

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 البلدية  
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا  
Faculté de Technologie

قسم الآلية والكهروتقني  
Département d'Automatique et Electrotechnique



## Mémoire de Master

Mention : automatique

Spécialité : automatique et systèmes

présenté par :

Yacine Benaouda

# *Création d'une IHM pour la supervision et la commande d'un coffret de control d'un IAT en protocole de communication IEC 60870-5-101*

Proposé par : Mme Salah Samia & Mr. RAHMANI Maamar

# *Remerciement*

*Tout d'abord,*

*Je tiens à exprimer notre profonde gratitude Mr. RAHMANI Maamar, mon Co-promoteur qui a dirigé mon stage Pour sa grande contribution et encore plus de patience lors de mon stage chez Sonelgaz. Je n'aurais pas pu faire ça sans lui.*

*Je tiens également à remercier ma promotrice Mme Salah Samia pour sa contribution et sa direction très utiles lors de la réalisation de ce mémoire*

*Nous remercions aussi les enseignants qui nous ont suivis de notre cursus universitaire et particulièrement nous remercions aussi L'ensemble des membres du jury.*

*Et enfin, Je tiens à remercier ma famille et mes amis pour leur soutien. Surtout mon père ma mère et ma grand-mère.*

تستخدم شركات توزيع الكهرباء معدات تحكم عن بعد تتوافق مع بروتوكولات اتصالات معينة. تم تنفيذ هذا العمل لإنشاء واجهة بين الإنسان والآلة لاستخدامها في الإشراف على هذه المعدات ومراقبتها بشكل صحيح دون الحاجة إلى غرفة التحكم المركزية في نظام سكاذا.

كلمات البحث: معدات التحكم عن بعد. سكاذا. واجهة الإنسان والآلة ؛ بروتوكولات الاتصال

---

**Résumé :**

Les entreprises de distribution d'électricité utilisent des équipements de télé conduite conformes à certains protocoles de communication. Ce travail a été fait pour créer une interface homme-machine à utiliser pour superviser et surveiller correctement ces équipements sans avoir besoin de la salle de contrôle du système SCADA central.

**Mots clés :** Équipement de télé conduite ; SCADA ; Interface Homme-machine ; Protocole de communication.

---

**Abstract :**

Electricity distribution companies use telecontrol equipment conforming to certain communications protocols. This work was done to create a human machine interface to be used to properly supervise and monitor these equipments without the need of the central SCADA system control room.

**Keywords :** Telecontrol equipment ; SCADA ; Human-machine interface ; Communication protocols.

---

# Listes des acronymes et abréviations:

**IHM:** Interface Homme-Machine.

**MT:** Moyen Tension.

**BT:** Basse Tension.

**IAT:** Interrupteur Aériens Télécommandé.

**ASDU:** Application Service Data Unit (Unité de données de service d'application).

**CASDU:** Common adress of ASDU (Adresse commune de l'ASDU).

**IOA:** Information Object Address (Adresse de l'objet d'information).

**SONELGAZ :** La Société Algérienne de Distribution de l'électricité et du gaz.

**SCADA :** Supervisory Control And Data Acquisition .

**IEC :** International Electronic Commission

**TCD:** Télé commande Double.

**TSD :** Télésignalisations Doubles .

**TSS :** Télésignalisation Simple.

**Trame :** Suite de bits ou caractères émis de façon ininterrompue par un équipement sur le réseau et dont l'ensemble constitue une information cohérente interprétable par le ou les destinataires (messages, questions ou réponses, diffusion d'une valeur...).

La longueur des trames est toujours limitée.

**Protocole de communication :** Désigne les règles de dialogue entre mêmes couches des entités communicantes.

# Liste des matières

Introduction général	3
Chapitre 1 Généralités sur le travail et les éléments utilisé	5
1.1 Introduction	6
1.2 Présentation de l'entreprise	6
1.2.1 L'Historique	6
1.2.2 Filiales de Sonalgaz	8
1.3 Le système Micro SCADA (Blida)	9
1.4 Le protocole IEC 60870-5-101	11
1.4.1 Les télécommandes et télésignalisations	12
a *Télécommande double (TCD)	12
b *Télésignalisations doubles (TSD)	12
c *Télésignalisation simple (TSS)	12
d *Télémesures	13
1.5 Poste MT/BT	13
1.6 Interrupteur aérien (IAT)	14
1.6.1 Le coffret de commande	14
1.7 IHM	17
1.8 Conclusion	18
Chapitre 2 Le Protocol IEC 60870-5-10120	
2.1 Introduction	21
2.2 Définitions	22
2.2.1 C'est quoi un Protocol de Communication?	22
2.2.2 EPA	23

a	Couche Physique	23
b	Couche de liaison	24
c	Couche d'application	24
2.2.3	ASDU	24
2.2.4	Mode Symétrique et Asymétrique	25
a	Mode Asymétrique	25
b	Mode Symétrique	25
2.2.5	Polling	26
2.2.6	Direction de contrôle et direction de surveillance	27
2.2.7	Adressage	28
a	Adresse de liaison	28
b	Adresse CASDU	28
c	Adresse d'objet d'information (IOA)	28
2.3	Les trames de données	29
2.3.1	Structures des trames	29
a	Les trames fixe	31
b	Les trames variable	35
c	Caractères uniques	38
2.4	Les différentes ASDU Utilisés et leurs structures:	38
2.4.1	Télésignalisation simple (TSS)	39
a	TYPE IDENT 1 : Informations ponctuelles sans horodatage	39
b	TYPE IDENT 2 : Informations ponctuelles avec horodatage CP24	41
c	TYPE IDENT 30 : Informations ponctuelles avec horodatage CP5642	
2.4.2	Télésignalisation Double (TSD)	42
2.4.3	Télémesures	43
a	TYPE IDENT 13: Valeur mesurée, nombre flottant court	44
2.4.4	Télécommande double (TCD)	44

2.5	Conclusion	47
Chapitre 3	IHM du Coffret de commande:	48
3.1	Introduction	49
3.2	Software utilisés	49
3.2.1	Visual Studios 2019 (Community Edition):	49
3.2.2	Langage de programmation C#	51
3.2.3	Création d'un projet et configuration d'une station de travail	51
a	Création d'un nouveau projet.	52
3.3	Hardware Utilisé	56
3.3.1	Coffret de commande de poste MT/BT	57
3.3.2	PC portable	59
3.3.3	Fil RS232	59
3.4	Principe de fonctionnement	60
3.4.1	Codage du le processus	61
a	-Initialisation de la station:	61
b	- Polling:	62
c	- Exécuter des commandes et recevoir des informations	63
3.5	Conclusion	65
Chapitre 4	Présentation de l'IHM créé Réalisation de la supervision de la station	66
4.1	Introduction	67
4.2	À la découverte de l'IHM	68
4.2.1	La boîte noire supérieure	70
4.2.2	Les paramètres	71
4.2.3	La partie de l'interrupteur	72
4.2.4	ADA (automate de défaut)	73
4.2.5	Les défauts	74
4.2.6	Les mesures	75

a	-Tension Réseau :	75	
b	-Tension Coffret :	75	
c	-Courant Voie A	75	
d	-Nmbr Manœuvre voie A	75	
4.2.7	Le reste des boutons	76	
a	-Connect	76	
b	-Disconnect	76	
c	-Gen Intero	76	
4.2.8	La boîte noire inférieure	77	
4.3	Conclusion	77	
	Conclusion générale	78	
	Bibliographie	79	



# Liste des Figures

Figure 1-1 Direction générale « SONELGAZ ».	5
Figure 1-2 L'organigramme du groupe « SONELGAZ ».	6
Figure 1-3 Architecture MICRO SCADA Blida.	8
Figure 1-4 Deux type de poste MT/BT	11
Figure 1-5 Installation d'un Interrupteur aérien (IAT)	12
Figure 1-6 Interrupteur aérien IAT Avec coffret installé	13
Figure 1-7 photo de le coffret que nous avons utilisée pour notre travail	14
Figure 1-8 Les éléments dans l'intérieur de coffret	15
Figure 1-9 Un IHM utilisé dans le centre de commande SCADA Sonelgaz	16
Figure 2-1 les 3 couches dans EPA	20
Figure 2-2 Liaisons multipoint	22
Figure 2-3 Liaisons point-a-point	23
Figure 2-4 Exemple de configuration multipoint de stations esclaves et maîtres	24
Figure 2-5 CASDU de taille 2 octets	25
Figure 2-6 IOA de taille 3 octets	25
Figure 2-7 Séquence binaire d'une trame IEC 60870-5-101	27
Figure 2-8 La format des trames FT1.2	28
Figure 2-9 La structure Spécifique d'une trame fixe	29
Figure 2-10 Le champ de contrôle et ses différents bits et codes de fonction possibles	31
Figure 2-11 Exemple d'une trame fixe avec explications Les trames variable:	32
Figure 2-12 Structure général d'une trame variable	33
Figure 2-13 Exemple d'un trame variable avec explication	34
Figure 2-14 Liste montrant une grande variété d'ASDU	35
Figure 2-15 Structures Général d'ASDU TI 1 partie 1	36
Figure 2-16 Structures Général d'ASDU TI 1 partie 2	37
Figure 2-17 Structure général d'ASDU TI 2	38
Figure 2-18 Structure général d'ASDU 30	39
Figure 2-19 Interprétation des le s 2 bits DIP	40
Figure 2-20 Structure général d'ASDU 13	41
Figure 2-21 Structure général d'ASDU 46	42

Figure 2-22 Sémantiques de l'octet de commande	43
Figure 3-1 Capture d'écran de l'IDE de Visual Studios, qui comprend des légendes indiquant où se trouvent les fonctionnalités clés	47
Figure 3-2 Illustration montrant comment IntelliSense affiche une liste de membres pour un type	47
Figure 3-3 Fenêtre de démarrage de Visual Studio	49
Figure 3-4 Fenêtre de création de projet	50
Figure 3-5 Fenêtre de configuration du projet	51
Figure 3-6 Projet Winforms	52
Figure 3-7 Project winforms coté Code	53
Figure 3-8 Coffret de commande	54
Figure 3-9 Le panneau avant de Coffret de commande	55
Figure 3-10 Fil RS232	56
Figure 3-11 Schéma de fonctionnement	57
Figure 3-12 Interrupteur aérien IAT Avec coffret installé	58
Figure 3-13 Schéma d'initialisation de la station	59
Figure 3-14 Schéma de polling	60
Figure 3-15 Schéma de la structure d'une commande	61
Figure 4-1 Le poste de travail pour la durée de ce projet	66
Figure 4-2 Le code source de l'IHM	64
Figure 4-3 L'IHM créé	65
Figure 4-4 La boîte noire supérieure	66
Figure 4-5 Les paramètres	67
Figure 4-6 La partie interrupteur	68
Figure 4-7 Les différentes représentations de l'interrupteur	69
Figure 4-8 La partie ADA	69
Figure 4-9 Les défauts	70
Figure 4-10 Les mesures	71
Figure 4-11 Le reste des boutons	72
Figure 4-12 La boîte noire inférieure	73

## Introduction général

L'énergie électrique est de nos jours, un élément incontournable dans la vie quotidienne de pratiquement tous les habitants de la planète.

La consommation de cette l'énergie est assurée par les points de production, les points de transport et de distribution. Elle est acheminée aux points de consommation quasi exclusivement par des réseaux électriques.

En Algérie, Le fournisseur national d'électricité n'est autre que Sonelgaz, qui profite d'équipements de télé conduite pour superviser et contrôler son réseau électrique. Constituant ce que l'on appelle un système SCADA (Supervisory Control And Data Acquisition).

Ces équipements de télé contrôle utilisés dans le système SCADA, se conforment à certains protocoles de communication afin d'obtenir une communication efficace, sécurisée et précise. L'IEC 60870-5-101 est l'un de ces protocoles.

Un protocole de communication est un système de règles qui permet à deux entités ou plus d'un système de communication et de transmettre des informations via tout type de variation d'une quantité physique. Le protocole définit les règles, la syntaxe, la sémantique et la synchronisation de la communication et les éventuelles méthodes de récupération d'erreur. Les protocoles peuvent être mis en œuvre par du matériel, des logiciels ou une combinaison des deux.

En raison de l'importance des équipement de télé contrôle dans un système SCADA du réseau électrique, il existe un besoin d' une application utilisateur capable de contrôler et de surveiller l'équipement à l'aide du protocole de communication IEC 60870-5-101. Une interface qui permet à un utilisateur formé de superviser et de surveiller librement et facilement certains équipements dans certaines centrales. Ou comme il est plus communément connu dans l'industrie, une interface homme-machine ou IHM.

# Chapitre 1 Généralités sur le travail et les éléments utilisé

---

En effet, dans ce travail nous nous intéressons principalement à la conception et le développement d'une application IHM capable de superviser et de contrôler un coffret de commande de marque ENSTO utilisé dans les SIA des stations MT/BT à de nombreux endroits du réseau électrique de Sonelgaz.

Le présent mémoire est structuré en quatre chapitres :

\***Le premier chapitre** sert à donner des informations générales sur le travail et les éléments utilisés .

\* **Le deuxième chapitre** est consacré à l'exploration et à la présentation du protocole de communication IEC 60780-5-101.

\* **Le troisième chapitre** porte sur le processus de création de l'IHM et présente les outils utilisés pour sa création.

\* **Le quatrième chapitre** traite de la présentation de l'IHM créé, de l'exploration et de l'explication de ses différentes parties, de leurs utilisations et de leur signification.

À la fin, nous terminerons par une conclusion générale qui discutera de ce que nous avons créé, de ses avantages et de ses éventuelles implémentations futures.

**Chapitre 1 Généralités sur le travail et les  
éléments utilisé**

## 1.1 Introduction

L'énergie électrique est un élément fondamental dans la société moderne. En effet la supervision et l'entretien du réseau électrique sont devenus une responsabilité très importante et cruciale. En Algérie, cette responsabilité incombe à la société Sonelgaz.

La capacité de couper certaines zones du réseau électrique dans les zones résidentielles, se fait généralement avec l'utilisation d'un IAT à un poste MT/BT. D'où l'intérêt des dispositifs de télésurveillance et de télécommande qui permettent de contrôler l'état des réseaux et d'agir avec rapidité en évitant autant que possible les déplacements coûteux en temps d'intervention. Pour cela, sonelgaz nous a proposé ce stage qui a pour objectif de créer une interface homme machine.

Dans ce chapitre, nous visons à présenter l'environnement dans lequel nous avons travaillé au cours de ce mémoire ainsi les matériaux et les concepts utilisés tels que le coffret de commande et le protocole de communication IEC 60870-5-101.

## 1.2 Présentation de l'entreprise

### 1.2.1 L'Historique

**De EGA à SONELGAZ d'aujourd'hui : Plus d'un demi siècle d'existence...** En 1947 est créé l'établissement public « électricité et gaz d'Algérie » par abréviation EGA, auquel est confié le monopole de la production, du transport et de la distribution de l'électricité et du gaz.

EGA regroupe les anciennes entreprises de production et de distribution, de statut privé, notamment Lebon et Cie et SAE (Société Algérienne de l'électricité et du gaz), tombant sous le coup de la loi de nationalisation de 1946, promulguée par l'état français.

## Chapitre 1 Généralités sur le travail et les éléments utilisé

---

**Un défi relevé en 1962 déjà...** EGA est pris en charge par l'état algérien indépendant ; en quelques années grâce à un formidable effort de formation, l'encadrement et le personnel algérien assurent effectivement le fonctionnement de l'établissement.

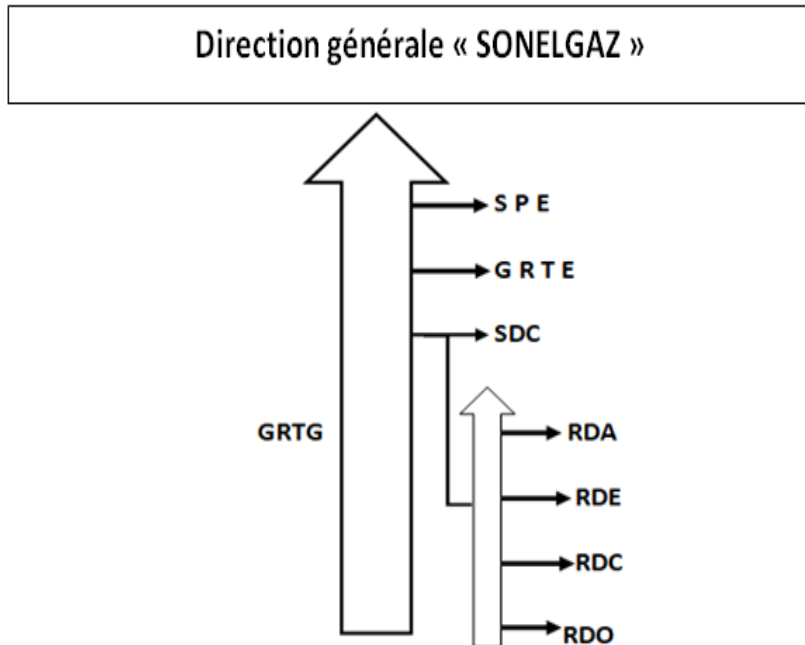
**Soutenir le développement économique et social,** En **1969**, EGA devient SONELGAZ, devient société nationale de l'électricité et du gaz ; à ce moment aujourd'hui l'entreprise est de taille importante dont le personnel est de quelques 6000 agents.

La transformation de la société avait pour objectif de conférer à l'entreprise les capacités organisationnelles et gestionnaires pour accompagner et soutenir le développement économique du pays. Il s'agit notamment du développement industriel, et de l'accès du plus grand nombre à l'énergie électrique. (L'électrification rurale), projet inscrits dans le plan de développement élaboré par les autorités publiques.



**Figure 1-1** Direction générale « SONELGAZ ».

## 1.2.2 Filiales de Sonalgaz



**Figure 1-2 L'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).



### 1.3 Le système Micro SCADA (Blida)

Le système micro SCADA est un système de télé conduite décentralise des réseaux électriques, qui permet de réduire les temps d'interruption et augmenter la qualité de l'énergie.

Pour cela il y a un matériel et toute une installation avec un mécanisme nécessaire qui permet de télécommander à distance les interrupteurs de coupure du réseau, et cela à partir d'un pc de commande. Sans avoir à se déplacer, l'exploitant peut en permanence contrôler et intervenir sur le fonctionnement de son réseau suite à un défaut il est possible de changer rapidement le schéma d'exploitation du réseau afin de rendre minimale la partie de réseau non alimentée, et cela en consultant à distance les indicateurs de localisation de défaut.

Il gère tous les ouvrages importants de la zone d'exploitation où il est implanté :

- Les postes HTB/HTA
- Les postes HTA/BT
- Les postes HTA/HTA
- Les interrupteurs aériens télécommandent (IAT)

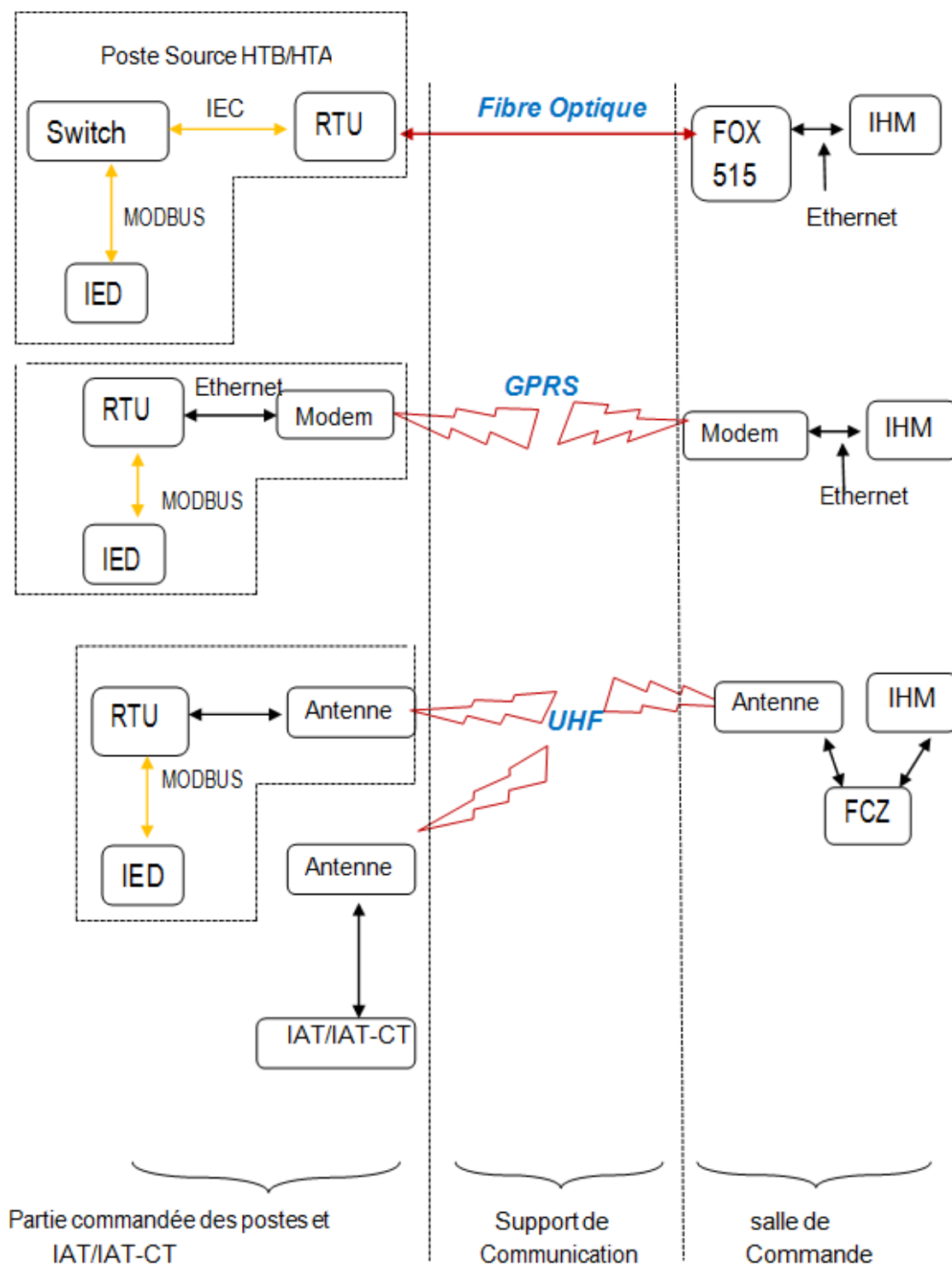


Figure 1-3 Architecture MICRO SCADA Blida.

