

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 الآلية و الألكترونيقتي
Département d'Automatique et d'Électrotechnique



Mémoire de Master

Filière : **Automatique**

Spécialité : **Automatique et informatique industriel**

Présenté par

DIBOUCHE AYMANE

&

AMARI ABDELKADER

Programmation et supervision du système DMS (Distribution Managment System)

Proposé par : Mr. **Benselama Zoubir**

Année Universitaire 2020-2021

Dédicace

*Je dédie ce modeste travail au meilleur des pères, celui qui
s'est toujours sacrifié pour me voir réussir*

*A la lumière de mes jours, ma vie et mon bonheur ;
maman que j'adore.*

Que dieu leur procure une bonne sainte et une longue vie.

*A mes frères et sœur dont le plaisir leur revient en premier
lieu pour leurs conseils, aides, et encouragements*

A tous mes amis et à tout qui me sont cher

Dibouche Aymane

Dédicace

*C'est avec joie et plaisir que je dédie ce modeste travail aux
personnes très chères à ma cour*

*A mes très chers parents, vous m'avez donné tout l'amour la
courage et la patience ainsi toutes les moyennes durant la
période des études, je vous remercie de toutes vos efforts.*

A ma grand-mère que dieux lui sont protégé

*A tous mes amis de près ou de loin que ma grande famille,
je vous Remercie pour tous*

Abdelkader Amari

Remerciement

*Après avoir rendu grâce à Dieu le tout puissant et le
miséricordieux, qui nous a donné la force et la patience
d'accomplir ce projet de fin d'étude, nous tenant à remercier
en premier lieu nos parents de nous avoir encouragé, sans
eux, on n'en serait pas là*

*On remercie notre promoteur Mr Z. Benslama pour ses
précieus conseils et son aide durant toute la période du travail*

*Nos vifs remerciements vont également à remercier tous les
membres du jury d'avoir accepté d'examiner ce travail.*

*Nous exprimons notre pleine gratitude à toutes les personnes
ayants fourni des efforts pour nous donner un enseignant de
qualité durant notre cursus universitaire.*

Table des matières

Introduction générale	1
Chapitre 1 : Présentation du réseau électrique et présentation de l'entrepris	
1.1 Introduction.....	2
1.2 Représentation de l'entreprise	2
1.2.1 Groupe SONELGAZ.....	2
1.2.2 Société de distribution d'Electricité et du Gaz de centre RDC.....	4
1.2.3 Organigramme de la direction de la société de distribution de Blida.....	5
1.3 Hiérarchisation du réseau électrique.....	7
1.3.1 Différents niveaux de tension.....	8
1,3,2 Caractéristiques des grandeurs dimensionnèrent.....	8
1.4 Différents réseaux électriques.....	8
1.4.1 Réseau de distribution.....	9
1.4.2 Postes source HTB/HTA.....	9
1.4.3 Réseaux HTA.....	10
1.4.4 Caractéristiques des transformateurs HTA/BTA.....	11
1.4.5 Modes de distribution des réseaux HTA.....	11
1.5 Historique.....	12
1.6 Réseaux ruraux.....	14
1.6.1 Réseaux en double dérivation simple.....	14
1.6.2 Réseaux en dérivation multiples.....	15
1.6.3 Structures en coupure d'artère.....	16
1.6.4 Réseaux urbains.....	17
1.7 Différents éléments du poste HTB/HTA.....	18
1.8 Conclusion.....	22
Chapitre 2 : Carte d'acquisition et logiciels nécessaire pour la réalisation du projet	
2.1 Introduction.....	23
2.2 Carte d'acquisition et de commande ARDUINO.....	23

2.2.1	Architecture de la carte (Hardware).....	23
2.2.2	Définition du circuit imprimé.....	23
2.2.3	Schéma synoptique.....	24
2.2.4	Différents types d'Arduino.....	24
2.2.5	Présentation d'Arduino Méga 2560.....	25
2.2.6	Périphériques.....	26
2.2.7	Caractéristiques de l'Arduino Méga 2560.....	26
2.3	Interface de programmation IDE (Software).....	27
2.3.1	Installation.....	27
2.3.2	Description de l'IDE.....	29
2.3.3	Structure du programme.....	32
2.3.4	Langage de programmation.....	33
2.3.5	Compilation et Téléversement.....	34
2.3.6	Principaux fonctionnements.....	35
2.3.7	Communication séries(ARDUINO).....	36
2.3.8	Schéma de la maquette de simulation.....	40
2.4	Logiciel Processing.....	40
2.4.1	Bases de Processing.....	42
2.4.2	Communication sériele.....	47
2.4.3	Code pour Processing.....	47
2.5	Conclusions.....	47
Chapitre 3 : Réalisation et simulation		
3.1	Introduction.....	48
3.2	Synoptique.....	48
3.3	Matériels utilisés.....	49
3.3.1	Bouton poussoir.....	49
3.3.2	LED.....	50
3.3.3	Capteur de courant ACS 712.....	51

3.4 Fenêtres de programmation.....	53
3.5 Interface de la simulation (SCADA)	54
3.6 Poste électrique sur l'écran SCADA.....	55
3.7 Modes de fonctionnement.....	55
3.8 Communication PROCESSING_ARDUINO.....	56
3.9 Organigramme.....	57
3.1 Conclusion.....	57
Conclusion générale	59
Références bibliographiques	60

Liste des figures

Chapitre 1 : Présentation du réseau électrique et présentation de l'entrepris

Figure 1.1 Organigramme du groupe « SONELGAZ ».....	3
Figure 1.2 Direction générale « SONELGAZ ».....	4
Figure 1.3 Organigramme de La direction de la société de distribution de Blida.....	5
Figure 1.4 Organigramme de la division technique électricité.....	6
Figure 1.5 Direction de Distribution de l'électricité et du gaz du Centre (RDC).....	7
Figure 1.6 Réseau électrique Algérien.....	7
Figure 1.7 Poste source simple antenne.....	9
Figure 1.8 Poste source double antenne-simple jeu de barre.....	9
Figure 1.9 Double antenne- double jeu de barre.....	10
Figure 1.10 Réseau rural radial.....	14
Figure 1.11 Réseau en double dérivation.....	15
Figure 1.12 Double dérivation multiple.....	15
Figure 1.13 Poste de coupure.....	16
Figure 1.14 Maille.....	17
Figure1.15 Transformateurs de puissance.....	19
Figure1.16 Disjoncteurs de courant.....	19
Figure1.17 Sélectionneurs.....	20
Figure1.18 Transformateurs de courant.....	20
Figure 1.19 Sélectionneur mise à terre.....	21
Figure1.20 Parafoudre mise à terre.....	22
Chapitre 2 : Carte d'acquisition et logiciels nécessaire pour la réalisation du projet	
Figure 2.1 Schéma Synoptique de la carte Arduino.....	24
Figure 2.2 Différents types d'Arduino.....	24
Figure 2.3 Caractéristiques d'une carte Arduino Méga 2560.....	26
Figure 2.4 Icône d'Installation.....	27
Figure 2.5 Mettre à jour le pilote.....	28
Figure 2.6 Sélectionner le pilote.....	28
Figure 2.7 Pilote mis à jour.....	29
Figure 2.8 IDE.....	30

Figure 2.9 Barre d'Action.....	30
Figure 2.10 Barre des erreurs.....	31
Figure 2.11 Structure du programme.....	32
Figure 2.12 Définition des constantes et des variables.....	32
Figure 2.13 Configuration des Entrées/Sorties.....	33
Figure 2.14 Programmation des interactions et comportements.....	33
Figure 2.15 Compilation.....	34
Figure 2.16 Sélection de la carte et du port.....	35
Figure 2.17 Communication arduino et ordinateur.....	36
Figure 2.18 Communication Arduino – Arduino.....	37
Figure 2.19 RX et TX.....	39
Figure 2.20 Câbles (USB, série).....	39
Figure 2.21 Simulation avec logiciel Isis.....	40
Figure 2.22 Fonctionnement de logiciel.....	41
Figure 2.23 Fenêtre principale et la fenêtre de visualisation.....	42
Figure 2.24 Langage de programmation.....	43
Figure 2.25 Coordonnées dans l'espace.....	45
Chapitres 3 : Réalisation et simulation	
Figure 3.1 Synoptique d'un système.....	48
Figure 3.2 Exemple d'un bouton poussoir.....	49
Figure 3.3 Schéma d'un bouton poussoir.....	49
Figure 3.4 Représentation d'une LED.....	50
Figure 3.5 LED avec une résistance 330 mΩ soudé.....	50
Figure 3.6 Capteur de courant acs 712 30A.....	51
Figure 3.7 Diagramme des broches du capteur acs 712.....	52
Figure 3.8 Fenêtres des programmes.....	53
Figure 3.9 Interface graphique.....	54
Figure 3.10 Interface graphique d'une poste électrique.....	55
Figure 3.11 fonction pour lire les données.....	56
Figure 3.12 fonction qui envoie des commandes.....	56
Figure 3.13 Organigramme de fonctionnement de l'interface.....	57

Liste des tableaux

Tableau 1.1 Différents niveaux de tension en Algérie.....	8
Tableau 1.2 Avantages et inconvénients de la structure en coupure d'artère.....	17
Tableau 2.1 Récapitulatif des différents types d'Arduino.....	25
Tableau 2.2 Formes et instruction.....	45
Tableau 2.3 Définition des couleur et instruction.....	46
Tableau 2.4 Ecrire du texte.....	46

Liste des abréviations

AT	Auto transformateur.
BT	Basse tension.
Disj	Disjoncteur.
DMS	Distribution Managment System.
GRTE	Société de Gestion du Réseau de Transport de l'Electricité.
HTA	Haut Tension Niveau A.
HTB	Haut Tension Niveau B.
KV	Kilovolt.
MΩ	Mili ohm.
MT	Moyenne Tension.
PC	Partie Commande.
RTU	Unité Terminal Unit.
RDC	Société de Distribution d'Electricité et du Gaz de Centre.
RMS	Root mean aquare.
RDE	Société Algérienne de Distribution de l'électricité et du gaz de l'Est.
RDO	Société Algérienne de Distribution de l'électricité et du gaz de l'Ouest.
SCADA	Supervisory Control and Data Acquisition.
SPE	Société de Production de l'Electricité.
TT	Transformateur de tension.
TP	Transformateur de puissance.
TC	Transformateur de courant.

ملخص

يؤدي التمدد المستمر لشبكة الجهد العالي إلى زيادة الحمل على شبكة التوزيع مما يؤدي إلى حدوث أعطال وانهيار في محولات الطاقة. ينتج عن هذا انقطاعات متكررة لشبكة الطاقة المحلية. وعلى هذا الأساس أجرينا دراسة حول الأنظمة واجهة المستخدمة في مراكز التحكم في محطات توليد الطاقة من أجل اقتراح تحسين كفاءة صيانة الشبكة. لهذا قمنا بإضافة لأتمتة نظام التحكم في محولات التوزيع المتاحة وطورنا نظام المراقبة والتحكم عن بعد.

باستخدام البرامج المناسبة. على هذا الأساس، قمنا بمحاكاة تطبيقنا باستخدام بطاقة اكتساب وتحكم مرتبطة SCADA بأداة التصور بناءً على حالات الخطأ المختلفة التي قد تظهر.

الكلمات المفتاحية : معالجة الشبكة الكهربائية. سكاذا. المحولات الكهربائية.

Résumé

L'expansion continue du réseau de haute tension augmente la charge sur le réseau de distribution, ce qui entraîne des défauts et des pannes au niveau des transformateurs de puissance. Il en résulte des interruptions fréquentes du réseau électrique local. C'est sur cette base que nous avons mené une étude sur les systèmes utilisés dans les centres de contrôle des centrales électriques afin de proposer une amélioration sur l'efficacité de la maintenance du réseau. Pour cela nous avons ajouté une interface pour automatiser le système de contrôle des transformateurs de distribution disponibles et développer un système de surveillance et de contrôle à distance SCADA, en utilisant un logiciel approprié. Sur cette optique, nous avons aussi simuler notre application à l'aide d'une carte d'acquisition et de commande associée à un outil de visualisation en s'appuyant sur les différentes situations de pannes qui puissent apparaître.

Mot clés : Processing, Scada, reseau électrique.

Abstract

The continuous expansion of the high voltage network increases the load on the distribution network, resulting in defects and failures at the power transformers. This results a frequent interruption of the local electrical network. It is on this basis that we conducted a

study on the systems used in the control centers of power plants to propose an improvement on the efficiency of the maintenance of the network. For this we have added an interface to automate the control system for available distribution transformers and developed a Scada remote monitoring and control system using appropriate software. On this basis, we have simulated our application using an acquisition card and control associates with a visualization tool based on the different siding situations that can appear.

Keyword : Processing, Scada, electrical network.

Introduction générale

L'homme vit aujourd'hui avec un ensemble d'outils conviviaux auxquelles il ne peut pas renoncer. Parmi ces choses, l'électricité, que l'homme en est venu à considérer comme l'une des priorités de sa vie quotidienne qu'il ne peut pas abandonner.

Le problème est que l'électricité est devenue une partie intégrante de la vie humaine à l'heure actuelle et qu'on ne peut s'en passer. Elle est devenue utilisée dans divers domaines à tous les temps elle est devenue indispensable à l'homme. En cas de problème, beaucoup d'activités peuvent être perturbées en causant beaucoup de pertes économiques ceci a entraîné un contrôle strict du réseau. Parmi les problèmes les plus importants auxquels le distributeur est confronté est le problème d'interruption, et afin de réduire ce problème, des techniques avancées ont été utilisées pour remédier aux impératifs qui peuvent apparaître tels que la surveillance du bon débit de courant via le réseau, le facteur temps d'intervention et sécurité des membres de la maintenance qui se tiennent sur le processus d'inspection manuelle pour trouver un site Erreur. Pour répondre à ces aléas la société de distribution (Sonelgaz) a installé un système DMS afin de maintenir la sécurité des travailleurs, la maintenance et d'économiser du temps et des pertes économiques que nous nous proposons de le développer en ajoutant une interface graphique pour rendre le contrôle du système convivial.

Pour cela, nous avons suivi un plan explicitant clairement la problématique et la solution s'articulant autour de trois chapitres.

Chapitre 1 : Présentation du réseau électrique et présentation de l'entreprise.

Chapitre 2 : La carte d'acquisition et logiciels nécessaires pour la réalisation du projet.

Chapitre 3 : La réalisation et simulation.

Chapitre 1

Présentation du réseau électrique et présentation de l'entreprise

1.1 Introduction

Nous allons présenter dans ce chapitre des notions générales sur les réseaux électriques, et on donnera un aperçu sur les différents réseaux électrique et postes électriques, architectures et fonctionnements, du fait que notre étude sera portée sur un système de Télé conduite par le système SCADA du réseau électrique moyenne HTA a basse tension BT de la SONELGAZ SDC.

1.2 Représentation de l'entreprise

L'objectif que nous nous sommes fixés pour notre projet de fin d'étude est de réaliser un système fonctionnel d'acquisition et d'affichage d'information pour une commande à distance d'action pour activer un actionneur ou pour un dépannage. Pour cela nous nous sommes dirigés vers le groupe SONELGAZ qui offre une plateforme idéale pour atteindre notre objectif.

1.2.1 Groupe SONELGAZ

En 1969, SONELGAZ était déjà une grande entreprise avec environ 6 000 agents. Il a servi 700 000 clients. La SONELGAZ a contribué à la construction de l'infrastructure économique nationale. Elle monopolise la production, le transport, la distribution, l'importation et l'exportation d'électricité et de gaz artificiel. Compte tenu de la diversité des missions de la SONELGAZ, sa structure est aujourd'hui un groupe industriel composé de 35 filiales et 5 holdings. Avec des ressources humaines bien formées et qualifiées, le groupe occupe une position dominante dans l'économie nationale et est chargé de fournir de l'électricité à plus de 6 millions de foyers et 3 millions de foyers en gaz naturel, soit un taux d'électrification supérieur à 99%, le taux de couverture géographique et 60% de pénétration du gaz. Ainsi, la filiale métier de base assure la production, le transport et la distribution d'électricité. Et du gaz par canalisations [21].

On compte :

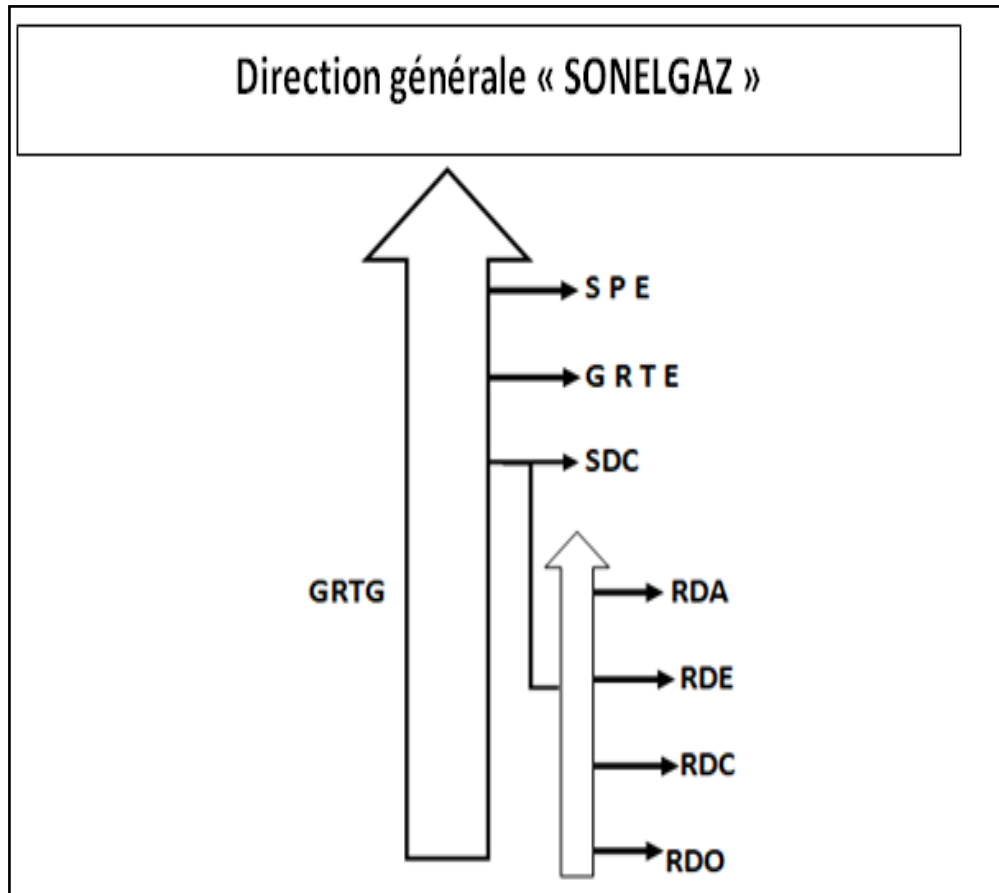


Figure 1.1 Organigramme du groupe « SONELGAZ »

- La Société Algérienne de Production de l'Électricité (SPE).
- La Société Algérienne de Gestion du Réseau de Transport de l'Électricité (GRTE).
- L'Opérateur Système électrique (OS), chargée de la conduite du système Production /Transport de l'électricité.
- La Société Algérienne de Distribution de l'électricité et du gaz d'Alger (RDA).
- La Société Algérienne de Distribution de l'électricité et du gaz du Centre (RDC).
- La Société Algérienne de Distribution de l'électricité et du gaz de l'Est (RDE).
- La Société Algérienne de Distribution de l'électricité et du gaz de l'Ouest (RDO).



