

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mouloud MAMMERI, Tizi-Ouzou



Faculté de Génie Electrique et d'Informatique
Département d'Automatique

Mémoire de Fin d'Etudes

En vue de l'obtention du diplôme

D'Ingénieur d'Etat en Automatique

Thème

**Conception d'un système de supervision pour un
bâtiment intelligent.**

Proposé par : M^r A. YOUNSI

Dirigé par : M^r BENSIDHOM

Soutenu le : 15 / 07 /2010

Présenté par :

M^{elle} BOUKHELF Ouerdia

M^{elle} DJELILI Lilya

Promotion 2010

Ce travail a été préparé à :Shneider Electric(Route d'Ouled Fayet Dely Ibrahim Alger)

REMERCEMENTS

Nous remercions avant tout, le bon « DIEU » tout puissant de nous donner la santé, le courage et la volonté pour finir ce travail.

Nos remerciements s'adressent à nos familles et surtout nos parents.

Nous tenons à remercier tous ceux qui nous ont aidés à réaliser ce travail notamment notre promoteur M^r BENSIDHOUM.

Nos remerciements les plus chaleureux s'adressent à M^r CHARIF-M pour le temps qu'il nous a consacré, son aide, ces orientations et encouragements et surtout pour sa patience tout au long de ce travail.

Nos gratitudes s'adressent à M^r CHOUARBI-H, le co-promoteur A-YOUNSI, ainsi qu'à M^r MOSSAB-A.

Nous remercions les membres de jury, qui nous feront l'honneur de participer au jugement de ce travail.

Enfin, nous tenons à remercier, tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.

DEDICACE

Je dédie ce modeste travail :

*A mes chers parents pour leurs sacrifices
qu'ils ont fournis pour mes études, surtout ma mère qui est la source de
Résistance et d'espérance, que dieu les protège.*

A mes frères Ali et sa femme Nabila et leur fils Abdou

A Moustapha et sa femme Naima.

A Hamid et boudjamma que j'aime beaucoup.

*A mes sœurs : Noura, Malika et son mari Djafar et leurs enfants Hiba, HOda, Ilias et
Abdou*

A Fatiha et son mari Sid Alli et leur fille Nour

*A ma tante Ouerdia et son mari brahim et leur fils
Didine et Mira*

A mes amis surtout Hanane, Karima, Samira ...

*A tous mes cousins et cousines, particulièrement à Ouerda
Et Fatima.*

A ma binôme : Lilya

A tous ceux qui me sont chers.

Ouerdia

DEDICACES

Je dédie ce modeste travail :

A tous ceux qui me sont chers.

A ma très chère grand-mère au grand cœur rempli d'amour FATIMA qui ne quitte jamais mes pensées, Que dieu te garde pour nous qui t'aimons tant.

Prompte rétablissement Mani chérie.

A Celle à qui mon cœur depuis ma naissance n'a pu qu'éprouver qu'amour et reconnaissance, à celle qui a donné un sens à mon existence À ma chère Mère.

A mon père, pour son amour et son soutien moral depuis mon enfance.

Père merci.

A ma petite princesse Mélina que j'aime tant.

A mon adorable frère et très chère sœurs, pour leurs soutient tout au long de mon PFE.

A ma chère Ouerdia.

A toute ma famille, et le reste de mes amis.

Lilya

Sommaire

Introduction générale.....	01
Plan de Travail	02
Chapitre I : Le Bâtiment Intelligent	
Introduction.....	03
I-1 Le bâtiment intelligent	03
I-1-1 Définition du bâtiment intelligent.....	03
I-1-2 Avantages des bâtiments intelligents	05
I-2 Gestion Technique Centralisée (GTC) et Gestion Technique des Bâtiments (GTB).....	06
I-2-1 Définition de la GTC.....	06
I-2-2 Définition de la GTB.....	06
I-3 Principales boucles de régulation et fonctions dans les bâtiments Intelligents.....	07
I-3-1 Contrôle de l'éclairage	07
I-3-2 Contrôle de Chauffage, ventilation et climatisation (CVC) et qualité de l'air intérieur	07
I-3-3 Sécurité et Alarmes.....	08
I-4 Les systèmes CVC	09
I-4-1 Système de climatisation.....	09
I-4-2 Ventilation	09
I-5 Rôle de la GTB dans la promotion de l'efficacité énergétique	12
I-5-1 Avantages de l'efficacité énergétique	13
I-6 Equipements dans la GTB.....	14
I-7 Le protocole de communication	14
I-7-1 Définition de Protocole MODBUS.....	15
I-7-2 Norme OSI	17
I-7-3 Etude du réseau.....	17
I-7-3-1 Ethernet	18
I-7-3-2 Ethernet TCP/IP.....	19
I-7-3-3 Le protocole Modbus sur TCP/IP.....	20
Conclusion	22