## 1.4 Le protocole IEC 60870-5-101

[2]

C'est la norme internationale spécifiée par la Commission Electrotechnique Internationale. Le protocole IEC 101 spécifie le codage des données et les règles d'échange de ces données entre deux équipements.

Il utilise une interface de canal de contrôle à distance série asynchrone standard entre DTE et DCE. Le standard est adapté à de multiples configurations, telles que point à point, étoile, et point à multipoint,.... etc.

Le protocole IEC101 est basé sur le modèle de référence à 3 couches qui est une version simplifiée du modèle ISO à 7 couches.

Les 3 couches utilisées sont les couches : Physique, Liaison. Application.

Ce protocole permet de fonctionner selon 2 modes de transmission :

Les échanges peuvent être de **type asymétrique** (mode maître - esclave) ou de **type symétrique** (mode maître – maître).

Dans le mode asymétrique, le Superviseur est le maître et les équipements à commander, entant qu'esclave, se limite à répondre aux demandes du maître. Dans le mode symétrique, chaque équipement peut initier un dialogue.

Le protocole IEC-101 spécifie les données qui peuvent être échangées et la forme sous laquelle elles sont transmises. Parmi les nombreuses informations auxquelles le protocole donne accès, on trouve:

-Des signalisations (simples ou doubles).

-Des mesures (suivant plusieurs formats).

-Des compteurs.

## 1.4.1 Les télécommandes et télésignalisations

Le système gère les données suivantes:

**a** \**Télécommande double (TCD)*

-commande d'ouverture et de fermeture les interrupteurs aériens (IAT/IAT-CT) et les disjoncteurs.

**b** \**Télésignalisations doubles (TSD)*

-Etats des disjoncteurs et interrupteurs (Ouverture / Fermeture)

**c** \**Télésignalisation simple (TSS)*

-Signalisations automatismes : en service, hors service, automatisme activé

-Reset des détecteurs de défaut

-Signalisations complémentaires disponibles (ouverture porte, ...)

-Manque alimentation AC

-Manque alimentation AC temporisée (2 ou 4heures)

-Défaut chargeur

-Défaut batterie

-Défaut alimentation motorisation 48 Vcc

-Défaut commande

-Position interrupteur incohérente

-Mode de d'exploitation Local/distant

-Signalisation des détecteurs de courants de défaut : défaut phase-phase et phase-terre

-position sectionneur de terre des interrupteurs

-Etat de communication : date de la dernière connexion

## *d* \*Télémesures

Parmi les télémesures, on a:

- Intensité en tête des départs et au niveau des arrivées 30 et 10 KV
- Puissance active
- fréquence (linéaire)
- Mesure du courant ou de la tension

## 1.5 Poste MT/BT

[1]



**Figure 1-4 Deux type de poste MT/BT**

[1]Les réseaux de distribution basse tension sont alimentés à partir des postes de transformation MT/BT qui représentent le dernier échelon de transformation ramenant à la basse tension et qui assurent la protection et la surveillance du réseau. Il existe trois types des postes MT/BT: le poste sur poteau, le poste sous enveloppe préfabriqué et le poste maçonné, Nous ne nous intéressons qu'à la première puisqu'elle se rapporte à notre travail. Dans ces postes, Sonelgaz utilise généralement soit des IACM (interrupteur aérien mécanique) ou des IAT (interrupteur aérien Télécommander). En raison de leur supériorité, Les IAT coûtent plus cher donc il sont pas utilisé partout dans les réseaux. La plupart des IAT ou IACM du réseau de distribution électrique de Sonelgaz peuvent être manipulés sur 25 Ampères de courant.

## 1.6 Interrupteur aérien (IAT)

[1]

C'est des interrupteurs tri phases installés dans les réseaux aériens de distribution moyen tension, en zone rurale ou semi urbain, ils construits autour d'une coupure en ambiance SF6 sous enveloppe métallique pouvant couper 400 ou 630 A, sous tension jusqu'à 38Kv, et qui est associe avec des coffrets de commande et une antenne.

Installation

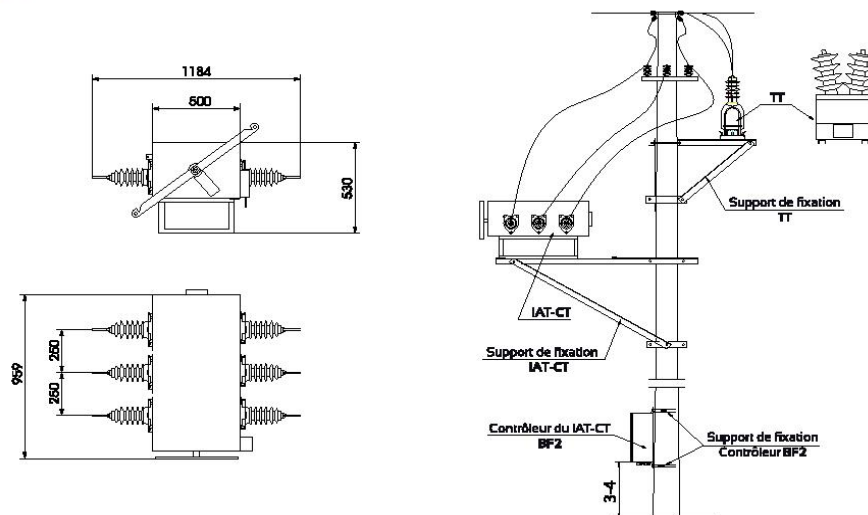


Figure 1-5 Installation d'un Interrupteur aérien (IAT)

### 1.6.1 Le coffret de commande



**Figure 1-6. Interrupteur aérien IAT Avec coffret installé**

Cet élément sert de poste extérieur dans un système SCADA, il reçoit les commandes d'une unité de commande et renvoie les mesures. Ce boîtier est relié à un relais qui éteint et allume le disjoncteur d'un poste MT/BT. Le Coffret peut être télécommandé avec une liaison à la forme d'une antenne (fig 1.6 ) ou par une connexion série comme RS-232.

Le coffret de commande a un microprocesseur pour l'automatisation de la ligne. Ce dernier est conçu pour un montage sûr Poteau extérieur. Il est normalement monté à un niveau bas sur le poteau pour faciliter l'accès. Le coffret de commande est muni d'un relais pour la détection des défauts de phase et de terre provenant des éléments de détection de courant TC, et la détection de la chute de tension par les éléments de détection de tension. Les détails du journal des défauts et des événements peuvent être stockés et transmis au SCADA via un modem GSM/GPRS. Celui que nous utilisons est fabriqué par l'ENSTO et s'appelle ITI 2012-fig1.7 le voici ci-dessous:



Figure 1-7 Le coffret que nous avons utilisée pour notre travail



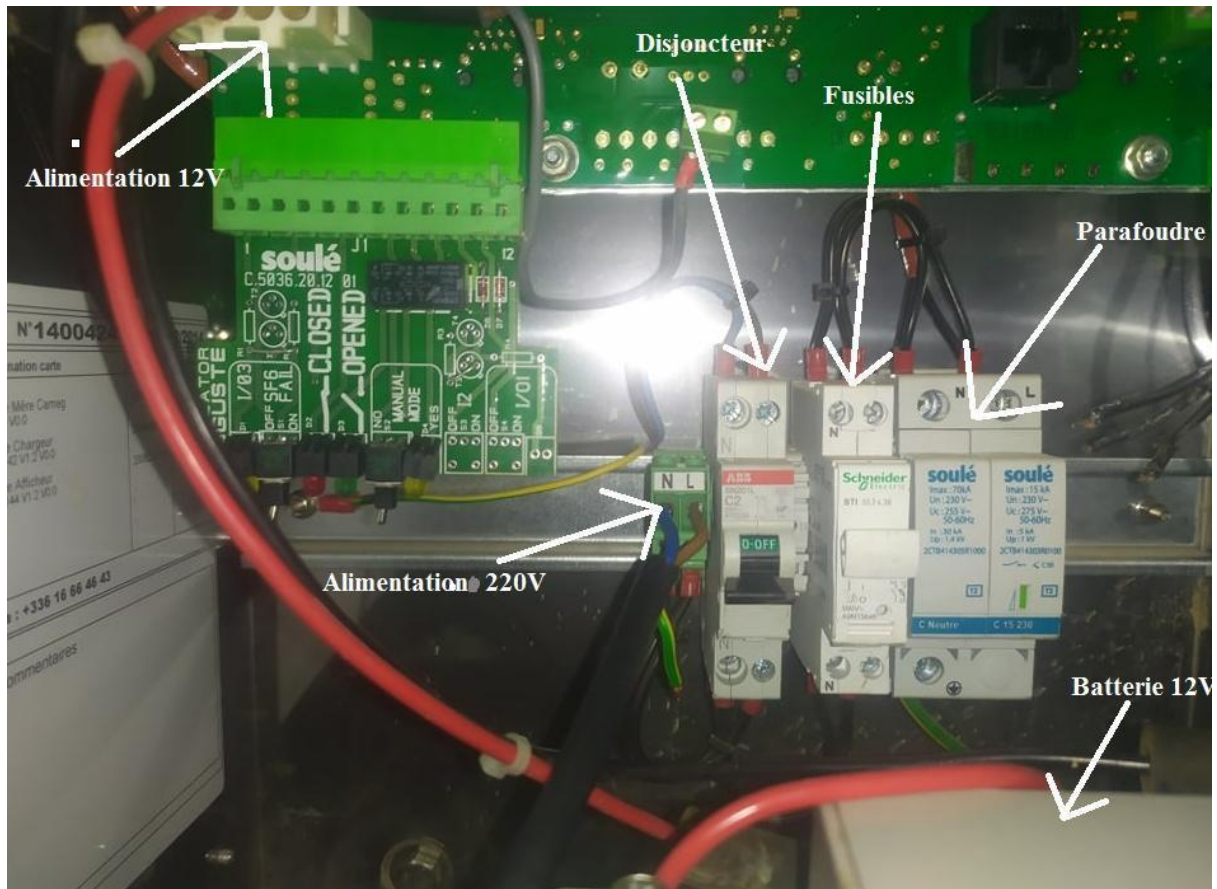


Figure 1-8 Les éléments du coffret

Dans l'installation réelle, l'alimentation 220V vient d'un transformateur relié aux réseaux MT.

le coffret peut commandé deux choses :

- 1- L'interrupteur du poste MT/BT qui se connecte ou se déconnecte essentiellement avec le réseau de distribution.
- 2- l'ADA qui est un système de détection automatique des pannes pour aider à la protection.

### 1.7 IHM

Une IHM est toute sorte d'interface qui permet à son utilisateur d'observer les caractéristiques importantes affichées sur la machine ou le processus auquel l'interface est connectée tout en

# Chapitre 1 Généralités sur le travail et les éléments utilisé

donnant également à l'utilisateur le contrôle (avec les autorisations appropriées si nécessaire) sur ladite machine ou ce processus.

Les IHM sont très utiles pour un contrôle et une surveillance rapides et faciles des équipements et sont essentiels dans un système SCADA comme celui utilisé par Sonelgaz montré dans la figure 1-9 .

[1]

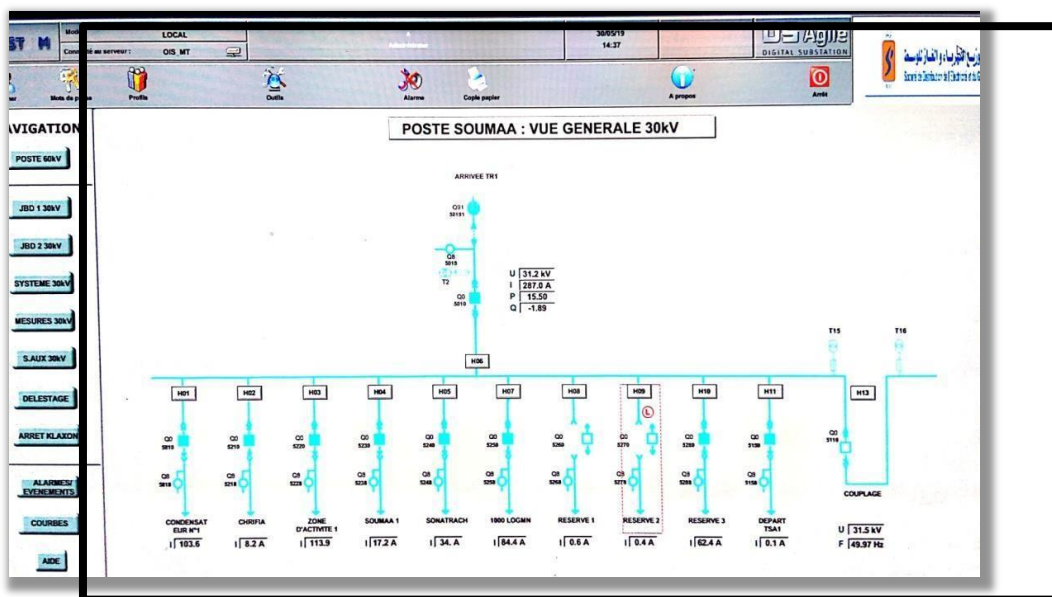


Figure 1-9 Un IHM utilisé dans le centre de commande SCADA Sonelgaz

Les IHM se présentent sous de nombreuses formes et tailles différentes avec différents degrés de fonctionnalité en fonction de la machine ou du processus auquel elle est connectée et du fabricant de l'IHM lui-même.

Cette interface apparemment simple permet à un utilisateur autorisé de contrôler à distance tous les IAT connectés au système SCADA, augmentant ainsi l'efficacité et la simplicité.

## 1.8 Conclusion

Dans ce chapitre, nous avons donné un aperçu général comme suit : l'historique de l'entreprise SONALGAZ , ses multiples activités et son organigramme, après on a passé à l'identification des différents éléments et équipement de la station caractéristiques, on a

# Chapitre 1 Généralités sur le travail et les éléments utilisé

---

introduire la société dans un peu de détail et aussi introduit les élément important dans notre travail.

Dans le chapitre suivant, nous allons bien approfondir dans le Protocol de communication IEC 60870-5-101.

**Chapitre 2 Le Protocol IEC 60870-5-101**

## 2.1 Introduction

Ce Protocole de communication a été créé par la CEI( IEC en anglais) en 1995, la Commission électrotechnique internationale, plus précisément par le comité numéro 57, à Genève en Suisse. Il a été créé en tant que norme complémentaire à la partie 5 pour les tâches de télé contrôle de base. Des parties importantes de la partie originale 5 seront également abordées dans ce chapitre.

Retour dans les années 60, les premiers jours des systèmes SCADA. Chaque fabricant avait sa propre méthode de communication entre ses équipements. Cela signifiait que l'interopérabilité entre les équipements de l'industrie était très limitée (interopérabilité signifiant la capacité des équipements ou des systèmes à fonctionner avec d'autres équipements ou systèmes fabriqués par différents fabricants).

Les utilisateurs ont été pour la plupart contraints de limiter leurs systèmes à n'utiliser que l'équipement d'un fournisseur spécifique.

Cela s'est bien sûr avéré très inefficace et la demande d'une solution à ce problème est donc apparue. Et à cette fin, l'CEI a créé un ensemble de règles et directives à suivre formant un protocole.

A noter que la CEI ne fournit aucune procédure de marquage pour indiquer son approbation et ne peut être rendue responsable d'aucun matériel déclaré conforme à l'une de ses normes.

La norme IEC 60870 partie 5, connue sous le nom de protocoles de transmission, fournit un profil de communication pour l'envoi de messages de la télé contrôle de base entre deux systèmes, qui utilise des circuits de données permanents directement connectés entre les systèmes. L'IEC TC 57 WG3 a développé une norme de protocole pour le télé contrôle, la télé protection et les télécommunications associées pour les systèmes d'alimentation électrique. Le résultat de ce travail est la norme IEC 60870-5. Parmi les documents suivants, les cinq premiers documents précisent la base IEC 60870-5. Le reste sont des extensions : [2]

Formats de trame de transmission IEC 60870-5-1

Services de transmission de liaison de données IEC 60870-5-2

IEC 60870-5-3 Structure générale des données d'application

IEC 60870-5-4 Définition et codage des éléments d'information

Fonctions d'application de base IEC 60870-5-5

IEC 60870-5-6 Lignes directrices pour les tests de conformité pour les normes d'accompagnement CEI 60870-5.

IEC TS 60870-5-7 Extensions de sécurité aux protocoles IEC 60870-5-101.

Le Protocol IEC 60870-5 a tendance à être populaire dans les régions non anglophones telles que l'Europe et la région MENA.

## 2.2 Définitions

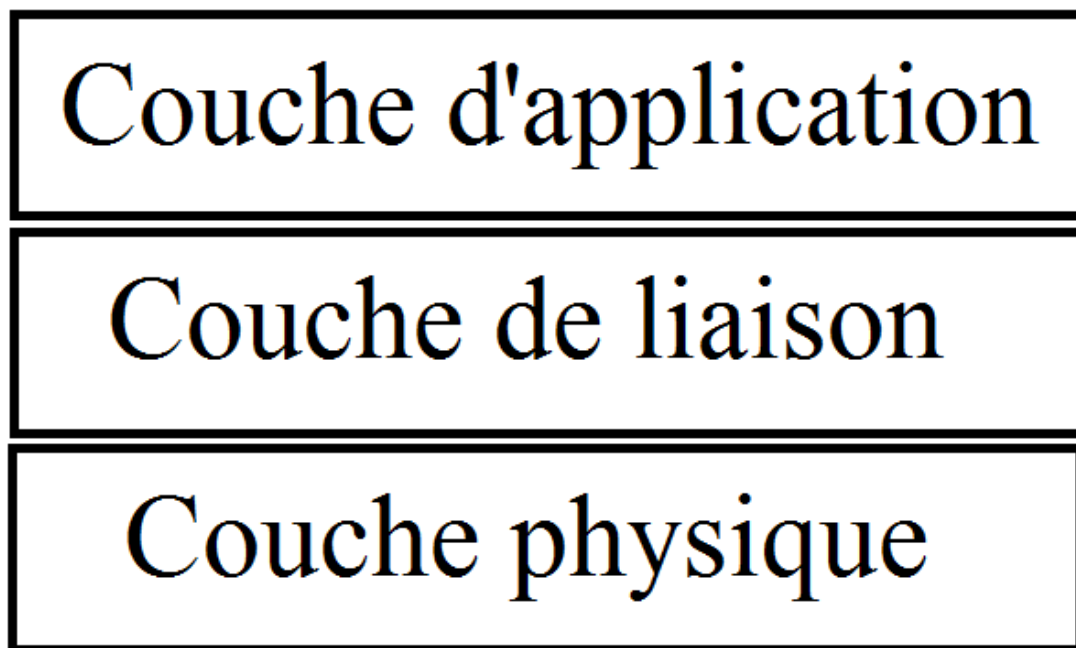
Avant d'entrer dans les détails, nous devons établir quelques définitions.

### 2.2.1 C'est quoi un Protocol de Communication?

[2]

Un protocole de communication est la manière dont les équipements communiquent entre eux ou avec l'ensemble d'un système, il est défini pour former les informations qui sont destinées à être communiquées, ainsi que les règles d'envoi de ces informations en fonction de la situation.

### 2.2.2 EPA



**Figure 2-1 les 3 couches dans EPA**

Il est basé sur les 7 couches ISO qui définissent clairement les différentes fonctions requises pour transporter des informations sur un réseau complexe. Toutes les couches ne sont pas requises dans les réseaux plus simples.