Figure 1.2 Direction générale « SONELGAZ » [18]

1.2.2 Société de Distribution d'Electricité et du Gaz de Centre RDC

Société de distribution d'électricité et de gaz naturel du Congo (RDC) Créée en janvier 2018 (changement d'organigramme), elle dispose d'un réseau électrique long, moyenne et basse tension (HTA/BT), et du même grand réseau de gaz naturel. La Société Centrale de distribution d'Electricité et de Gaz Naturel (RDC) est chargée de :

- Exploitation et maintenance des réseaux de distribution d'électricité et de gaz.
- Le développement des réseaux d'électricité et de gaz naturel permet des raccordements nouveau client.
- La commercialisation de l'électricité et du gaz naturel offre une qualité de service au moindre coût dans les meilleures conditions de sécurité.
- La société centrale de distribution d'électricité et de gaz naturel « RDC » met en œuvre un plan d'investissement avec un double objectif :
- Développer le réseau et répondre à la demande.

- Modernisation de l'exploitation et de la gestion. Dans ce cas, le bureau central de contrôle (SCADA) constitue un projet structurant pour l'amélioration de la conduite des réseaux et de l'amélioration de la qualité de service.

1.2.3. Organigramme de La direction de la société de distribution de Blida



Figure 1.3 Organigramme de La direction de la société de distribution de Blida

La direction de la société de distribution de Blida contient de plusieurs divisions lesquelles se complètent entre eux se compose de :

- **Le directeur** : c'est le premier responsable pour gérer l'entreprise économique et Financial, et prend toutes les décisions administratives.
- **Secrétariat** : c'est l'intermédiaire entre le directeur et le personnel et aussi les clients. Il s'occupe de la réception des plaintes des clients et les transmettre au directeur.
- **Attaché juridique** : son rôle axé sur la défense les intérêts de l'entreprise de manière le recours à la justice s'il y a des plaintes contre elle ou bien a déposé une plainte contre les clients par Ex : vol d'électricité et le gaz.
- **L'ingénieur de sécurité** : effectue une étude de l'environnement de travail afin de bien équiper le personnel pour assurer sa sécurité.

- **Division ressource humain** : cette section est d'une grande importance en raison de sa relation avec les autres divisions mais aussi avec la clientèle.
- **Division technique électrique** : cette section étudie le réseau électrique afin d'améliorer la qualité des services et de la maintenance.

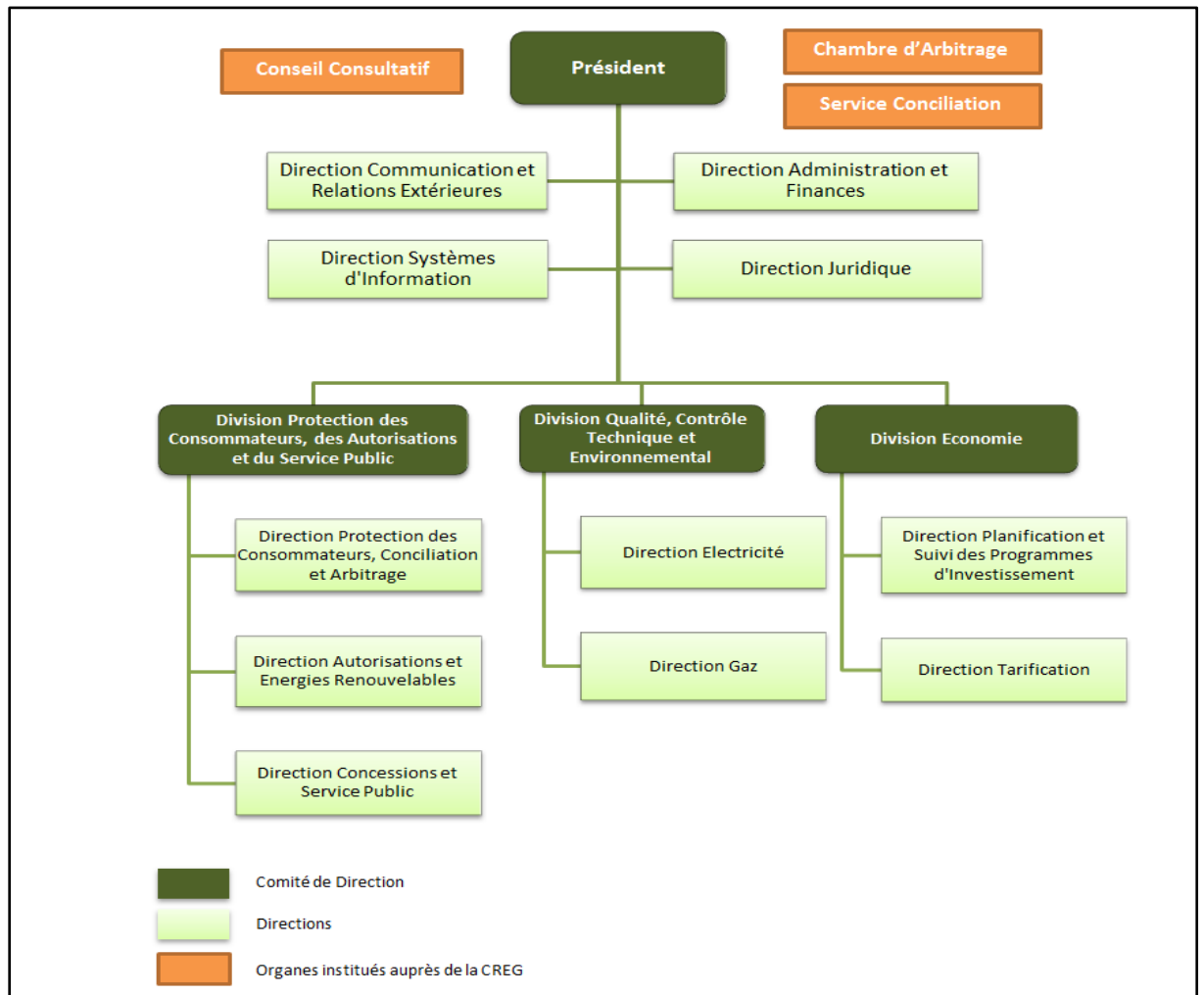


Figure 1.4 Organigramme de la division technique électricité [22]

- **Division études et exécution des travaux** : faire l'étude des demandes des clients et d'exécution (réalisation le raccordement l'électricité et le gaz).
- **Division relation commercial** : cette section d'une grande importance en raison de sa relation avec les autres divisions et aussi il possède des tâches techniques et gestionnaires.



Figure 1.5 Direction de Distribution de l'électricité et du gaz du Centre (RDC) [17]

1.3 Hiérarchisation du réseau électrique

Le réseau électrique national est hiérarchisé en trois parties dont les fonctions actuelles sont très différentes. Tout d'abord, le réseau de transport a pour rôle de transporter l'énergie en très haute tension depuis les centres de productions jusqu'au premières zones de consommation. Le réseau de répartition alimente directement les gros consommateurs industriels puis achemine l'énergie jusqu'aux réseaux de distribution chargés d'approvisionner les consommateurs moyenne et basse tension [3].

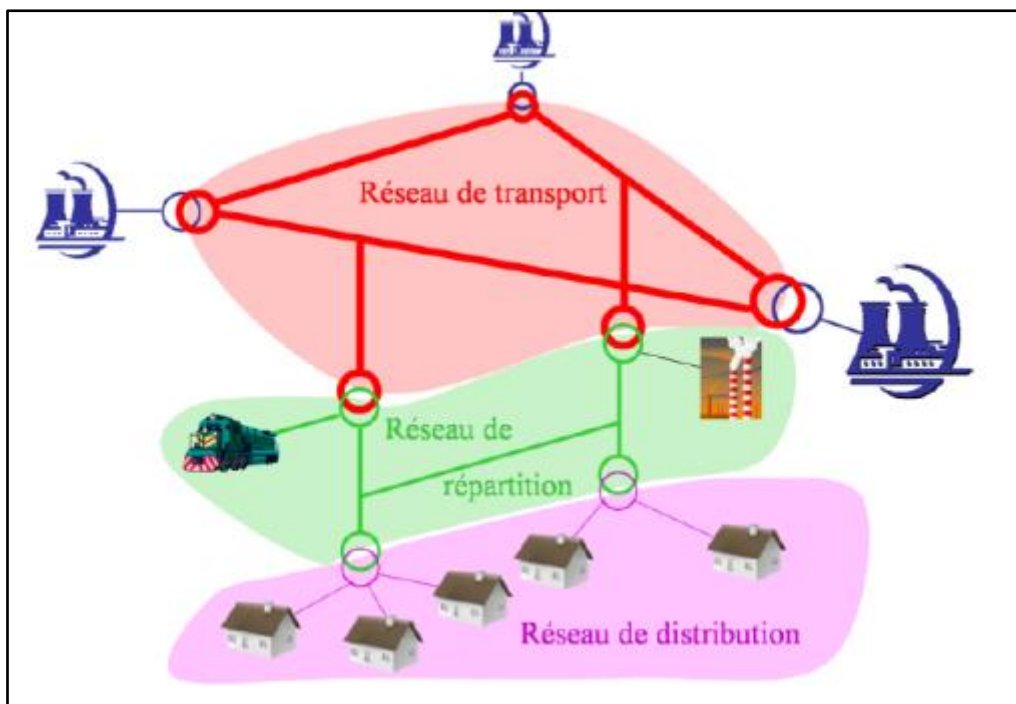


Figure 1.6 Réseau électrique Algérien [3]

Ces trois niveaux de réseau sont délimités grâce à des transformateurs. Ces transformateurs permettent d'acheminer l'énergie à différents niveaux de tension

1.3.1 Différents niveaux de tension en Algérie

Les différents niveaux de tensions sont indiqués dans le tableau 1-1. Nous utiliserons par la suite ces abréviations. Ainsi le réseau de transport et de répartition se situe au niveau de la HTB. Le gestionnaire du réseau de transport est responsable, à l'échelle nationale, de l'équilibre production/consommation et du respect des échanges transfrontaliers. Le réseau de distribution est au niveau de la HTA et de la BTA. Le rôle du gestionnaire du réseau de distribution est l'entretien et la gestion de ce dernier et aussi l'alimentation des clients de type petites et moyennes entreprises et résidentiels au travers du réseau moyenne tension (HTA) et du réseau basse tension (BTA) [1].

U<50 V	50V<U<500 V	500V<U<1K V	1KV<U<50K V	50KV<U<100K V	U>100K V
TBT	BT	MT	HT	THT	THT
TBT	BTA	BTB	HTA	HTB	HTB

Tableau 1.1 Différents niveaux de tension en Algérie [3].

1.3.2 Caractéristiques des grandeurs dimensionnèrent

Les conducteurs aériens et souterrains permettent de transporter l'énergie électrique. Cependant, le coût de la matière première et l'encombrement du territoire font qu'il n'est pas possible d'investir dans d'immenses plaques de conducteurs métalliques. Ainsi, les tailles de conducteurs limitées créent des imperfections sur le réseau comme la chute de tension, les puissances limitées dans le réseau et enfin les pertes. Nous allons rappeler la définition de ces imperfections dans les paragraphes qui suivent [3].

1.4 Différents réseaux électriques

Le réseau électrique est divisé en plusieurs niveaux, il ne suffit pas de générer du courant dans la centrale, il faut aussi fournir des courants de différentes tensions et puissances aux utilisateurs finaux [2].

1.4.1 Réseau de distribution HTA

Le réseau de distribution moyenne tension commence à partir du poste source HTB/HTA d'où partent plusieurs départs HTA constitués d'un ensemble de conducteurs et d'appareils de coupure qui alimentent les charges moyenne tension ou les postes de distribution publique (HTA/BTA) [4].

1.4.2 Postes source HTB/HTA

Les postes source HTB/HTA sont parfois alimentés en antenne mais, le plus souvent, ils sont alimentés avec un jeu de barre recevant plusieurs arrivées (ou lignes) HTB. Un ou plusieurs transformateurs HTB/HTA sont raccordés sur ces jeux de barre HTB simples ou multiples. A l'aval de ces transformateurs, des départs moyens tension partent d'un ou plusieurs jeux de barres HTA. les différents postes source HTB/HTA. Un départ est la portion du réseau dont l'extrémité amont est un poste source HTB/HTA et l'extrémité aval un organe de coupure normalement ouvert (si c'est un départ bouclé à un autre) et le dernier poste de distribution publique HTA/BTA le cas échéant [3].

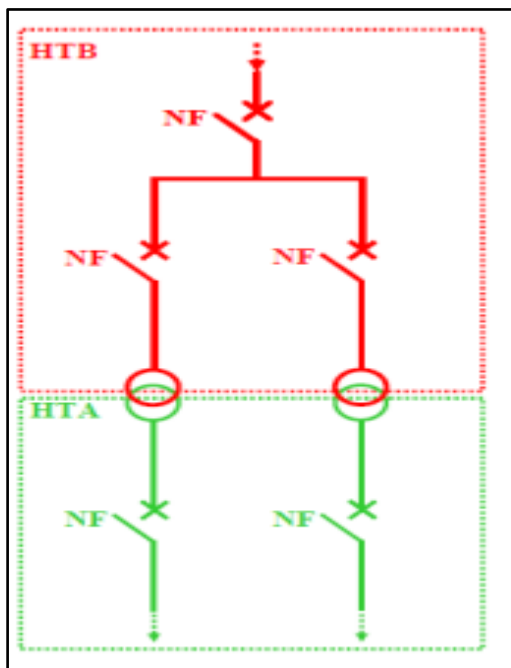


Figure1.7 Poste source simple antenne

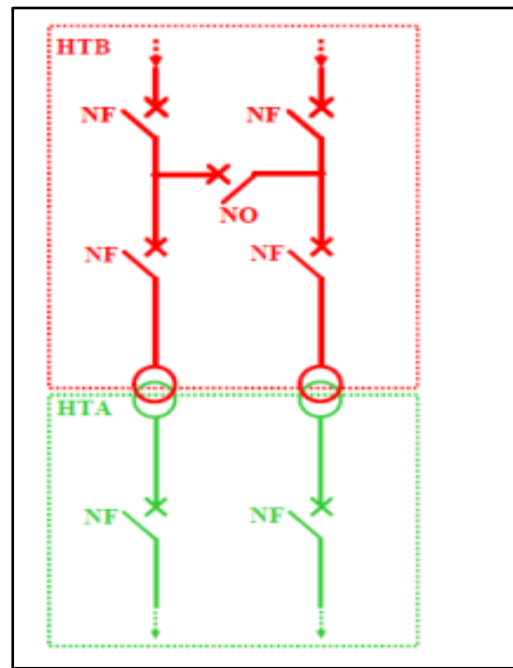


Figure1.8 Poste source double antenne-simple jeu de barre

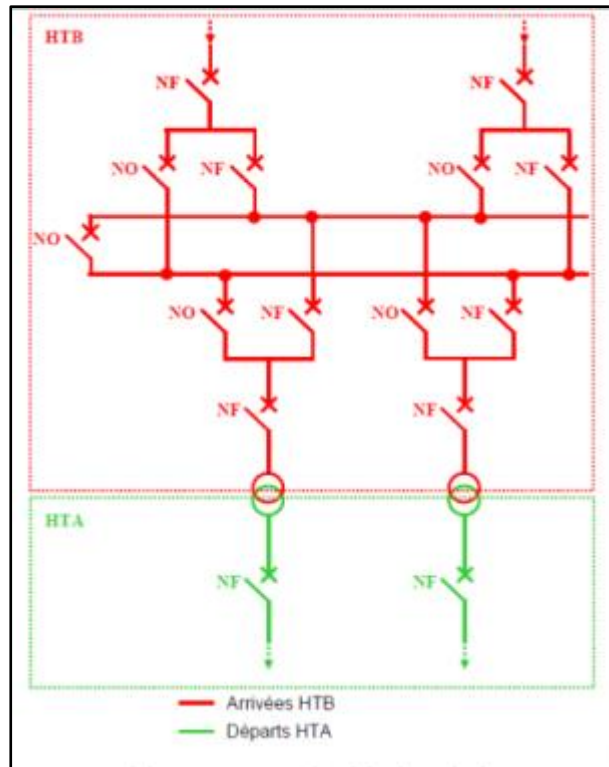


Figure1.9 Double antenne- double jeu de barre

1.4.3 Réseaux HTA

Les réseaux HTA sont, soit réalisés avec des câbles souterrains quasiment toujours bouclés mais exploités en radial (réseaux urbains principalement), soit réalisés avec des lignes aériennes, elle aussi le plus souvent bouclables (réseaux ruraux). Il reste quelques structures aériennes en antenne sans secours possible. Avant, les réseaux étaient en grande majorité construits avec des lignes aériennes. Mais des contraintes liées à l'esthétique, à la fiabilité et à l'encombrement ont poussé au développement des câbles souterrains. Par ailleurs, des techniques modernes de pose mécanisée des câbles souterrains ont rendu leur utilisation plus compétitive [5].