Chapitre II : Présentation du Logiciel Unity Pro

Introduction.....	23
II-1 Présentation de l'interface utilisateur d'Unity Pro	24
II-2 Création d'un nouveau projet sous Unity Pro	26
II-2-1 Configuration matérielle	26
II-2-1-1 Choix d'une CPU	26
II-2-1-2 Configuration d'un rack	27
II-2-1-3 Choix du module d'alimentation	28
II-2-1-4 Choix de modules d'Entrée/sortie, de communication et de modules métiers	29
II-2-2 Configuration d'un réseau	31
II-2-3 Définition et édition des variables	33
II-2-3-1 Définitions	33
II-2-3-2 Adressage des variables dans Unity Pro	34
II-3 Exemple d'application	34
II-3-1 Cahier de charge	34
II-3-2 Traduction du GRAFCET en équations logiques	37
II-3-3 Programmation de l'exemple en FBD	37
Conclusion	39

Chapitre III : Introduction au Logiciel SCADA Vijeo Citect

Introduction.....	40
III-1 Définition de la supervision.....	40
III-2 Fonctions de supervision	40
III -3 Avantage de la supervision	40
III-4 Présentation de logiciel utilisé (Vegio Citect)	41
III - 5 Caractéristiques et avantages du logiciel Vijeo Citect	41
III-6 Outils et composantes de Vijeo Citect	42
III-6-1 Outils de configuration	42
III-6-1-1 Explorateur Citect	42
III-6-1-2 Editeur de projet	42
III-6-1-3 Editeur graphique Citect	43
III-6-2 Outils d'exécution	43
III-6-2-1 Vijeo Citect Client Web	43

III-6-2-2 Gestionnaire d'exécution	43
III-6-2-3 Analyseur de processus.....	43
III-6-3 Les pilotes	44
III-7 Eléments d'un projet	44
III-7-1 Eléments graphiques.....	45
III-7-2 Les tags (Variables).....	46
III-7-3 Les Alarmes	46
III-7-4 Éléments système	46
III-7-5 Eléments de communication	47
III-7-6 Eléments de serveurs d'E/S	48
III-8 Création et configuration d'un nouveau projet sous Vijeo Citect.....	48
III-8-1 Création d'un nouveau projet.....	49
III-8-2 Configuration du nouveau projet	49
III-8-2-1 Définition de cluster et des serveurs.....	49
III-8-2-2 Exécution de l'Assistant de Configuration de Poste	51
III-9 Les graphiques	52
III-9-1 Création de la page principale.....	53
III-9-2 Création des génies et des super génies.....	56
III-9-2-1 Création des génies des capteurs de mouvement	57
III-9-2-2Création des génies et des supers génies des boutons poussoirs.....	61
III-9-2-3Création du super génie	62
Conclusion	66
IV-1 : Introduction et descriptif général de l'application.....	67
IV-1-1 Description de la tour	67
IV-2 Description de l'installation électrique.....	69
IV-2-1 Le Poste de livraison	69
IV-2-2 Les transformateurs	71
IV-2-3 Le TGBT 1	71
IV-2-4 Le TGBT 2.....	73
IV-2-5 Le Tableau divisionnaire d'éclairage	73
IV-2-6 Le Tableau Général du Groupe Electrogène TGGE	74
IV-2-6 Les alarmes	75
IV-3 description du système d'éclairage	77