L'IEC propose une norme simplifiée à 3 couches pour des implémentations plus basiques, "Enhanced Performance Architecture", EPA. Ces couches sont:

*a*      ***Couche Physique***

La couche physique définit les moyens de transmettre un flux de bits bruts. Dans ce protocole, les données sont envoyées sous la forme de 1 bit de démarrage, 8 bits de données, 1 bit d'arrêt, 1 bit de parité, via un RS-232, CCITT V.11 ou CCITT V.24 par exemple (ces 3 sont des normes de télécommunication recommandé par l'UIT-T).

Les configurations de réseau fixes suivantes sont prises en charge :

- point à point;
- plusieurs point à point ;
- étoile multipoints ;
- ligne multipoints ;
- bague multipoint.

Cette couche est aussi concerné par le Baud rate. Le Baud rate est le débit auquel les informations sont transférées dans un canal de communication.

### ***b*** ***Couche de liaison***

La couche de liaison est chargée de reconnaître les adresses, de contrôler l'accès à la couche physique, d'assembler et de désassembler les trames, d'envoyer/recevoir des erreurs de trame et est capable de gérer différentes tailles de trame. Cette couche est également capable de gestion de collisions si nécessaire.

Dans la CEI 870-5-4, les formats de trame définis sont FT1.1, FT1.2, FT2, FT3. pour ce protocole compagnon, nous utilisons le format FT1.2.

La couche de liaison des données peut envoyer un message avec ou sans réponse attendue et envoyer une demande avec une réponse de données ou NAK (aucune donnée disponible) attendue.

### ***c*** ***Couche d'application***

La couche application est l'endroit où les ASDU à l'intérieur des trames sont traitées et traduites en commandes, en signaux ou en mesures.

## **2.2.3 ASDU**

Contenu dans chaque trame, un ASDU est un concept de la couche application, il comprend toutes les informations nécessaires concernant un objet de données plus l'objet lui-même.

Ils sont les principaux éléments constitutifs de ce protocole.

Il existe de nombreux types d'ASDU, jusqu'à environ 126 qui sont tous définis dans le protocole. Il existe également la possibilité de créer des ASDU personnalisés, mais ce n'est pas recommandé pour des raisons d'interopérabilité.

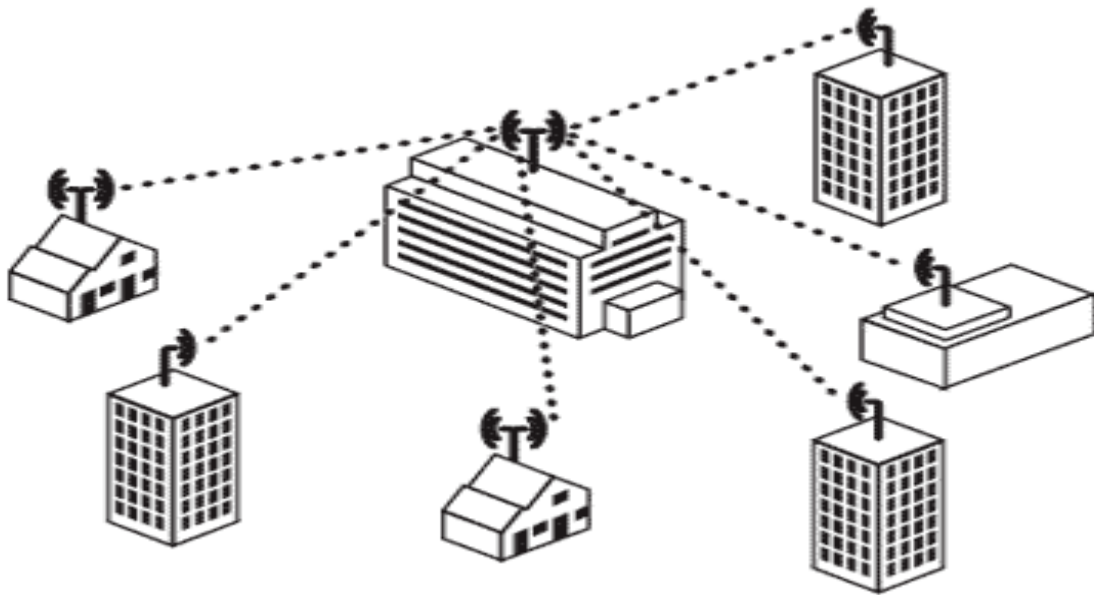


### 2.2.4 Mode Symétrique et Asymétrique

Dans ce protocole, la couche liaison prend en charge deux modes de communication.

#### *a Mode Asymétrique*

Dans ce cas, seul le poste maître peut initier la communication, cela simplifie notre système puisqu'il n'y a pas besoin de gestion des collisions. Le mode Asymétrique peut utiliser liaisons point à point ou multipoints.



**Figure 2-2 Liaisons multipoint**

#### *b Mode Symétrique*

En revanche, Le mode symétrique est limité aux liaisons point à point, mais il peut prendre en charge les messages non sollicités d'un esclave tant que les deux stations peuvent initier la communication.

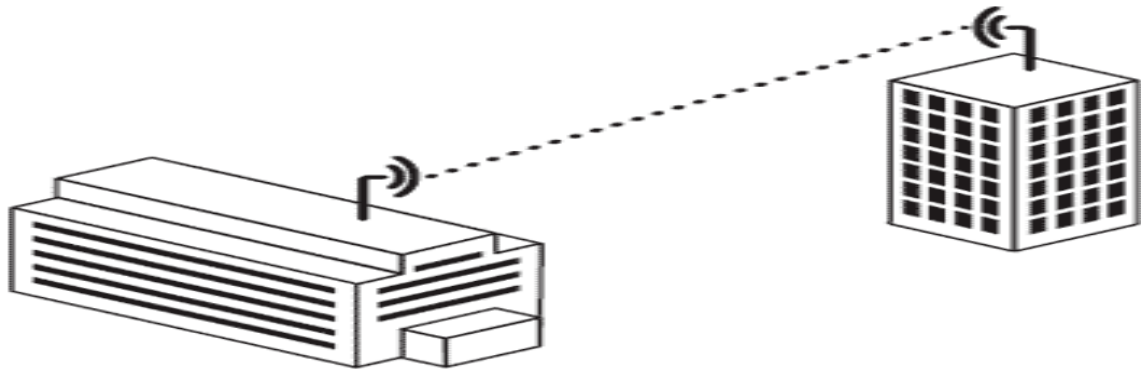


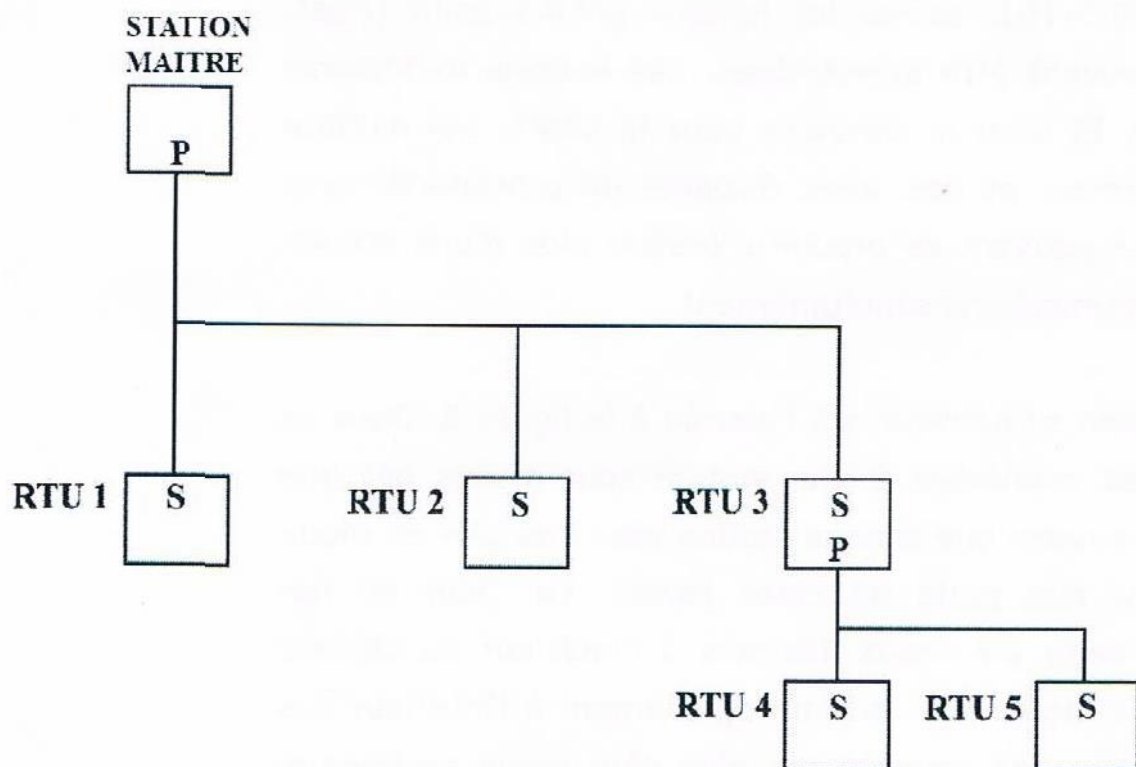
Figure 2-3 Liaisons point-a-point

### 2.2.5 Polling

En mode asymétrique, un esclave ne peut pas initier de communication avec le maître. Cela signifie que la station contrôlée ne peut pas envoyer d'informations précieuses ou éventuellement urgentes de manière non sollicitée. Donc, pour être sûr de recevoir ces informations pendant qu'elles sont fraîches, la station maître envoie périodiquement des requêtes de données à la station esclave, pendant une période de  $T$  paramétrable.

## 2.2.6 Direction de contrôle et direction de surveillance

Dans la CEI 60870-5, il existe une hiérarchie structurée supposée, dans laquelle pour deux stations données, il existe une station de contrôle (Primaire) et une station contrôlée (Secondaire). Les messages en provenance d'un poste contrôlé sont dits dans le sens "surveillance", tandis que les messages en provenance du poste de contrôle sont dits dans le sens "contrôle".



**Figure 2-4 Exemple de configuration multipoint de stations esclaves et maîtres**

Si une station envoie à la fois des données de surveillance et des commandes, on dit qu'elle fonctionne en bi-mode mais cela nécessite l'utilisation de l'adresse d'origine. Cette option existe seulement dans le mode symétrique.

### 2.2.7 Adressage

Il existe 3 types d'adresses dans le protocole IEC 60870-5, L'adresse de liaison, l'adresse CASDU et l'adresse d'objet d'information (généralement appelée IOA).

#### *a Adresse de liaison*

L'adresse de liaison représente la connexion entre la station commandée et la station de commande, elle est nécessaire dans une configuration multipoint qui est utilisée avec le mode asymétrique pour que la station <<maître>> s'adresse à la bonne station <<esclave>>. Sa taille peut varier de 0 à 2 octets

Etant donné qu'en mode équilibré seule la configuration point à point est autorisée, cet adressage devient redondant mais il peut être utilisé à des fins de sécurité.

#### *b Adresse CASDU*

L'adresse CASDU représente l'adresse de la station contrôlée à la fois dans le sens du contrôle et dans celui de la surveillance. Sa taille peut varier de 1 à 3 octets.

L'adresse CASDU combinée à l'IOA donne l'adresse spécifique de tout élément dans un système complet.

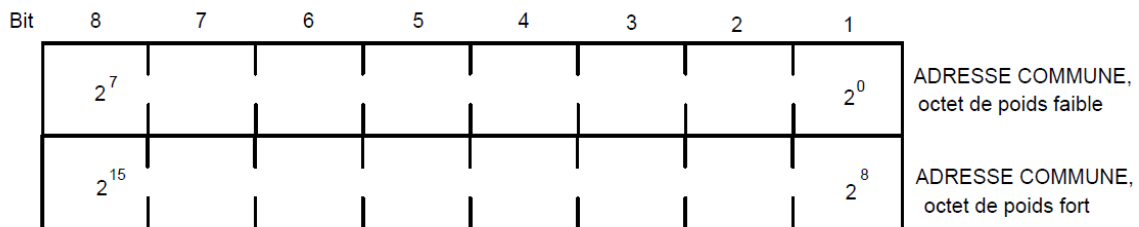
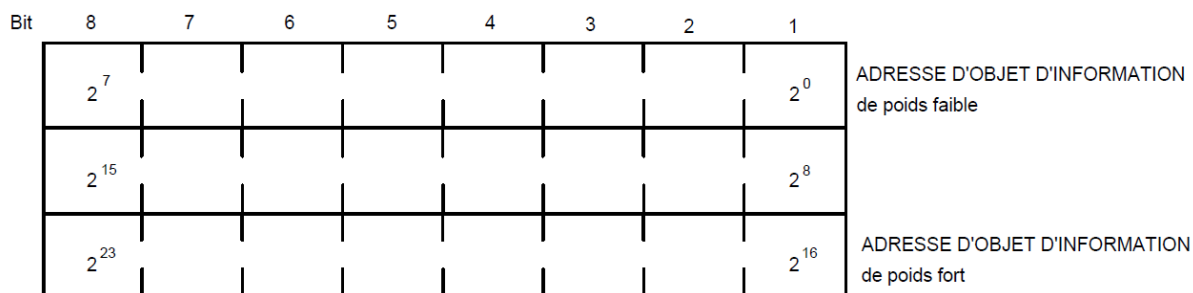


Figure 2-5 CASDU de taille 2 octets

#### *c Adresse d'objet d'information (IOA)*

Cette adresse représente l'adresse d'un élément à l'intérieur d'une station et est utilisée avec l'adresse CASDU pour former l'adresse d'un élément dans un système complet. L'adresse de l'objet d'information est utilisée comme adresse de destination dans le sens contrôle et comme adresse source dans le sens moniteur.

L'adresse d'objet d'information (IOA) a une longueur de 1 à 3 octets et peut être fournie une fois dans un ASDA. Cela permet une transmission efficace de blocs d'informations séquentiels.



**Figure 2-6 IOA de taille 3 octets**

Chaque type de données a sa propre adresse IOA de base spécifique configurée par l'utilisateur, et l'adresse de chaque membre suivant de ce groupe de types de données ajoute un décalage à cette adresse de base.

Exemple:

Si l'adresse de base TSS1 est 150, l'adresse TSS2 est 151, l'adresse TSS3 est 152, et ainsi de suite.

## 2.3 Les trames de données

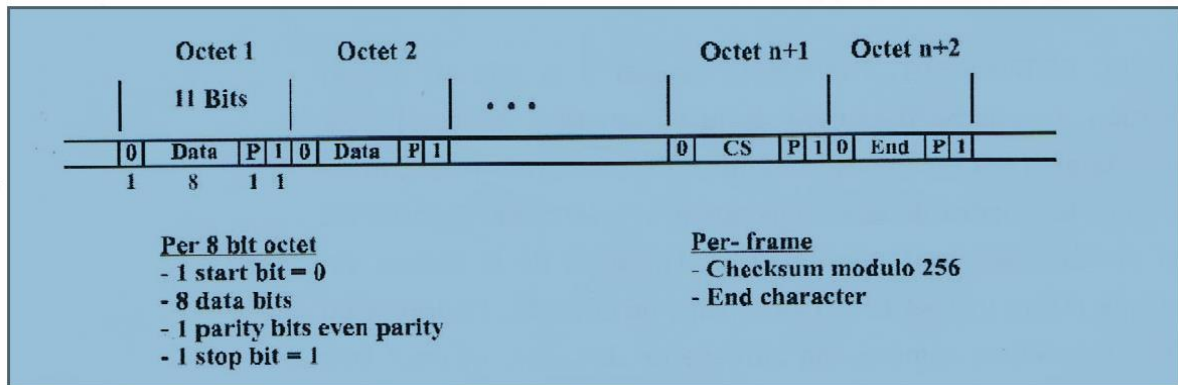
[3]

Une trame de données est le moyen le plus courant de stocker des données et, en général, est la structure de données la plus souvent utilisée pour l'analyse de données et c'est la forme sous laquelle les messages sont transmis.

### 2.3.1 Structures des trames

Dans IEC 60870-5-101, Une trame de données porte l'adresse de liaison et certaines informations de contrôle, ainsi qu'un objet de données qui se traduit différemment selon le type d'ASDU contenu dans la trame.

Dans IEC 101 les trames sont de format FT1.2 ce qui signifie que nous avons trois types des trames. Trames fixes et trames variables et des caractères uniques pour confirmer ou confirmer négativement, "Acknowledgement" (ACK) ou "negative Acknowledgement" (NACK).



**Figure 2-7 Séquence binaire d'une trame IEC 60870-5-101**

La figure 2-7 montre la configuration binaire réelle qui serait vue sur le canal physique, interprétée de gauche à droite, Cette représentation inclut les bits de départ et d'arrêt qui sont transmits avec chaque octet ou "byte" de la trame.

l'image suivante montre la structure de chaque type de trame:

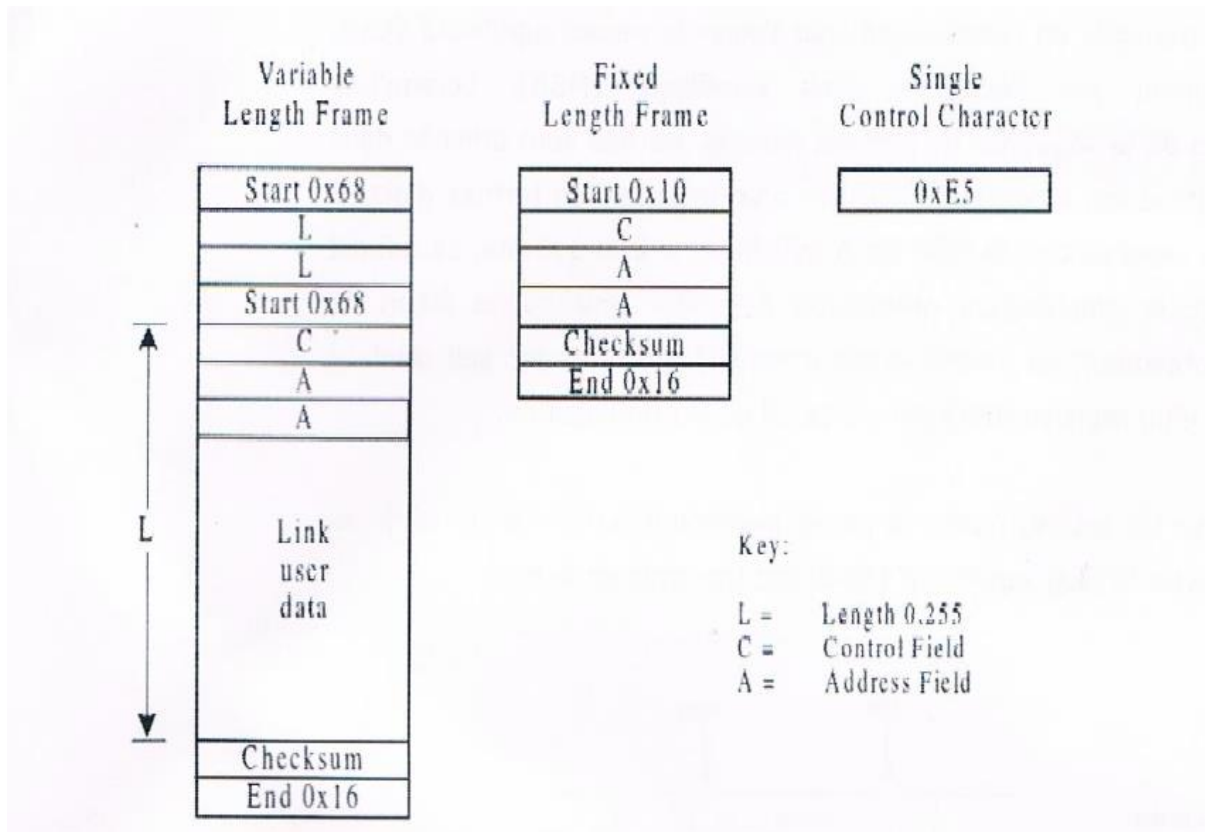


Figure 2-8 La format des trames FT1.2

Trame variable à gauche, Trame fixe au milieu, Caractère singulier à droite.

*a Les trames fixe*

Les trames fixes sont exactement ce qu'elles suggèrent, leur taille est cohérente dans tout un système, entre les systèmes sa taille peut aller de 4 à 6 octets selon la taille de l'adresse du lien puisque c'est le seul adressage inclus dans ce type de trame.

Les trames fixes sont utilisées pour envoyer des demandes, des confirmations et des services généraux de couche de liaison.