Le long des liaisons HTA, on trouve :

- Pour les câbles souterrains, des postes de transformation maçonnés HTA/BTA raccordés en coupure d'artère ou en double dérivation,
- Pour les lignes aériennes, des postes de transformation simplifiés raccordés par des

piquages en antenne alimentant, soit des transformateurs aériens sur poteaux, soit des transformateurs sous abris dits « cabine ou bas de poteau »[3].

- A l'aval de ces transformateurs HTA/BTA, on trouve principalement des réseaux BTA non bouclés sauf quelques exceptions en réseaux urbains. Parfois, ces réseaux sont très courts comme, par exemple, dans les zones rurales d'Amérique du Nord où le distributeur alimente directement en biphasé haute tension de quelques kilovolts des charges dispersées ponctuelles allant de 10 à 20 voire 30 kW. Les branchements très courts en basse tension viennent alors directement de la transformatrice moyenne tension (MV)/basse tension (LV).

En Algérie, les longueurs de ces réseaux basse tension varient mais excèdent rarement deux kilomètres [5].

1.4.4 Caractéristiques des transformateurs HTA/BTA

Voici les puissances caractéristiques que l'on va retrouver principalement en Algérie :

- Aérien : 50, 100, 160, 250 kVA (avec un $U_{cc} = 4\%$ en valeur typique) avec des régleurs hors charge $\pm 2.5\%$ de la tension nominale.
- Souterrain : 250, 400, 630, 1000 kVA (avec un niveau de tension de court-circuit : $4\% \leq U_{cc} \leq 7\%$) avec des régleurs hors charges : $\pm 2.5\%$ et/ou $\pm 5\%$ de la tension nominale. Ces régleurs hors charge permettent d'ajuster, lorsque le poste n'est pas alimenté, le niveau de tension au niveau de la BT. Ces réglages sont importants car ils fixent le rapport de transformation entre les niveaux de tension HTA et BT en fonction de la chute de tension en amont du poste de distribution public mais sont actuellement peu commodes car réglables uniquement lorsque le réseau n'est pas connecté [7].

1.4.5 Modes de distribution des réseaux HTA

L'architecture des réseaux de distribution ainsi que le placement des appareils de coupure dépend de plusieurs paramètres technico-économiques comme le type de zone (rurale ou urbaine), la qualité de service désirée (temps moyen de coupure par client) et donc l'investissement que l'on est prêt à engager. Dans la suite de ce chapitre, nous allons décrire

la structure générale des réseaux de distribution ainsi que les modes de distribution associés [3].

1.5 Historique

Les premiers systèmes de distribution HTA au début du XX^{ème} siècle étaient constitués des réseaux triphasés courts et le plus souvent aériens. Le neutre du réseau était souvent isolé de la terre. Les défauts non permanents susceptibles de se produire étaient en général auto extincteurs (qui disparaissent spontanément en un temps très court, inférieur à 100 ms, sans provoquer le déclenchement des organes de protection du réseau) car la capacité du réseau était très faible [8].

Puis, le courant de défaut se mit à augmenter en raison de :

- L'augmentation des niveaux de tension.
- L'utilisation de câbles souterrains.
- La diminution des longueurs de réseaux issus d'une même source.
- L'interconnexion d'ensembles initialement séparés.
- Il constitua alors un danger pour l'homme à cause des élévations locales de potentiel qu'il provoquait.

Deux grandes tendances apparurent : le système nord-américain et le système européen[9].

- En Amérique du Nord, le neutre fut directement relié à la terre régulièrement pour avoir de forts courants de court-circuit et ainsi pouvoir réaliser une détection et isolation des défauts reposant sur des logiques de protection de type fusibles, en Europe du Nord, le neutre fut mis à la terre par une réactance accordée à la capacité du réseau pour compenser le courant de défaut monophasé et pour pouvoir, le cas échéant, continuer d'alimenter les charges lors de défauts monophasés.
- En Europe du Sud, le neutre fut raccordé par une impédance pour réduire la valeur du courant de défaut à une valeur raisonnable [11].
- Pour le reste du monde, l'une de ces tendances précédentes furent adoptées.
- Le choix des modes de distribution de l'électricité de par le monde dépend de deux principaux critères : un critère géographique et humain (étendue du territoire, répartition de la population et les puissances à desservir entre autres) et un critère historique (les savoirs faire introduits pas les zones d'influence ou colonies) [11].

L'avantage du triphasé est qu'il permet le transport de la même quantité d'énergie avec une section conductrice totale plus petite qu'en monophasé. Ainsi toutes les structures rencontrées dans le monde ont au moins leur ossature principale en triphasée. Il peut cependant s'avérer que l'alimentation monophasée soit économiquement intéressante dans le cas, par exemple, de charges faibles et dispersées.

Il existe deux principaux systèmes de distribution de par le monde : le système Nord-Américain et le système européen [10]. Dans un premier temps, nous allons définir les différentes structures HTA rencontrées, puis nous ferons un tour du monde des modes de distribution puis nous détaillerons quelques structures particulières. Nous verrons dans cette section que la structure d'un réseau de distribution est à la fois image des contraintes d'exploitation, du matériel disponible et des pratiques d'exploitation héritées de l'histoire du pays [3].

La qualité de service est encore plus importante en milieu urbain notamment à cause de certaines infrastructures qui ne doivent pas être déconnectées comme par exemple les hôpitaux. Le réseau est donc très souvent enterré à cause de ces contraintes d'encombrement et de qualité. La principale différence entre les réseaux aériens et souterrains provient du fait que si le défaut est moins fréquent en souterrain, il est en revanche plus long à réparer. Par ailleurs, un nombre plus important de clients peut être affecté [5].

L'exploitation est généralement en boucle ouverte. Cependant, on trouve une exploitation maillée à l'est de l'Allemagne (ex RDA). On trouve deux classes de structures : les structures à une voie d'alimentation ou encore appelées structures radiales ou en antenne (généralement dans les réseaux aériens) et les structures à deux voies d'alimentation ou en double dérivation [12].

1.6 Réseaux ruraux

En milieu rural, on trouve des architectures arborescentes bouclables mais exploitées en radial. Les boucles peuvent se situer entre les postes HTB/HTA ou entre départs voisins (du même poste source).

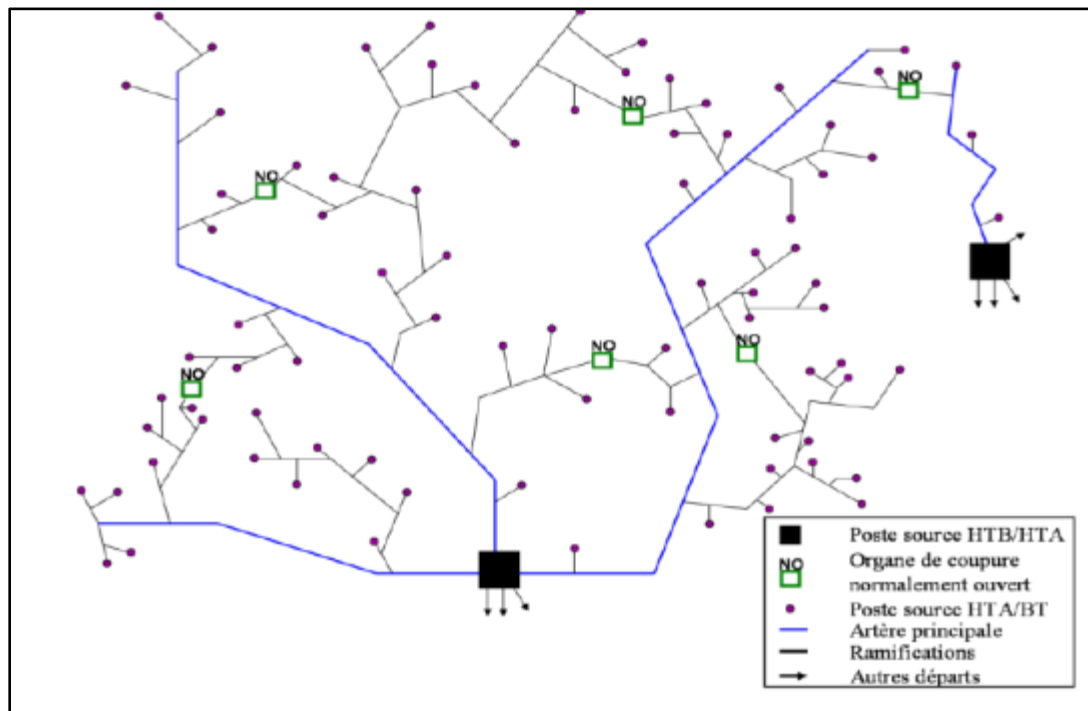


Figure 1.10 Réseau rural radial

1.6.1 Réseaux en double dérivation simple

Le réseau radial en antenne est doublé à partir du jeu de barre du poste source HTB/HTA.

Ainsi chaque transformateur HTA/BTA est connecté à un câble normal et à un câble de secours par le biais de dispositifs inverseurs.

C'est une structure difficilement exploitable manuellement mais facilement automatisable malgré les coûts [3].

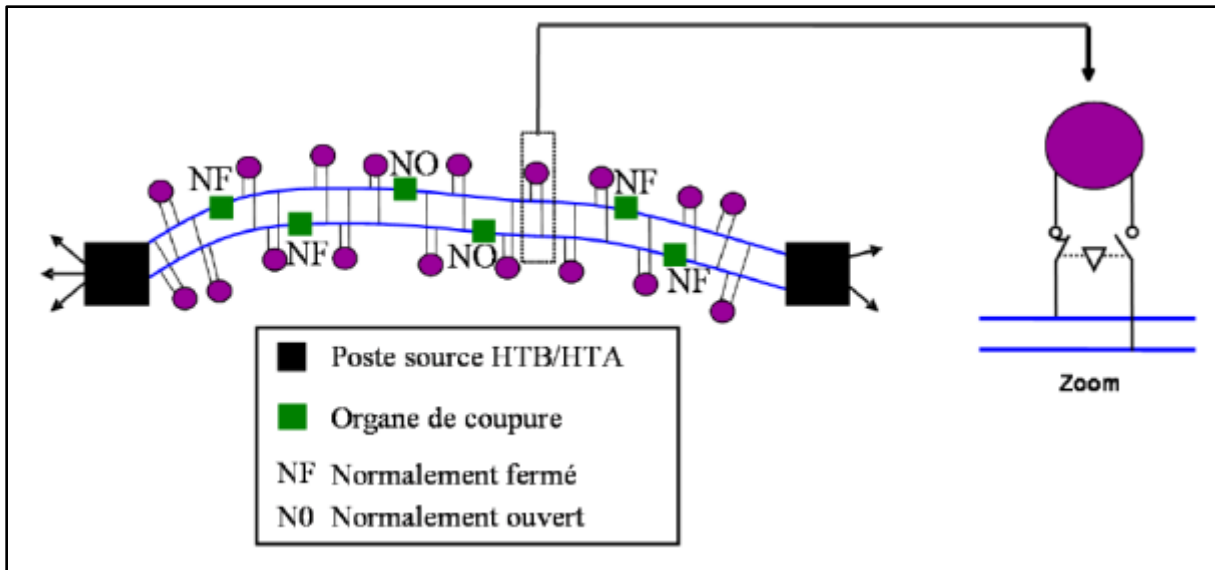


Figure 1.11 Réseau en double dérivation

1.6.2 Réseaux en dérivation multiples

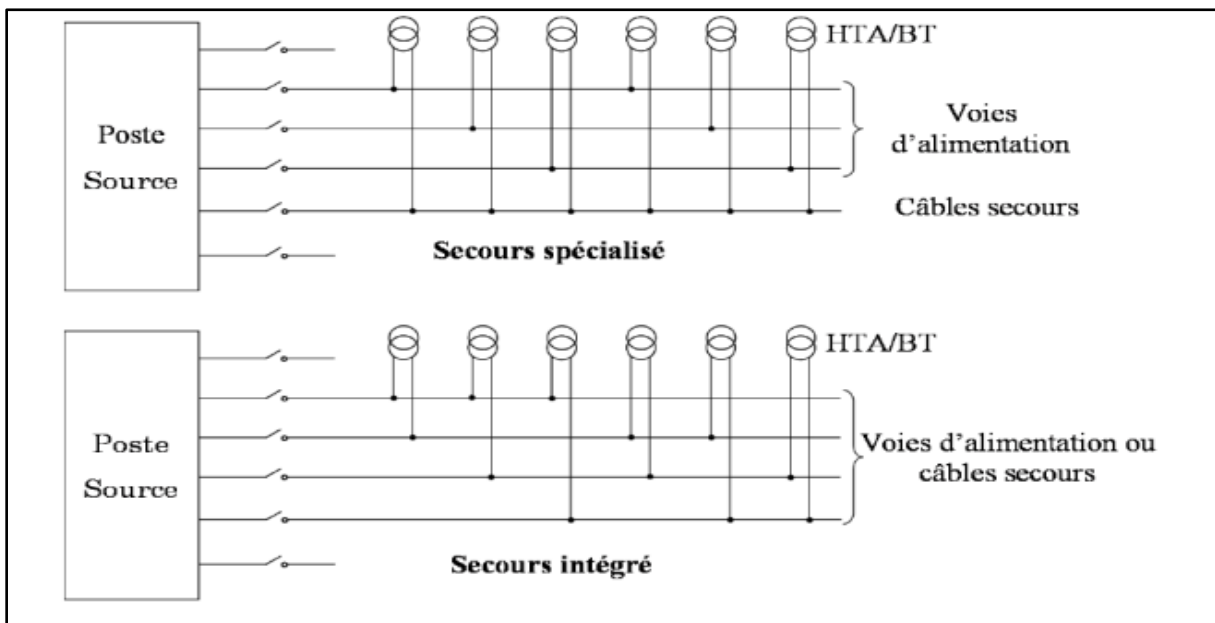


Figure 1.12 Double dérivation multiple

On trouve cette structure dans la capitale française (Paris). Chaque poste source HTA/BTA ne dispose que de deux voies d'alimentation mais ces deux voies sont raccordées alternativement à entre trois et six câbles partant du poste source HTB/HTA. En cas de défaut,

la charge peut être basculée soit sur un câble secours, soit sur les autres câbles (secours intégré) comme le montre la Figure 1.12.

Dans tous les cas, on a intérêt à mettre un poste de coupure réparti tous les 10 à 15 postes HTA/BTA de façon à limiter les manœuvres lors de l'élimination du défaut.

Donne un exemple d'un poste de coupure sur deux départs d'une structure en double dérivation multiple de la Figure 1.13 [3].

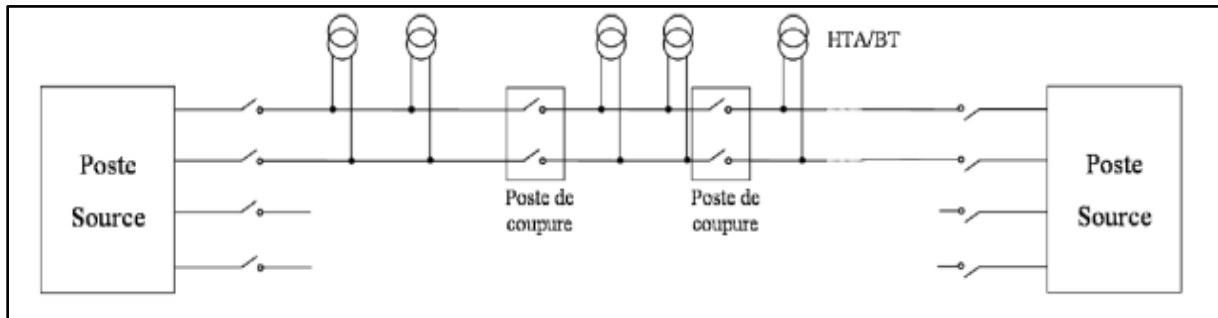


Figure 1.13 Poste de coupure

Dans le cas de fortes densités de charges ou quand une qualité de service accrue est demandée, cette structure peut être envisagée [3].

1.6.3 Structures en coupure d'artère

Dans cette architecture, un câble part d'un poste source HTB/HTA, passe successivement par les postes HTA/BTA à desservir avant de rejoindre soit un autre poste source HTB/HTA, soit un départ différent du même poste source HTB/HTA, soit un câble secours.

Au niveau des postes HTA/BTA, des interrupteurs sont placés de part et d'autre des postes sources. Ils sont tous normalement fermés sauf un qui permet l'exploitation radiale. Ainsi en cas de défaut sur un tronçon de câble, on peut l'isoler en ouvrant les deux interrupteurs qui l'encadrent.

La fermeture de l'interrupteur normalement ouvert permet la réalimentation du reste des charges non touchées par ce défaut. Le Tableau 1-2 résume les avantages et inconvénients de la coupure d'artère [3].

La coupure d'artère présente plusieurs variantes que nous allons détailler ci-dessous.

Avantages	Inconvénients
<ul style="list-style-type: none">▪ Plus économique que la double dérivation▪ Exploitation manuelle : temps d'intervention d'environ 1h	Automatisation coûteuse et difficile mais qui sera amenée à se développer en raison de la baisse des coûts de transmission et des systèmes de gestion centralisée.

