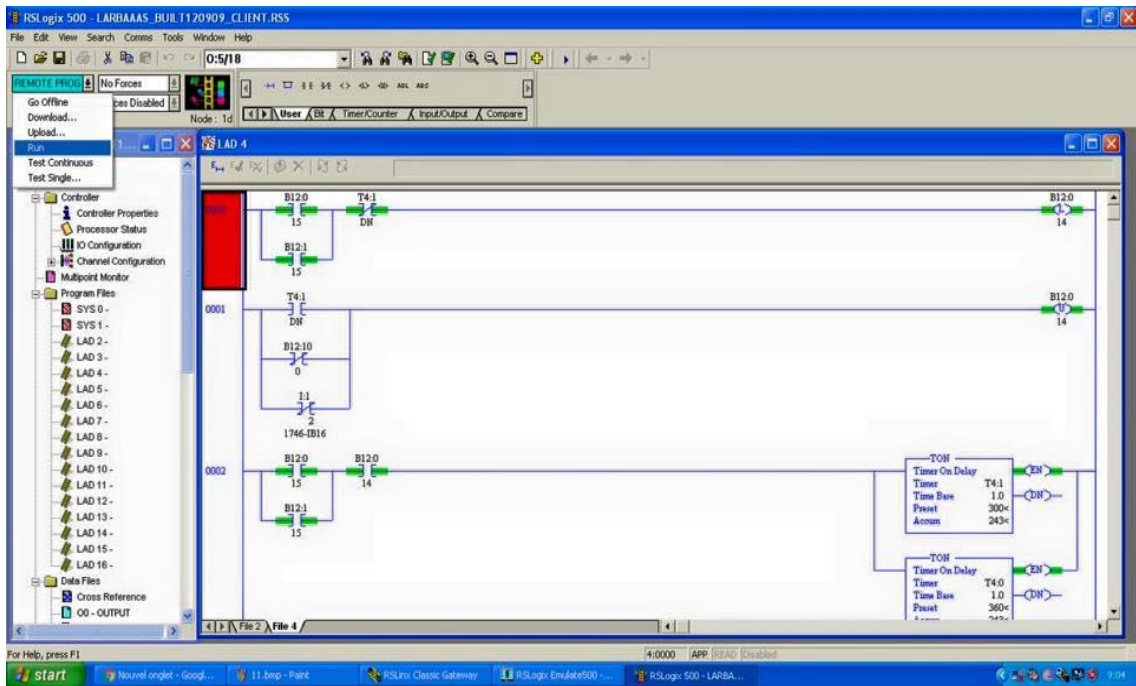


Figure 61 : Passage au mode RUN.

L'environnement de simulation est enfin mis en place. La procédure de création de l'émulateur et la configuration de ce dernier est achevée de manière satisfaisante.

## 5.5 Conclusion

Dans ce chapitre nous avons présenté la méthodologie que nous avons adoptée pour réaliser la partie simulation. Cette partie malgré qu'elle n'a pas été parmi les objectifs primaires de notre projet. Elle nous a permis d'avoir une vision claire sur la démarche à suivre lors de l'utilisation d'un API Allen Bradley, du fait qu'elle a ressorti les besoins en termes de logiciels en soulignant les étapes principales pour réussir à tester un système SCADA.

# Conclusion générale

---

Tout d'abord, nous tenons à souligner l'expérience très enrichissante acquise au cours de ce stage au sein de la centrale électrique de LARBAA, pour l'élaboration de ce projet de fin d'étude. Le travail qui nous a été demandé consiste à réaliser un système pour la supervision d'un API Allen Bradley de l'unité réchauffeur gaz à bain marie.

Nous avons réalisé un système de supervision qui consiste, à se connecter à l'automate via le protocole DF1 pour lire des entrées sorties qui sont en relation avec le processus de réchauffement du gaz. Nous avons non seulement arrivé à lire des entrées sorties depuis l'automate mais aussi à les modifier et à garder l'historique des alarmes. Il suffit qu'un opérateur utilisant notre application choisit le port adéquat, la vitesse de transmission, la parité et le checksum et la connexion est établie.

Néanmoins, durant la réalisation de ce projet plusieurs difficultés ont été rencontrées. Sans être exhaustif, nous citons :

- Le driver de communication, entre l'API et le PC, que nous devons utiliser n'est pas distribué par le constructeur, cela nous a pris un mois de recherche pour trouver une implémentation libre de ce driver.
- Comme les produits Allen Bradley appartiennent à un constructeur international, nous avons trouvé très peu de documentation francophone.
- L'une des principales difficultés à laquelle nous avons été confrontés, a été l'identification des entrées sorties de l'API car nous n'avons pas eu un accès tout à fait libre à l'API et le programme de ce dernier n'était pas commenté par ses développeurs.
- La diversité des logiciels SCADA a rendu le choix, de logiciel convenable pour le développement de notre système, très laborieux.

- Après avoir mis tous le paquet pour la réalisation de notre application et lorsque nous avons voulu la testée nous étions heurté par une décision, que nous jugeons irresponsable, de la part du directeur de la centrale électrique qui a interdit le test de notre application sur l'API réel. Ce qui nous a obligé à intégrer une partie simulation qui n'était pas prévu au début de ce projet en un laps de temps très court.
- Pour la réalisation de ce projet nous avons dû apprendre à manipuler cinq nouveaux logiciels : RSLogix 500 (Ladder), Visual Studio Express (VB), RSLinx (connexion et configuration du driver), Null-modem emulator com0com (utilisé pour fournir une paire d'interface série virtuelle pour RSEmulate) et le logiciel RSEmulate (pour tester le fonctionnement de notre programme sous l'API virtuel).

Au cours de ce projet nous avons acquis beaucoup de connaissances que ce soit sur le plan hard ou sur le plan soft. Ce projet nous a été triplement bénéfique du fait que :

- En premier lieu, nous avons eu la chance d'apprendre le langage de programmation informatique Visual Basic qui est un langage très puissant ainsi que d'avoir une idée sur le langage de programmation des API Ladder, d'interagir directement ou indirectement avec des équipements industriels, de comprendre leur rôle dans le processus de réchauffement et même de manipuler l'API via ces logiciels de base.
- En deuxième lieu, nous avons appris le principe des systèmes SCADA et les bonnes habitudes pour la création des interfaces graphique. Sans oublié la manipulation du logiciel de simulation et aussi les principes de base du protocole DF1.
- En dernier lieu, nous avons découvert la suite logicielle Allen Bradley (englobant RSLogix, RSLinx et RSEmulate) qui permet de faire beaucoup de chose dans le développement des programmes API. Ceci représente un véritable plus pour nous et nous ouvre d'autres perspectives d'avenir.

Bien que nous avons obtenu des résultats satisfaisant une application, avant qu'elle soit testée par un nombre représentatif d'utilisateurs, n'est jamais complètement terminée. Nous pouvons alors citer les perspectives et les développements suivants :

- Adapter notre application pour pouvoir superviser d'autres API Allen Bradley au niveau de la centrale.
- Mettre à jours l'application si d'autres fonctionnalités s'imposent au cours du temps.
- Tester des protocoles de communication distante pour mettre l'application sur le réseau.

# Annexe A - visuel studio express

---

## Définition

Microsoft a créé une suite logicielle nommée « Visual Studio », qui rassemble Visual Basic, Visual C++, Visual C#, ...etc. Cette suite est payante mais heureusement, l'éditeur nous propose généreusement une version « Express » gratuite de chaque logiciel de cette suite.

Nous allons donc utiliser Visual Basic 2010 Express (les étudiants peuvent toujours récupérer une version de Visual Studio 2010 sur la MSDN pour étudiants).