0	Start byte 1 (0x10)				Control field	
1	RES	PRM	FCB / ACD	FCV / DFC		Function code
2	Link address (0, 1 or 2 octets)					
n-1	Checksum					
n	Stop byte (0x16)					

**Figure 2-9 La structure Spécifique d'une trame fixe**

Voici les définitions des bits de champ de contrôle extraites de la documentation

**-RES/ DIR :**

**Transmission asymétrique :** Réserve

**Transmission symétrique:** Direction de transmission physique ( direction )

**DIR = 1 :** Données de la station de contrôle ( Maître ) à la station contrôlée ( Esclave ).

**DIR = 0 :** Données de la station contrôlée ( esclave ) à la station de contrôle ( maître ).

**-PRM Message primaire :** Bit primaire (message primaire)

**0** = message d'une station secondaire (contrôlée).

**1** = message d'une station principale (contrôleuse).

**FCB Message primaire :** bit de comptage de trames

Bit en alternance pour les messages SEND/ CONFIRM ou REQUEST/RESPOND consécutifs par station. La station primaire alterne le bit FCB pour chaque nouveau SEND/CONFIRM ou REQUEST/RESPOND service de transmission dirigé vers la même station secondaire.

Si une réponse attendue a expiré (manquante), alors les mêmes services SEND/CONFIRM et REQUEST/RESPOND sont répétés avec le même bit de comptage de trames.

Dans le cas d'une commande reset, le bit FCB est toujours à zéro, et à réception de ces commandes, la station secondaire attend toujours que le prochain message primaire soit mis à FCV = valid (FCV = 1 ) afin d'avoir la valeur inverse de FCB (c'est-à-dire FCB = 1).

**Message principal FCV :** bit de comptage de trames valide .



0 = fonction alternée et bit FCB invalide

1 = fonction alternée et bit FCB valide

Services SEND/NO REPLY, messages diffusés et autres services de transmission qui tolèrent la duplication

ou l'effacement de la sortie d'information n'alterne pas le bit FCB et l'indique par un bit FCB effacé.

### **ACD/ RES**

**Transmission symétrique** : Réserve

**Transmission Asymétrique** : Demande d'accès

**Message secondaire** : Demande d'accès

Il existe deux classes de données de message fournies, à savoir la classe 1 et la classe 2. Les stations secondaires indiquent aux stations primaires à l'aide du bit ACD qu'il existe des données de classe 1 prêtes à être transmises.

**0** = pas de demande d'accès lors de la transmission de données de classe 1

**1** = demande d'accès lors de la transmission de données de classe 1

La transmission de données de classe 1 est normalement utilisée pour des événements ou pour des messages de haute priorité. Classe 2

la transmission de données est normalement utilisée pour la transmission cyclique ou pour une faible priorité.

**Message secondaire DFC** : contrôle de flux de données

**0** = plus de messages sont acceptés

**1** = d'autres messages peuvent provoquer un débordement de données

Les stations secondaires ( répondantes ) indiquent par ce message à la station émettrice ( principale ) qu'un

la succession immédiate d'un autre message peut provoquer un débordement de la mémoire tampon.

**Function code:**

Il représente la fonction de la trame qui est envoyée dans 4 bits

**T101 – CONTROL FIELD (unbalanced mode)**

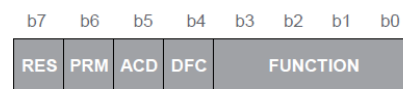
● **Master → Slave**



- **RES** : =0, Reserved
- **PRM** : =1, Primary (Initiating station)
- **FCB** : Frame Count Bit = 0/1 alternatively
- **FCV** : Frame Count Validity = Validity of FCB.
  
- **FUNCTION** :

  - 0 : Reset of link (FCV:0)
  - 1 : Reset of user process (FCV:0)
  - 3 : User data with confirmation (FCV:1)
  - 4 : User data without confirmation (FCV:0)
  - 9 : Request status of link (FCV:0)
  - 10 : Request class 1 data (FCV:1)
  - 11 : Request class 2 data (FCV:1)

Slave → Master



- RES** : =0, Reserved
- PRM** : =0, Secondary (responding station)
- ACD** : = 1, Access demand Class 1 Data
- DFC** : =1, risk of Data Flow Control
  
- FUNCTION** :

  - 0 : ACK, Positive acknowledgement
  - 1 : NACK, message not accepted, link busy
  - 8 : User Data response
  - 9 : NACK, Data not available
  - 11 : Status of link or access demand

**Figure 2-10 Le champ de contrôle et ses différents bits et codes de fonction possibles**

Voici un Exemple d'une trame fixe:

**Sample 1**

10 49 0C 00 55 16

<b>LPDU bytes</b>	<b>Explanation</b>
10	Start byte: frame with fixed length
49	Control filed: PRM-bit set (frame from primary station), function code = 9 (link status)
0C 00	Link address (2 octets) = 12 dec.
55	Checksum
16	Stop byte

**Figure 2-11 Exemple d'une trame fixe*****b Les trames variable***

Les trames variables sont utilisées pour transmettre les données utilisateur entre la station de contrôle et la station contrôlée, elles peuvent atteindre jusqu'à 255 octets.

Puisqu'ils sont responsables du transport des données utilisateur, ils sont le type de trame qui contient ASDUS, il ne peut y avoir qu'un seul ASDU par trame.

0	Start byte 1 (0x68)					Header
1	Block length					
2	Block length (copy)					
3	Start byte 2 (0x68)					
4	RE S	PR M	FCB / ACD	FCV / DFC	Function code	Control field
5	Link address fields (0, 1 or 2 octets)					
	<u>Type identification</u>					DATA UNIT IDENTIFIER
	SQ	Number of objects				
	T	P/N	<u>Cause of transmission (COT)</u>			
	Originator address (ORG, 0 or 1 octets)					
	ASDU address fields (1 or 2 octets)					
	Information object address fields (IOA) (1,2 or 3 octets)					
	Object information					Info-object
n-1	Checksum					Tail
n	Stop byte (0x16)					

**Figure 2-12 Structure général d'une trame variable**

**Data Unit Identifier combiné avec info-object donne l'ASDU.**

Une **checksum** est une technique utilisée pour déterminer l'authenticité des données reçues, c'est-à-dire pour détecter s'il y a eu une erreur de transmission. ... Lorsque le récepteur obtient les données, il calcule la somme de contrôle des données reçues en utilisant le même algorithme et la compare avec la somme de contrôle transmise. Dans IEC 60870-5, checksum est égales tout simplement a la somme des octets après Header ( et devant checksum bien sure) .

**Bloc length** donne la longueur des données, il est répété pour la fiabilité.

Exemple d'une trame variable:

**Sample 3**

68 0B 0B 68 08 0C 00 65 01 0A 0C 00 00 00 05 95 16

<b>LPDU bytes</b>	<b>Explanation</b>
68	Start byte 1: frame with variable length
0B 0B	Block length and block length copy
68	Start byte 2
08	Control field: PRM-bit not set (frame from secondary station), function code = 8 (user data)
0C 00	Link address (2 octets) = 12 dec.
65	Type identification: C_CI_NA_1 (counter interrogation)
01	Number of objects = 1
0A	Cause of transmission = 10 (activation confirmation)
0C 00	Common ASDU address (2 octets) = 12 dec.
00 00	Object address (2 octets) = 0
05	Counter interrogation request qualifier = 5 (general counter interrogation)
95	Checksum
16	Stop byte

**Figure 2-13 Exemple d'une trame variable avec explication**

*c Caractères uniques*

Des caractères uniques sont utilisés pour confirmer la couche de liaison et les transmissions de données utilisateur.

- 0xE5 (acquiescement positif) ;
- 0xA2 (accusé de réception négatif) .

## 2.4 Les différentes ASDU Utilisés et leurs structures:

[3]

### ● TYPE IDENTIFICATION

#### ● Slave → Master

<0>	:= not defined	
<1>	:= single-point information	M_SP_NA_1
<2>	:= single-point information with time tag	M_SP_TA_1
<3>	:= double-point information	M_DP_NA_1
<4>	:= double-point information with time tag	M_DP_TA_1
<5>	:= step position information	M_ST_NA_1
<6>	:= step position information with time tag	M_ST_TA_1
<7>	:= bitstring of 32 bit	M_BO_NA_1
<8>	:= bitstring of 32 bit with time tag	M_BO_TA_1
<9>	:= measured value, normalized value	M_ME_NA_1
<10>	:= measured value, normalized value with time tag	M_ME_TA_1
<11>	:= measured value, scaled value	M_ME_NB_1
<12>	:= measured value, scaled value with time tag	M_ME_TB_1
<13>	:= measured value, short floating point number	M_ME_NC_1
<14>	:= measured value, short floating point number with time tag	M_ME_TC_1
<15>	:= integrated totals	M_IT_NA_1
<16>	:= integrated totals with time tag	M_IT_TA_1
<17>	:= event of protection equipment with time tag	M_EP_TA_1
<18>	:= packed start events of protection equipment with time tag	M_EP_TB_1
<19>	:= packed output circuit information of protection equipment with time tag	M_EP_TC_1
<20>	:= packed single-point information with status change detection	M_PS_NA_1
<21>	:= measured value, normalized value without quality descriptor	M_ME_ND_1
<30>	:= single-point information with time tag CP56Time2a	M_SP_TB_1
<31>	:= double-point information with time tag CP56Time2a	M_DP_TB_1
<32>	:= step position information with time tag CP56Time2a	M_ST_TB_1
<33>	:= bitstring of 32 bits with time tag CP56Time2a	M_BO_TB_1
<34>	:= measured value, normalized value with time tag CP56Time2a	M_ME_TD_1
<35>	:= measured value, scaled value with time tag CP56Time2a	M_ME_TE_1
<36>	:= measured value, short floating point number with time tag CP56Time2a	M_ME_TF_1
<37>	:= integrated totals with time tag CP56Time2a	M_IT_TB_1
<38>	:= event of protection equipment with time tag CP56Time2a	M_EP_TD_1
<39>	:= packed start events of protection equipment with time tag CP56Time2a	M_EP_TE_1
<40>	:= packed output circuit information of protection equipment with time tag CP56Time2a	M_EP_TF_1

#### ● Master → Slave

<45>	:= single command	C_SC_NA_1
<46>	:= double command	C_DC_NA_1
<47>	:= regulating step command	C_RC_NA_1
<48>	:= set point command, normalized value	C_SE_NA_1
<49>	:= set point command, scaled value	C_SE_NB_1
<50>	:= set point command, short floating point number	C_SE_NC_1
<51>	:= bitstring of 32 bits	C_BO_NA_1
<100>	:= interrogation command	C_IC_NA_1
<101>	:= counter interrogation command	C_CI_NA_1
<102>	:= read command	C_RD_NA_1
<103>	:= clock synchronization command	C_CS_NA_1
<104>	:= test command	C_TS_NA_1
<105>	:= reset process command	C_RP_NA_1
<106>	:= delay acquisition command	C_CD_NA_1
<110>	:= parameter of measured value, normalized value	P_ME_NA_1
<111>	:= parameter of measured value, scaled value	P_ME_NB_1
<112>	:= parameter of measured value, short floating point number	P_ME_NC_1
<113>	:= parameter activation	P_AC_NA_1
<120>	:= file ready	F_FR_NA_1
<121>	:= section ready	F_SR_NA_1
<122>	:= call directory, select file, call file, call section	F_SC_NA_1
<123>	:= last section, last segment	F_LS_NA_1
<124>	:= ack file, ack section	F_AF_NA_1
<125>	:= segment	F_SG_NA_1
<126>	:= directory	F_DR_TA_1

Figure 2-14 Liste montrant une grande variété d'ASDU

Il existe de nombreux types d'ASDU, jusqu'à 127 (avec la possibilité de créer des ASDU personnalisés. Comme la plupart d'entre eux ne sont pas utilisés dans notre projet, nous n'en explorerons que quelques-uns)

Nous les diviserons en fonction du type de données utilisateur dans lequel ils sont inclus dans notre travail.

### 2.4.1 Télésignalisation simple (TSS)

TSS est Envoyé par la station contrôlé, Cela inclut les ASDU de TI (Type Identification) 1, 3, et 30.

donne une information binaire dans 1 bit , 0 = ON et 1= OFF.

*a* **TYPE IDENT 1 : Informations ponctuelles sans horodatage**

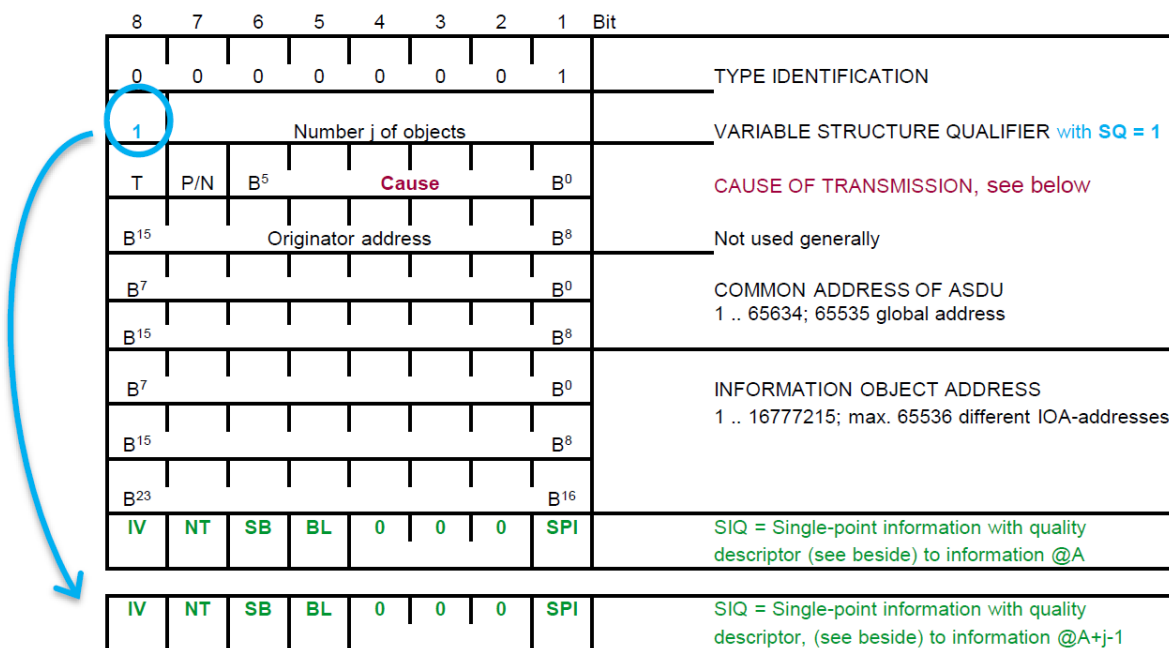


Figure 2-15 Structures Général d'ASDU TI 1 partie 1

**SIQ (Single-point information ) :**

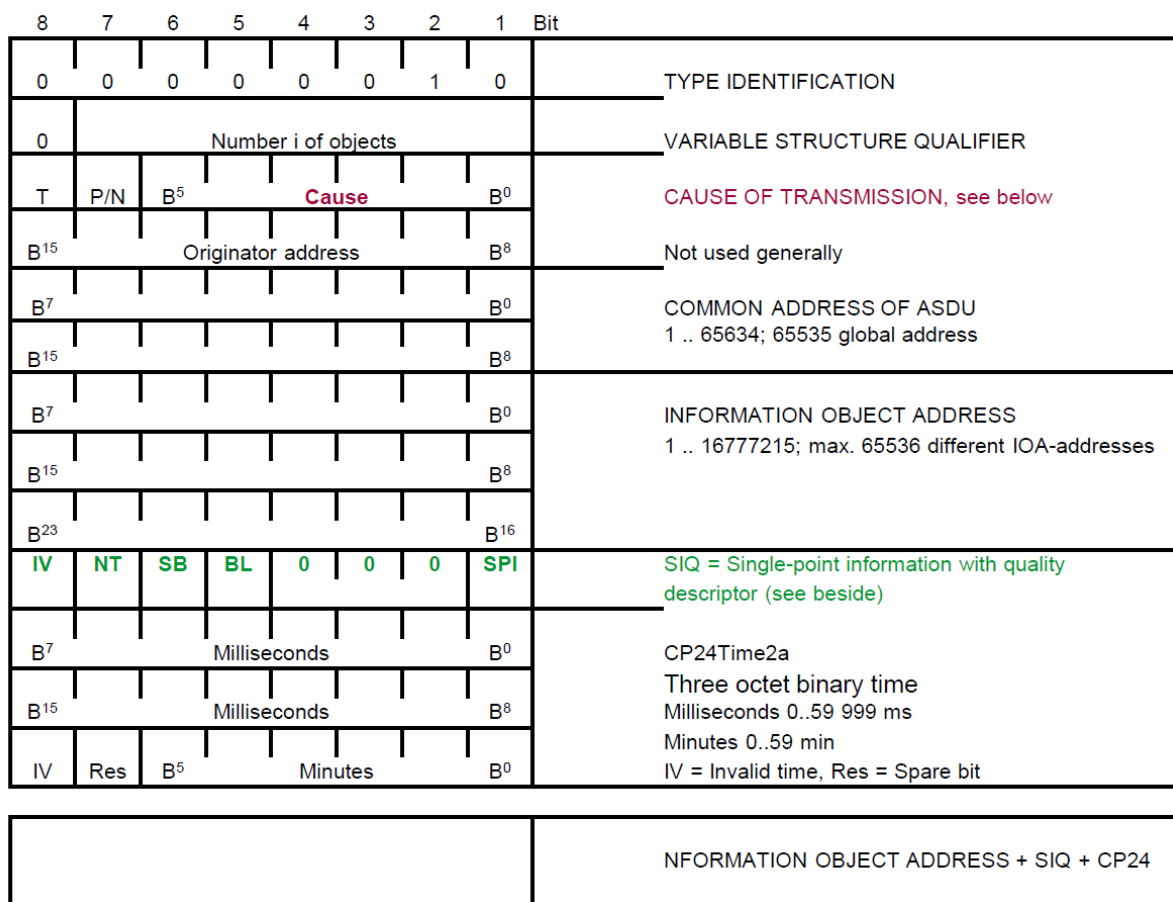
<b>SPI</b>	<0>	:=	OFF
	<1>	:=	ON
<b>BL</b>	<0>	:=	not blocked
	<1>	:=	blocked
<b>SB</b>	<0>	:=	not substituted
	<1>	:=	substituted
<b>NT</b>	<0>	:=	topical
	<1>	:=	not topical
<b>IV</b>	<0>	:=	valid
	<1>	:=	invalid

**Figure 2-16 Structures Général d'ASDU TI 1 partie 2**

Si SQ= 1, l'objet informationnel est une séquence, sinon les objets sont tous au singulier.