Tableau 1.2 Avantages et inconvénients de la structure en coupure d'artère

1.6.4 Réseaux urbains

A. Maille

Cette structure est composée de boucles alimentées directement par les sources HTB/HTA ou via des postes têtes de boucle reliés aux sources HTB/HTA par des conducteurs de section importante appelés câbles de structure. Les postes têtes de boucle ont la même structure que les postes source sauf qu'il n'y a pas de transformateur HTB/HTA. Des liaisons inter-boucles permettent le report de charge d'une boucle sur l'autre en cas de perte d'un câble de structure [3].

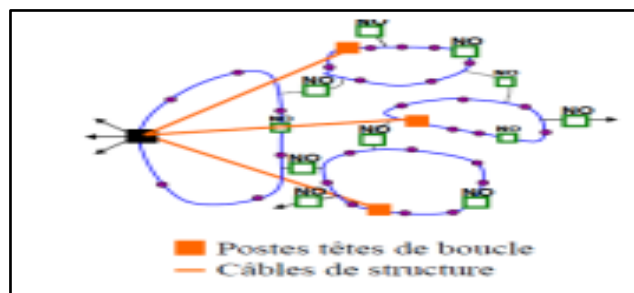


Figure 1.14 Maille

1.7 Différents éléments du poste HTB/HTA

B : jeu de barre

TP : transformateur de puissance

Dj : disjoncteur

S : sectionneur

ST : sectionneur mise à la terre

Tc : transformateur de courant

Tt : transformateur de tension

P : parafoudre

- **Jeu de barre**

Un jeu de barres est un ouvrage électrique triphasé dominant sur la longueur du poste. Il permet de relier entre eux les départs de même tension qui y aboutit. Un poste électrique peut être doté d'un, deux, voire trois jeux de barres pour une tension donnée.

Les jeux de barres sont typiquement soit des barres plates, soit des tubes.

- **Transformateur de puissance**

Le transformateur de puissance est un équipement électrique indispensable au fonctionnement du réseau électrique. Il est défini par la Commission électrotechnique internationale comme suit : « Un dispositif statique à deux ou plusieurs enroulements, par induction électromagnétique, convertit le système de tension et de courant alternatif en un autre système de tension et de courant, qui est généralement de valeur différente, et la fréquence est la même.



Figure1.15 Transformateurs de puissance

- **Disjoncteur**

Un disjoncteur est un appareil qui peut couper des courants importants, qu'il s'agisse de courant normal ou de courant de défaut. Par conséquent, il peut être utilisé comme un grand interrupteur pour un contrôle sur site ou une télécommande via des boutons. De plus, dès que le courant circulant dépasse une valeur prédéterminée, le disjoncteur ouvre automatiquement le circuit. Lorsqu'il est utilisé pour couper un fort courant de court-circuit, sa fonction est la même que celle d'un fusible, mais il est plus sûr à utiliser et n'a pas besoin d'être remplacé après chaque coupure.



Figure1.16 Disjoncteurs de courant

- **Sectionneur**

Les sectionneurs permettent d'ouvrir ou de fermer le circuit d'appareils électriques à vide. Ils n'ont pas de pouvoir de coupure. Ils ne permettent d'ouvrir le circuit qu'en l'absence de courant.



Figure1.17 Sectionneurs

- **Transformateur de courant**

Selon la définition de la Commission électrotechnique internationale, un transformateur de courant est « un transformateur de mesure dans lequel le courant secondaire est, dans les conditions normales d'emploi, pratiquement proportionnel au courant primaire et déphasé par rapport à celui-ci d'un angle voisin de zéro pour un sens approprié des connexions ». La caractéristique la plus importante d'un transformateur de courant est donc son rapport de transformation, exprimé par exemple sous la forme 400A /1A. L'équipement de mesure connecté à son secondaire est en général un ampèremètre, mais on peut également brancher un wattmètre ou des relais de protection. Tous sont conçus pour mesurer des courants de quelques ampères.



Figure1.18 Transformateurs de courant

- **Sélectionneur mise à terre**

L'interrupteur de mise à la terre est un interrupteur de sécurité qui permet d'isoler le circuit. En raison de la mise à la terre, il peut empêcher toute tension d'apparaître sur le circuit lors de la maintenance.



Figure 1.19 Sélectionneur mise à terre

- **Transformateur de tension**

Selon la définition donnée par la Commission électrotechnique internationale, un transformateur de tension est un « transformateur de mesure dans lequel la tension secondaire est, dans les conditions normales d'emploi, pratiquement proportionnelle à la tension primaire et déphasée par rapport à celle-ci d'un angle voisin de zéro, pour un sens approprié des connexions ».

Ils ont de types, selon leur raccordement :

- Phase/phase : primaire raccordé entre deux phases.
- Phase/terre : primaire raccordé entre une phase et la terre.

- **Parafoudre**

Les parafoudres sont des appareils destinés à limiter la surtension imposée aux transformateurs, instruments et machines électriques par la foudre et par les manœuvres de commutation. La partie supérieure du parafoudre est reliée à un des conducteurs de la ligne à protéger et la partie inférieure est connectée au sol par une mise à la terre de faible résistance.



Figure1.20 Parafoudre mise à terre

1.8. Conclusion

Dans ce chapitre, nous avons couvert une série d'éléments qui régissent les réseaux électriques en général en décrivant et en définissant le sujet et en identifiant les types de réseaux constitués de différents niveaux de tension.

En plus de présenter les stations dont les structures et le rôle ont été révélés dans la livraison, la distribution et la conversion de l'énergie électrique de la haute à la basse tension afin de mettre en œuvre les travaux de réseau électrique requis et de les exploiter à distance. Le but de notre projet de fin d'étude est de fournir une solution de gestion des défauts du réseau électrique, le chapitre suivant expliquera l'élément central de notre projet à savoir la carte d'acquisition est l'élément de contrôle qui n'est rien de plus que la carte de prototypage ARDUINO.

Chapitre 2

Carte d'acquisition et logiciels nécessaire pour la réalisation du projet

2.1 Introduction

Après avoir détaillé la problématique posée à savoir la proposition d'une solution permettant de répondre en temps réel aux Pannes pouvant affecter le réseau, nous aborderons dans ce chapitre le cœur qui nous permettra de détecter et de transmettre à la centrale les pannes principales pouvant apparaître dans le réseau, à savoir la carte de prototypes ARDUINO ainsi que le logiciel qui nous permettra de réaliser l'interface graphique.

2.2 Carte d'acquisition et de commande ARDUINO

Les cartes Arduino possèdent de nombreuses entrées-sorties. Le choix du type de carte Arduino s'effectue en fonction des besoins de votre projet.

2.2.1 Architecture de la carte (Hardware)

Différents composants électroniques sont nécessaires pour la réalisation de la Carte Arduino, ces derniers sont soudés sur un circuit imprimé, mais en premier lieu, définissant le circuit imprimé [13].

2.2.2 Définition du circuit imprimé

Le circuit imprimé est un support plan, flexible ou rigide, généralement composé d'époxy 1 ou de fibre de verre, il possède des pistes électriques disposées sur une ou plusieurs couches (en surface et/ou en interne) qui permettent la mise en relation électrique des composants électroniques. Chaque piste relie tel composant à tel autre, de façon à créer un système électronique qui fonctionne et qui réalise les opérations demandées, ce système porte le nom de carte électronique.

Evidemment, tous les composants d'une carte électronique ne sont pas forcément reliés entre eux. Le câblage des composants suit un plan spécifique à chaque carte électronique, qui se nomme le schéma électronique. En ce qui nous concerne, les composants utilisés forment trois circuits reliés entre eux, un circuit pour l'alimentation, un autre pour la communication avec la partie Software, et le dernier s'agit du noyau, le cœur de la carte. Le schéma synoptique ci-dessous nous donne une idée générale sur l'architecture de la carte Arduino [13].

2.2.3 Schéma synoptique

Les principaux éléments constituant la carte de prototypage ARDUINO et représenté à travers ce synoptique

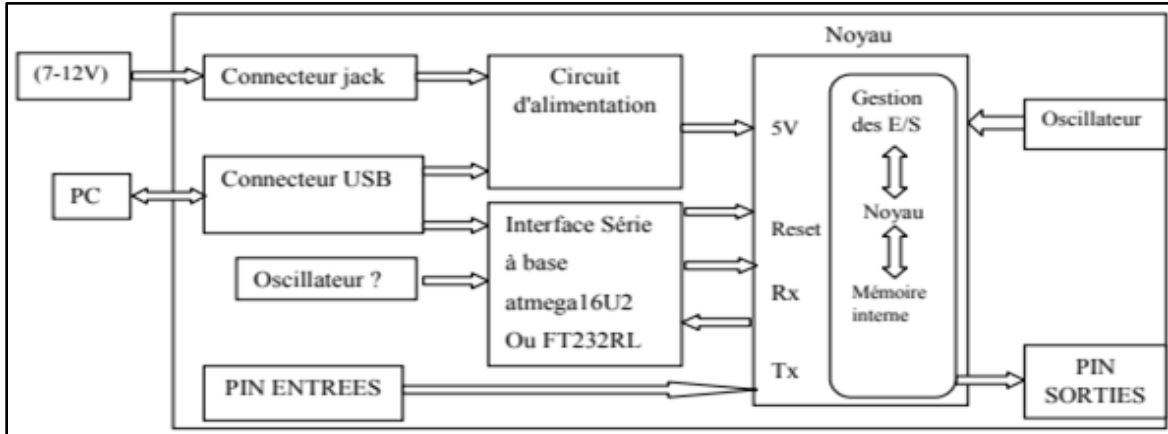


Figure 2.1 Schéma Synoptique de la carte Arduino [13]

2.2.4 Différents types d'Arduino

Il existe plusieurs type d'Arduino qui sont représenter dans la figure ci-dessous :

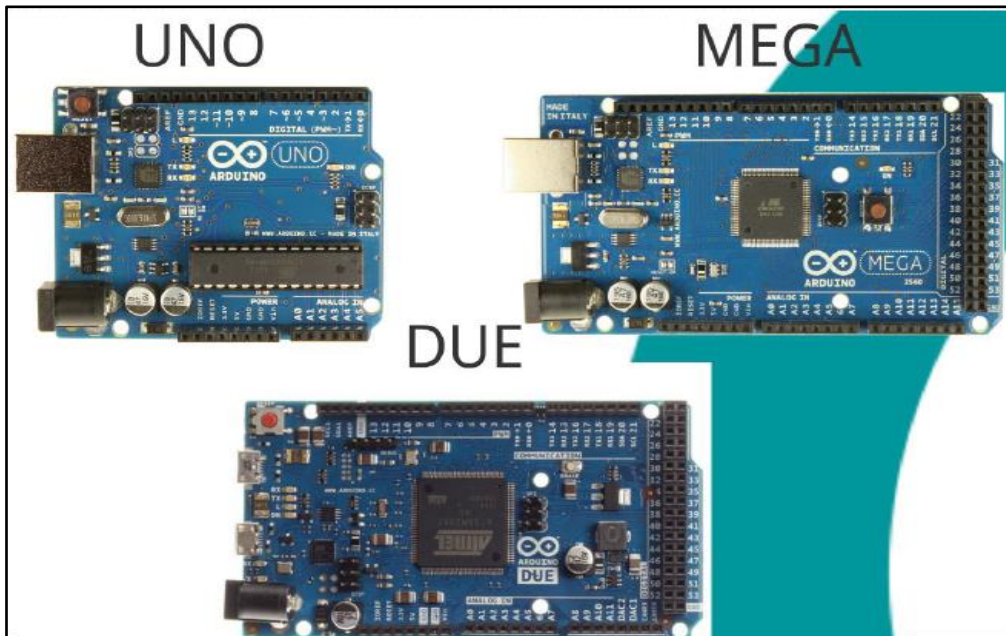


Figure 2.2 Différents types d'Arduino

2.2.5 Présentation d'Arduino Méga 2560

La carte Arduino repose sur un circuit intégré (un mini-ordinateur appelé également microcontrôleur) associée à des entrées et sorties qui permettent à l'utilisateur de brancher différents types d'éléments externes :

Côtés entrés, des capteurs qui collectent des informations sur leur environnement comme la variation de température via une sonde thermique, le mouvement via un détecteur de présence ou un accéléromètre, le contact via un bouton-poussoir, etc.

Côtés sortis, des actionneurs qui agissent sur le monde physique telle une petite lampe qui produit de la lumière, un moteur qui actionne un bras articulé, etc. Comme le logiciel Arduino, le circuit électronique de cette plaquette est libre et ses plans sont disponibles sur internet. On peut donc les étudier et créer des dérivés.

Plusieurs constructeurs proposent ainsi différents modèles de circuits électroniques programmables et utilisables avec le logiciel Arduino. Il existe plusieurs variétés de cartes Arduino [6].

ARDUINO	Microcontrôleur	Flash Ko	E/S Numériques	PWM	Entrée analogique	Type D'interface USB
Diecimila	ATmega168	16	14	6	6	FTDI
Due	ATmel SAM3X8E	512	54	12	12	ATmega16u2
Duemilanove	ATmega168/328P	16/32	14	6	6	FTDI
Esplora	ATmega32U4	32	N/A	N/A	N/A	ATmega32U4
Leonardo	ATmega32U4	32	20	7	12	ATmega32U4
Méga2560	ATmega2560	256	54	15	16	ATmega8U2
Uno	ATmega328P	32	14	6	6	ATmega16U2

Tableau 2.1 Récapitulatif des différents types d'Arduino

2.2.6 Périphériques

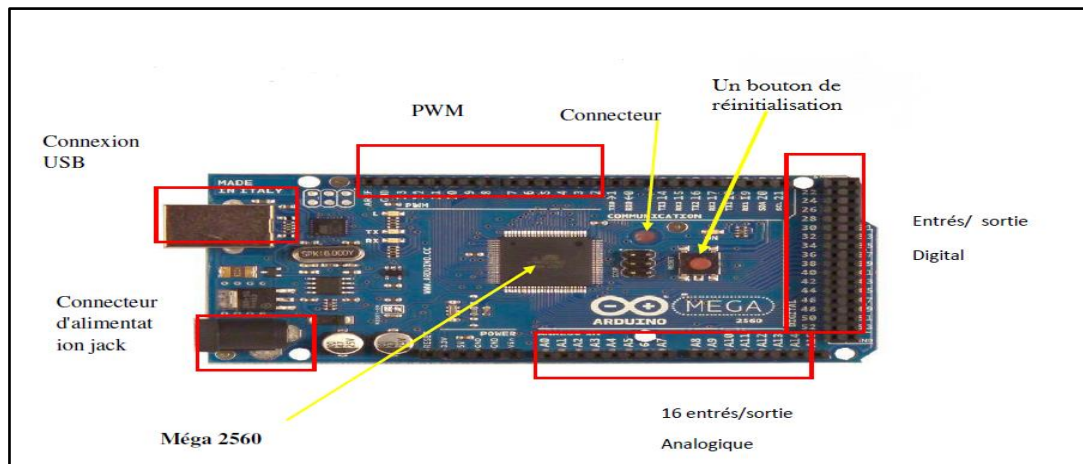


Figure 2.3 Caractéristiques d'une carte Arduino Méga 2560 [20]

2.2.7 Caractéristiques de l'Arduino Méga 2560

Cette carte dispose de 54 broches numériques d'entrées/sorties (dont 14 peuvent être utilisées en sortie PWM (largeur d'impulsion modulée) ;

- De 16 entrées analogiques (qui peuvent également être utilisées en broches entrées/sorties numériques).
- De 4 UART (port série matériel).
- D'un quartz 16Mhz.
- D'une connexion USB.
- D'un connecteur d'alimentation jack.
- D'un connecteur ICSP (programmation "in-circuit").
- Et d'un bouton de réinitialisation (reset).

Elle contient tout ce qui est nécessaire pour le fonctionnement du microcontrôleur ; Pour pouvoir l'utiliser et se lancer, il suffit simplement de la connecter à un ordinateur à l'aide d'un câble USB (ou de l'alimenter avec un adaptateur secteur ou une pile, mais ceci n'est pas indispensable, l'alimentation étant fournie par le port USB).