## Installation de Visual Basic 2010 Express

Passons immédiatement au téléchargement du petit logiciel intermédiaire, qui ne pèse que quelques Mo et qui va télécharger Visual Basic 2010 Express. Vous lancez donc le programme et arrivez à la première page (figure suivante).



Figure A. 1 : Installation de Visual Basic 2010 Express.

Sur cette page, l'installateur nous propose déjà une case à cocher. Si nous autorisons Microsoft à récupérer des informations sur notre ordinateur et des statistiques pour ses bases de données.

Cliquons ensuite sur le bouton « Suivant ». Lisons puis acceptons les termes du contrat de licence. Cela fait, appuyez une nouvelle fois sur « Suivant ».

Comme pour n'importe quelle autre installation, nous choisissons le dossier dans lequel nous souhaitons que le logiciel s'installe. Nous cliquons ensuite sur « Installer ». Une nouvelle page apparaît. Elle indique la progression du téléchargement du logiciel, le taux de transfert et la partie du programme en cours d'installation, comme à la figure suivante.

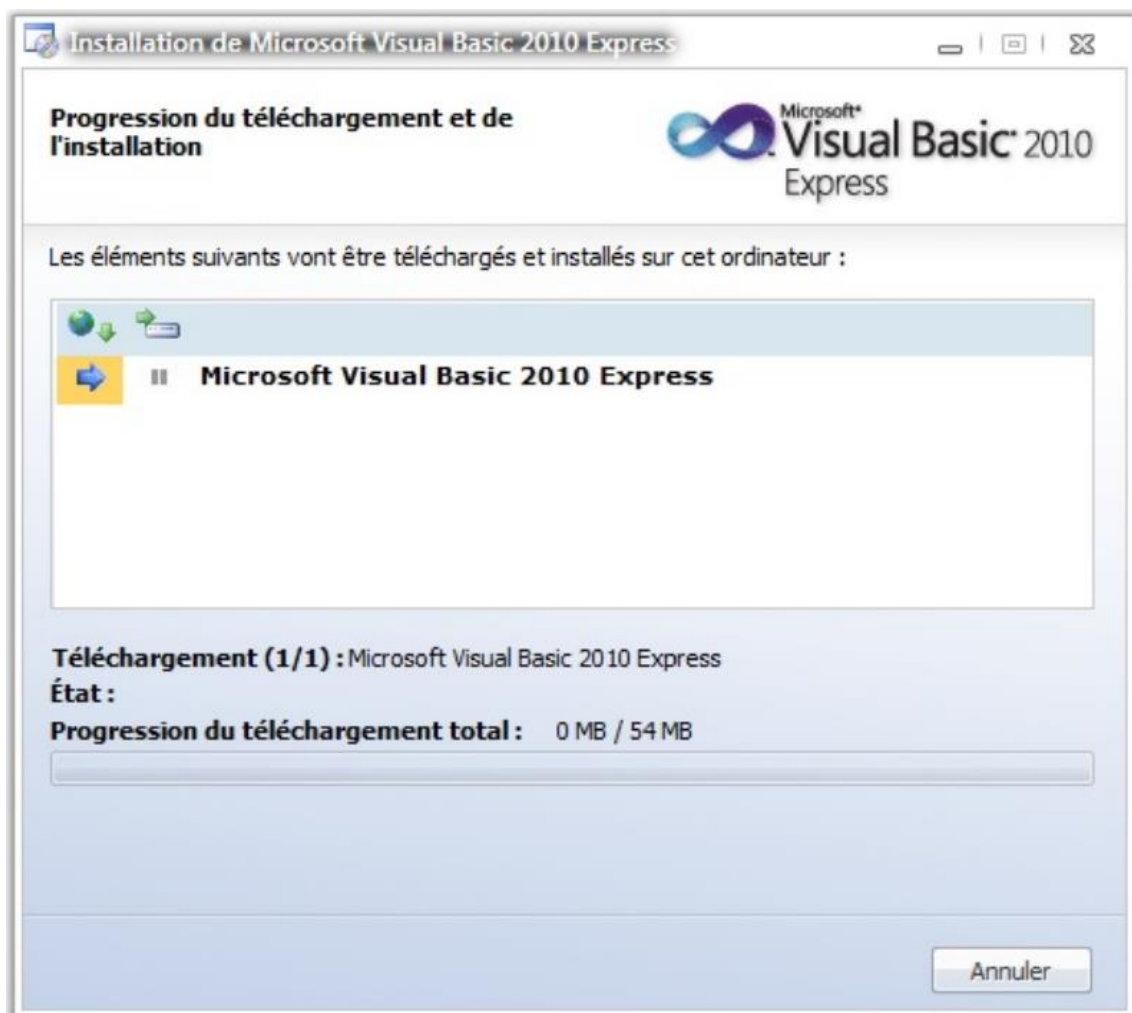


Figure A. 2 : Visual Basic 2010 Express est en cours de téléchargement.

Il ne nous reste plus qu'à attendre la fin du téléchargement, suivi de l'installation. Une fois cela terminé, nous voilà avec Visual Basic 2010 Express installé.



## L'interface de VB 2010 Express

Ce dernier est un IDE (environnement de développement intégré) qui rassemble les fonctions de conception, édition de code, compilation et débogage. Lors du premier lancement, nous constatons qu'un petit temps de chargement apparaît : le logiciel configure l'interface pour la première fois.

## Page d'accueil

Nous voici sur la page de démarrage du logiciel (Figure suivante). Nous pouvons la parcourir, elle contient des informations utiles aux développeurs et conservera l'historique de vos projets récents.

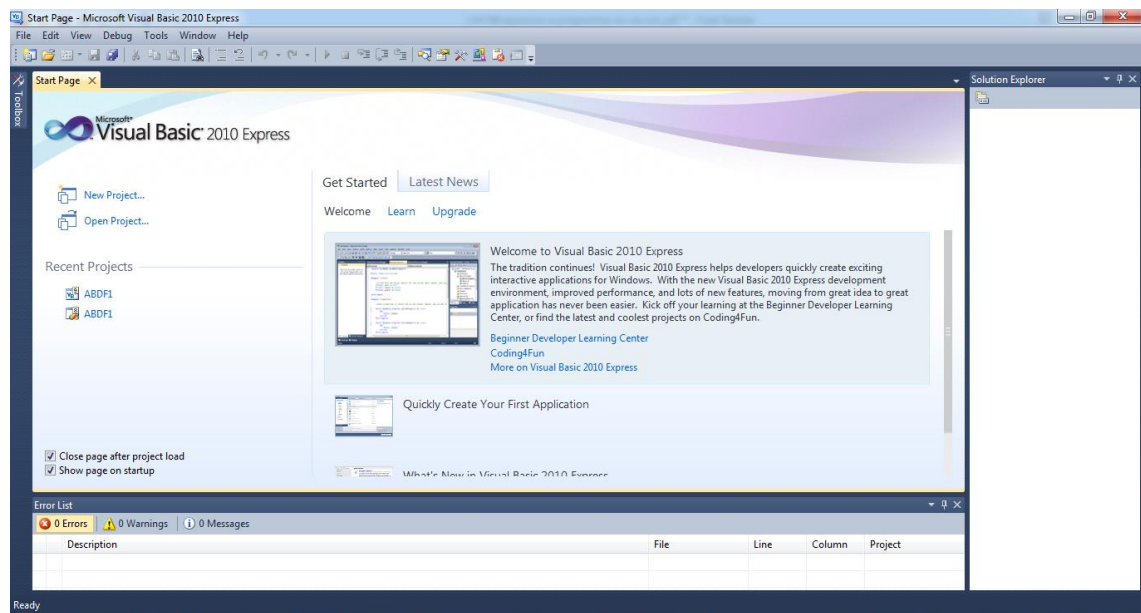


Figure A. 3 : Page de démarrage.