**b TYPE IDENT 2 : Informations ponctuelles avec horodatage CP24**



**Figure 2-17 Structure général d'ASDU TI 2**

C'est fondamentalement la même chose que TI 1 sauf avec une étiquette de temps ajoutée de 3 bits

*c* **TYPE IDENT 30 : Informations ponctuelles avec horodatage CP56**

8	7	6	5	4	3	2	1	Bit	
0	0	0	1	1	1	1	0		TYPE IDENTIFICATION
0	Number i of objects								VARIABLE STRUCTURE QUALIFIER
T	P/N	B <sup>5</sup>	Cause				B <sup>0</sup>		CAUSE OF TRANSMISSION, see below
B <sup>15</sup> Originator address B <sup>8</sup>									Not used generally
B <sup>7</sup> COMMON ADDRESS OF ASDU B <sup>0</sup>									COMMON ADDRESS OF ASDU 1 .. 65634; 65535 global address
B <sup>15</sup> INFORMATION OBJECT ADDRESS B <sup>8</sup>									INFORMATION OBJECT ADDRESS 1 .. 16777215; max. 65536 different IOA-address
B <sup>7</sup>									
B <sup>15</sup>									
B <sup>23</sup> B <sup>16</sup>									SIQ = Single-point information with quality descriptor (see beside)
IV	NT	SB	BL	0	0	0	SPI		
B <sup>7</sup> Milliseconds B <sup>0</sup>									CP56Time2a seven bytes binary time
B <sup>15</sup> Milliseconds B <sup>8</sup>									Milliseconds 0..59 999 ms
IV	Res1	B <sup>5</sup>	Minutes				B <sup>0</sup>		Minutes 0..59 min
SU	Res2	B <sup>4</sup>	Hours				B <sup>0</sup>		Hours 0..23 h
B <sup>2</sup> Day of week B <sup>0</sup>		B <sup>4</sup> Day of mouths					B <sup>0</sup>		SU = 1 for summer time
Res3		B <sup>3</sup> Mouths					B <sup>0</sup>		Days of month 1..31
							B <sup>0</sup>		Days of week 1..7
							B <sup>0</sup>		Months 1..12
Res4	B <sup>6</sup>	Years						B <sup>0</sup>	Years 0..99

**Figure 2-18 Structure général d'ASDU 30**

C'est fondamentalement la même chose que TI 2 sauf avec une étiquette de temps de 7 bits

### 2.4.2 Télésignalisation Double (TSD)

Comme la télésignalisation simple mais avec de une point double d'information (DPI) a place de SPI.

donne une information binaire dans 0 bit

0 = Intermédiaire

1 = OFF

2 = ON

3 = Intermédiaire

### DIQ (Double-point information) :

DPI

<0>	:=	Indeterminate or intermediate state
<1>	:=	OFF
<2>	:=	ON
<3>	:=	Indeterminate state

**Figure 2-19** Interprétation des les 2 bits DIP

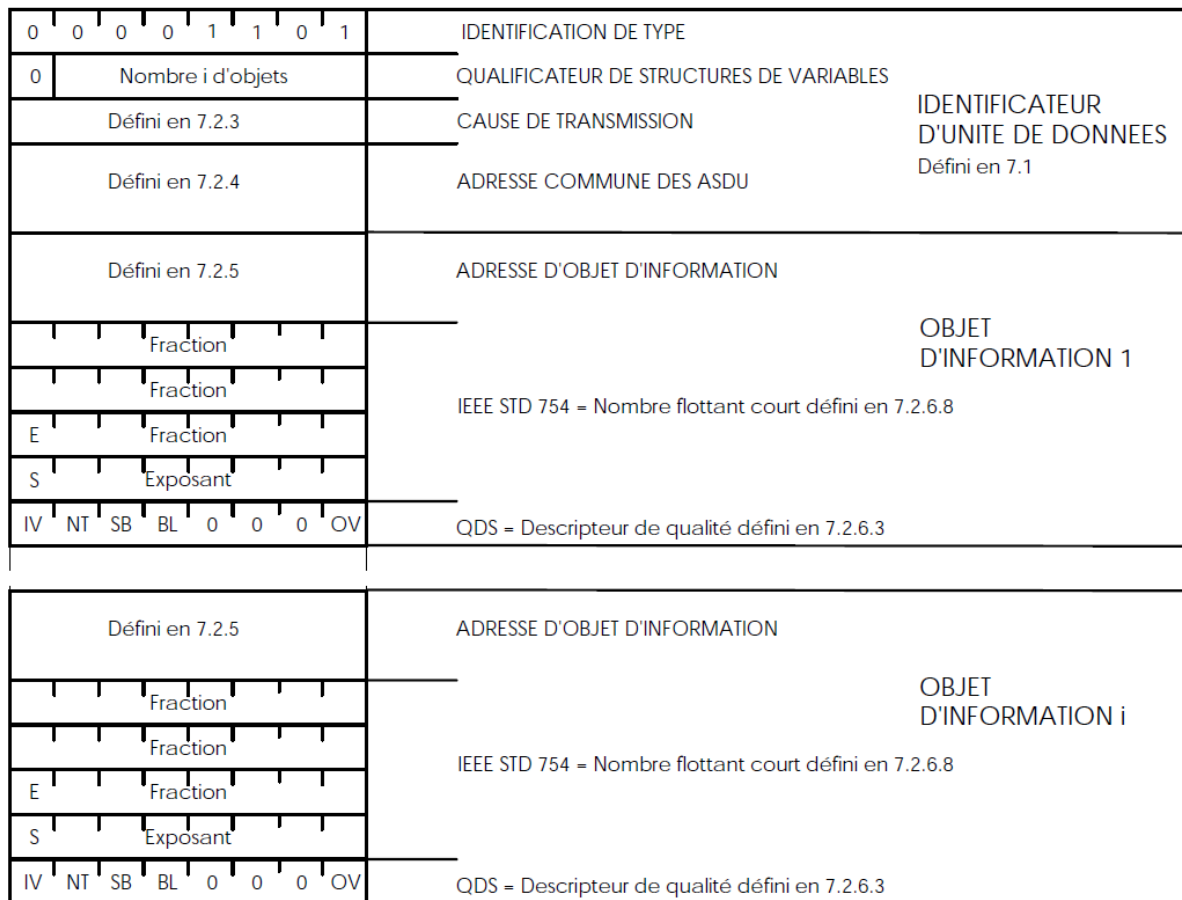
Cela inclut les ASDU de TI (Type Identification) 2, 4, et 31.

### 2.4.3 Télémessures

Cela inclus plusieurs ASDU mais on travaille seulement avec TI 13 pour le moment, il est utilisé pour donner des valeurs de mesure spécifiques

*a* **TYPE IDENT 13: Valeur mesurée, nombre flottant court**

**Séquence d'objets d'information (SQ=0)**



**Figure 2-20 Structure général d'ASDU 13**

### 2.4.4 Télécommande double (TCD)

Envoyé par la station de contrôle, cela inclut le ASDU 46

8	7	6	5	4	3	2	1	Bit	
0	0	1	0	1	1	1	0		TYPE IDENTIFICATION
0	0	0	0	0	0	0	1		VARIABLE STRUCTURE QUALIFIER
T	P/N	Cause				B <sup>0</sup>			CAUSE OF TRANSMISSION, see below
B <sup>15</sup> Originator address B <sup>8</sup>									Not used generally
B <sup>7</sup> B <sup>0</sup>									COMMON ADDRESS OF ASDU 1 .. 65634; 65535 global address
B <sup>15</sup> B <sup>8</sup>									INFORMATION OBJECT ADDRESS 1 .. 16777215; max. 65536 different IOA-addresses
B <sup>7</sup> B <sup>0</sup>									
B <sup>15</sup> B <sup>8</sup>									
B <sup>23</sup> B <sup>16</sup>									DCO = Double command, see beside
S/E	QU				DCS				

Figure 2-21 Structure général d'ASDU 46

**DCO (Double command state) :****DCS**

- <0> := not permitted
- <1> := OFF
- <2> := ON
- <3> := not permitted

**QU**

- <0> := no additional definition (1)
- <1> := short pulse duration  
(circuit-breaker), duration  
determined by a system  
parameter in the outstation
- <2> := long pulse duration, duration  
determined by a system parameter  
in the outstation
- <3> := persistent output
- <4..8> := reserved for standard  
definitions of this companion  
standard (compatible range)
- <9..15> := reserved for the  
selection of other predefined  
functions (2)
- <16..31> := reserved for special  
use (private range)

**S/E**

- <0> := Execute
- <1> := Select

**Figure 2-22 Sémantiques de l'octet de commande**

## 2.5 Conclusion

Le protocole IEC 60870-5 est un protocole couramment utilisé pour les systèmes d'alimentation. Nous avons exploré ses bases et un peu de ses spécificités et de sa sémantique qui sont importantes pour le travail.

Dans le prochain chapitre, nous commencerons à présenter notre travail et à expliquer ses différentes parties

**Chapitre 3 l'IHM du Coffret de commande:**



## 3.1 Introduction

Créer une interface homme-machine(IHM) valide n'est pas exactement une tâche simple, cela nécessite une bonne compréhension de la machine avec laquelle nous essayons d'interfacer et du type de communication spécifique (protocole IEC 101 dans notre cas) entre notre machine et la machine hôte pour notre IHM (généralement un PC), et bien sûr les compétences de programmation nécessaires pour le rendre fonctionnel.

Dans ce chapitre, nous allons explorer la création de cette interface et les étapes nécessaires à suivre.

## 3.2 Software utilisés

Pour la réalisation de cette application, nous avons choisi le langage de programmation connu sous le nom de "C#" et en utilisant la plateforme appelée "Visual Studios 2019".

Notre choix s'est porté sur le C# comme langage de programmation car il est polyvalent et prend en charge la création d'application graphique avec "Winforms" et sur la plateforme Visual studios 2019 car c'est le moyen le plus pratique pour écrire un code en C#.

### 3.2.1 Visual Studios 2019 (Community Edition):

[4]

L'environnement de développement intégré de Visual Studio est une plateforme de lancement créative avec laquelle on peut modifier, déboguer et générer un code, puis publier une application. Un environnement de développement intégré (IDE) est un programme riche en fonctionnalités qui peut être utilisé pour de nombreux aspects du développement de logiciels. Au-delà de l'éditeur et du débogueur standard fournis par la plupart des IDE, Visual Studio inclut des compilateurs, des outils de complétion de code, des concepteurs graphiques et de nombreuses autres fonctionnalités afin de faciliter le processus de développement logiciel.

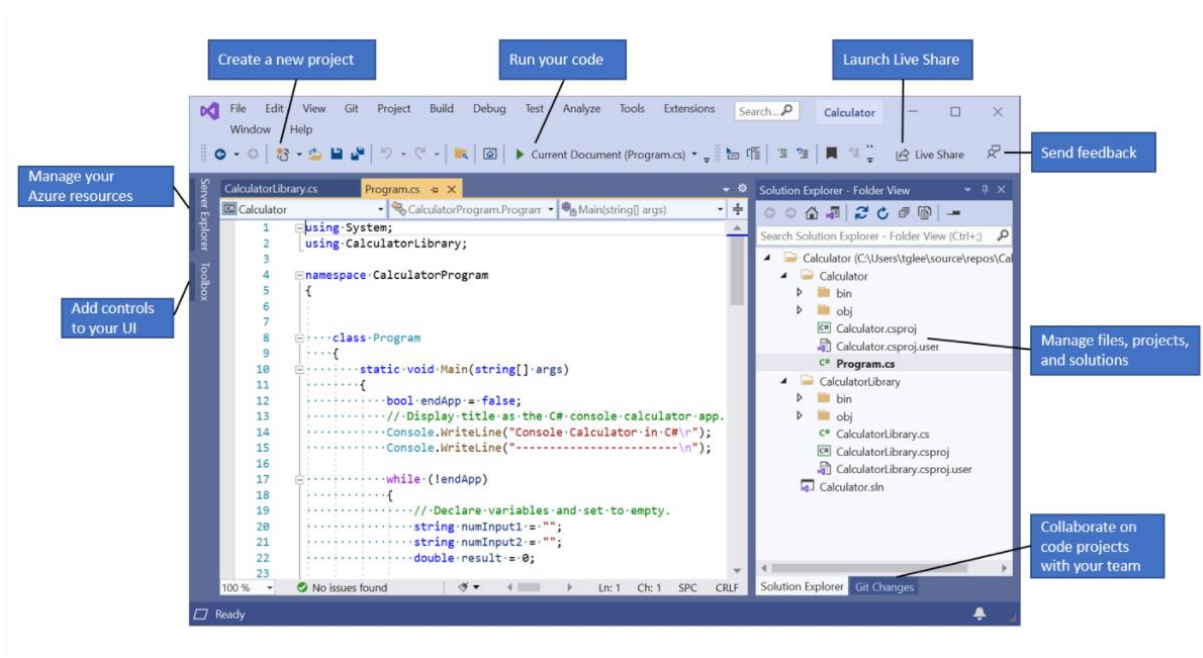


Figure 3-1 Capture d'écran de l'IDE de Visual Studios, qui comprend des légendes indiquant où se trouvent les fonctionnalités clés

Pour notre travail, la plus grande caractéristique visée est appelée "Intellisense".

IntelliSense est un terme désignant un ensemble de fonctionnalités qui affiche des informations sur le code directement dans l'éditeur et, dans certains cas, écrit de petits morceaux de code. C'est comme avoir une documentation de base en ligne dans l'éditeur, ce qui nous a permis d'éviter la recherche des informations de type ailleurs. Les fonctionnalités d'IntelliSense varient selon la langue.

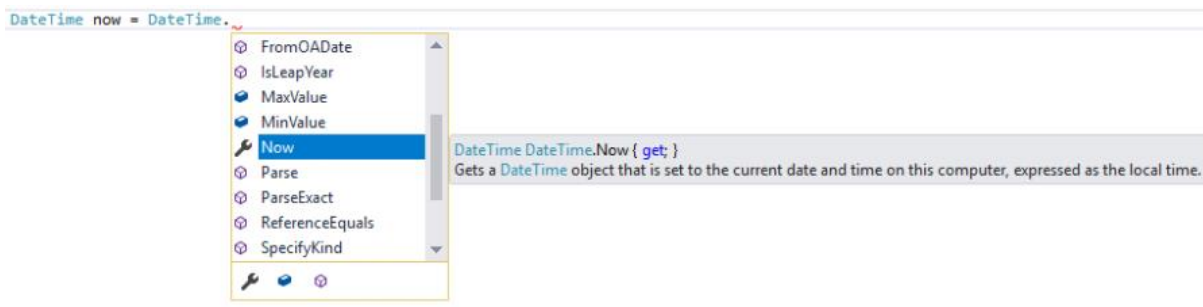


Figure 3-2 Illustration montrant comment IntelliSense affiche une liste de membres pour un type

Elle est incroyablement utile et pratique et a rendu le codage beaucoup plus facile et beaucoup plus productif.

### **3.2.2 Langage de programmation C#**

[5]

Le C# est un langage de programmation développé et lancé par Microsoft en 2001. C# est un langage simple, moderne et orienté objet qui offre aux développeurs modernes la flexibilité et les fonctionnalités nécessaires pour créer des logiciels qui non seulement fonctionnent aujourd'hui, mais seront applicables pendant des années dans le futur.

J'ai choisi ce langage pour sa relative simplicité et sa syntaxe facile à apprendre, la cerise sur le gâteau étant l'option "winforms", ce qui rend la conception d'un programme d'interface très simple.

### **3.2.3 Création d'un projet et configuration d'une station de travail**

[4]

Pour créer un projet, il faut suivre les étapes suivantes :

Ouverture du Visual Studio.

Lorsque l'on lance Visual studio, la fenêtre de démarrage s'affiche avec différentes options :

- Cloner un référentiel (copier du code à partir d'un site Web comme GitHub)
- Ouverture d'un projet récent
- Création d'un nouveau projet.

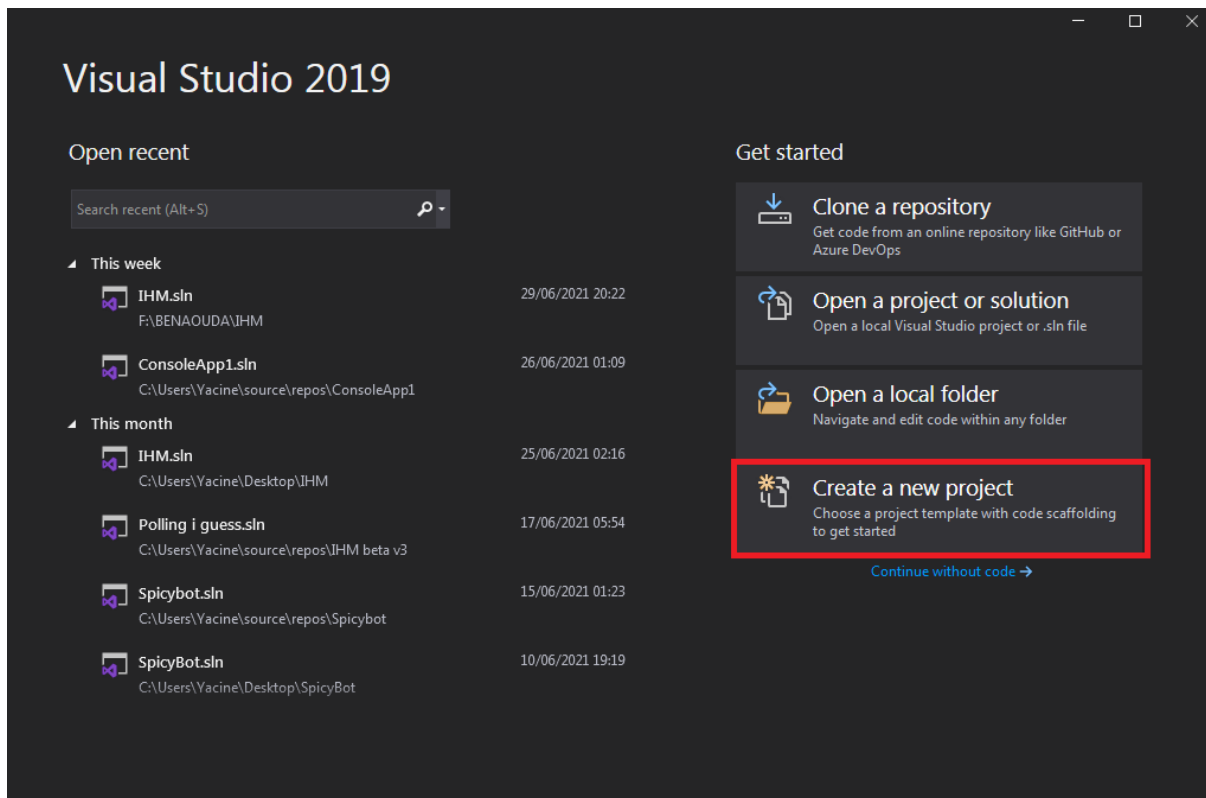


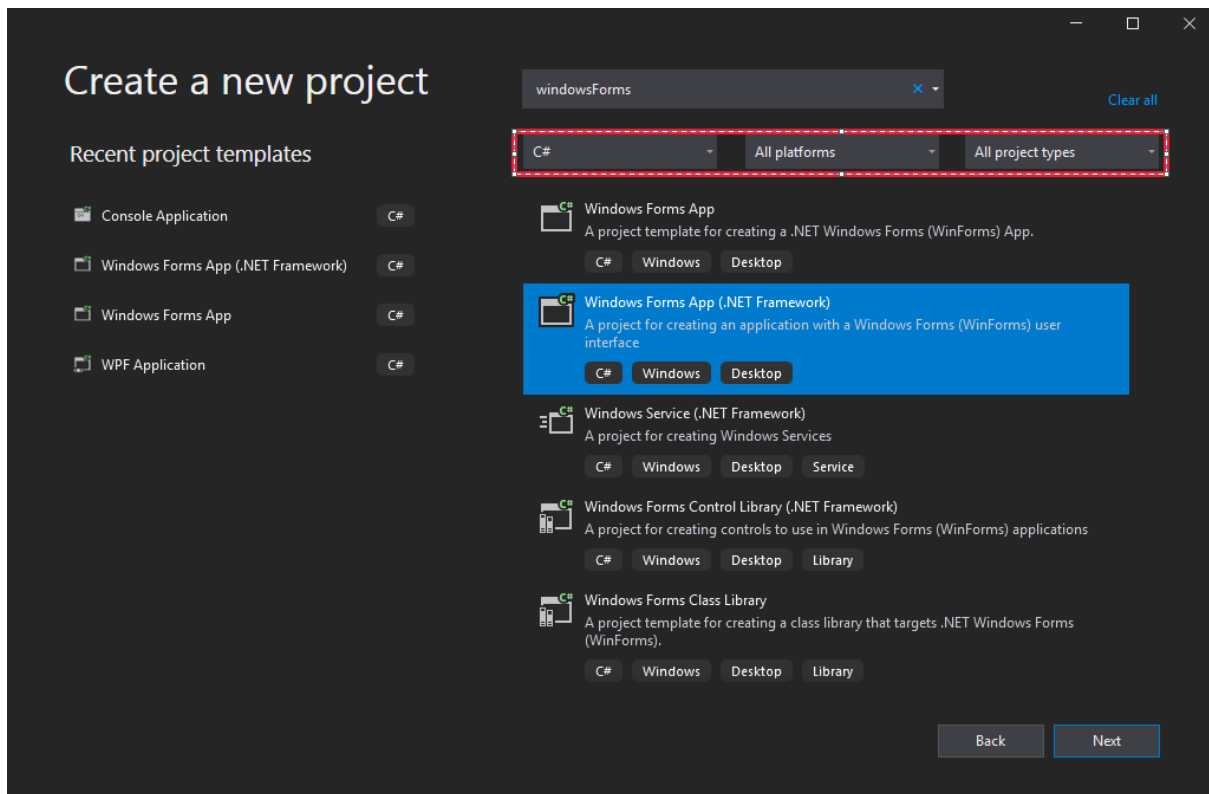
Figure 3-3 Fenêtre de démarrage de Visual Studio

*a*      **Création d'un nouveau projet.**

Pour créer un nouveau projet, il faut sélectionner l'action «**Créer un projet**». La fenêtre s'affiche et présente plusieurs *modèles*. Un modèle contient les fichiers et les paramètres de base requis pour un type de projet donné.