2.3 Interface de programmation IDE (Software)

L'interface de programmation Arduino :

2.3.1 Installation

L'installation de l'interface de programmation Arduino est relativement simple et possible sur les plates-formes Windows, Mac OS X et Linux. L'environnement de programmation Arduino (IDE en anglais) est une application écrite en Java et inspirée du langage Processing. Pour télécharger le fichier d'installation, il suffit de se rendre sur le site officiel Arduino. Pour sélectionner une version, on doit cliquer sur le nom qui correspond à notre système d'exploitation et sauvegarder sur l'ordinateur le fichier correspondant. Il est à noter que la version anglaise contient habituellement des mises à jour plus récentes que la version française. L'environnement Arduino est en open source, donc facilement téléchargeable, une fois téléchargé avec le lien On le décompresse, puis on le copie sur l'ordinateur, ensuite on lance l'application Arduino Double -cliquer sur :

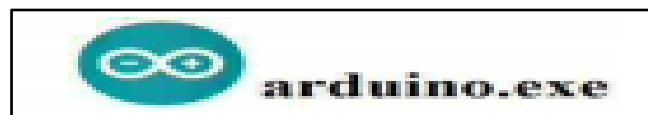


Figure 2.4 Icône d'Installation

L'installation de l'IDE sur les différents systèmes d'exploitation se ressemble beaucoup Sauf pour les pilotes (ou driver) requis pour la partie raccordement, en effet :

- a) Sous Windows, durant la connexion de la carte interface, il est question de démarrage du processus d'installation du driver, mais pour l'installer, il faut se rendre à la Gestion de l'ordinateur, puis sélectionner Gestion des périphériques, le pilote non installé sera dans la rubrique autres périphérique avec un point d'exclamation en jaune, là on devra appuyer sur mettre à jour le pilote, une autre fenêtre va s'ouvrir :

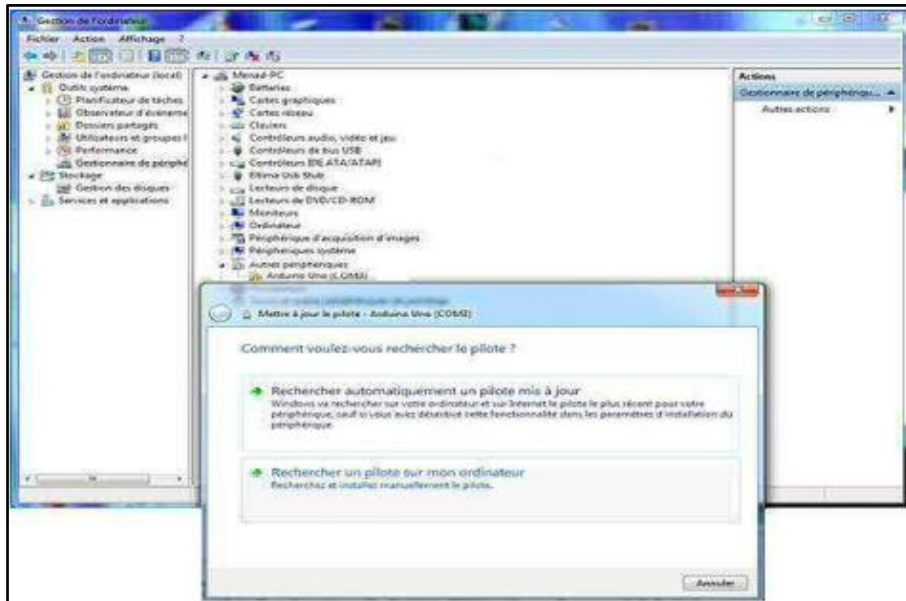


Figure 2.5 Mettre à jour du pilote

Là, on clique sur Rechercher pilote sur mon ordinateur, puis une autre fenêtre va apparaître :

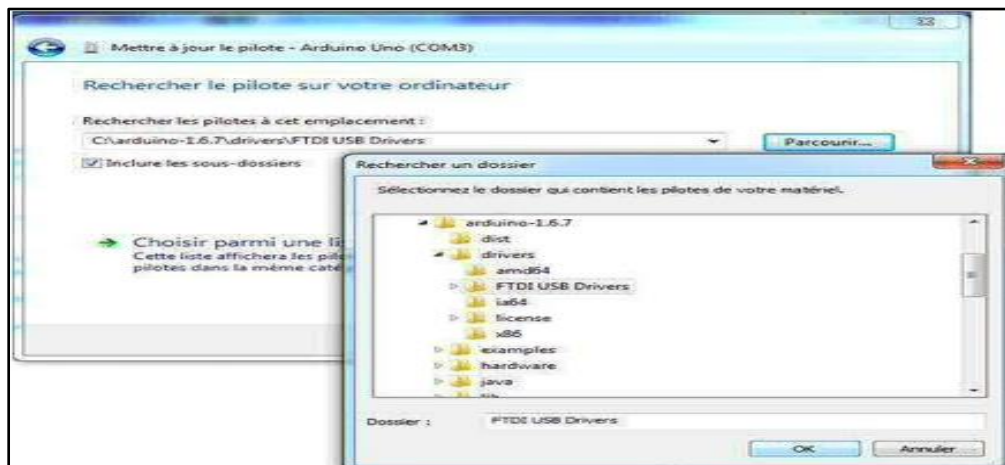


Figure 2.6 Sélectionner le pilote

On appuis sur Parcourir..., puis on ouvre le dossier Arduino-1.6.7 ou la version de l'IDE installée_puis Drivers et en fin, on sélectionne FTDI USB Drivers.

Cette fenêtre apparaîtra si le pilote est installé correctement.

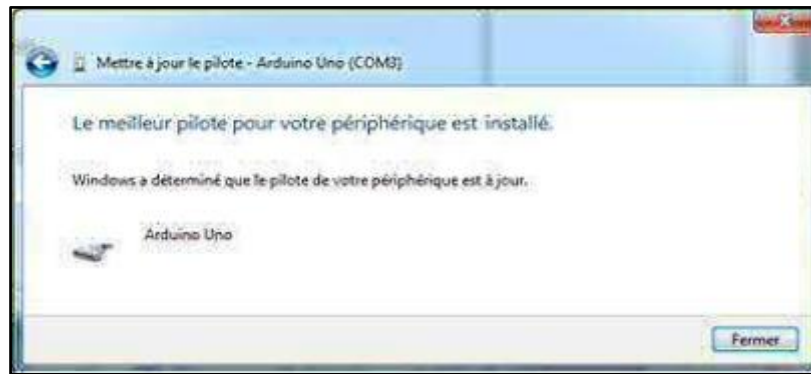


Figure 2.7 Pilote mis à jour.

Si le pilote n'est pas présent dans le dossier de l'IDE, il faut le télécharger.

b) Sous MAC OS X, Les pilotes sont déjà préexistants, mais pour les versions autres que l'UNO et la Mega2560, une nouvelle fenêtre apparaît indiquant qu'une nouvelle interface réseau a été détectée, on doit alors sélectionner « Préférence Réseau », à l'ouverture, on clique sur suivant, en fin on peut lancer l'application Arduino.

c) Sous GNU/Linux, La procédure d'installation sous GNU/Linux dépend de la distribution utilisée et l'exécution du programme dépend également de l'installation d'autres pilotes ou logiciels. Pour les utilisateurs de Linux.

2.3.2 Description de l'IDE

L'IDE est un logiciel de programmation qui permet d'écrire, de modifier un programme et de le convertir en une série d'instructions compréhensibles pour la carte. Il programme par code, contenant une cinquantaine de commandes différentes. A l'ouverture, l'interface visuelle du logiciel contient le menu, des boutons de commande en haut, une page blanche vierge, une bande noire en bas, comme ceci :

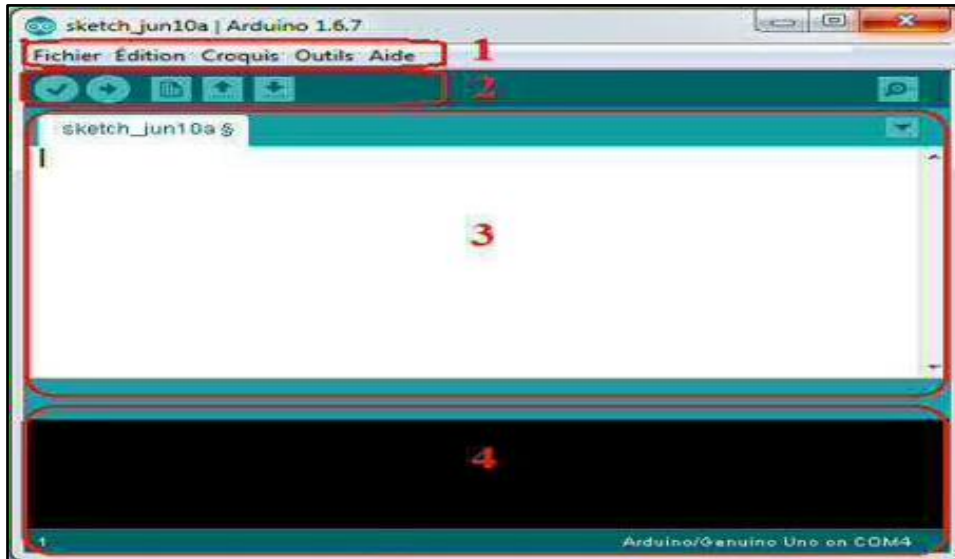



Figure 2.8 IDE

- Menu : Les différents éléments du menu permettent de créer de nouveaux sketches (programmes), de les sauvegarder, et de gérer les préférences du logiciel et les paramètres de communication avec votre carte Arduino. Le menu comprend :
Fichier : pour créer, sauvegarder en spécifiant la destination, et d'appeler un programme, Edition : Pour couper, copier, coller, supprimer, sélectionner, ...etc. ;
Croquis : regroupe les fichiers réalisées
Outils : pour spécifier le type de la carte, le port série, formater, recharger et réparer l'encodage, graver la séquence d'initialisation, de la carte branchée sur l'ordinateur.



Figure 2.9 Barre d'Action

- 1)  Bouton Verify (Compiler) : il permet de vérifier le programme pour trouver d'éventuelles erreurs. Cette procédure prend un certain temps d'exécution et lorsque elle est terminée, elle affiche un message de type « Binary sketch size : ... » Indiquant la taille du programme téléversé.

- 2)  Bouton Upload (Téléverser) : ce bouton permet de compiler et téléverser le programme sur la carte Arduino.
 - 3)  Bouton New (Nouveau) : ce bouton permet d'ouvrir une nouvelle fenêtre de programmation.
 - 4)  Bouton Open (Ouvrir) : il fait apparaître un menu qui permet d'ouvrir un programme qui figure dans le dossier de travail ou des exemples de programmes intégrés au logiciel.
 - 5)  Bouton Save (Sauvegarder) : il permet de sauvegarder le programme.
 - 6)  Bouton Serial Monitor (Moniteur sériel) : ce bouton fait apparaître le moniteur série.
- Fenêtre de Programmation : est l'éditeur où s'écrit le programme, chaque logiciel obéit à quelques notions pour pouvoir bien structurer le programme à fin de le compiler et éviter les erreurs de syntaxe et autres.
 - Barre des erreurs : La barre des erreurs affiche les erreurs faites au cours du programme, comme l'oubli d'un point-virgule, le manque d'une accolade ou toute autre erreur dans les instructions. Un exemple de ce genre s'affiche en cas d'erreurs :

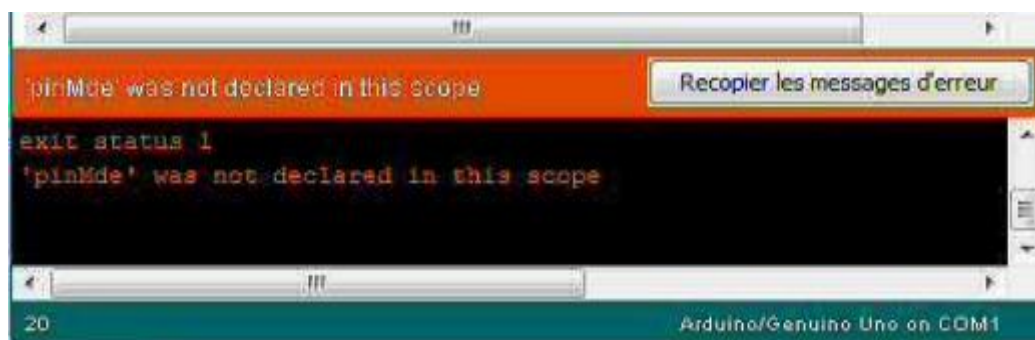


Figure 2.10 Barre des erreurs

2.3.3 Structure du programme

Le programme comporte trois phases consécutives :

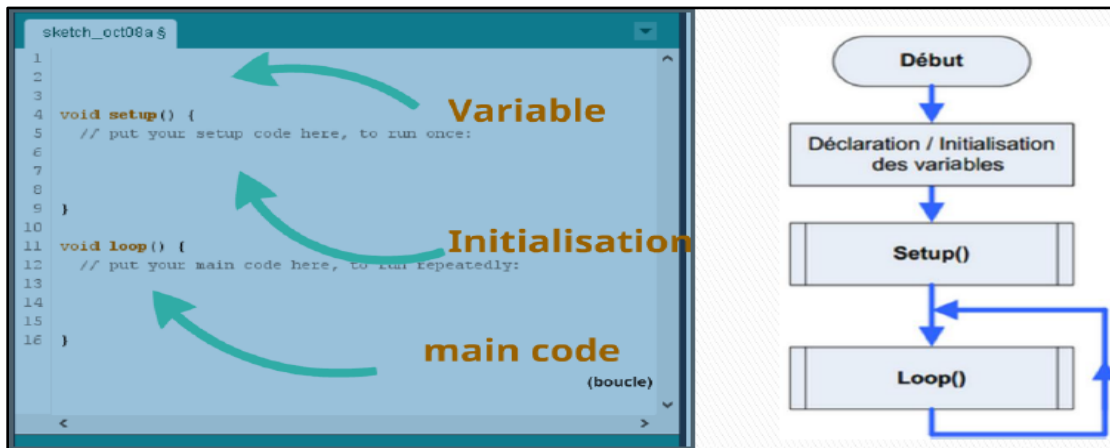


Figure 2.11 Structure du programme.

a) Définition des constantes et des variables

Cette partie est optionnelle, chaque entrée et sortie est définie et déclarée, en lui donnant un nom arbitraire et en lui affectant le numéro de l'entrée ou celui de la sortie voulue, sans oublier de préciser le type de la variable.

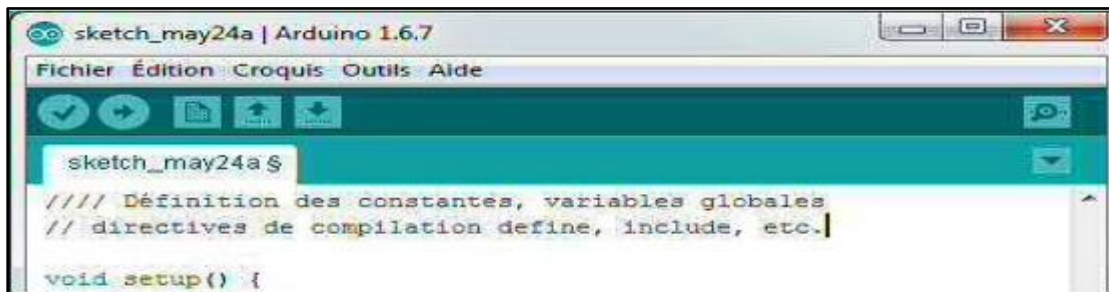


Figure 2.12 Définition des constantes et des variables.

b) Configuration des entrées/sorties

Les instructions viennent après le <<void setup ()>>, après avoir ouvert une accolade, on peut manipuler les broches de la carte en les configurant comme étant des entrées ou des sorties, selon les besoins. Les entrées analogiques pour les capteurs par exemple, ne sont soumises à aucune configuration, car la carte possède 6 entrées analogiques qui ne font que cela.

```
void setup() {  
  // initialisation des ressources de la carte,  
  // configuration des entrées/sorties,  
  // définition de la vitesse de fonctionnement du port série, etc.  
  // setup() n'est exécuté qu'une seule fois.  
}
```

Figure 2.13 Configuration des Entrées/Sorties

c) Programmation des interactions et comportements

Celles-ci viennent après le <<void loop ()>>, c'est la partie principalement, ou on rédige les instructions et les opérations comme la lecture des données, les boucles, les affectations, etc. Chacune d'elle doit obligatoirement finir par un point-virgule.

```
void loop() {  
  // les instructions contenues ici sont exécutées indéfiniment en boucle  
  // Seule une coupure de l'alimentation de la carte ou un appui sur le bouton Reset  
  // permet de quitter le programme.  
}
```

Figure 2.14 Programmation des interactions et comportements

d) Commentaires

Comme chaque IDE, des commentaires peuvent être ajoutés au programme. Dans la configuration des entrées/sorties, les commentaires doivent être écrits après un slash ou une étoile ou les deux, tandis que sur une ligne de code, on les écrit après deux slashes.

2.3.4 Langage de programmation

- Comme nous l'avons mentionné, le langage de l'IDE Arduino est un mélange entre le C et le C++, il possède un jeu d'instruction très riche. Ces instructions décrivent :
- Les données, qui peuvent être numériques (byte, int, word,), logiques (booleen, ...), sous forme de tableaux (array), caractères ou chaînes de caractères (char, string), ou constantes particulières (True/False, HIGH/LOW) ...etc.

- Les fonctions arithmétiques et mathématiques : comme les fameuses quatre opérations arithmétiques (+, -, * et /) simples ou composées, les fonctions mathématiques (abs, min, max,...) et trigonométriques (cos, sin, ...).
- Les opérateurs logiques (&&, !, ...) et les opérateurs de comparaisons (=, , ..).
- Les structures de contrôle comme les boucles (for, while,..), les prises de décision (if-else,...), les sauts (break, goto, continue,...).
- Gestions du temps (delay) et des entrées /sorties numérique (pinMode, digitalWrite/Read) ou analogique (analogRead/Write).
- Fonctions diverses pour générer des nombres aléatoires (random, randomSeed), et pour manipuler des bits (low/highByte, bitRead/Write/Set/Clear,), ainsi que pour gérer les interruptions (attach/detach/noInterrupt).
- Gestion du port série (Serial.begin/.end/.available/.read/.print,...etc.).

2.3.5 Compilation et Téléversement

Une fois le programme terminé et vérifié, on passe au test, en appuyant sur le bouton Compiler, une barre de progression s'affiche au-dessus de la barre des erreurs, s'il n'y a pas d'erreur, on verra s'afficher le message « Compilation terminée », suivi de la taille du programme. Dans le cas contraire, un message d'erreur s'affiche.

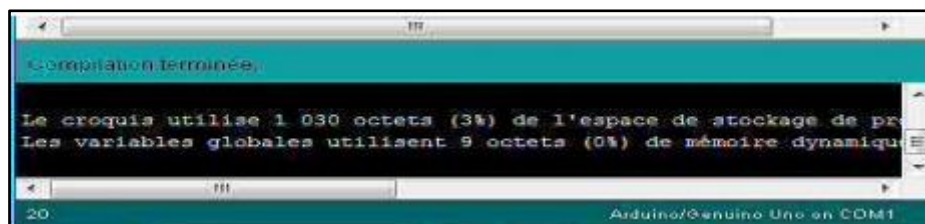


Figure 2.15 Compilation

Une fois la compilation terminée, le programme sera sauvegardé. Mais avant de téléverser, il faut :

- Réinitialiser la carte en appuyant sur le bouton de réinitialisation.
- La brancher sur l'ordinateur. Une fois branchée, on clique sur le bouton « Outils ».
- On sélectionne « Type de carte », puis on coche la carte correspondante.