## Nouveau projet

Pour créer un projet Windows Forms (Figure suivante). Trois solutions s'offrent à nous: cliquer sur le bouton « Nouveau projet », se rendre dans le menu Fichier > Nouveau projet, ou utiliser le raccourci clavier CTRL + N.

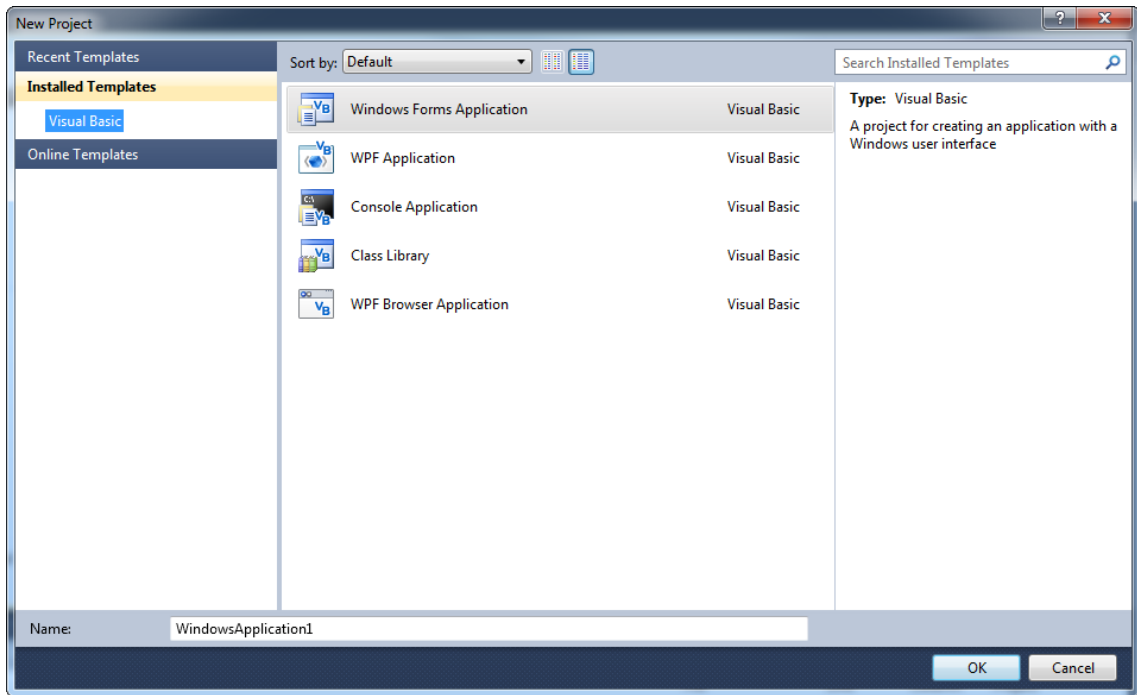


Figure A. 4 : Créer un nouveau projet.

Nous cliquons donc sur l'icône correspondant à « **Application Windows Forms** ». Nous saisissons un nom de projet dans la case « Nom ». Nous cliquons ensuite sur «OK», et nous voici dans un nouveau projet. Nous pouvons remarquer que beaucoup plus de choses s'offrent à nous (voir figure suivante).

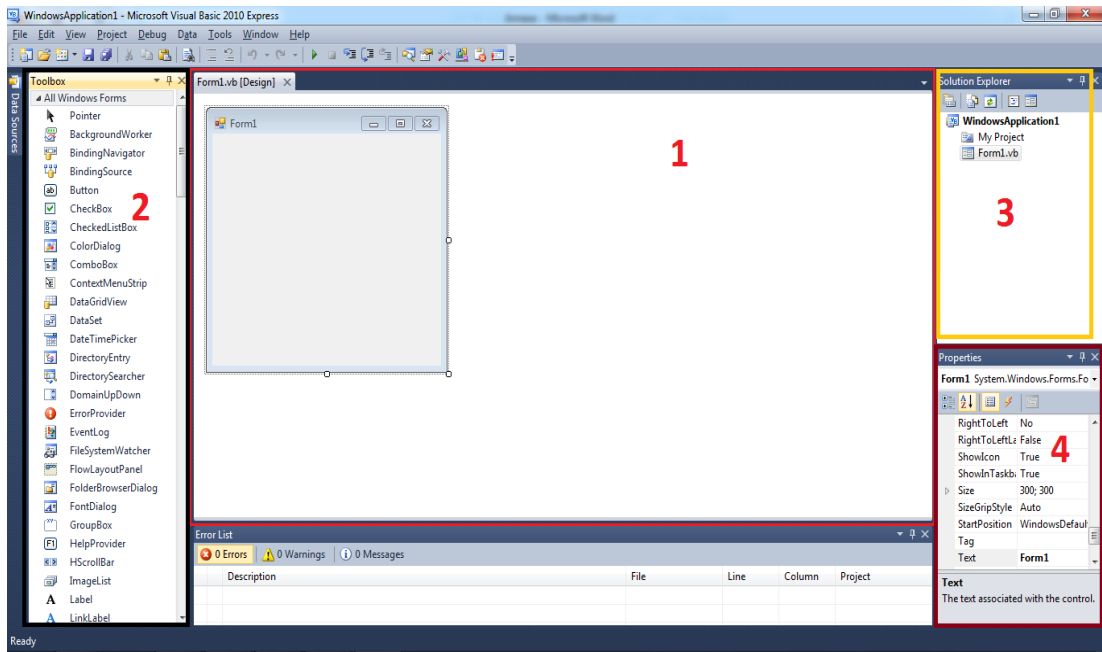


Figure A. 5 : Projet « Application Windows Forms ».

## **Espace de travail**

La partie (1) correspond à notre espace de travail : c'est ici que nous allons créer nos fenêtres, entrer nos lignes de code, etc.

Sur la gauche de l'interface, nous avons accès à la boîte à outils. Pour afficher cette boîte, nous devons cliquer sur le petit onglet qui dépasse sur la gauche. Une fois la boîte sortie, cliquons sur la punaise pour la fixer et la maintenir visible. La boîte à outils (2) nous sera d'une grande utilité lorsque nous créerons la partie graphique de nos applications, mais inutile lors de l'écriture du code VB.

## **Fenêtre de solutions**

De l'autre côté de notre écran, nous remarquons la fenêtre de solutions (3) : elle récapitule l'arborescence de notre projet. Elle servira également à gérer les bases de données.

## **Fenêtre des propriétés**

Autre partie essentielle : la fenêtre des propriétés (4) qui va nous permettre, en mode conception, de *modifier les propriétés* de nos objets.

# Annexe B – RSLinx Classic

---

## Définition

RSLinx Classic pour réseaux et dispositifs Rockwell Automation est une solution complète de communications industrielle pour les systèmes d'exploitation Windows.

Il fournit aux automates programmables Allen Bradley l'accès à une grande variété d'applications Rockwell Software et Allen-Bradley : des applications de programmation et de configuration de dispositifs, telle que RSLogix, aux applications IHM telles que RSVIEW32, en passant par nos propres application d'acquisition de données utilisant Microsoft Office, des pages Web ou Visual Basic®.

RSLinx Classic comprend également des techniques avancées d'optimisation de données avancée et contient un ensemble de diagnostics. Son interface de programmation d'applications (API) prend en charge les applications personnalisées développées avec RSLinx Classic SDK. RSLinx Classic est un serveur compatible avec OPC et un serveur DDE.

## Installation du logiciel RSLinx Classic

Pour installer RSLinx Classic, procédez comme suit :

- 1 – lancez l'installable de RSLinx Classic: le programme de configuration démarre automatiquement et l'écran d'accueil RSLinx Classic s'affiche (Figure B. 1).