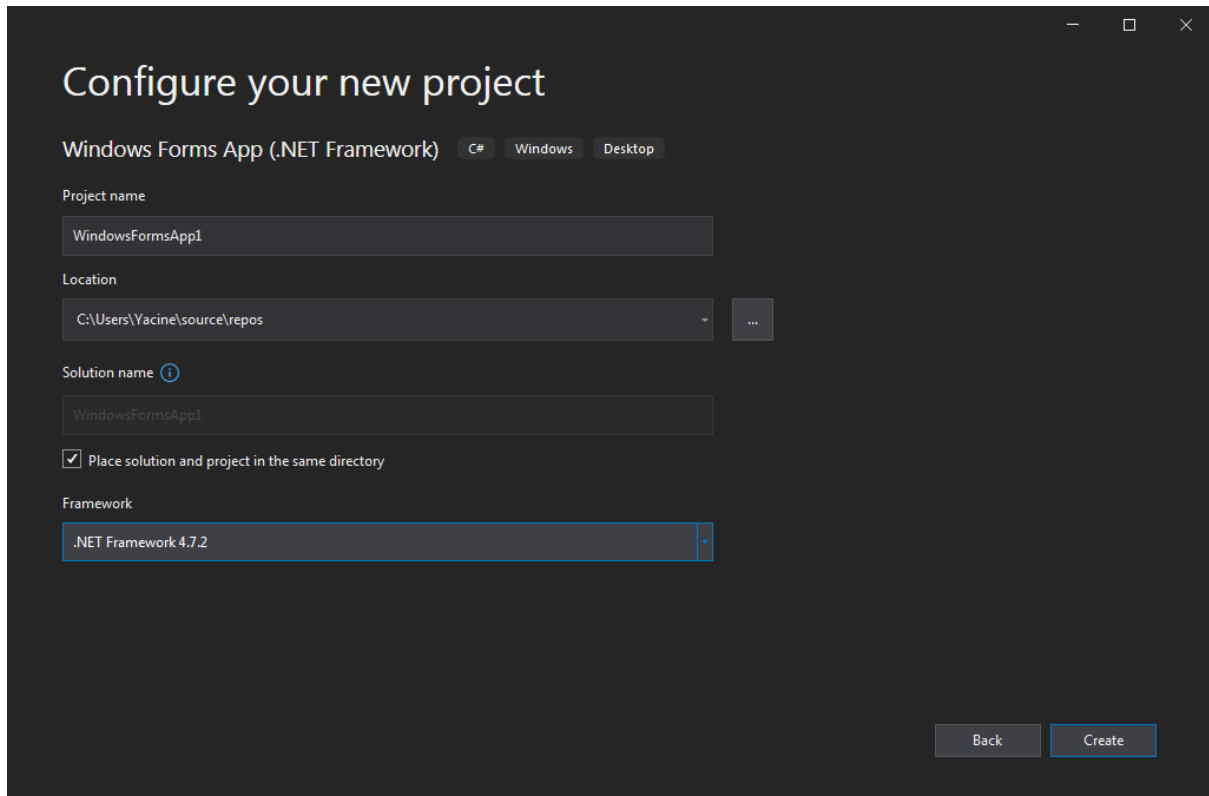
Pour rechercher le modèle souhaité, on tape **console .net core** dans la zone de recherche. La liste des modèles disponibles est automatiquement filtrée en fonction des mots clés que nous avons entrés. On peut filtrer davantage les résultats du modèle en choisissant **C#** dans la liste déroulante **tous les langages, Windows** dans la liste **toutes les plateformes** et la **console** dans la liste **tous les types de projets**.

On sélectionne le modèle **Windows Forms App (.NET Framework)**, puis on clique sur **suivant**.



**Figure 3-4** Fenêtre de création de projet

Dans la fenêtre **configurer un nouveau projet**, on peut donner un nom au projet puis choisir un chemin où il sera enregistré. Une fois que les informations sont entrées, il suffit de cliquer sur le bouton «créer».



**Figure 3-5** Fenêtre de configuration du projet

Une fois le projet créé, on obtient la fenêtre suivante:

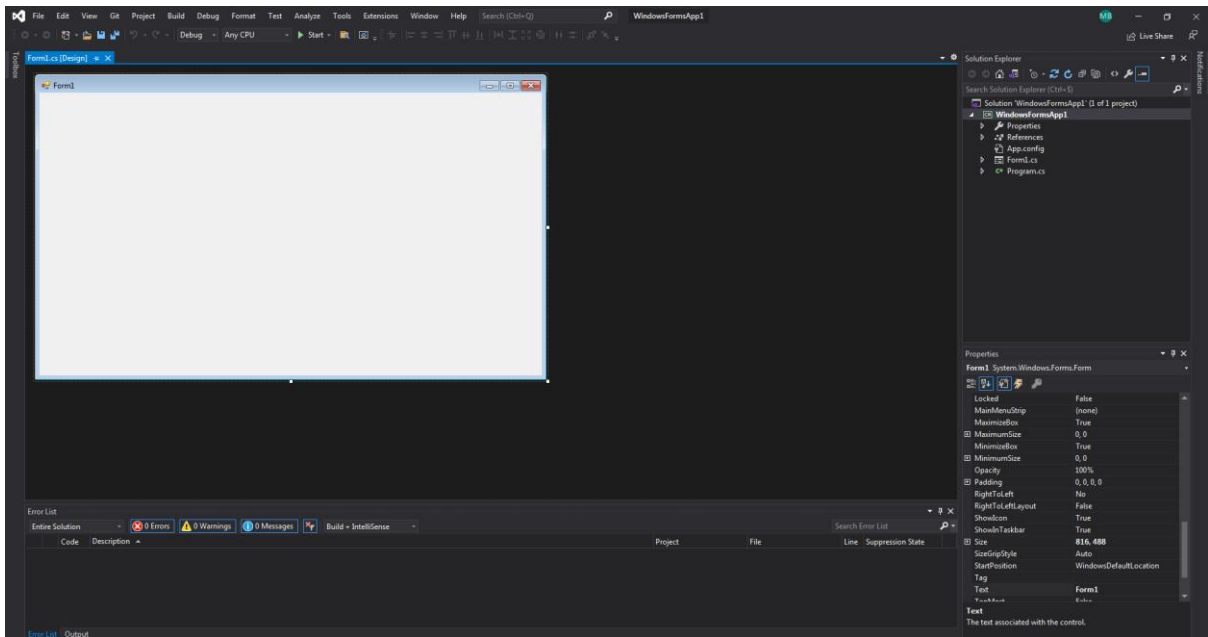


Figure 3-6 Projet Winforms

Un double clic sur la fenêtre sur la fenêtre créée nommée Form 1 montré dans Figure 3.6.

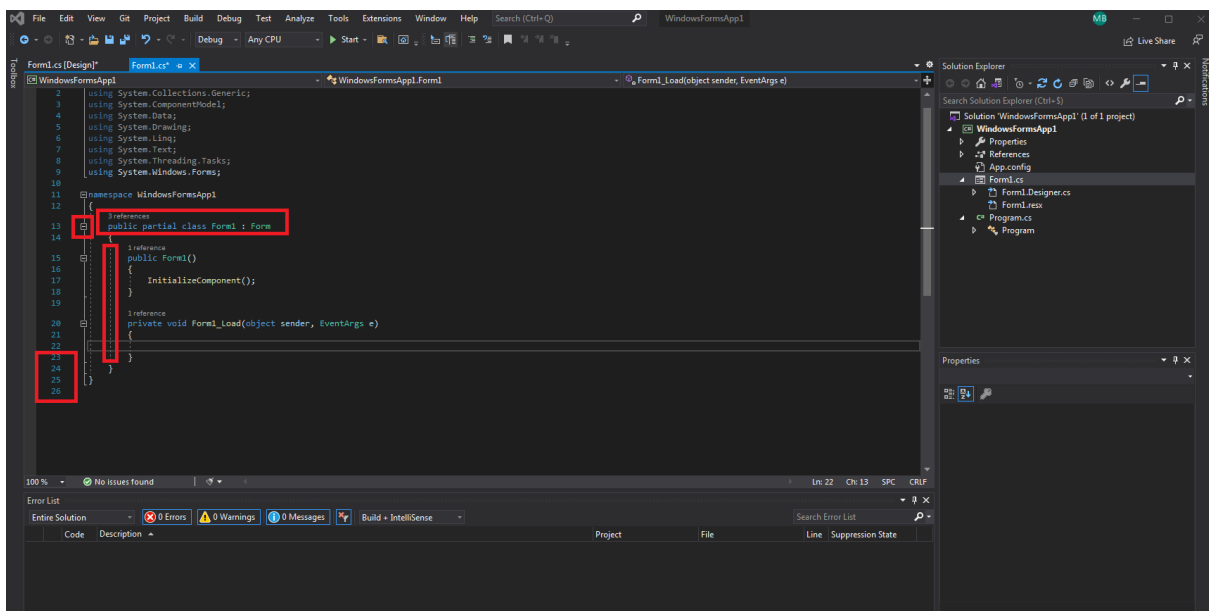


Figure 3-7 Project winforms coté Code

Le code C# de l'application s'affiche dans la fenêtre de l'éditeur, qui occupe la plus grande partie de l'espace. Notez que le texte est automatiquement colorisé pour indiquer différentes parties du code, telles que les mots-clés et les types. De plus, de petites lignes pointillées verticales dans le code indiquent les accolades qui correspondent, et les numéros de ligne aident à localiser le code plus tard. nous pouvons choisir les petits signes moins encadrés pour réduire ou développer des blocs de code. Cette fonctionnalité de contour de code permet à un utilisateur de masquer le code dont vous n'avez pas besoin, ce qui aide à minimiser l'encombrement à l'écran. Les fichiers de projet sont répertoriés sur le côté droit dans une fenêtre appelée "Solutions explorer".

Après avoir parcouru tout cela, il est maintenant possible de commencer à coder le programme pour suivre le processus souhaité.

### **3.3 Hardware Utilisé**

Après avoir présenté le logiciel. Nous allons maintenant aborder le côté matériel et présenter les outils et les machines que nous avons utilisés tels que le boîtier de commande, L'ordinateur portable et le câble RS 232.



**3.3.1 Coffret de commande de poste MT/BT**



**Figure 3-8 Coffret de commande**

C'est la machine avec laquelle nous voulons créer une interface homme-machine avec laquelle communiquer. Elle est fabriquée par l'ENSTO et elle est conforme au protocole IEC 60870-5-101.

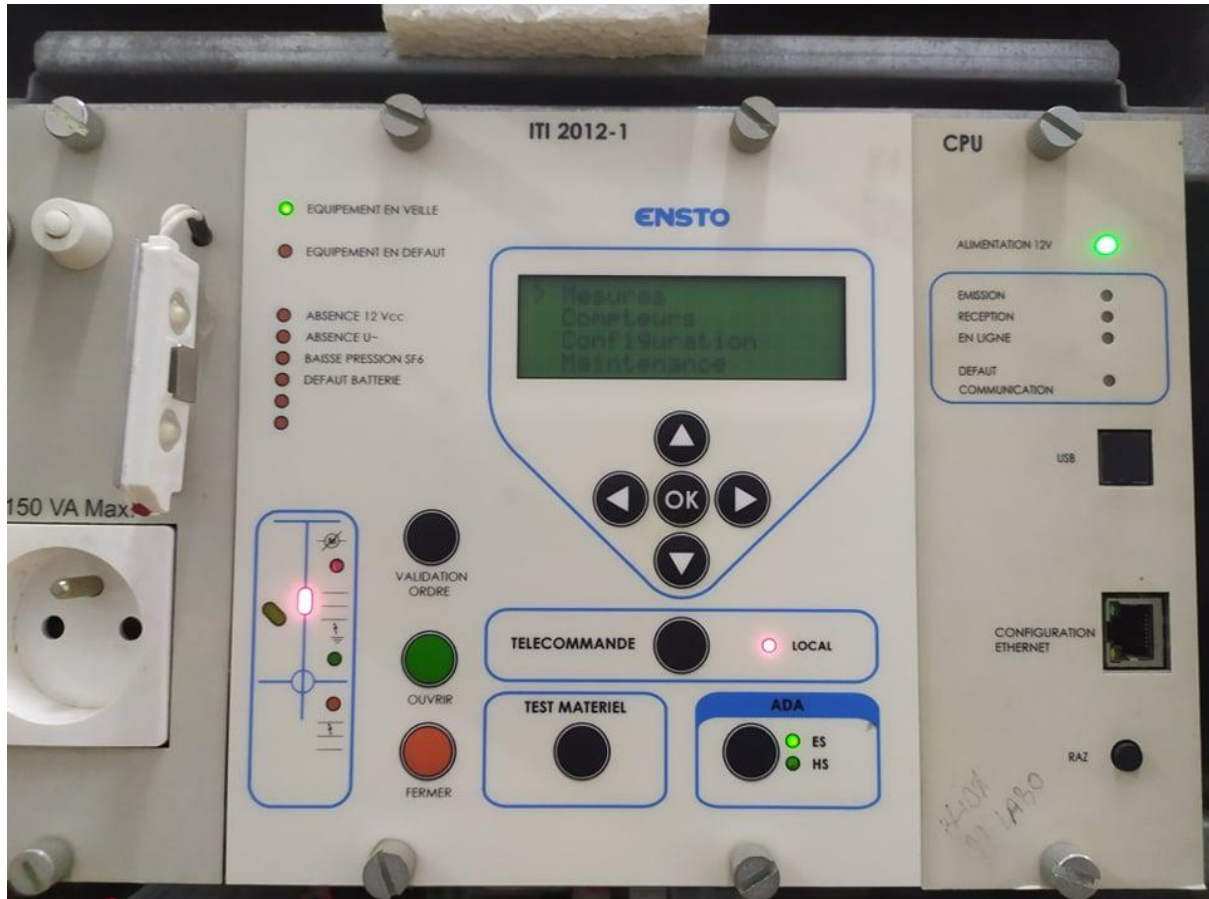


Figure 3-9 Le panneau avant de Coffret de commande

Comme le montre l'image ci-dessus (Figure 3-9) . Le panneau avant de ce coffret de commande nous montre l'état de l'interrupteur et de l'ADA et tous les autres états utiles, et nous permet également de commander manuellement l'interrupteur ou l'ADA En maintenant enfoncé le bouton "Validation d'ordre" et en appuyant sur "Ouvrir" ou "fermer".

La zone des défauts se trouve en haut à gauche, au milieu se trouve une interface qui permet aux utilisateurs de vérifier manuellement des éléments tels que les mesures de tension et d'ampérage, la date et l'heure, les compteurs et d'autres informations sur ce coffret de commande spécifique.

### 3.3.2 PC portable

Un ordinateur portable est utilisé pour héberger l'application. Fonctionnellement parlant, tout appareil compatible avec le code peut fonctionner mais pour plus de simplicité, on travaille uniquement avec un ordinateur portable.

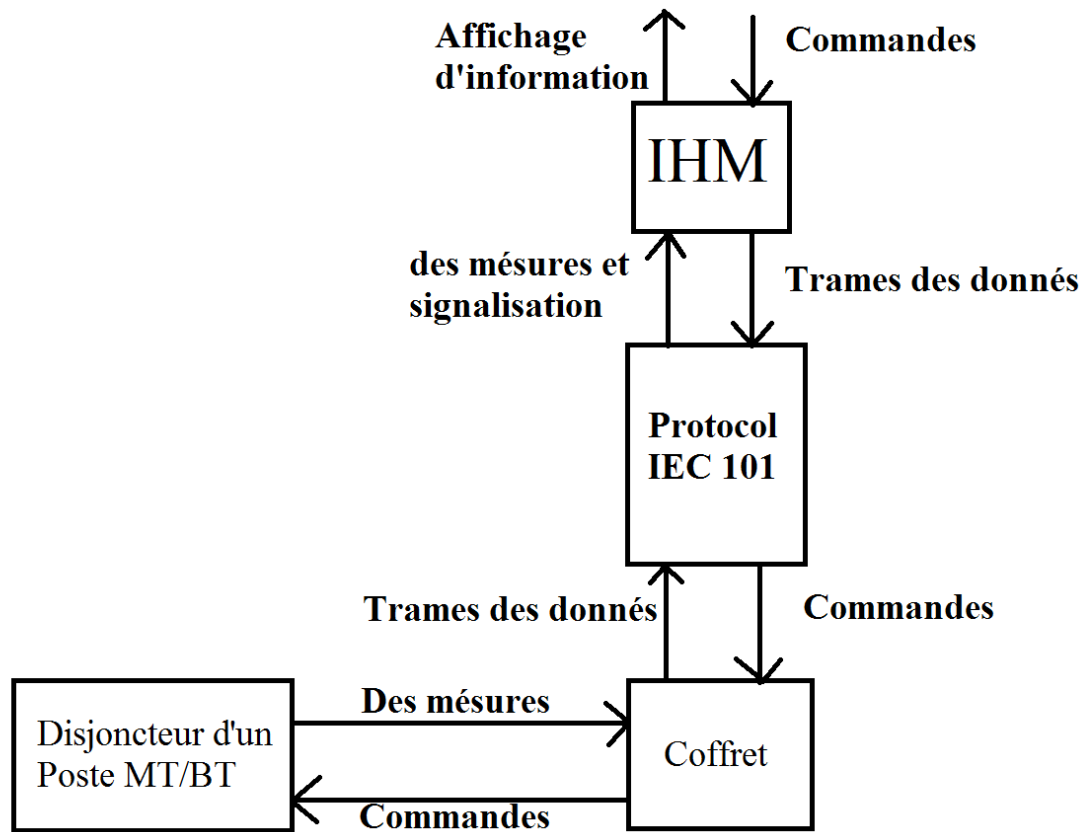
### 3.3.3 Fil RS232



**Figure 3-10 Fil RS232**

Un RS 232 est utilisé pour la communication série entre Le coffret et l'ordinateur portable (Connecté à un adaptateur RS 232/USB car l'ordinateur portable n'a pas les ports DB9 appropriés).

### 3.4 Principe de fonctionnement



**Figure 3-11 Schéma de fonctionnement**

Comme le montre le schéma ci-dessus, le but de l'IHM est d'envoyer des commandes et d'afficher des informations, puis à l'aide de la couche application, ces commandes sont transformées en trames de données qui sont traduites à l'aide du protocole IEC 101 en commandes.

A noter que la transformation en trames de données se fait au niveau de la couche application de chacun de l'IHM et du coffret et le bloc de protocole n'est que virtuel.

Le coffret lui-même envoie des commandes au disjoncteur de poste MT/BT et en reçoit des mesures analogiques.

Le disjoncteur d'un Post MT/BT est essentiellement un gros interrupteur installé dans la ligne électrique et contrôlé par le boîtier de commande qui est à son tour contrôlé par notre IHM.



Figure 3-12 Interrupteur aérien IAT Avec coffret installé

### 3.4.1 Codage du le processus

Après avoir créé le projet winforms en c#, nous pouvons commencer à coder notre programme pour suivre le processus souhaité, qui comprend :

*a -Initialisation de la station:*

[3]

Le lien de la station de contrôle établit la connexion avec le lien de l'IED en transmettant une "demande d'état du lien" qui doit être répondu par une réponse "Etat du lien" (éventuellement "Confirmer ACK" ou un "caractère unique") de la part du IED. La station de contrôle doit alors transmettre un "Reset of remote link" auquel doit répondre un "Confirm ACK" (éventuellement un "single character") qui confirme la condition de démarrage de la couche liaison du maître.

Immédiatement après cette phase de connexion :

Une requête GI est envoyée à l'IED (ASDU 100 « Commande d'interrogation » et 101 « Commande d'interrogation de compteur » selon configuration).

Une fois cette séquence d'initialisation effectuée, la station de contrôle peut démarrer l'interrogation de classe 2 pour le rapport de données.