IV-3-1 Description du système d'éclairage intérieur.....	77
IV-3-1-1 Gestion de l'éclairage en fonction de la présence et de l'apport en éclairage naturelle	79
IV-3-2 Descriptif du système d'éclairage extérieur.....	82
IV-4 Centrales de Traitement d'Air (CTA)	84
IV-4-1 Présentation	84
Conclusion	87
Conclusion générale	88
Bibliographie	
Annexe	

Liste des figures

- Figure I-1 : Niveaux fondamentaux du bâtiment intelligent
- Figure I-2 : Exemple d'une CTA appliquée à un local
- Figure I-3 : Exemple d'un système VAV mono-gaine
- Figure I-4 : Courbes caractéristiques d'un ventilateur centrifuge à aubes inclinées vers l'arrière
- Figure I-5 : Point de fonctionnement d'un ventilateur
- Figure I-6 : Le principe mono maître du protocole MODBUS
- Figure I-7 : Principe des échanges MODBUS
- Figure I-8 : Les sept couches du modèle OSI
- Figure I-9 : Service et protocoles associés à Ethernet
- Figure I-10 : Communication avec Ethernet TCP/IP
- Figure I-11 : Principe d'encapsulation
- Figure I-12 : Communication via Ethernet TCP/IP et Modbus
- Figure II-1 : Interface utilisateur
- Figure II-2 : Interface utilisateur (suite)
- Figure II-3 : les différentes vues du navigateur de projet
- Figure II-4 : Vue structurelle du navigateur de projet
- Figure II-5 : vue fonctionnelle du navigateur de projet
- Figure II-6 : choix d'une CPU
- Figure II-7 : Remplacement d'une CPU
- Figure II-8 : Ajout d'un rack
- Figure II-9 : choix d'une alimentation
- Figure II-10 : exemple de choix d'un module analogique
- Figure II-11 : Fenêtre décrivant le module d'entrée/sortie choisi
- Figure II-12 : Exemple de configuration d'une voie
- Figure II-13 : Création d'un nouveau réseau logique
- Figure II-14 : Configuration du réseau logique
- Figure II-15 : Association d'un réseau logique à un module de communication
- Figure II-16 : Représentation du bureau
- Figure II-17 : GRAFCET de l'exemple du bureau
- Figure III-1 Eléments d'un projet sous Vijeo Citect
- Figure III-2 : Définition d'un nouveau cluster

Figure III-3 : Définition des serveurs sur Vijeo Citect

Figure III-4 : Choix du périphérique d'E/S

Figure III-5 : Configuration de l'adresse et du protocole de communication

Figure III-6 : Ecran de visualisation de l'application

Figure III-7 : Définition de tags

Figure III-8: Création d'un nouveau génie

Figure III-9: Ajout d'un texte sous le symbole du capteur

Figure III-10: Edition de l'apparence du texte

Figure III-11 : Popup permettant d'associer un tag à un génie

Figure III-12 : Programmation de la couleur du texte

Figure III-13: Sauvegarde d'un génie, association à une librairie et à un projet

Figure III-14: Coller un génie dans la page principale

Figure III-15: Configuration du génie

Figure III-16: Configuration du bouton d'allumage dans le super génie

Figure III-17 : Configuration du bouton quitter dans le super génie

Figure III-18: Lier un super génie à un génie

Figure III-19: Aperçu de l'exécution de l'application, appel d'un super génie

Figure IV-1: Page d'Accueil du projet

Figure IV-2 : Synoptique général de l'installation électrique

Figure IV-3 : Schéma électrique du poste de livraison

Figure IV-4: Schéma électrique du tableau TGBT 1

Figure IV-5 : Schéma électrique du TGBT 2

Figure IV-6 : Le tableau divisionnaire pour l'éclairage du premier étage

Figure IV-7: Schéma électrique du Tableau Général du Groupe Electrogène

Figure IV-8 Page d'alarme

Figure IV-9 : Page sommaire des alarmes

Figure IV-10 Page de commande d'éclairage du premier étage

Figure IV-11 : Vue de l'étage 1

Figure IV-12 : Grafctet du cahier de charge de l'éclairage de la zone 2B

Figure IV-13 : Gestion de l'éclairage de la zone 2B en fonction de la présence et de
l'éclairage naturel