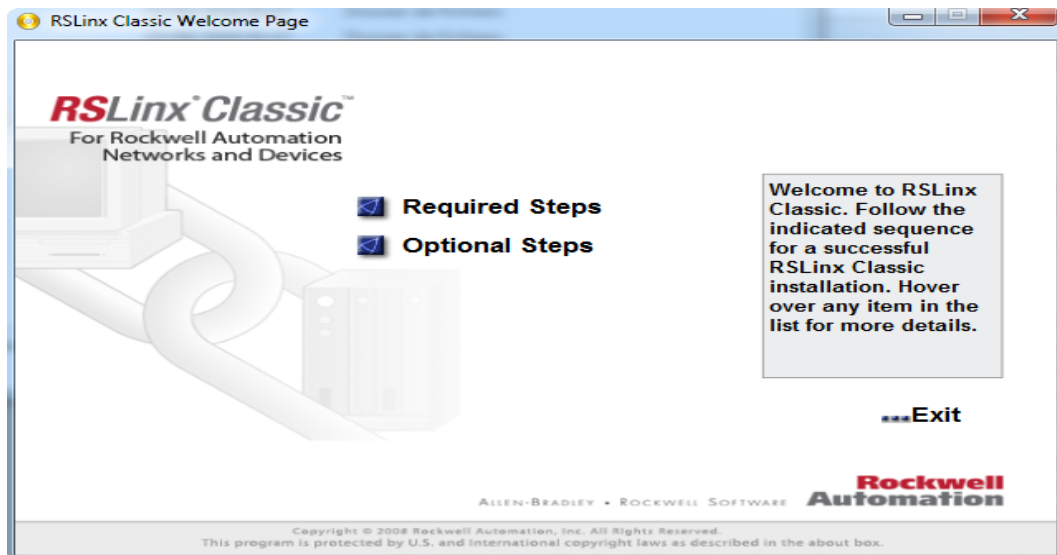


Figure B. 1 : La boîte de dialogue Installation RSLinx Classic.

2- Suivez les instructions qui s'affichent à l'écran : Dans la boîte de dialogue Bienvenue (Welcome), cliquer sur « Required Steps », puis choisissez **Install RSLinx Classic** (Figure B. 2).

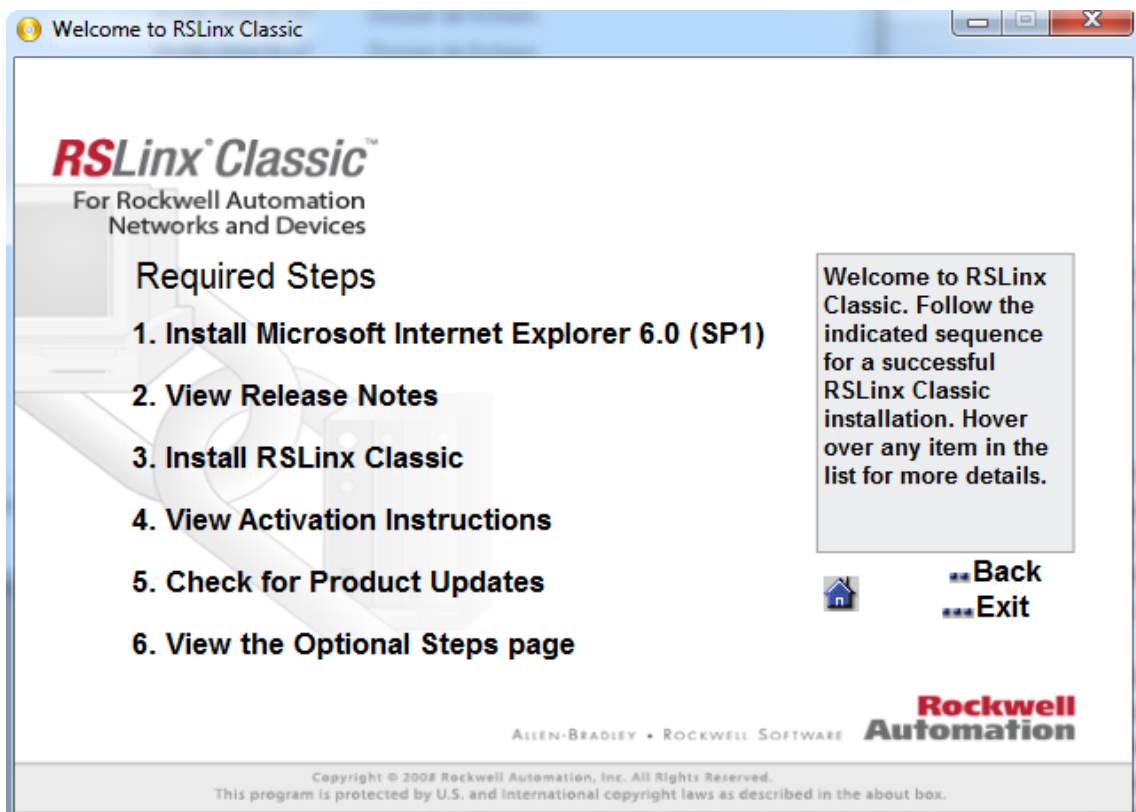


Figure B. 2 : Required Steps.

3- Dans la boîte de dialogue Bienvenue (Figure B. 3), lisez l'introduction à RSLinx Classic, puis cliquez sur **Suivant**.

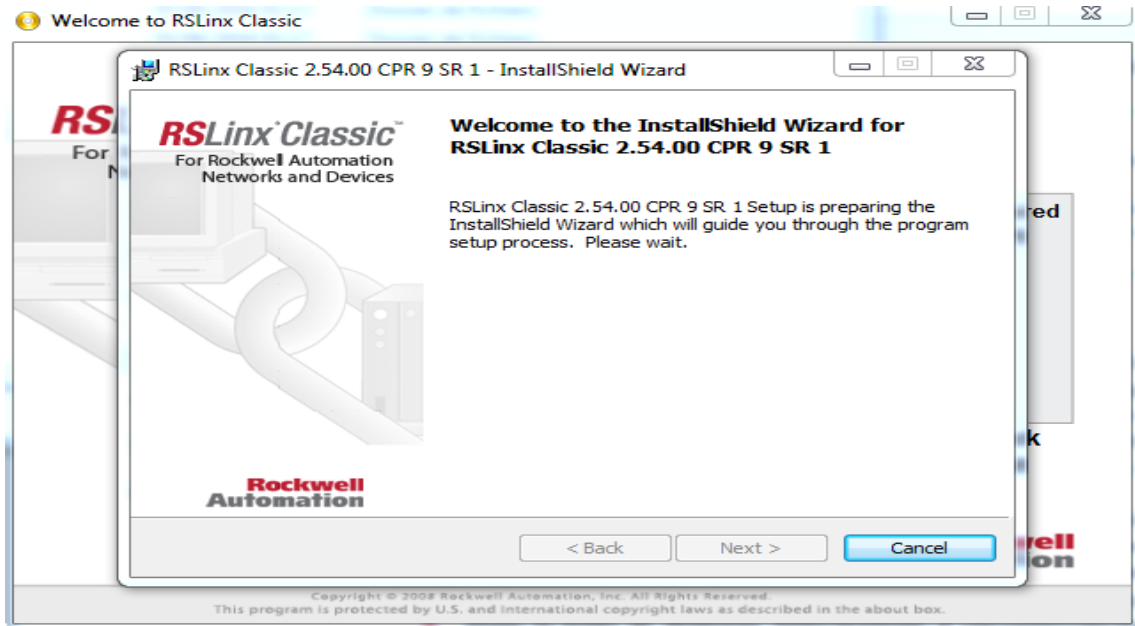


Figure B. 3 : La boîte de dialogue bienvenue.