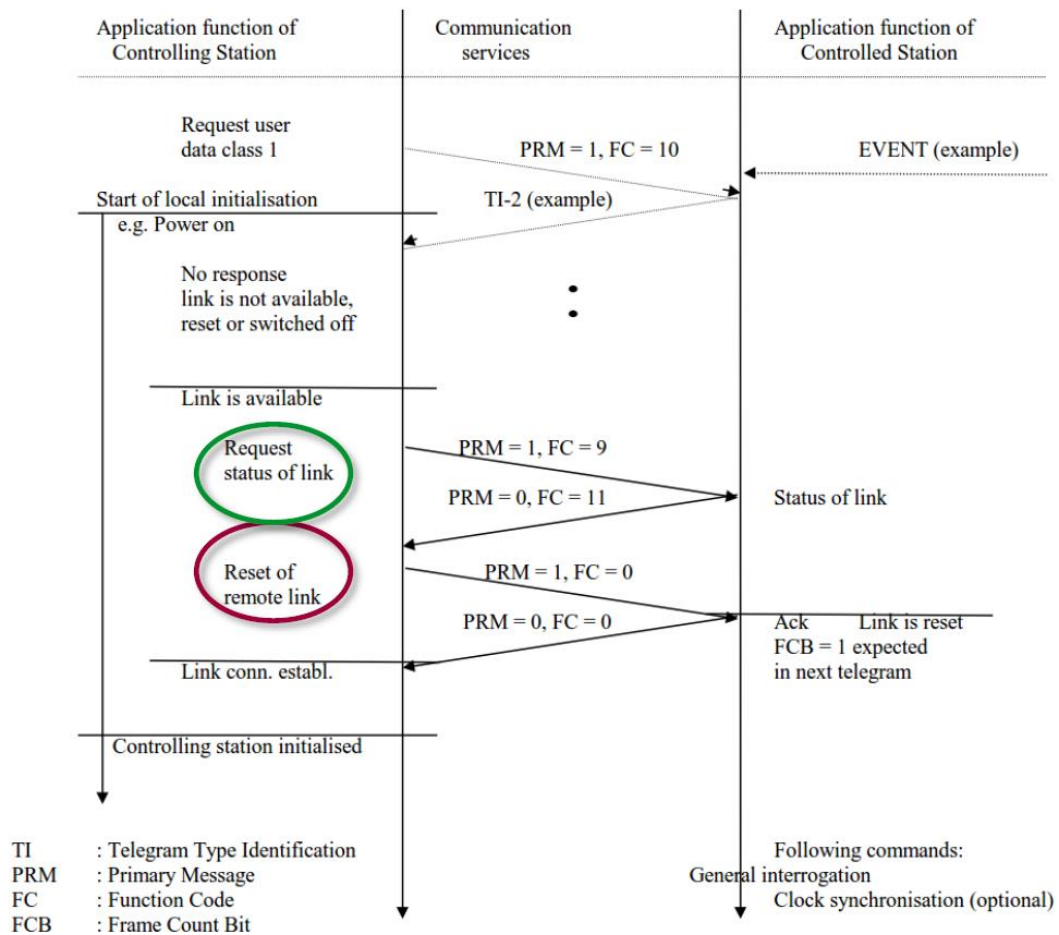


Figure 3-13 Schéma d'initialisation de la station

***b - Polling:***

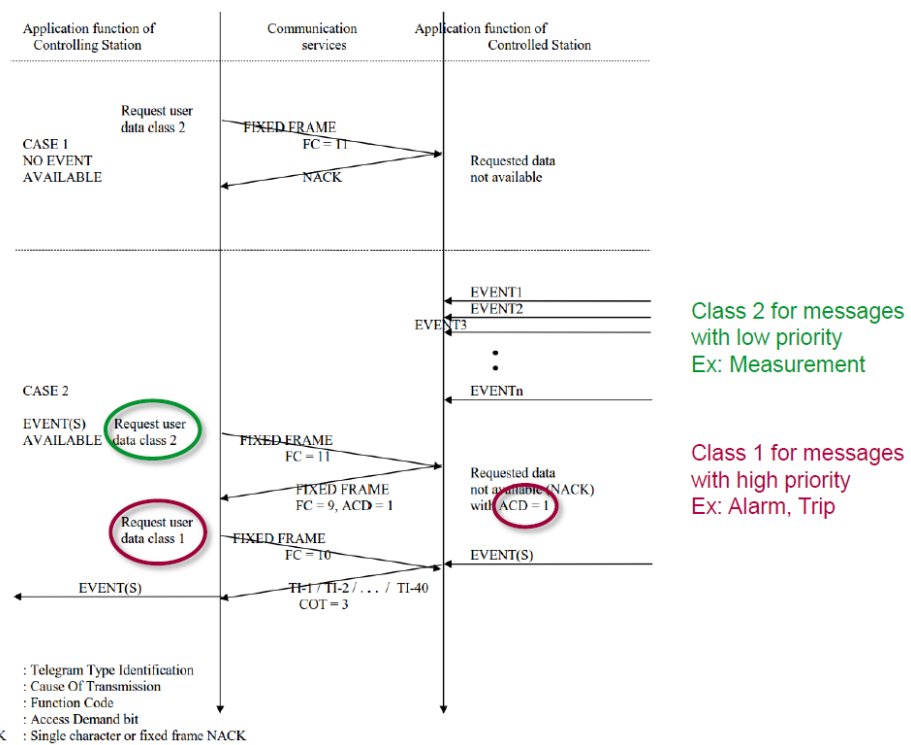
[3]

Une fois la communication entre la station contrôlée et la station de contrôle est initialisée. Elles commencent maintenant à communiquer entre eux par le "polling".

Comme le fonctionnement est en mode déséquilibré, la station contrôlée ne peut signaler aucune alarme ou mesure sans avoir été préalablement interrogée. Il faut donc utiliser le

"polling", dans lequel la station de contrôle envoie des requêtes de données de classe 2 ou 1 à caractère cyclique (période T modifiable).

Au départ, la station de contrôle envoie des requêtes de données de classe 2, si le bit ACD dans la réponse renvoyée par la station contrôlée est à zéro, elle répète la même opération jusqu'à ce que le bit ACD soit égal à 1, dans laquelle elle envoie maintenant une requête de données de classe 1, et ainsi de suite.



Schneider Electric - DUVIGNAU Jean-Gabriel – Training course T101/T104 Rev D - March 2018

**Figure 3-14 Schéma de polling**

*c - Exécuter des commandes et recevoir des informations*

[3]

Puisque notre programme fonctionne comme la station de contrôle. Il doit pouvoir envoyer des commandes et afficher les résultats et la télémétrie dans un format facilement digestible.

Grâce à notre connaissance du protocole IEC 60870-101-5 (chapitre 2), nous pouvons coder notre programme pour envoyer les ASDU correct et convertir les trames reçues en informations utiles.

Les commandes sont utilisées dans les systèmes de télé conduite pour provoquer un changement d'état d'un équipement opérationnel.

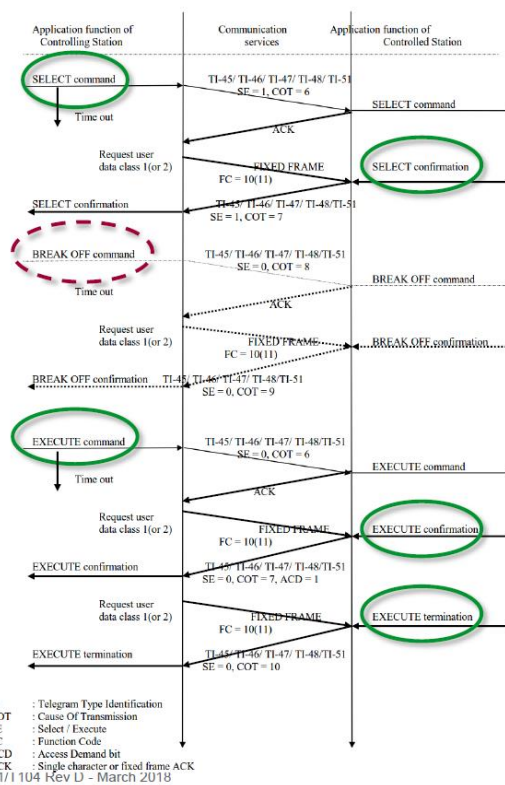
Il existe deux procédures standard pour la transmission des commandes :

- Sélection de la commande avant l'exécution
- Commande d'exécution directe

### ● Unbalanced Mode

COT 6 : Activation

COT 8 : Desactivation



Schneider Electric - DUVIGNAU Jean-Gabriel – Training course T107/1104 Rev U - March 2018

87

**Figure 3-15 Schéma de la structure d'une commande**

Le système SONELGAZ SCADA ne transmet généralement que des commandes par la méthode "Sélectionner avant d'exécuter". C'est donc la seule méthode de transmission des commande utilisé dans ce projet.



### 3.5 Conclusion

Dans ce chapitre, Nous avons présenté les différentes parties nécessaires à la création de notre IHM, et les processus qui doivent être codés pour que notre IHM suivre conformément le protocole IEC 101 pour le contrôle et la supervision de la station contrôlé. Nous avons aussi donné un aperçu des blocs utilisés lors de la programmation.

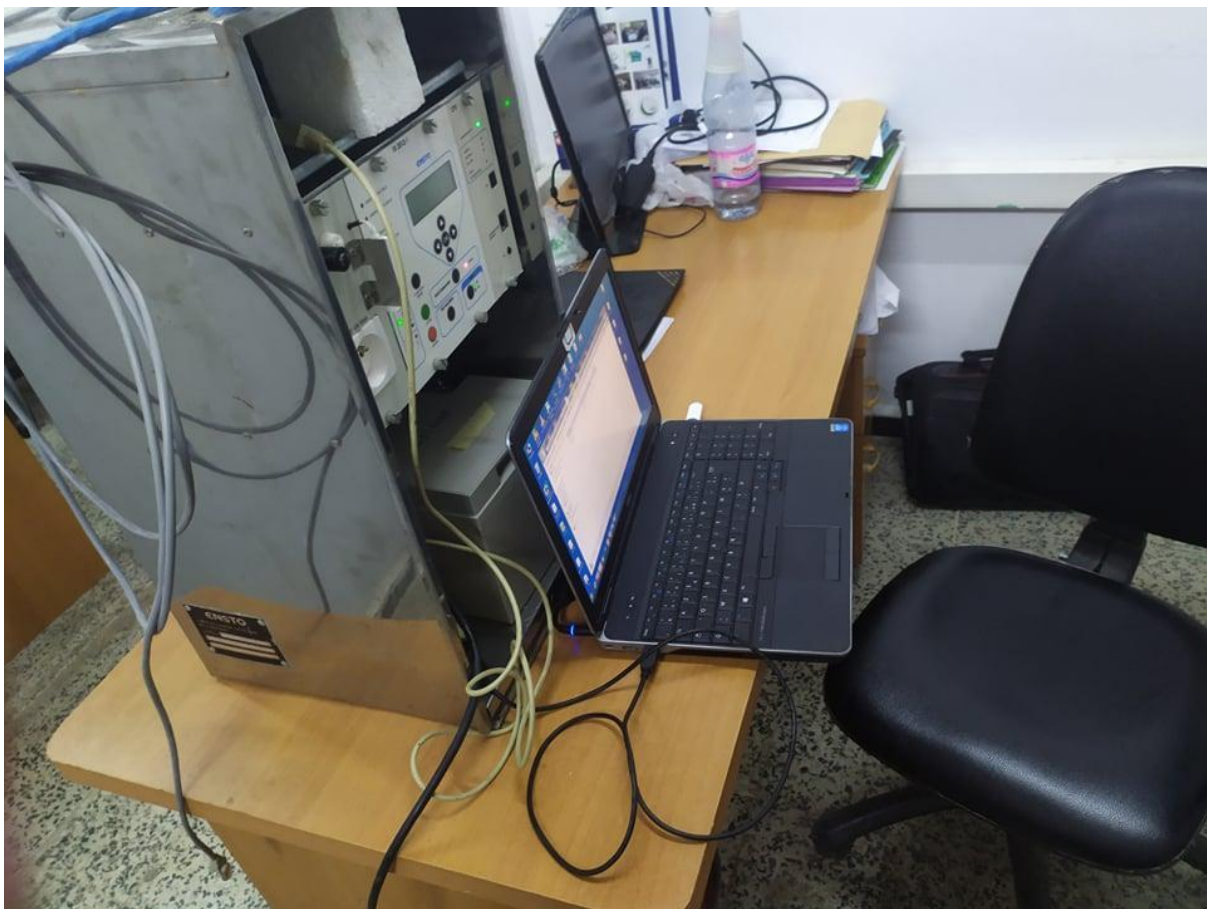
La création d'une Interface Homme Machine (IHM) exige non seulement une bonne connaissance de la procédure et des étapes de la création de la supervision, mais aussi du langage avec lequel est programmé afin d'établir une communication correcte.

Dans le prochain chapitre, nous montrerons les résultats de notre travail.

**Chapitre 4 Présentation de l'IHM créé**  
**Réalisation de la supervision de la station**

## 4.1 Introduction

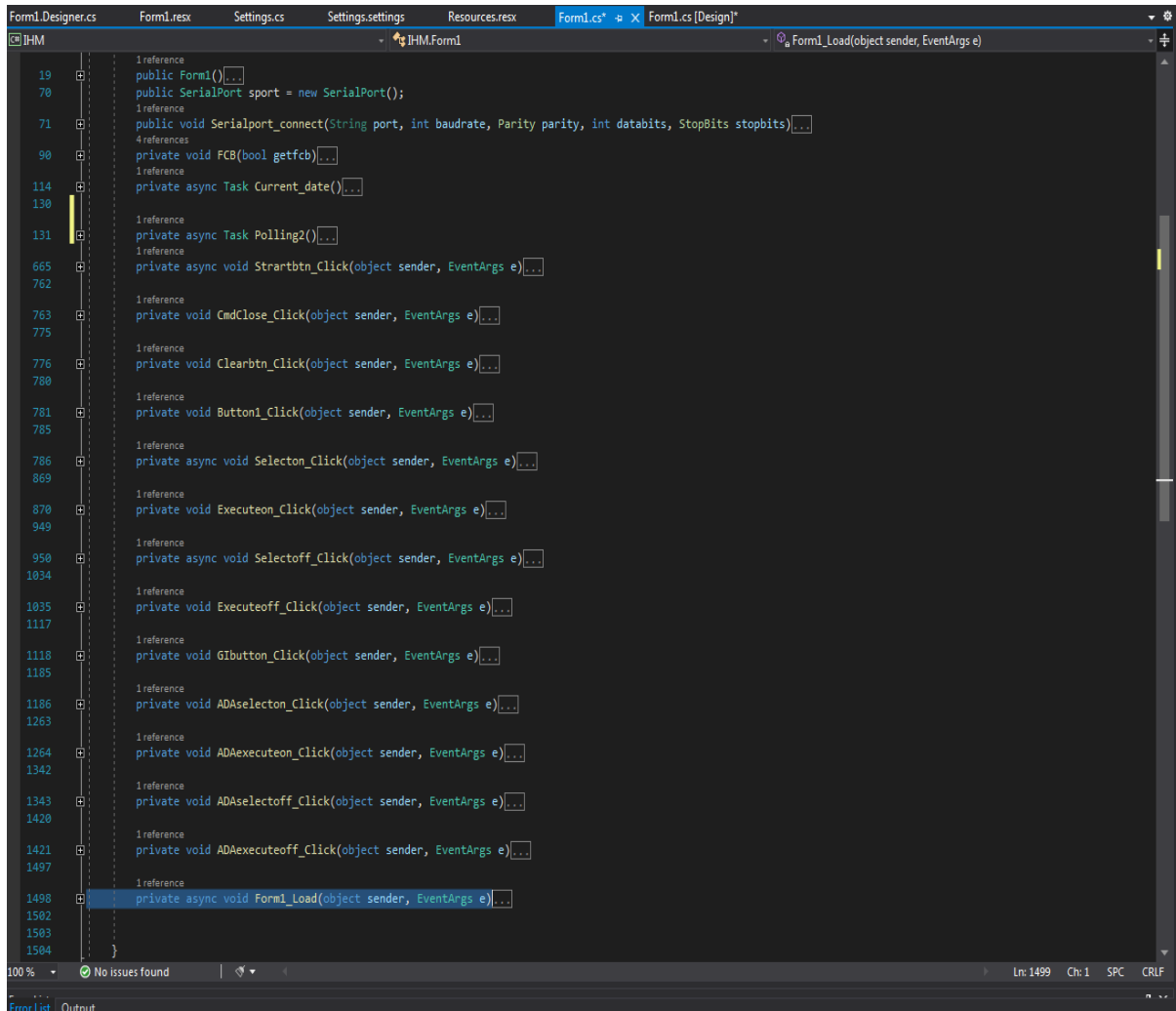
Lorsque la complexité des installations augmente et que les machines doivent répondre à des spécifications de fonctionnalité toujours plus sévères, l'opérateur a besoin d'un maximum de transparence pour la supervision et le contrôle de ces installations. Cette transparence s'obtient au moyen de l'interface homme machine (IHM). Après avoir présenté tous les logiciels et matériels utilisés pour créer cette interface. Et après avoir établi le processus requis pour être codé, nous allons dans ce chapitre présenter le programme résultant réalisé après une quantité importante de codage (environ 1500 lignes de code à plus précis).



**Figure 4-1** Le poste de travail

C'est là (Figure 4-1) que le projet a été développé et testé. Et ce les pages suivantes sont toutes les résultats dudit projet.

## 4.2 À la découverte de l'IHM



```
19 public Form1()...
70 public SerialPort sport = new SerialPort();
71 public void Serialport_connect(String port, int baudrate, Parity parity, int databits, StopBits stopbits)...
90 private void FCB(bool getfcb)...
114 private async Task Current_date()...
130
131 private async Task Polling2()...
132
665 private async void Strartbtn_Click(object sender, EventArgs e)...
762
763 private void CmdClose_Click(object sender, EventArgs e)...
775
776 private void Clearbtn_Click(object sender, EventArgs e)...
780
781 private void Button1_Click(object sender, EventArgs e)...
785
786 private async void Selecton_Click(object sender, EventArgs e)...
869
870 private void Executeon_Click(object sender, EventArgs e)...
949
950 private async void Selectoff_Click(object sender, EventArgs e)...
1034
1035 private void Executeoff_Click(object sender, EventArgs e)...
1117
1118 private void GIbutton_Click(object sender, EventArgs e)...
1185
1186 private void ADAselcton_Click(object sender, EventArgs e)...
1263
1264 private void ADAexecuteon_Click(object sender, EventArgs e)...
1342
1343 private void ADAselctoff_Click(object sender, EventArgs e)...
1420
1421 private void ADAexecuteoff_Click(object sender, EventArgs e)...
1497
1498 private async void Form1_Load(object sender, EventArgs e)...
1502
1503
1504 }
```

Figure 4-2 Le code source de l'IHM

Chaque ligne de code dans l'image ci-dessus comprend une certaine fonctions utilisées, le tout compressé en une seule ligne pour faciliter la présentation.

L'image ci-dessous (Figure 4-3) le programme créé dans ce projet.

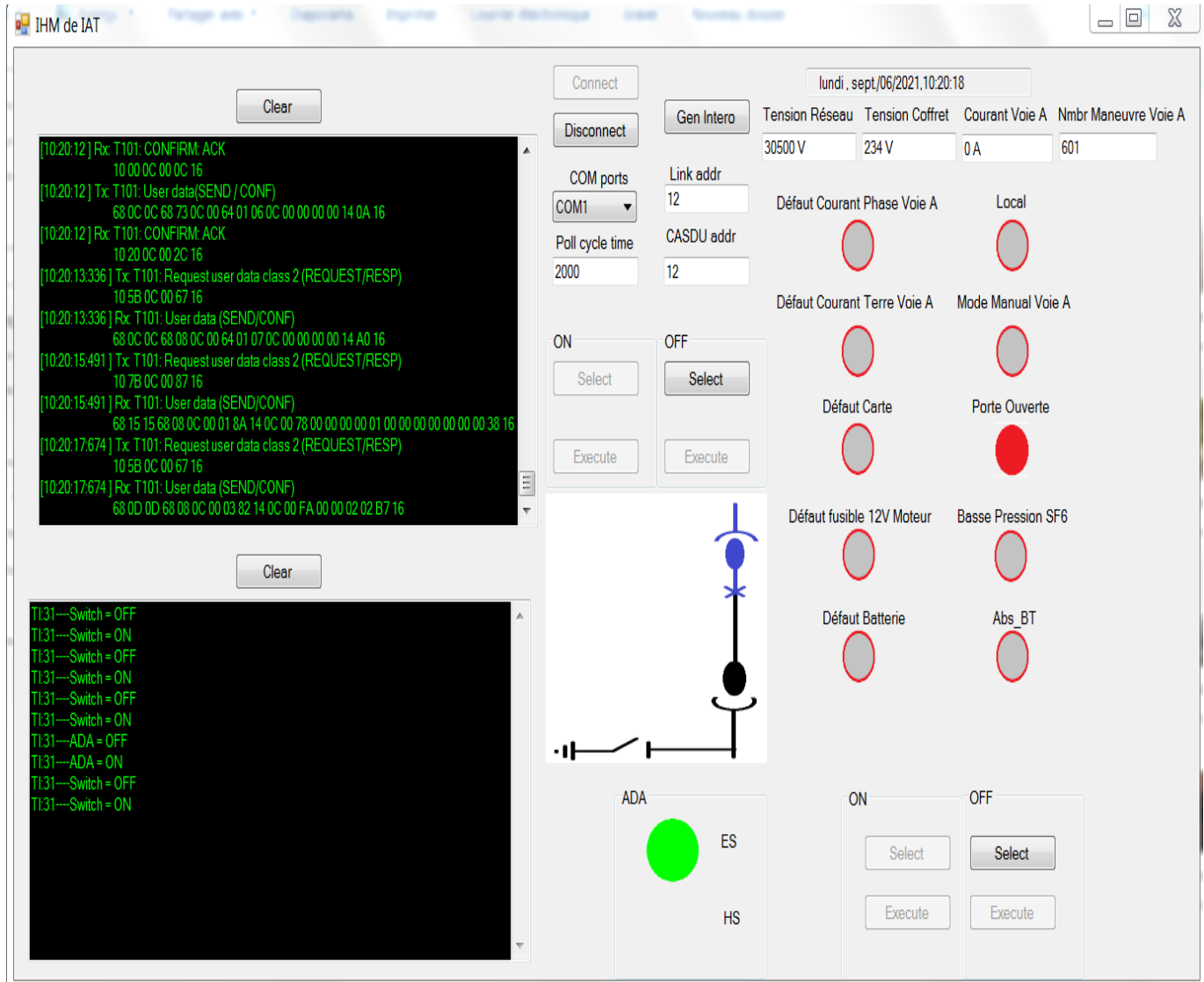
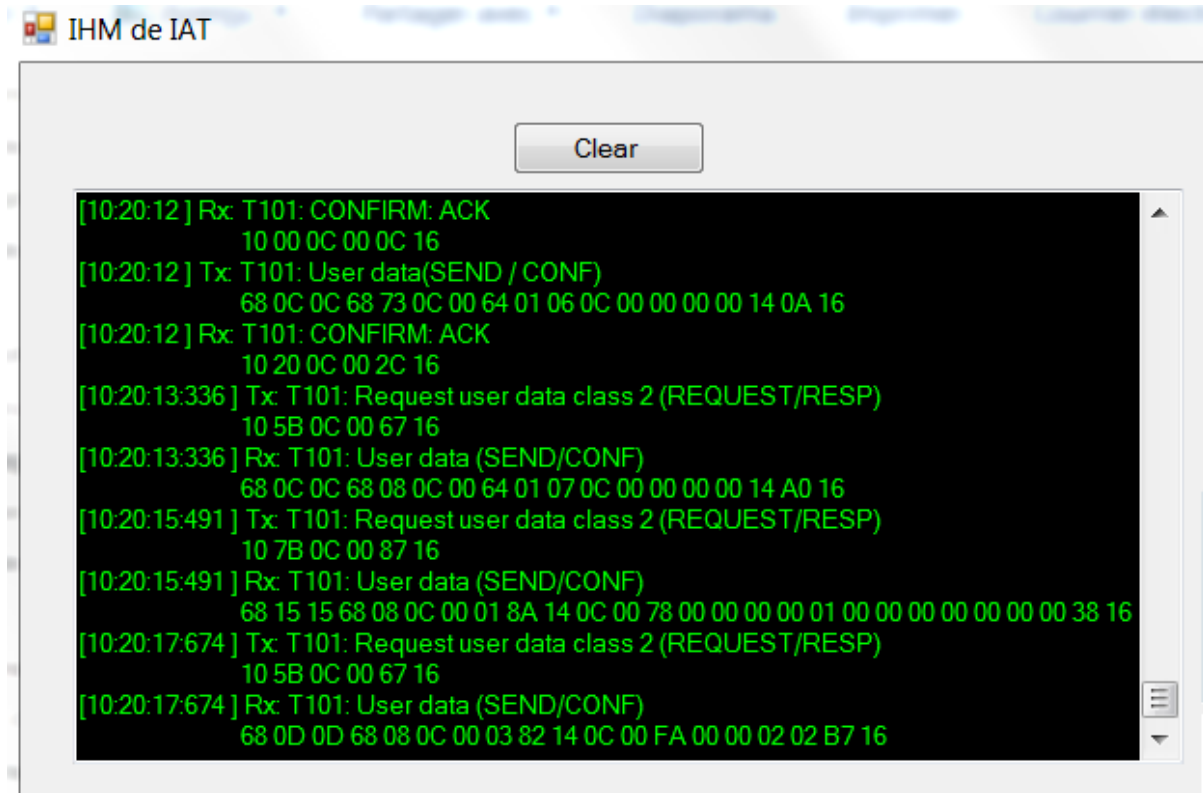


Figure 4-3 L'IHM créé

À la première vue, cela peut apparaître comme un fouillis de symboles et de mots, mais je vais décomposer ce que fait chaque partie.