- Toujours dans le menu « Outils », on clique sur « Serial port », pour sélectionner le type de la connexion.
- On télécharge le programme en cliquant sur « Téléverser » plus haut sur la barre d'action, le message « Compilation du croquis » s'affiche au-dessus de la barre des erreurs, ce message signifie que le programme est en train d'être traduit en langage machine. Puis, on verra « Téléversement... », qui veut dire que le programme se charge dans la carte. La LED LX clignote dans le cas où la carte n'est pas défectueuse.

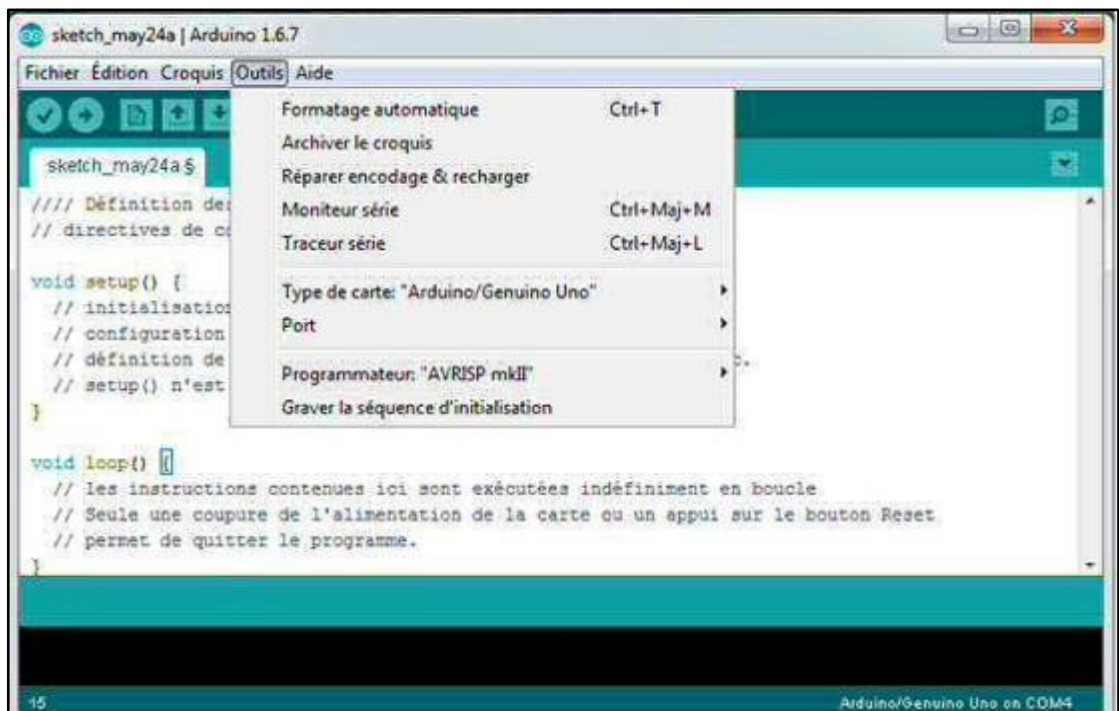


Figure 2.16 Sélection de la carte et du port

- Pour connaître le type de port, on doit se rendre dans le « Panneau de configuration », on sélectionne « Gestionnaire des périphérique », puis on regarde la ligne « Port (COM et LPT), et là on peut voir le nom de l'Arduino suivi du type de la connexion entre parenthèses, par exemple « Arduino Mēga (COM3).

2.3.6 Principaux fonctionnements

Les principales fonctionnalités de l'interface de l'application Arduino vous permet de créer et éditer un programme (appelé sketch) qui sera compilé puis téléversé sur la carte Arduino. Ainsi, lorsque vous apportez des changements sur le code, ces changements ne seront effectifs qu'une fois le programme téléversé sur la carte. Il est à noter que ce manuel fait

référence à la version en anglais de ce logiciel puisqu'elle comporte habituellement des mises à jour plus récentes que la version en français.

Que les non-anglophones se rassurent : le nombre réduit de fonctionnalités et l'utilisation d'icônes rendent l'interface du logiciel simple d'utilisation.

2.3.7 Communication séries (ARDUINO)

La communication série est indispensable pour dialoguer avec votre Arduino puisque c'est le mode de transmission utilisé pour communiquer avec la carte.

A. Bibliothèque Serial

La bibliothèque Serial (incluse dans le langage Arduino) est utilisée pour les communications séries asynchrones basées sur le circuit UART.

Ce circuit utilise des ports numériques de la carte (voir Câblage), que l'on ne peut par conséquent plus utiliser en tant qu'entrées ou sorties numériques.

Grâce à cette bibliothèque, il est possible d'établir des communications simples entre :

- Un Arduino et un ordinateur.
- Deux Arduino.
- Un Arduino et tout autre périphérique compatible UART [19].

• **Arduino_ordinateur**

Le port USB de la carte <Arduino> utilise lui aussi le circuit UART : pour communiquer avec l'ordinateur relié à la carte, il suffit de le relier via un port USB.



Figure 2.17 Communication Arduino et ordinateur

- **Communication Arduino – Arduino**

Pour relier entre eux deux <Arduino> par des câbles, il faut utiliser les ports numériques associés aux signaux (TX) et (RX), et mettre en commun leur masse (GND).

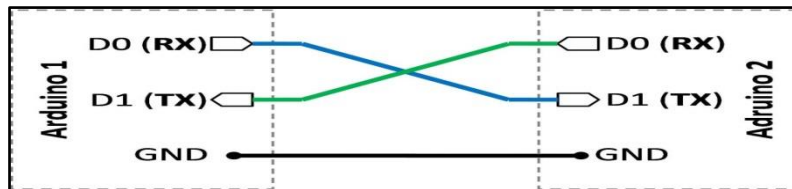


Figure 2.18 Communication Arduino – Arduino

- Lorsque le câble USB est relié à un ordinateur, le port Serial est automatiquement occupé par cette connexion. Il est impossible de l'utiliser avec un autre appareil [19].

- **Programmation**

Avant de pouvoir utiliser un port série (quel que soit le média) il faut l'initialiser, ce qui doit se faire dans la fonction

setup ():

```
void setup() {
```

```
    // Initialisation du port série
```

```
    Serial.begin(9600); // 9600 la vitesse de transmission
```

```
}
```

- **Écriture**

Du point de vue de l'Arduino, écrire, c'est envoyer des données vers le périphérique connecté.

On distinguera principalement 3 fonctions pour écrire sur le port série :

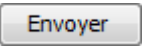
- Serial.write() // écriture d'un octet ou d'une chaîne de caractères.
- Serial.print() // écriture de n'importe quelle valeur convertible en chaîne de caractères.
- Serial.println() // comme.
- Serial.print() // mais avec un retour à la ligne à la fin.

Exemple

```
void loop() {  
  
Serial.print("Arduino :"); // Écriture d'une ligne de texte ...  
  
Serial.println(" l'essentiel !"); // ... avec retour à la ligne  
  
delay(1000);  
  
}
```

a. Lecture

Du point de vue de l'Arduino, écrire, c'est recevoir des données depuis le périphérique connecté.

Le moniteur série dispose d'une zone de saisie de texte, associée à un bouton  pour envoyer le message saisi.

Du côté de l'Arduino (le récepteur), les données sont d'abord stockées dans une zone mémoire spéciale appelée *serial buffer*.

• Fonctions de lecture de la bibliothèque Serial

```
Serial.read() //Lit un octet (byte) du buffer (et l'en retire).  
Serial.readString() // lit le buffer comme une chaîne de caractère (String).
```

Exemple

```
void loop() {  
  
// si le buffer n'est pas vide ...  
while (Serial.available() > 0) {  
int v = Serial.parseInt();  
Serial.print("Vous avez saisi : ");  
Serial.println(v);  
}  
} [19].
```

B. Câblage

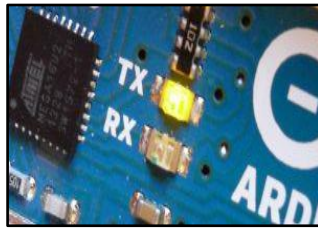


Figure 2.19 RX et TX

- Pins : 0 (RX) et 1 (TX) sont utilisés pour la réception (RX) et d'émission (TX) des données en série TTL. Ceux - ci sont connectés aux broches correspondantes du port USB-TTL du processeur ATmega16U2 ou la puces FT232RL.

La carte Arduino Mega dispose de trois ports série supplémentaires :

- Serial1 : 19 (RX) et 18 (TX).
- Serial2 : 17 (RX) et 16 (TX).
- Serial3 : 15 (RX) et 14 (TX).

La communication peut se faire par différents médias

- Câbles (USB, série, ...)

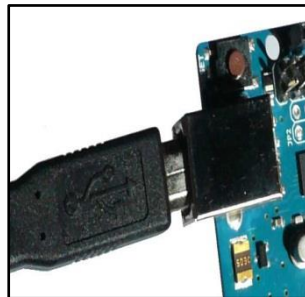


Figure 2.20 Câbles (USB, série)

2.3.8 Schéma de la maquette de simulation

Pour mieux prédire le comportement de notre maquette de simulation est exigée, on a réalisé ce Schéma suivant pour la version basic. Ce schéma a été réalisé avec le logiciel Isis Professional, qui est en même temps le logiciel que nous avons choisi pour simuler notre projet.

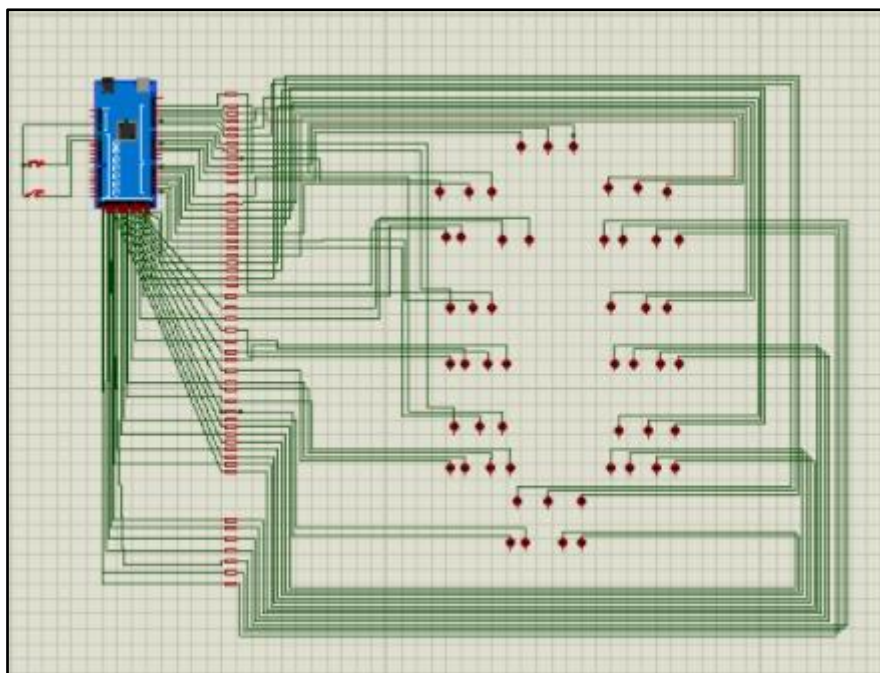


Figure 2.21 Simulation avec logiciel Isis

2.4 Logiciel Processing

Puisque l'objectif majeur de notre travail est de réaliser une interface graphique ; nous allons dans ce paragraphe présenter l'outil qui nous permettra de matérialiser notre but à savoir le logiciel processing [16].

Le logiciel Processing est un carnet de croquis logiciel flexible et un langage pour apprendre à coder dans le contexte des arts visuels. Depuis 2001, Processing fait la promotion de la maîtrise du logiciel dans les arts visuels et de la maîtrise visuelle dans la technologie. Il y a des

dizaines de milliers d'étudiants, d'artistes, de designers, de chercheurs et d'amateurs qui utilisent Processing pour l'apprentissage et le prototypage.[14].

Notre choix c'est diriger vers le logiciel Processing pour plusieurs raisons dans les principaux sont :

- Téléchargement gratuit et open source.
- Programmes interactifs avec sortie 2D, 3D, PDF ou SVG.
- Intégration OpenGL pour la 2D et la 3D accélérée.
- Pour GNU/Linux, Mac OS X, Windows, Android et ARM.
- Plus de 100 bibliothèques étendent le logiciel de base.
- Bien documenté , avec de nombreux livres disponibles.
- Processing est un logiciel libre (open-source) totalement gratuit pour l'utilisateur.

Le logiciel fonctionne sur Macintosh, sous Windows et sous Linux, en effet il est basé sur la plate-forme Java. Les programmes réalisés avec Processing peuvent être lus par les navigateurs internet équipés du plug-in java, mais aussi sous forme d'applications indépendantes pour Windows, Linux ou Mac (en réalité n'importe quelle machine disposant d'une Machine virtuelle Java) et également Android.[15]



Figure 2.22 fonctionnement de logiciel

2.4.1 Bases de Processing

Processing est très proche de celle du logiciel Arduino.

A. Interface

L'interface d'utilisation de Processing est composée de deux fenêtres distinctes : la fenêtre principale dans laquelle vous allez créer votre projet et la fenêtre de visualisation dans laquelle vos créations (dessins, animations, vidéos) apparaissent.

On trouve plus précisément les éléments suivants dans l'interface :

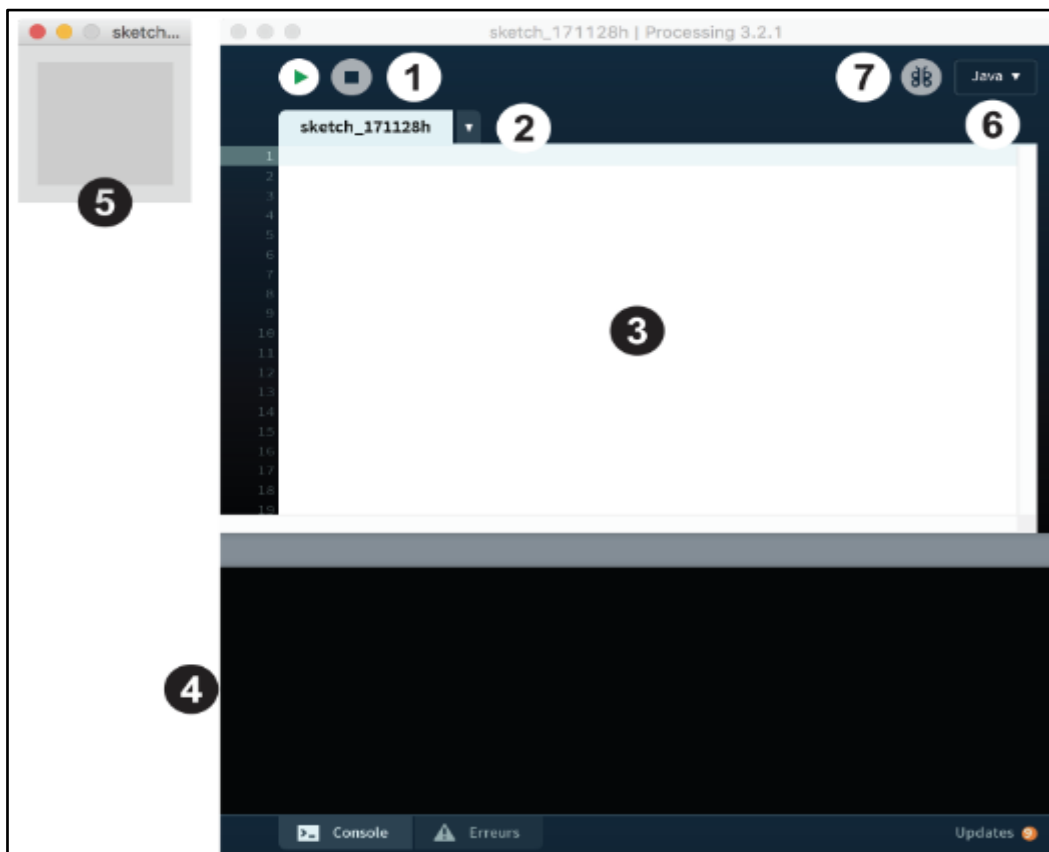


Figure 2.23 Fenêtre principale et la fenêtre de visualisation

1. Barre d'actions.
2. Barre d'onglets.
3. Zone d'édition (pour y saisir votre programme).

4. Console, qui comprend un onglet les messages affichés par programme un onglet pour les erreurs. Cette console indique aussi si des mises à jour (« *updates* ») sont disponibles pour les librairies et les modes.
5. Fenêtre de visualisation (espace de dessin).
6. Liste déroulante pour les modes.
7. Bouton pour activer le mode debug (pas à pas) [15].

B. Barre d'actions



Bouton "Run" : exécute votre sketch (votre programme).



Bouton "Stop" : arrête l'exécution de votre sketch.

Processing permet de travailler dans plusieurs modes, un mode permettant de programmer dans un environnement spécifique à chaque plateforme visée (ex : application, application pour tablette Android, ...). Ces modes peuvent être gérés depuis une interface spécifique, le « Contribution Manager ».

Vous pouvez changer ce mode à tout moment depuis l'interface, en ayant au préalable sauvegardé votre sketch. Pour plus d'informations sur le rôle des modes dans Processing, veuillez consulter le chapitre à ce sujet.

- **Langage de programmation**

Le langage de base de processing est le Java, auquel est ajouté des fonctions propres à Processing. Un programme contient un code minimal de 2 fonctions :

```
void setup() //fonction d'initialisation de la carte
{
}

void draw() //fonction principale, elle se répète à l'infini
{
}
```

Figure 2.24 Langage de programmation

- **Setup () et draw ()**

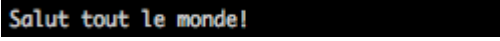
La fonction `setup ()` est appelée au démarrage du programme. Cette fonction est utilisée pour initialiser les variables, pour préparer la fenêtre de visualisation (l'espace de dessin), par exemple en lui donnant une taille au départ. La fonction `setup ()` n'est exécutée qu'une seule fois au lancement du programme.