Dans la boîte de dialogue Informations d'enregistrement client saisissez votre nom d'utilisateur, le nom de votre organisation et le numéro de série de votre logiciel RSLinx Classic, puis cliquez sur **Suivant**. Le logiciel RSLinx est à présent installer.

### Démarrage du logiciel RSLinx Classic

Pour démarrer le logiciel RSLinx Classic, cliquez sur Démarrer, puis sélectionnez Programmes > Rockwell Software > RSLinx > RSLinx Classic > vous obtenez (Figure B. 4).

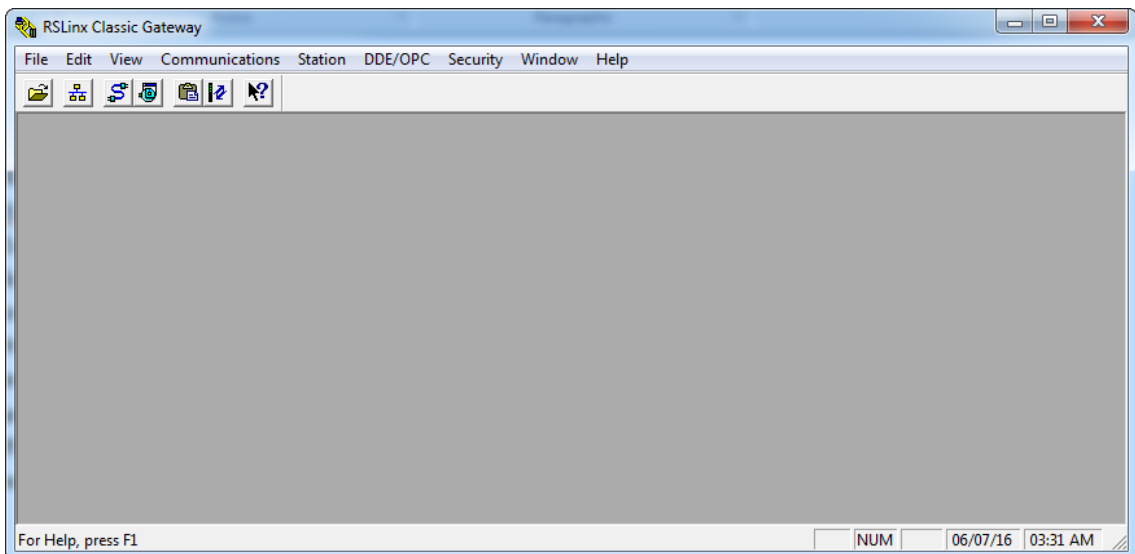


Figure B. 4 : page d'accueil de RSLinx Classic.

## Exploration de RSLinx Classic

### Barre de menus

Chaque menu contient des options pour effectuer les tâches indiquées dans la figure ci-dessous:

<b>Menu</b>	<b>Description</b>
Fichier	Créer et ouvrir des projets RSLinx.
Edition	Copier des liaisons DDE et OPC dans le Presse-papiers.
Affichage	Définir et modifier les affichages de l'interface RSLinx, ouvrir l'afficheur d'événements et sélectionner l'affichage RSWho.
Communications	Configurer des drivers, des sujets et d'autres options RSLinx, et afficher des diagnostics de drivers, de DDE et d'autres applications client.
Station	Effectuer des actions sur des compteurs de diagnostic et afficher le Contrôleur de données.
DDE/OPC	Configurer des sujets DDE/OPC et afficher des informations sur les événements et les diagnostics.
Sécurité	Définir un utilisateur de sécurité et des droits d'accès.
Fenêtre	Réorganiser des fenêtres RSLinx.
Aide	Afficher les options d'aide de RSLinx et des autres produits et services Rockwell Software.

Figure B. 5 : Fonctionnalités de la barre de menus.

### Barre d'outils

La barre d'outils contient des raccourcis vers plusieurs fonctions RSLinx Classic couramment utilisées. Chaque bouton de la barre d'outils est la représentation graphique d'une commande à laquelle on peut également accéder à partir de la barre de menus RSLinx Classic (Figure B. 7). Les éléments suivants apparaissent sur la barre d'outils de RSLinx Classic (Figure ci-dessous).



Figure B. 6 : Barre d'outils.








<b> Icône </b>	<b> Sélection par le menu </b>	<b> Description </b>
	Fichier > Ouvrir projet	Affiche les projets actuellement définis et vous permet d'ouvrir un projet DDE/OPC.
	Communications > RSWho	Ouvre une instance supplémentaire de RSWho (une instance est ouverte par défaut chaque fois que vous lancez RSLinx).
	Communications > Configuration des drivers.	Affiche les drivers RSLinx actuellement configurés et vous permet d'ajouter d'autres drivers à utiliser avec votre matériel.
	Communications > Diagnostics du driver	Affiche une liste des drivers actuellement configurés et permet de visualiser les informations de diagnostic pour chaque driver.
	Edition > Copier liaison DDE/OPC	Permet de créer une liaison DDE/OPC entre RSLinx et une application client telle que Microsoft Excel.
	DDE/OPC > Configuration du sujet	Vous permet de créer et de modifier un sujet DDE/OPC qui est un chemin spécifique vers un processeur.
	Aide > q u'est-ce que c'est è	Change le curseur en une flèche et un point d'interrogation pour vous indiquer que vous êtes en mode d'aide q u'est-ce que c'est è Cliquez sur un élément d'écran pour afficher le texte d'aide correspondant et pour quitter le mode d'aide q u'est-ce que c'est è

Figure B. 7: Description des fonctionnalités des icônes de la boîte d'outils.

### Espace de travail de l'application

L'espace de travail de l'application affiche des sous-fenêtres, telles que les boîtes de dialogue RSWho et RSLinx Classic.

### Barre d'état

La barre d'état au bas de l'écran de RSLinx Classic fournit des informations sur l'état actuel de votre système.

### Utilisation de RSWho

RSWho est la fenêtre principale de RSLinx Classic qui affiche les réseaux et les appareils de la même façon que l'Explorateur de Windows. En cliquant sur le bouton droit de la



sourie dans RSWho, vous aurez accès à une grande variété d'outils de configuration et de contrôle intégrés.

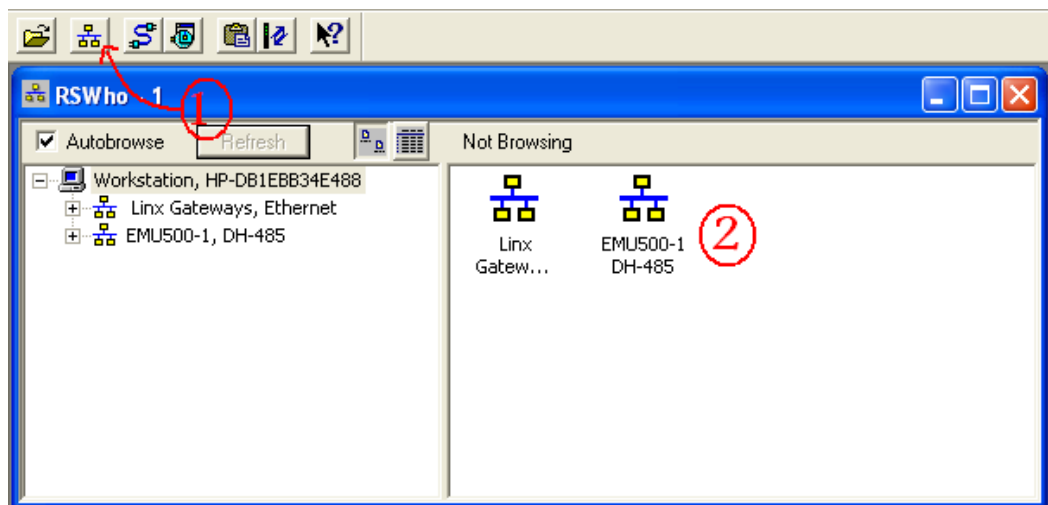


Figure B. 8 : Fenêtre de RSWho.