## 4.2.1 La boîte noire supérieure

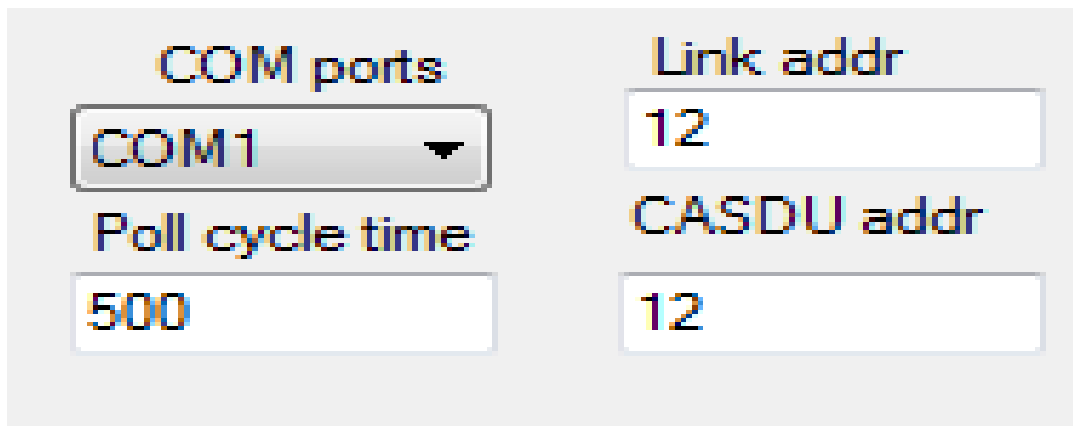


**Figure 4-4** La boîte noire supérieure

Cette boîte est faite pour afficher les trames (données) envoyées et reçues par l'IHM sous sa forme hexadécimale, elle est utile pour l'analyse et la recherche de bugs et d'erreurs. Cela a également été très utile lors de la création de l'IHM.

Le bouton "Clear" permet d'effacer tout le texte existant actuellement dans la zone de texte.

## 4.2.2 Les paramètres



The image shows a configuration window with four parameters arranged in a 2x2 grid:

<b>COM ports</b> COM1	<b>Link addr</b> 12
<b>Poll cycle time</b> 500	<b>CASDU addr</b> 12

Figure 4-5 Les paramètres

Celui-ci est assez explicite. c'est là que nous choisissons les paramètres de notre IHM.

"Link addr" et "CASDU addr" représentent L'adresse de liaison et l'adresse CASDU respectivement , Ils sont déjà expliquées au chapitre 2. Ce sont essentiellement des paramètres système pour notre coffret de commande spécifique.

"COM ports" sont l'endroit où nous choisissons le port utilisé pour se connecter avec le boîtier de commande. Les ports COM disponibles sont automatiquement détectés sur la machine utilisée.

Le «Poll cycle time» représente la période T du « Polling » en millisecondes.

### 4.2.3 La partie de l'interrupteur

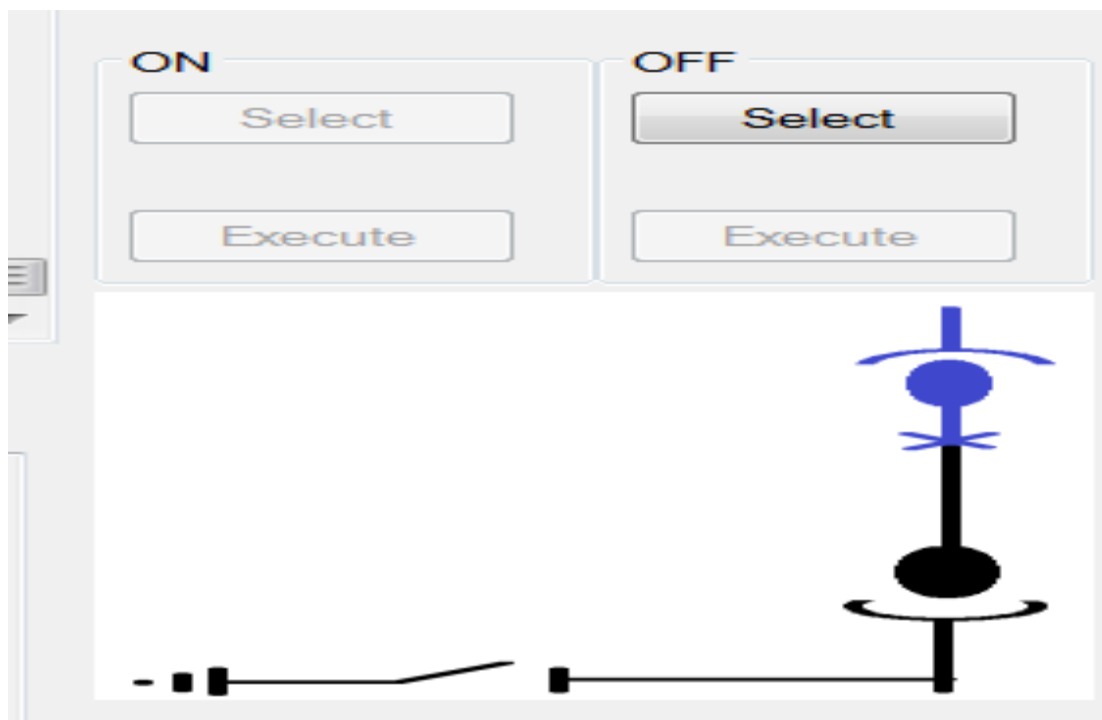
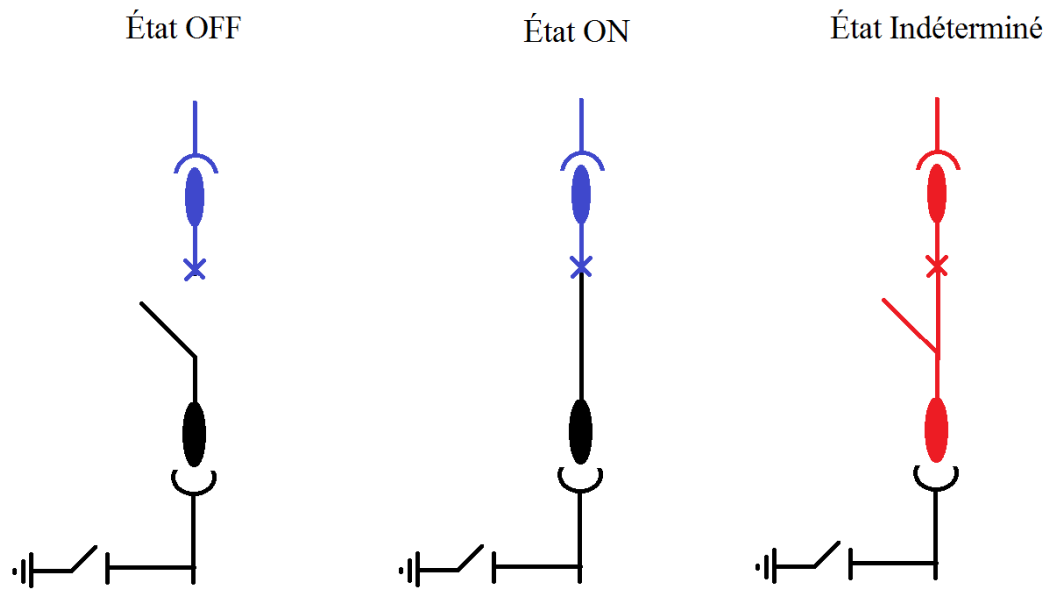


Figure 4-6 La partie interrupteur

C'est là que nous pouvons contrôler le commutateur IAT et afficher son état actuel. Dans le système SCADA de SONELGAZ, toutes les commandes sont "Sélectionner avant exécution", C'est pourquoi nous avons deux boutons pour allumer et éteindre l'interrupteur.

La figure 4.6, représente les boutons d'affichage des états de l'interrupteur, il peut avoir trois états : ON, OFF, Indéterminé.

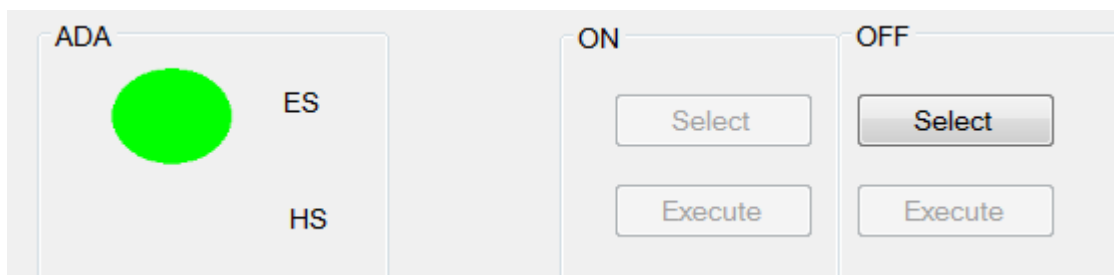




**Figure 4-7 Les différentes représentations de l'interrupteur**

Ces représentations suivent la norme IEC qui m'a été indiquée par l'ingénieur en chef de la salle SCADA de Sonelgaz.

#### 4.2.4 ADA (automate de défaut)



**Figure 4-8 La partie ADA**

ADA est un système de protection qui éteint automatiquement l'interrupteur en cas d'erreur critique. Ce système de protection peut être activé et désactivé et son état est également affiché.

ES: " En Service"

HS: " Hors Service"

### 4.2.5 Les défauts

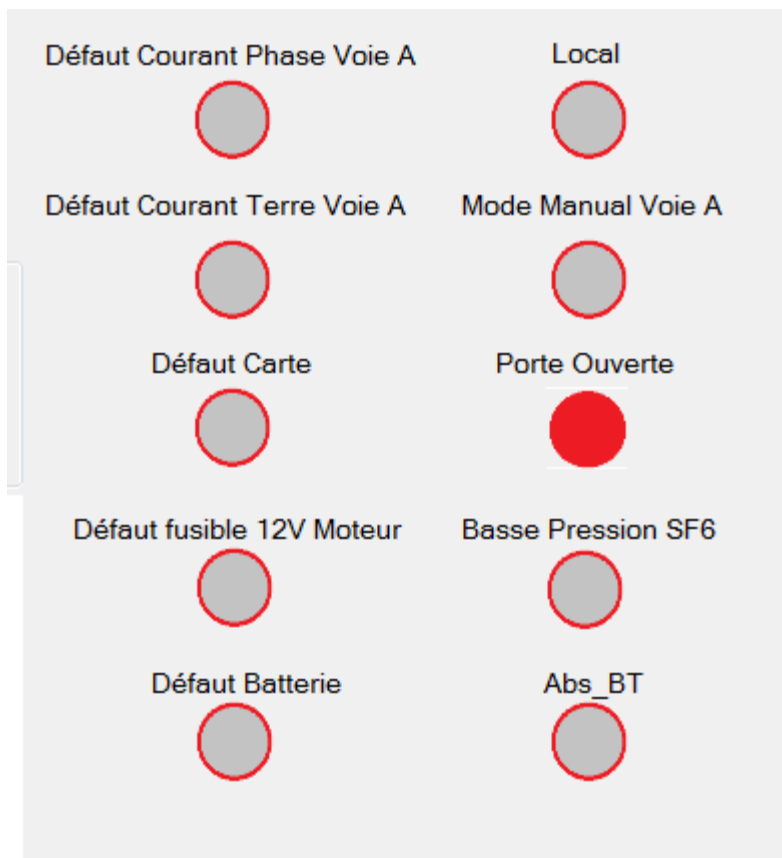


Figure 4-9 Les défauts

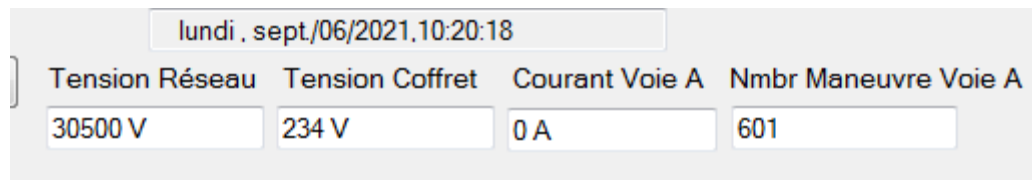
Ce sont les erreurs que le boîtier de contrôle surveille. Ils s'allument en rouge lorsqu'ils sont actifs, certaines erreurs ne sont pas des erreurs à proprement parler. Par exemple:

**-Local** : représente si le coffret de commande est contrôlé manuellement ou à distance ( ON = manuel , OFF = à distance).

-**Porte ouverte** : indique si la porte du coffret de commande est ouverte ou non.( ON = Ouverte

OFF = Fermée).

## 4.2.6 Les mesures



**Figure 4-10 Les mesures**

C'est là que 4 mesures sont affichées (avec l'heure et la date actuelles au-dessus).

**a -Tension Réseau :**

Il s'agit de la tension de la ligne électrique à laquelle notre IAT est connecté.

**b -Tension Coffret :**

Il s'agit de la tension fournie au boîtier de commande et mesurée par celui-ci.

Notez que le coffret de commande est alimenté par la ligne électrique via un transformateur avec un certain coefficient. Cela signifie que le coffret de commande n'a qu'à mesurer la tension à l'intérieur de lui-même et à la multiplier par le coefficient du transformateur pour obtenir la tension de ligne.

**c -Courant Voie A**

Représente la valeur du courant. puisque cela a été fait dans un laboratoire sans que le coffret de commande soit connecté à quoi que ce soit, Il reste à zéro.

**d -Nnbr Manœuvre voie A**

Cela représente un compteur qui monte à chaque fois que le commutateur est tourné. Ceci est utile à des fins de suivi, telles que des rapports annuels ou trimestriels.

## 4.2.7 Le reste des boutons

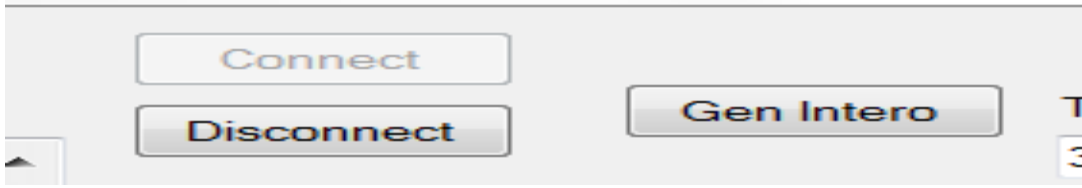


Figure 4-11 Le reste des boutons

### *a* -Connect

Ce bouton ouvre le port COM et commence immédiatement le processus d'initialisation qui marque le début d'une bonne communication entre la station contrôlée et la station de contrôle en utilisant le protocole IEC 101.

### *b* -Disconnect

Ce bouton ferme le port COM et met fin à la communication entre les deux stations.

Notez que toutes les trames envoyées mais n'ayant pas été lues en raison de la fermeture du port COM seront toujours mises en mémoire tampon et lues à la prochaine ouverture du port.

### *c* -Gen Intero

A l'appui sur ce bouton, Notre programme envoie une requête d'interrogation générale (ASDU 100) qui est reçue par le coffret de commande qui renvoie à son tour des informations sur l'état de ses pièces (L'interrupteur, L'ADA, Les erreurs, Les mesures).

Ce coffret est programmé pour envoyer ces mesures et informations toutes les 10 secondes ou plus à la station de contrôle (10 est le plus bas possible). Ce bouton est donc placé lorsque nous voulons envoyer une demande d'interrogation générale (ASDU 100) et recevoir des mesures avant que la période de 10 secondes (minimum) ne se soit écoulée.

## 4.2.8 La boîte noire inférieure



**Figure 4-12** La boîte noire inférieure

Cette zone de texte est l'endroit où l'historique d'utilisation est stocké, comme vous pouvez le voir, l'interrupteur a été activé et désactivé plusieurs fois. alors que l'ADA n'a été éteint et allumé qu'une seule fois

TI 31 représente le type d'ASDU à partir duquel ces informations ont été lues.

Notez qu'à l'avenir, il convient de faire en sorte que chaque entrée soit horodatée.

## 4.3 Conclusion

Dans ce chapitre, nous avons présenté les résultats de notre travail et exploré et expliqué les différentes parties de celui-ci et leurs fonctions.

Avec cette IHM terminée, il est désormais possible de contrôler et de surveiller ce coffret de commande ENSTO ou tout coffret de commande conforme au protocole IEC 60870-5-101 via un poste de commande à distance connecté par une communication Série.

## Conclusion générale

Ce projet de fin d'études était très intéressant dans l'ensemble. Il a permis de découvrir le monde professionnel et d'apprendre de nouvelles choses sur Sonelgaz et les réseaux électriques en général.

Nous avons créé une interface homme-machine fonctionnelle pour un boîtier de contrôle de marque ENSTO utilisé dans les IAT

Nous avons commencé cette thèse en introduisant des généralités sur le matériel sur lequel nous travaillons/avec lequel nous travaillons. Nous avons ensuite exploré un peu en profondeur le protocole CEI 60870-5-101, après quoi nous avons exploré le processus de création de l'interface homme-machine et les étapes à suivre tout en présentant le matériel et les logiciels utilisés pour créer l'IHM. Enfin, nous avons présenté l'IHM créée et expliqué ses différentes parties et leurs utilisations.

Avec la création de cette IHM, nous avons établi une communication série appropriée entre un boîtier de commande conforme à la norme IEC 101 et une machine secondaire (dans notre cas un ordinateur portable) qui utilise l'IHM. Avec cela, une station de contrôle peut être établie pour se connecter en série à un ou plusieurs boîtiers de contrôle et les superviser et les contrôler à distance sans avoir à dépendre de la salle de contrôle SCADA principale qui communique par antenne et ne doit pas être limitée par le débit de la bande passante.

En utilisant ce que nous avons fait et l'expertise en programmation apprise, il est possible en théorie de développer cela et d'essayer de concevoir et de créer un système SCADA fabriqué en Algérie. beaucoup trop pour embaucher des fabricants étrangers pour construire leurs systèmes SCADA et payer pour les mettre à jour/maintenir.

# Bibliographie

[1] Kasmi Athmane & Fellag Mohammed, , diplôme de Master : " Télécommande et supervision d'un poste de transformation électrique", Université de Saad Dahleb, Blida Année Universitaire 2018-2019.

[2] Norme Internationale IEC 6870-5-101 Première édition 1995-11.

[3] Training course-protocols communication-IEC 60870-5-101/IEC 60870-5-104,Schneider Electric - DUVIGNAU Jean-Gabriel – Training course T101/T104 Rev D - March 2018.

[4] <https://docs.microsoft.com/en-us/visualstudio/get-started/csharp/visual-studio-ide?view=vs-2019>.

[5] <https://www.pluralsight.com/blog/software-development/everything-you-need-to-know-about-c->