La fonction `draw ()` s'exécute en boucle infinie, permettant de créer l'interface graphique. L'exécution de cette boucle s'effectue par défaut au rythme de 30 fois par seconde, jusqu'à ce que l'utilisateur arrête le programme. La fonction `frameRate(x)` permet toutefois de modifier cette valeur (x étant le nombre de répétitions par seconde) [15].

- **Affichage dans la console**

La console (la zone 4 dans le schéma de l'interface présenté en début de ce chapitre) permet d'afficher du texte brut à des fins de test et de débogage (correction d'erreurs). Pour afficher quelque chose dans cette zone, il faut utiliser la méthode `println ()`;

```
Println ("Salut tout le monde !");
```



```
Println (1000);
```



- **Espace de dessin**

L'espace de dessin constitue la fenêtre de visualisation où s'affichera vos réalisations dans Processing en 2 ou 3 dimensions [15].

- **Coordonnées dans l'espace**

En dimensions (2D), on utilise deux axes de coordonnées x et y correspondant respectivement à la largeur et à la hauteur. Le coin en haut à gauche correspond aux valeurs $x=0$ et $y=0$.

Instruction	Définition
size (largeur,hauteur)	taille de la fenêtre de dessin, par défaut de 100x100 pixels

Figure 2.25 Coordonnées dans l'espace

En 3 dimensions (3D), en plus des deux axes de coordonnées, on a un troisième axe de coordonnées z , exprimant la profondeur : dans ce cas précis, on utilise la commande size avec un troisième paramètre indiquant que l'on travaille dans un espace 3D :

size(100. 100.P3D) ;

- Formes

Forme	Instruction	
le point	point(x, y)	
la ligne	line(xA, yA, xB, yB)	
le rectangle	rect(x, y, largeur, hauteur)	
l'ellipse	ellipse(x, y, largeur, hauteur)	
le triangle	triangle(x1, y1, x2, y2, x3, y3)	
l'arc	arc(x, y, largeur, hauteur, début, fin)	
le quadrilatère	quad(x1, y1, x2, y2, x3, y3, x4, y4) les quatre paires de coordonnées x et y sont données dans le sens horaire.	

Tableau 2.2 Formes et instruction

- Couleurs

Instruction	Définition
background(x)	couleur de la fenêtre en niveaux de gris; x de 0 (noir) à 255 (blanc)
background(R,G,B)	couleur de la fenêtre en RGB (de 0 à 255 par couleur)
stroke(x)	couleur de tracé d'une forme en niveaux de gris; x de 0 (noir) à 255 (blanc)
stroke(R,G,B)	couleur de tracé d'une forme en RGB (de 0 à 255 par couleur)
stroke(R,G,B,x)	couleur de tracé d'une forme en RGB (de 0 à 255 par couleur), x opacité de la couleur (de 0 le plus clair à 255 le plus foncé)
strokeWeight(x)	fixe la largeur du trait de tracé en pixels
noStroke()	contours invisibles
fill(x)	couleur de remplissage d'une forme en niveaux de gris; x de 0 (noir) à 255 (blanc)
fill(R,G,B)	couleur de remplissage d'une forme en RGB (de 0 à 255 par couleur)
fill(R,G,B,x)	couleur de remplissage d'une forme en RGB (de 0 à 255 par couleur), x opacité de la couleur (de 0 le plus clair à 255 le plus foncé)
noFill()	couleur de remplissage invisible

Remarque 1 : il est toujours possible de définir les couleurs en notation hexadécimale, par exemple #ffcc33.
Remarque 2 : par défaut, toute modification de style (couleur de remplissage ou de contour, épaisseur ou forme de trait) s'appliquera à tout ce que vous dessinerez ensuite. Pour limiter la portée de ces modifications, vous pouvez les encadrer par les commandes pushStyle() et popStyle().

Tableau2.3 Définition des Couleurs et instruction

- Texte

Il est possible d'écrire du texte sur le dessin avec l'instruction « text »

Instruction	Définition
text (message, x, y)	affiche le message entre " " aux coordonnées x,y

Remarque : la seule possibilité d'effacer ou de modifier un texte est de réécrire par-dessus.

Tableau2.4 Ecrire du texte

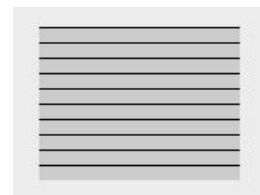
- Répétitions

La boucle << for >> :

```
for (int i = NombreDeDépart; i < NombreMaximal; i = i + INCREMENT) { }
```

exemple des lignes:

```
for (int i = 0; i < 100; i = i + 10) {  
    Line (0, i, 100, i);  
} [15]
```



2.4.2 Communication série

Les ordinateurs communiquent entre eux à l'aide d'impulsions électriques à travers des fils. Les impulsions sont envoyées les unes après les autres, dans une série de bits. Un bit peut être soit vrai, soit faux. C'est le chiffre un ou zéro. Les bits sont souvent groupés en paquets de huit. On appelle ces paquets des octets. Un caractère de texte utilisant le jeu de caractères ASCII est exprimé sur huit bits.

Pour pouvoir échanger ces informations sous forme de bits avec l'extérieur, votre ordinateur utilise une ou plusieurs entrées physiques (des ports en jargon informatique) lui permettant de se connecter à des appareils périphériques (imprimante, souris, clavier, modem, etc.).

Tout cela pour vous dire que les impulsions électriques et les instructions informatiques ne sont pas si éloignées que cela dans un ordinateur et qu'il est possible de communiquer facilement avec un Arduino à l'aide de simples caractères textuels [15].

2.4.3 Code pour Processing

Envoyer et recevoir du sériel avec Processing est assez facile avec la librairie `processing.Serial`. Il faut spécifier le bon numéro de port série (le numéro de ce port, peut-être 0, 1, 2...), ainsi que la bonne vitesse. Ici, notre vitesse est de 9600 bits par seconde.

2.5 Conclusion

Dans ce chapitre nous avons présenté la carte d'acquisition et de transmission de données ainsi que le logiciel nécessaire pour le développement de l'interface graphique de supervision du réseau électrique pour l'intervention à distance.

Le chapitre suivant nous allons aborder la réalisation et simulation pratique de notre projet de fin d'étude.

Chapitre 3

Réalisation et Simulation

3.1 Introduction

Après avoir détailler la problématique de notre projet de fin d'étude ainsi que les éléments nécessaires pour atteindre notre objectif à savoir la carte de prototypage ARDUINO ainsi que le logiciel nécessaire pour la réalisation de l'interface graphique, Pour faciliter le contrôle du système de contrôle à distance DMS. Nous passons dans ce qui suit à la réalisation de notre système en représentant notre synoptique et en détaillant chaque élément de ce dernier ; des composants utilisés et des programmes implémentés pour la réalisation de l'interface graphique.

3.2 Synoptique

Ce synoptique représente une simulation de notre problématique on a représenté un seul poste qui peut être décuplé autant de fois que l'on veut suivant la taille du réseau. Il est représenté par trois parties principales (pc, carte d'acquisition, entrees/sorties) afin que l'on puisse contrôler toutes les sorties des LEDs via l'interface de contrôle illustrée sur PC, en utilisant la communication entre la carte d'acquisition `_ARDUINO_` et le PC.

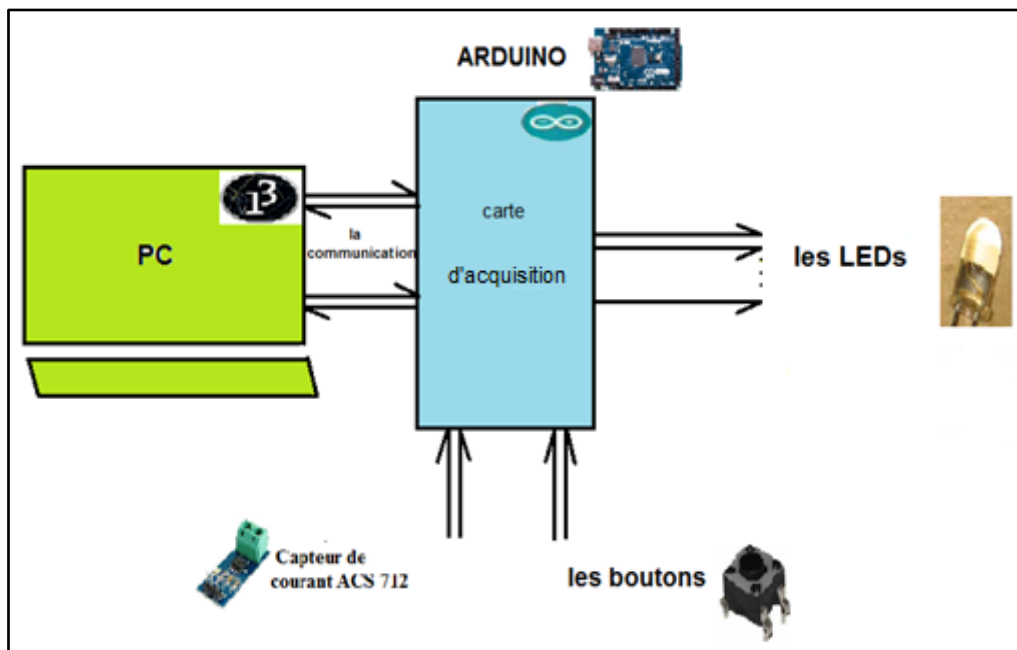


Figure 3.1 Synoptique d'un système

3.3 Matériels utilisés

Pour la réalisation de notre système nous avons utilisé les matériels suivants :

3.3.1 Bouton poussoir

Un bouton poussoir peut se trouver sous plusieurs formes dans le commerce. Ceux qui nous intéressent pour le moment c'est le bouton avec 4 pattes, une forme carrée, et un rond au centre qui est le bouton lui-même.



Figure 3.2 Exemple d'un bouton poussoir

Le principe de ce bouton est que lorsque l'on appuie, le courant passe, et lorsque l'on relâche et bien... le courant ne passe plus.

Le bouton poussoir sert de pont. S'il est levé, le courant ne passe pas, on dit que le circuit est ouvert. S'il est baissé et donc relie les deux contacts, le courant passe, on dit que le circuit est fermé.

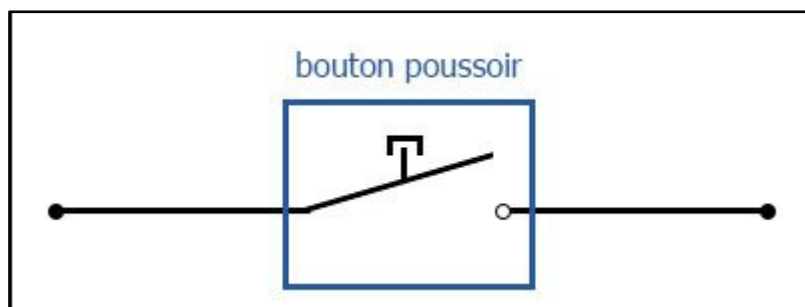


Figure 3.3 Schéma d'un bouton poussoir

3.3.2 LED

Tout d'abord LED c'est de l'anglais (Light Emitting Diode), en français on dit plutôt DEL (Diode à Émission de Lumière). Donc une LED est une diode qui produit de la lumière.

Une diode est un dipôle (c'est-à-dire un composant électrique qui se branche avec 2 bornes) qui ne laisse passer le courant que dans un seul sens. On appelle ça un semi-conducteur. La diode et la LED sont des semi-conducteurs.

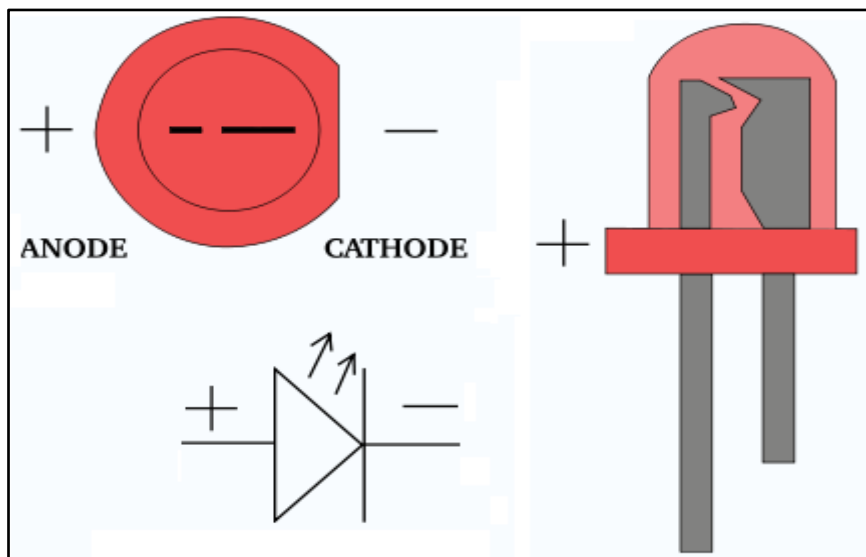


Figure 3.4 Représentation d'une LED

De plus elle ne doit pas être traversée par un courant trop fort, c'est pour cela qu'il est indispensable de brancher une résistance de 330 mΩ en série avec la LED. On notera qu'une résistance n'a pas de sens de branchement.

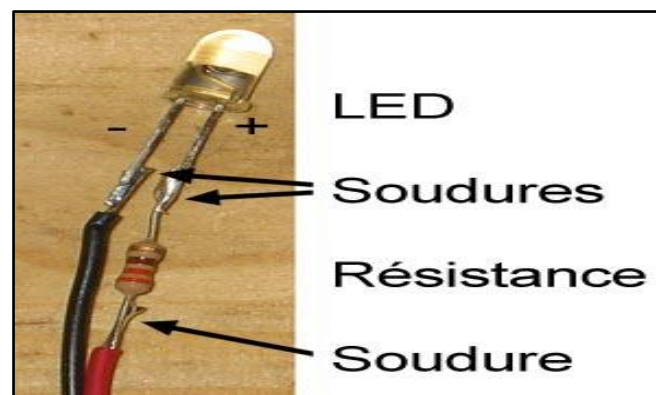


Figure 3.5 LED avec une résistance 330 mΩ soudé

3.3.3 Capteur de courant ACS 712

Acs712 est un capteur de courant à effet Hall. Il peut mesurer à la fois le courant continu et le courant alternatif. Il s'agit d'un capteur de type linéaire. Il s'agit d'un circuit intégré très célèbre conçu par Allegro. Il a des caractéristiques d'annulation du bruit, un temps de réponse très élevé. L'erreur de sortie est d'environ 1,5%, mais elle peut être traitée avec une programmation intelligente et en multipliant la valeur mesurée avec l'erreur standard du capteur. Si vous donnez un courant continu à son entrée, cela donnera une tension continue proportionnelle à la sortie du capteur et si vous fournissez un courant alternatif à l'entrée de acs712, cela vous donnera une tension alternative proportionnelle à la sortie. Le terme proportionnel dépend de la sensibilité de sortie du capteur.

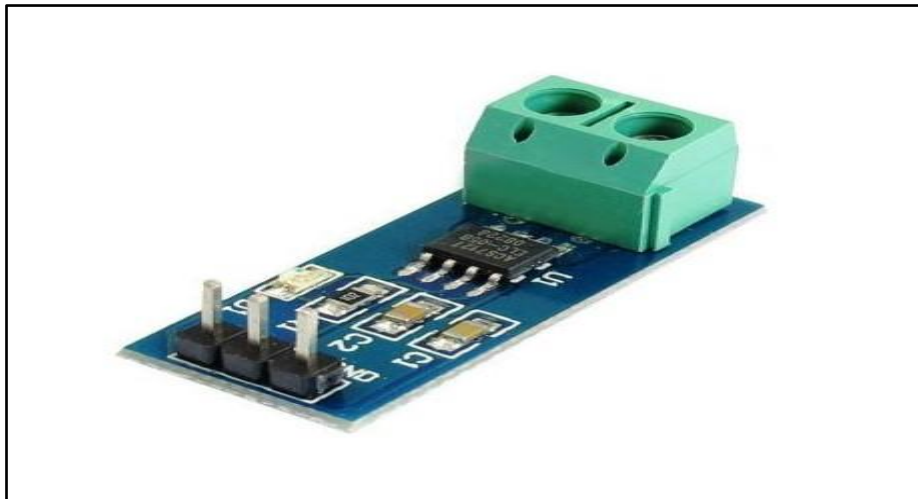


Figure 3.6 Capteur de courant acs 712 30A

a) Fonctionnement du capteur de courant acs712

Ce capteur acs712 se compose d'un circuit à effet Hall linéaire avec un chemin de conduction en cuivre. Le chemin de conduction du cuivre est situé autour de la surface de la filière. Lorsque du courant alternatif ou continu traverse un chemin de conduction en cuivre, il produit un champ magnétique. Ce champ électromagnétique interagit avec le capteur à effet Hall. Le circuit à effet Hall convertit ce champ électromagnétique déposé en tension proportionnelle DC ou AC (dans notre cas c'est le courant alternatif AC) selon le type de courant d'entrée. Cette tension de sortie est mesurée à l'aide d'arduino ou de tout

microcontrôleur. Après avoir mesuré cette tension, nous la reconvertissons en courant à l'aide d'équations de sensibilité.

b) Diagramme des broches du capteur de courant à effet Hall ACS712

La broche du capteur de courant acs712 est donnée ci-dessous. Les broches numéro 1, 2 et 3, 4 sont utilisées pour l'échantillonnage actuel. En d'autres termes. Vous connecterez ces broches en série avec la charge dont vous souhaitez mesurer le courant.