Le panneau gauche de RSWho est l'arborescence, qui affiche les réseaux et les appareils. Le panneau droit est la liste, où vous pouvez voir tous les membres d'un groupe. Un groupe est soit un réseau soit un appareil qui est une passerelle. Cliquez sur la liste avec le bouton droit de la souris sur liste et choisissez une option d'affichage de grandes icônes ou détaillée.

# Annexe C – RSLogix Emulate 500

---

## Définition

La famille de produits RSLogix offre plusieurs solutions aux divers aspects de programmation logique à relais. Parmi elles, RSLogix Emulate 500 sous Windows, qui émulent une ou plusieurs stations processeur SLC 500 ou PLC-5. RSLogix Emulate scrute la logique à relais comme un véritable processeur et vous permet de tester et de déboguer les programmes en logique à relais depuis votre bureau avant le démarrage. Il permet de :

- Exécuter votre programme Ladder sur votre PC.
- Mettre à jour vos tables de données.
- Rapprocher ce qui va ce passé lorsque vous télécharger votre programme dans un processeur PLC 500 physique.

## Installation du logiciel RSLogix Emulate 500

Pour installer RSLogix Emulate 500, procédez comme suit :

- 1- Lancez l'exécutable de RSLogix Emulate 500, Le programme d'installation démarre automatiquement et la boîte de dialogue d'accueil s'affiche.
- 2- Avant de passer à l'installation vous devez lire les notes concernant le logiciel, alors cliquez sur « View RSLogix Emulate 500 Documentation » puis cliquez sur le bouton « Read RSLogix Emulate 500 Release notes », quand vous terminez cliquez sur « Back ».
- 3- Cliquez sur « Install RSLogix Emulate 500 », et suivez les instructions qui s'affichent à l'écran.
- 4- **Dans la boîte de dialogue d'accueil** : lisez les informations d'introduction, puis cliquez sur Suivant.

- 5- **Dans la boîte de dialogue Contrat de licence logiciel** : lisez entièrement le contrat de licence du logiciel. Cliquez sur Oui pour accepter et continuer l'installation ou sur Non pour décliner et quitter l'installation. Puis cliquez sur Suivant.
- 6- **Dans la boîte de dialogue Informations d'enregistrement** : tapez votre nom, celui de votre société, le numéro d'identification de support du logiciel RSLogix Emulate 500, puis cliquez sur Suivant.
- 7- **Dans la boîte de dialogue Type d'installation** : Sélectionnez soit une installation complète (recommandée) soit une installation personnalisée puis cliquez sur Suivant.
- 8- **Dans la boîte de dialogue d'installation** : Consultez les informations expliquant comment contacter Rockwell Software, puis cliquez sur **Install**.
- 9- **Dans la boîte de dialogue annonçant la fin de l'installation** : Sélectionnez les options d'activation et d'affichage des notices et cliquez sur Terminer.

## Démarrer le logiciel RSLogix Emulate 500

Pour démarrer le logiciel RSLogix Emulate 500, cliquez sur Démarrer, puis sélectionnez Programmes > Rockwell Software > le fichier RSLogix Emulate 500 > l'exécutable RSLogix Emulate 500.

## Exploration de RSLogix Emulate 500

Les commandes de base de RSLogix Emulate 500 se trouvent sur la barre d'icônes ( ):

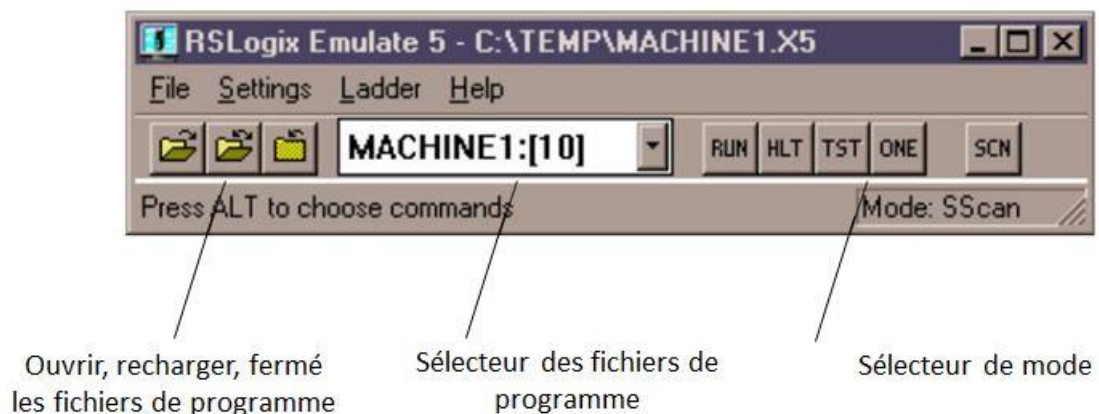


Figure C. 1: barre d'icônes RSLogix Emulate 500.

## Annexe D – Null-modem Emulator (com0com)

---

### Définition

Le logiciel Null-modem emulator (com0com) est un driver windows (en mode noyau) disponible librement sous licence GPL, qui permet de créer un nombre illimité de paires virtuelles de ports COM et de les utilisées pour connecter n'importe quelle application utilisant un port COM avec une autre et vice versa.

Chaque paire de port COM fournit deux ports COM avec des noms par défaut commençant par CNCA0 et CNCB0. La sortie d'un port est l'entrée du l'autre port et inversement.

### Installation de Com0com

Pour installer le logiciel com0com, procédez de la manière suivante :

- Au début, exécutez l'installable (setup.exe), une fenêtre d'accueil (Figure D. 1) va s'afficher pour vous guidez au long de toutes les étapes d'installation. Cliquez sur Next.



Figure D. 1: La fenêtre bienvenue de l'application.

- Sélectionnez "Install the software automatically (Recommended)" et cliquez sur next. Le processus d'installation se lance à la fin vous obtenez le résultat indiqué à la (Figure D. 2), cachez launch Setup pour lancer l'application directement :



Figure D. 2 : dernière fenêtre dans le processus d'installation de com0com.

Une paire de ports COM avec les noms CNCA0 et CNCB0 sera disponible par défaut dans le logiciel après l'installation (Figure D. 3), vous pouvez créer des nouvelles paires en utilisant le bouton « Add Pair ».

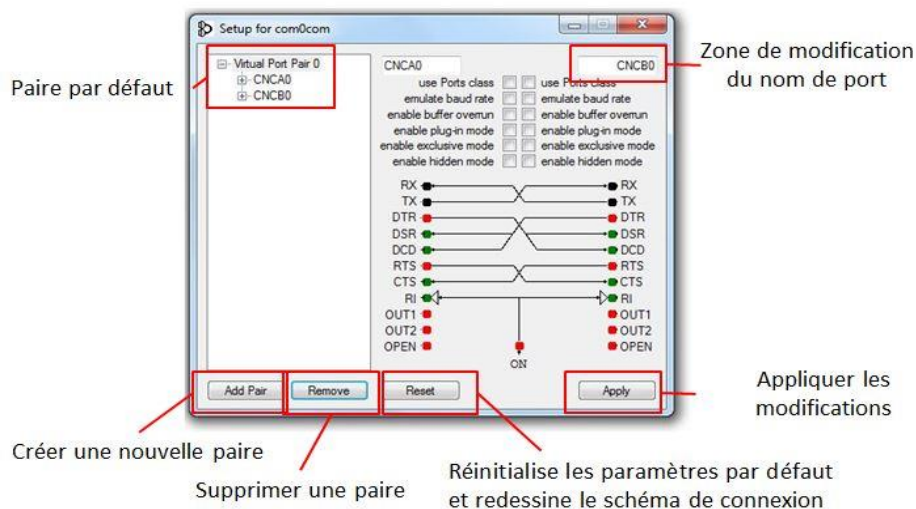


Figure D. 3 : page d'accueil de com0com.

## Annexe E – RSLogix 500

---

### Définition

Le logiciel RSLogix 500 est un logiciel de programmation de logique à relais sous Windows pour les processeurs SLC 500 et MicroLogix.

Les fonctions du logiciel RSLogix 500 comprennent :

- un éditeur de logique à relais à structure libre qui vous permet de vous concentrer sur la logique de l'application plutôt que sur la syntaxe pendant que vous écrivez votre programme.
- un vérificateur de projet puissant qui sert à créer une liste d'erreurs que vous pouvez parcourir à loisir pour effectuer des corrections.
- une fonction d'édition "glisser-déplacer" pour rapidement déplacer des éléments de table de données d'un fichier de données à l'autre, des lignes d'un sous-programme ou d'un projet à l'autre ou des instructions d'une ligne à l'autre dans un même projet.
- une fonction rechercher et remplacer pour rapidement modifier les instances d'une adresse ou d'un symbole particuliers.
- une surveillance personnalisée des données pour visualiser des éléments de données distincts et observer les interactions.
- une interface pointer-cliquer, appelée arborescence de projet, qui vous permet d'accéder à tous les dossiers et fichiers que contient votre projet.
- une fonction de tendance et d'historique permettant de surveiller et d'afficher les données de procédé.
- Des bibliothèques SLC servant à stocker et à récupérer des portions de la logique à relais pour les réutiliser dans l'un des logiciels de programmation de SLC de Rockwell Software.

- un utilitaire de comparaison permettant de visualiser graphiquement les différences dans un projet.

## Méthodes d'installation

Pour installer le logiciel RSLogix 500, procédez de la manière suivante :

- 10- Lancez l'exécutable de RSLogix 500, Le programme d'installation démarre automatiquement et la boîte de dialogue d'accueil s'affiche.
- 11- **Dans la boîte de dialogue d'accueil** : lisez les informations d'introduction, puis cliquez sur Suivant.
- 12- **Dans la boîte de dialogue Contrat de licence logiciel** : lisez entièrement le contrat de licence du logiciel. Cliquez sur Oui pour accepter et continuer l'installation ou sur Non pour décliner et quitter l'installation.
- 13- **Dans la boîte de dialogue Informations d'enregistrement** : tapez votre nom, celui de votre société, le numéro d'identification de support du logiciel RSLogix500, puis cliquez sur Suivant.
- 14- **Dans la boîte de dialogue Sélectionner un dossier** : sélectionnez un emplacement de répertoire pour les fichiers d'application de RSLogix 500 et Acceptez l'emplacement du répertoire de destination par défaut, ou sélectionnez un nouveau répertoire dans lequel seront copiés les fichiers de RSLogix 500, puis cliquez sur Suivant. Une boîte de dialogue apparaît pour indiquer qu'un sous-répertoire RSLogix 500 sera créé dans le répertoire de destination spécifié. Cliquez sur **Oui** pour confirmer, ou sur **Non** pour quitter.
- 15- **Dans la boîte de dialogue Sélectionner des composants** : Sélectionnez les options d'installation. Dans le volet de gauche, sélectionnez le(s) produit(s) RSLogix 500 que vous voulez installer. Dans le volet de droite, sélectionnez le(s) composant(s) de chaque produit que vous voulez installer. Cliquez sur Suivant.
- 16- **Dans la rubrique Spécifiez la rubrique de menu Démarrer** : Acceptez le dossier de programme par défaut ou saisissez le nom de dossier programme dans lequel vous voulez placer les icônes de l'application RSLogix 500. Cliquez sur Suivant. Une boîte de dialogue apparaît et indique l'emplacement spécifié dans

le menu de démarrage pour les icônes de RSLogix500. Cliquez sur **Oui** pour confirmer, ou sur **Non** pour quitter.

- 17- **Dans la boîte de dialogue de RSLogix 500** : Confirmez vos sélections et cliquez sur Suivant. La boîte de dialogue d'installation s'affiche pendant que les fichiers sont copiés sur le disque dur.
- 18- **Dans la boîte de dialogue d'installation** : Consultez les informations expliquant comment contacter Rockwell Software, puis cliquez sur Suivant.
- 19- **Dans la boîte de dialogue annonçant la fin de l'installation** : Sélectionnez les options d'activation et d'affichage des notices et cliquez sur Terminer. Pour démarrer l'activation, insérer la disquette-maître dans le lecteur 3.5 pouces.
- 20- **Dans la boîte de dialogue EVMOVE** : Suivez les instructions qui apparaissent à l'écran pour activer le logiciel RSLogix 500.
- 21- **Dans la boîte de dialogue Redémarrez Windows** : Spécifiez l'option de redémarrage de votre système d'exploitation et cliquez sur Fin. L'installation est terminée.

### **Exploration de RSLogix 500**

Pour parcourir plus facilement les différentes fenêtres et barres d'outils de RSLogix 500, vous devriez tout d'abord prendre connaissance de ce qu'elles contiennent et des fonctions qu'elles permettent d'effectuer. Lorsque vous ouvrez un projet RSLogix 500, vous verrez la figure :



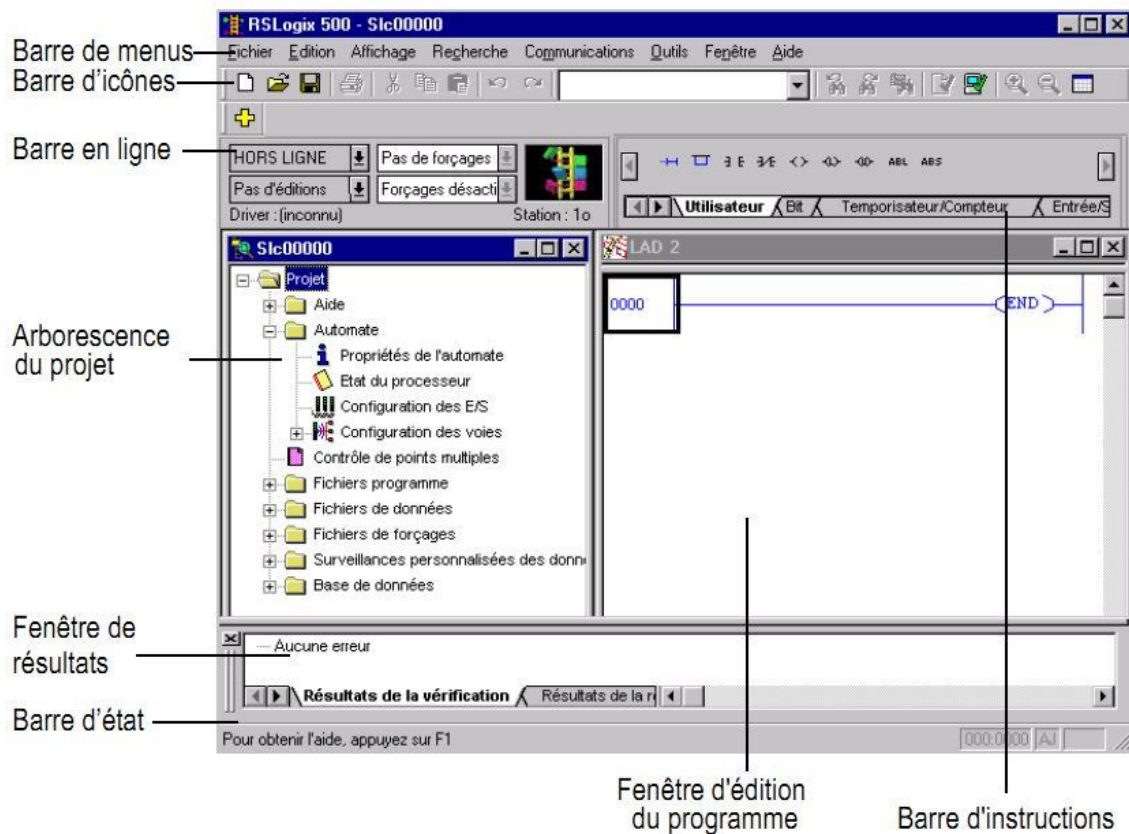


Figure B. 9 : Fenêtre de RSLogix 500.