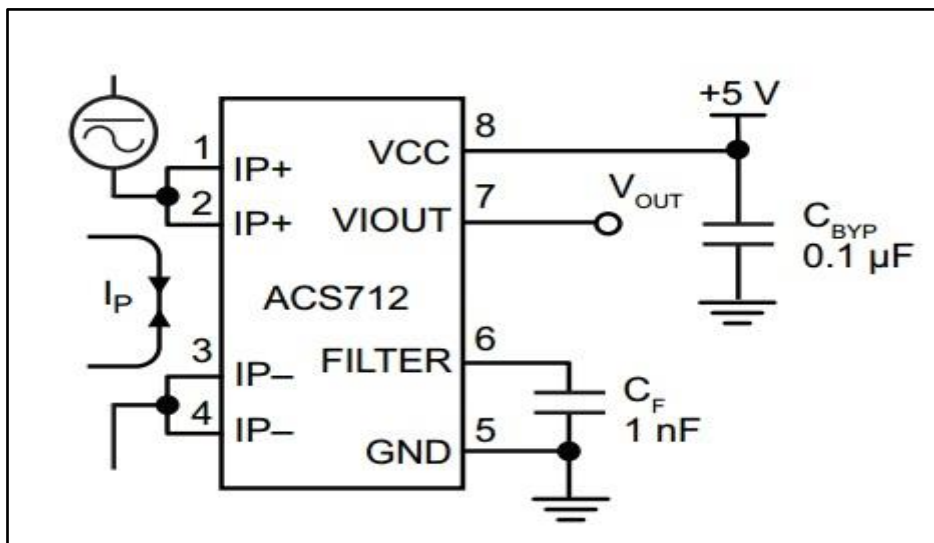


Figure 3.7 Diagramme des broches du capteur acs 712

Le numéro de broche est la connexion à la terre d'une alimentation 5 volts et le numéro de broche 6 est utilisé pour connecter le condensateur du filtre. Une borne du condensateur du filtre doit être connectée à la broche numéro 6 et l'autre borne doit être connectée à la masse. De même, la broche numéro 8 vcc est une broche d'alimentation et vous devez vous y connecter en 5 volts cc. La broche numéro 7 est la broche de sortie du capteur de courant acs712. À partir de la broche de sortie, nous mesurerons la tension à l'aide d'arduino et nous verrons plus tard comment le faire. La connexion charge en parallèle avec IP + et IP +, cela endommagera l'appareil (le capteur).

3.3.4 Plaque d'essai

Il s'agit d'une planche ou d'un morceau de plastique spécial utilisé pour installer et démonter facilement des composants électroniques sans avoir besoin de soudure, vous pouvez acheter n'importe quelle taille, de préférence une planche à demi-taille.

3.3.5 Multimètre

Mesure multi-usage et peut mesurer les résistances, la différence de tension et l'intensité du courant électrique par défaut.

3.3.6 Câble USB A male/B male

Ce câble fait la transmission des informations entre la carte ARDUINO et le pc, ainsi son alimentation.

3.3.8 Résistance

La résistance est un composant de base qui s'oppose au passage du courant, On s'en sert pour limiter des courants maximums mais aussi pour d'autres choses.

3.4 Fenêtres de programmation

On a 5 fenêtres de programmation :

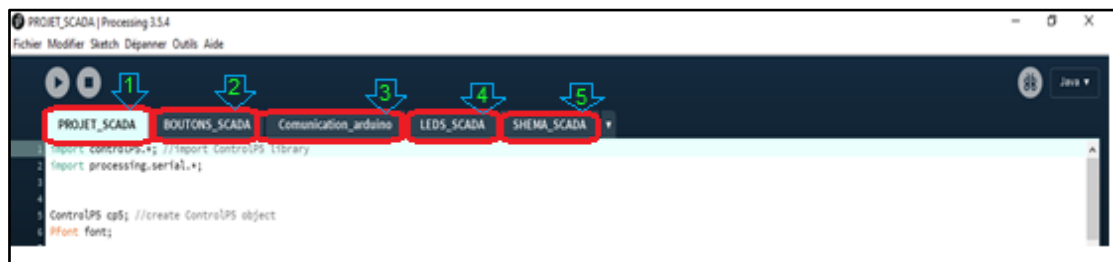


Figure 3.8 Fenêtres des programmes

1>> fenêtre de programme principale.

2>> fenêtre des boutons.

3>>fenêtre de la communication avec L'arduino.

4>> fenêtre des LEDs.

5>> fenêtre de schéma électrique.

3.5 Interface de la simulation (SCADA)

Après avoir appuyé sur ce bouton , cette interface apparaît

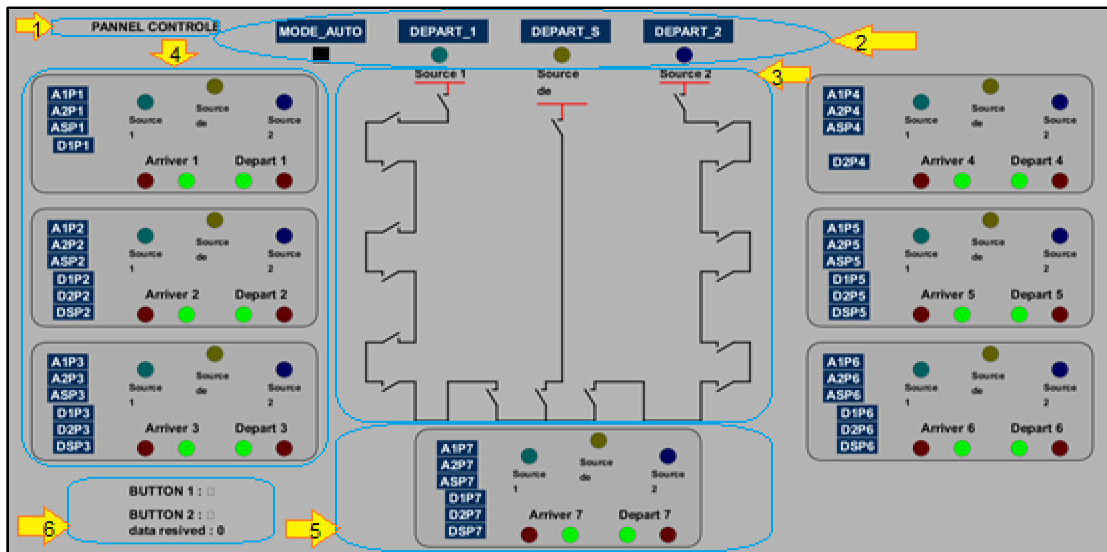


Figure 3.9 Interface graphique

1 >> le panel contrôle.

2>>les sources électriques<1, 2, source de secoure et bouton de mode automatique>.

3 >> le schéma électrique de réseau.

4 >>les postes électriques <1...6>.

5 >> poste électrique de couplage.

6 >> les boutons.

3.6 Poste électrique sur l'écran SCADA

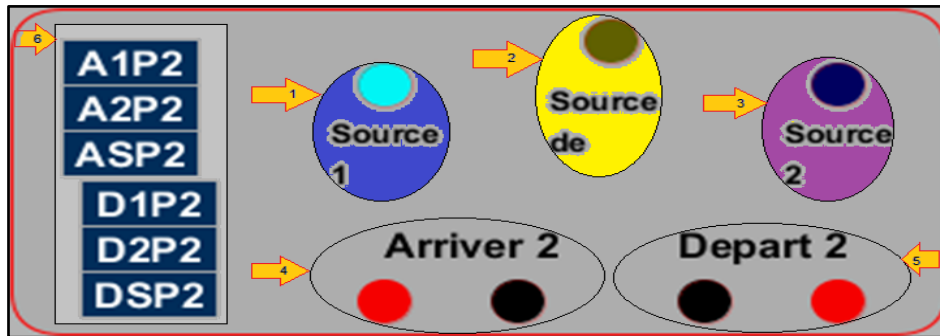


Figure 3.10 Interface graphique d'une poste électrique

1>> Source électriques 1

2>> Sources électrique de secoure.

3>> Source électriques 2.

4>> Arriver de courant de poste 2.

5>>Départ de courant de poste 2.

6>> Les Boutons : A1P2 : Arriver le courant de source 1 a Poste 2.

ASP2 : Arriver le courant de source Secoure a Poste 2.

D1P2 : Départ le courant de source 1 a Poste 2.

DSP2 : Départ le courant de source Secoure a Poste 2.

3.7 Modes de fonctionnement

On a deux modes de fonctionnement :

1. **Mode automatique** : Il fonctionne automatiquement et corrige tous les défauts _surcharge et Les coupures de courant électrique_ automatiquement sans intervention humaine.
2. **Mode manuel** : nous pouvons contrôler chaque partie du réseau électrique manuellement.

3.8 Communication PROCESSING_ARDUINO

La fonction pour lire les données de l'Arduino si il envoi quelque choses est :

```
void serialEvent(Serial myPort) {  
  // read a byte from the serial port:  
  inByte = myPort.read();  
  /* serialInArray[serialCount] = inByte;  
  serialCount++;*/  
}
```

Figure 3.11 fonction pour lire les données

La fonction qui envoie des commandes à l'Arduino est :

```
for(int i =0;i<52;i++){  
  word = word+storData[i];  
}  
myPort.write(word);
```

Figure 3.12 fonction qui envoie des commandes

3.9 Organigramme

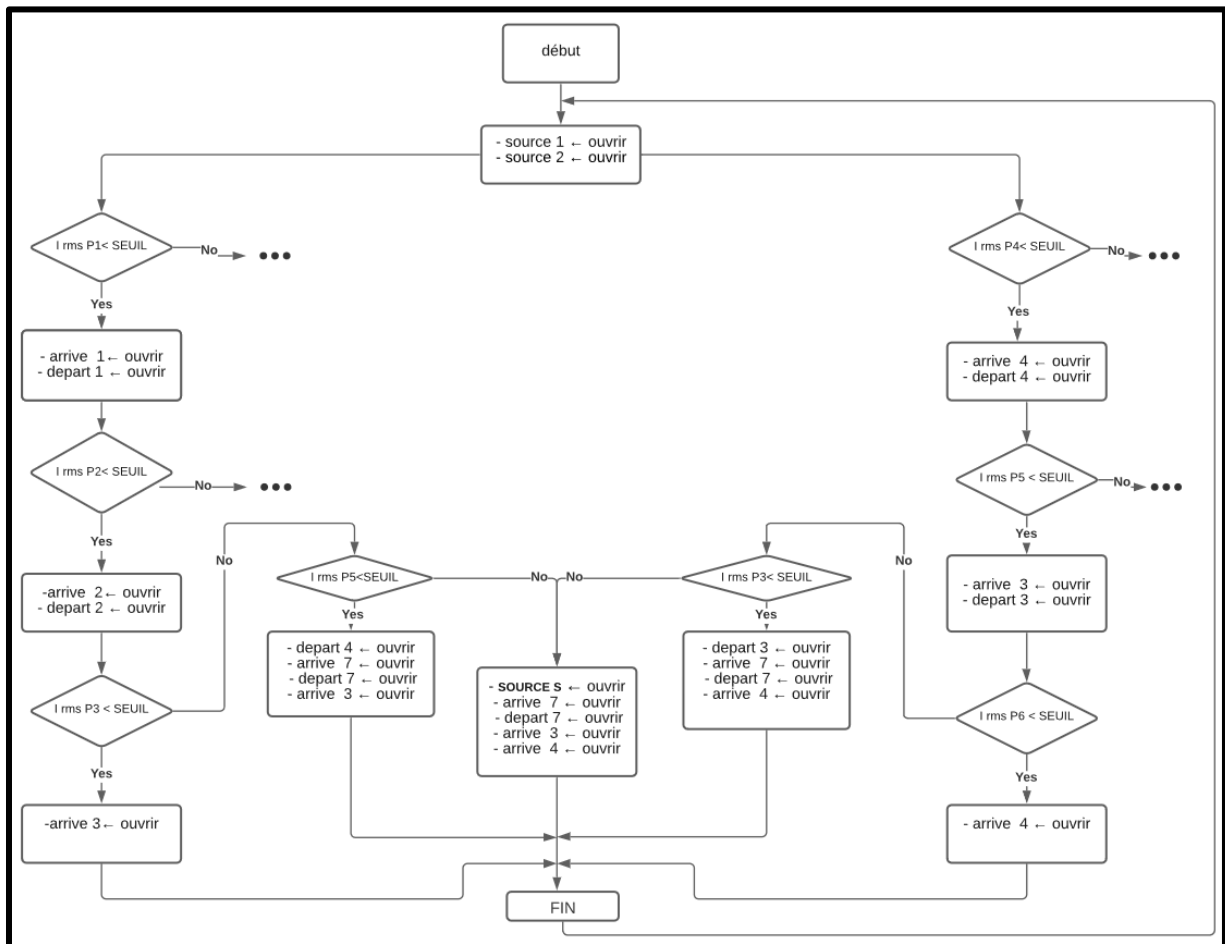


Figure 3.13 Organigramme de fonctionnement de l'interface

3.10 Conclusion

Après avoir positionné la problématique de notre projet de fin d'étude et détaillé les éléments nécessaires utiles pour notre objectif dans les chapitres précédents, nous avons abordé sereinement dans ce chapitre la réalisation.

Nous avons fait la réalisation et la simulation de l'interface graphique du système DMS, et traité la communication entre la carte ARDUINO et le logiciel PROCESSING.

Nous avons présenté la partie supervision qui est conçu sous le logiciel PROCESSING, ainsi que nous avons contrôlé le système DMS, et nous avons pu commander et supervisé les différents états des postes électriques à distance, pour intervenir à tout dysfonctionnements ou panne qui peut survenir.

Conclusion générale

Un bon réseau électrique doit être équipé d'un système de contrôle sélectif le plus large possible afin de s'assurer que ses paramètres sont suivis et extraits de la base de données pour un meilleur contrôle.

Pour cette raison, Sonelgaz a développé un système DMS, qui vise à contrôler automatiquement et à distance le réseau de distribution d'énergie moyenne et basse tension depuis le centre de contrôle. Afin de mieux comprendre le fonctionnement du système DMS. Nous avons fait un modèle qui est simulé à l'aide du programme Arduino.

Après l'étude du réseau électrique afin de déterminer les points du réseau pour un fonctionnement automatisé, nous avons créé une interface graphique du système DMS, gérons la communication entre la carte ARDUINO et le logiciel de traitement.

Nous avons introduit la partie de supervision conçue sous le programme de traitement PROCESSING afin de surveiller et contrôler automatiquement les composants du réseau ou les sous-stations en temps réel sans intervention manuelle. Cela permet de réduire la durée d'arrêt lors d'un accident, d'isoler le défaut, puis de réalimenter les pièces saines, en réduisant la charge si nécessaire, afin d'atteindre ces objectifs, l'interface montre l'état du réseau électrique en temps réel sous la forme des animations graphique en plus de ça des commentaires sont affichés lorsqu'un échec ou une panne survient pour faciliter la tâche sur le superviseur.

Référence bibliographie

[1] Raphael Caire, "Gestion de la production décentralisée dans les réseaux de distribution", Thèse de doctorat de l'Institut National Polytechnique de Grenoble, Décembre 2004.

[2] Wendy Carolina Briceno Vicente, "Modélisation des réseaux de distribution sous incertitudes", Thèse de doctorat de l'Institut National Polytechnique de Grenoble, septembre 2012.

[3] Marie-Cécile Alvarez-Hérault, " Architectures des réseaux de distribution du futur en présence de production décentralisée", Thèse de doctorat de l'Institut National Polytechnique de Grenoble, Décembre 2009.

[4] Karim TAREB, " REGLAGE DES PROTECTIONS DES RESEAUX HTA", Thèse de Magister de Université mouloud mammeri de tizi-ouzou 25 SEPTEMBRE 2017.

[5] P. Carrive, « Techniques de l'ingénieur N° D4 210 : Réseaux de distribution Structure et planification », décembre 1991.

[6] DJAFRI Menad, " Etude et Réalisation d'une Carte Arduino", Thèse de Master de Université A. MIRA DE BEJAIA 20 Juin 2016.

[7] <https://www.scribd.com/document/413880752/a-pdf> 22/06/2021.

[8] E. Gain, « Techniques de l'ingénieur N° D 4220 : Réseau de distribution Conception et dimensionnement », 1993.

[9] A. Charoy, « Compatibilité électromagnétique », Livre, Edition Dunod, 2005

- [10] D. Fulchiron, « Cahier Technique Schneider Electric N°203 : Choix de base des réseaux MT de distribution publique », mars 2001.
- [11] H. Persoz, « Histoire générale de l'électricité en France », Volume 3 : 1947-1987 », Tome 3, février 1984.
- [12] C. Puret, « Cahier Technique Schneider Electric N°155 : Les réseaux de distribution publique MT dans le monde », septembre 2001.
- [13] Livre ARDUINO, publié en ligne en 2011, écrit par : téléchargé depuis le lien www.flossmanualsfr.net/_booki/arduino/arduino.pdf, Avril 2016 .
- [14] Accident de train aux Pays-Bas *Le Monde*, 23 février 2016.
- [15] Lionel Tardy processing dans l'enseignement
<https://fr.flossmanuals.net/processing/linstallation-de-processing/> 06/07/2021.
- [16] <https://processing.org/> 24/06/2021.
- [17] <https://www.emploitic.com/entreprise/sadeg-societe-dpe-distribution-de-l-electricite-et-de-gaz/presentation/> ,21/06/2020.
- [18]<https://www.reporters.dz/travail-durant-le-confinement-sonelgaz-attribue-une-prime-exceptionnelle-de-risque-a-ses-employes/> 05/07/2021.
- [19] <https://arduino.blaisepascal.fr/bibliotheque-serial/> 05/07/2021.
- [20] https://arduino.blaisepascal.fr/wp-content/uploads/Thumb/Thumb_Serial-300x142.jpg/
04/07/2021.
- [21] <https://www.sonelgaz.dz/fr/category/qui-sommes-nous> 10/07/2021.
- [22] <https://www.creg.dz/index.php/presentation/organisation> 12/07/2021.