**Barre de menus :** Sélectionnez les fonctions des menus qui apparaissent quand vous cliquez sur chacune des sélections de cette barre.

**Barre d'icônes :** La barre des icônes contient plusieurs fonctions que vous utiliserez souvent lors du développement et des tests de votre programme de logique. Si vous voulez savoir ce que l'une ou l'autre des icônes représente, RSLogix 500 peut vous le dire. Placez simplement votre curseur sur l'icône. Une info-bulle apparaît et vous indique à quoi sert l'icône.

**Barre en ligne :** Apprenez le mode de fonctionnement de cette barre et voyez d'un coup d'œil si vous avez des éditions en ligne ou des forçages installés. Vous pouvez même visualiser le numéro du driver et de la station.

**Arborescence du projet :** Cette fenêtre regroupe tous les dossiers et fichiers que contient votre projet. Cliquez sur une icône de cette arborescence puis sur le bouton droit de la souris pour ouvrir le menu propre à l'icône sélectionnée. Par exemple, si

vous cliquez avec le bouton droit sur un fichier programme, vous verrez les options Renommer, Ouvrir, Masquer le fichier programme ou en Afficher les propriétés.

**Barre d'état :** Vous y trouverez des informations d'état constamment mises à jour et des invites selon votre utilisation du logiciel.

Fenêtre de résultats : les résultats d'une opération Rechercher ou d'une opération de vérification. Vous pouvez masquer cette fenêtre ou la séparer de la fenêtre d'application pour la positionner n'importe où sur votre écran.

**Fenêtre d'édition du programme :** Dans cette section de la fenêtre d'application, vous pouvez visualiser plusieurs fichiers programme en même temps. C'est là que vous éditez votre programme.

**Barre d'instructions :** Affiche des mnémoniques d'instructions dans des onglets de catégories. Lorsque vous cliquez sur un onglet de catégorie, la barre d'instructions située juste au-dessus se transforme et affiche cette catégorie d'instructions. Cliquez sur une instruction pour l'insérer dans votre programme.

# Bibliographie

---

- [1] Y.Soler, «PLANIFICATION ET SUIVI D'UN PROJET,» *Centre national de la recherche scientifique Direction des systèmes d'information* , 2001.
- [2] G. SONELGAZ, «Site officiel de la SONELGAZ,» [http ://www.sonelgaz.dz](http://www.sonelgaz.dz), 2016.
- [3] G. SONELGAZ, « Rapport d'activités et comptes de gestion consolidés,» 2014.
- [4] «Architecture système,» Document interne de la centrale électrique de LARBAA, LARBAA, 2015.
- [5] «Fiche technique de la centrale,» LARBAA, 2015.
- [6] «P&ID système du gaz naturel filtration et mesure global,» Document interne de la centrale, LARBAA, 2013.
- [7] W. bolton, *Automates Programmables Industriels*, Éditions Dunod, 2015.
- [8] M.Caussade, *Cours d'automatisme*, 2008.
- [9] B. Jrad, *Systèmes automatisés*.
- [10] B. Michel, «Automates programmables industriels,» *Techniques de l'ingénieur. Informatique industrielle*, vol. vol. 3, n° %1no S8015, pp. p. S8015. 1-S8015. 14, 2001.
- [11] A. GONZAGA, «LES AUTOMATES PROGRAMMABLES INDUSTRIELS,» 2004.
- [12] L.Bergounoux, «Les automates progrmmables industriels,» *Polytech Marseille*

*Département de Mécanique Energétique.*

- [13] P. Jargot, «Langages de programmation pour API. Norme IEC 1131-3,» *Techniques de l'ingénieur*, n° %1n° %1S 8030, p. p. 23, 1999.
- [14] M. d. I. santé, «Directives nationales de supervision des acteurs du système de,» *République du Bénin*, mai 2010.
- [15] G. Cladera, «Supervision en mode SaaS,» *RG supervision*, juin 2012.
- [16] P. BONNET, «Introduction à la supervision,» *Université lille1*, Novembre 2010.
- [17] M. D. Silveira., « sur la distribution avec redondance partielle de modeles a evenements discrets pour la supervision de procedes industriels,» *PhD thesis, L'Université Paul Sabatier*, octobre 2003.
- [18] I. Boualem, «Contribution à l'étude de la Supervision Industrielle Automatique,» *PhD thesis, Université M'hamed Bougara de Boumerdes*, 2009.
- [19] D. RACOCEANU, «Contribution à la Surveillance des Systèmes de Productionet Utilisant les Techniques de l'Intelligence Artifficielle,» *PhD thesis, Université de Franche-Comté*, 2006.
- [20] V. Cocquempot, «Contribution à la surveillance des systèmes industriels complexes,» *PhD thesis, UNIVERSITE DES SCIENCES ET TECHNOLOGIES DE LILLE*, decembre 2007.
- [21] M. Combacau, «Systèmes de production surs de fonctionnement,» mars 2000.
- [22] M. RAFIK, «Diagnostic industriel par Neuro-Flou-Application á un système de production,» *PhD thesis, Université de Batna*, mars 2008.
- [23] G. B. Carlos Daniel, «OutilsS pour l'aide á la supervision de procédés dans une architecture multiagents,» *PhD thesis, INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE*, septembre 2004.

- [24] T. E. BAHRI, La maintenance industrielle, Academia.edu.
- [25] R. HAMZA, «Modélisation, Surveillance d'une centrale à Beton par l'outil BOND GRAPH et validation en ligne par un système SCADA,» *PhD thesis, Université Sétif (Algérie), 2014.*
- [26] C. ABDELJAOUED, «Elaboration d'un système d'automatisme et de régulation d'une unité d'aérocondenseur de vapeur d'eau en remplacement d'un condenseur de vapeur à eau de mer,» *Université du 7 novembre à Carthage, 2010.*
- [27] A. Skaf, «Etude d'un système de supervision et de commande d'un procédé complexe comme un élément de base d'une organisation distribuée comprenant des machines et des hommes,» *PhD thesis, UNIVERSITE JOSEPH FOURIER, décembre 2001.*
- [28] O. Souli, «Mise en place d'un système de supervision open source,» *Technical report, Université Virtuelle de TUNIS, 2011.*
- [29] A. B. R. Software, «SLC 500 SYSTEMS SELECTION GUIDE "BULLETIN 1746 AND 1747",» [En ligne]. Available: <http://rockwellautomation.com/literature>).
- [30] Ingénieur, de, l'unité, réchauffeur, gaz, à, bain et marie, Interviewees, *Constituants de l'API.* [Interview]. Mars 2016.