Figure IV-14 : Tableau divisionnaire de l'éclairage extérieur

Figure IV-15 : Page de gestion de l'éclairage extérieur

Figure IV-16 : Accès à la page des CTA

Figure IV-17 : Choix d'une CTA

Figure IV-18: Synoptique d'une CTA

Liste des tableaux

Tableau II-1 : Adressage physique des entrées / sorties

Tableau II-2 : Paramètres de la temporisation AKF_TE

Tableau III-1 : liste des tags utilisés

Introduction générale

Introduction Générale

L'économie de l'énergie est devenue un enjeu stratégique à maîtriser dans le monde d'aujourd'hui, en effet, une utilisation accrue et non rationnelle de l'énergie conduit à des problèmes très sérieux. D'une part, nous avons le problème de l'épuisement des sources d'énergie devant le fait que le coût des énergies alternatives n'est pas encore maîtrisé, et d'autre part, une consommation non rationnelle de l'énergie engendrera aussi le problème de la pollution qui menace le sort de notre planète et des générations futures, or face à cette réalité, on se heurte à un dilemme, en effet :

1. D'une part, on estime que la demande de l'énergie à l'échelle mondiale va doubler dans l'horizon de 2050.
2. D'autre part, et afin de limiter le problème du réchauffement climatique, on estime que les émissions de CO₂ doivent être diminuées de moitié dans l'horizon de 2050.

Le bâtiment est l'un des secteurs qui consomme le plus d'énergie et ceci au niveau mondial. De ce fait, il constitue l'un des principaux émetteurs de gaz à effet de serre. De plus une consommation irrationnelle en énergie fait très vite grimper la facture énergétique qui provient essentiellement de l'éclairage, de la ventilation et de la climatisation. Dans le but d'améliorer l'efficacité énergétique des bâtiments, le concept de « Bâtiment Intelligent » a été créé.

En appliquant les principes de l'automatisation de bâtiments, cette consommation sera réduite de façon durable. La Gestion Technique des Bâtiments étant d'actualité, nous avons décidé de l'explorer et d'y apporter notre contribution en développant une application de supervision de l'installation électrique, d'éclairage (intérieur et extérieur) et de climatisation pour un bâtiment à six étages avec rez-de-chaussée et sous sol.

Pour ce faire, nous avons effectué un stage chez l'un des leaders mondiaux dans le domaine des bâtiments intelligents qui est Schneider Electric dont le rôle est d'apporter plusieurs solutions permettant la réduction de la facture énergétique.

Notre application s'articulant autour des automates programmables, nous avons débuté notre étude par un apprentissage du logiciel **Unity Pro** pour la programmation des automates programmables de la gamme Schneider Electric puis nous nous sommes familiarisé avec le logiciel **Vijeo Citect** qui est l'un des logiciels de conception de systèmes de supervision les plus connus et les plus utilisés par les professionnels de l'automatisation.

Plan de Travail

Nous avons organisé notre projet en quatre chapitres :

Le premier chapitre donne des généralités sur les bâtiments intelligents en explicitant les concepts de base de ce domaine qui nous aideront à mieux appréhender notre application.

Le deuxième chapitre est consacré au logiciel Unity Pro pour la programmation des automates programmables de la gamme Schneider Electric. Les différentes étapes de création d'un projet sous ce logiciel y sont clairement présentées et finalisées par un exemple d'application.

Le troisième chapitre traite quand à lui du logiciel de conception de systèmes de supervision Vijeo Citect. Là encore les différentes étapes de création d'un projet y sont définies et illustrées par un exemple d'application.

Dans le quatrième et dernier chapitre, nous avons exposé notre application de supervision de l'installation électrique, de l'éclairage et de la climatisation d'un bâtiment de six étages avec sous sol et rez-de-chaussée.

Enfin, nous terminons notre travail par une conclusion générale et quelques perspectives.

Chapitre I

Généralité sur les Bâtiments Intelligents et les organes communicants

Introduction :

Ces dernières années, la construction est devenue de plus en plus complexe. En effet, les exigences des clients augmentent en parallèle avec la pression sur les prix, surtout à cause des coûts d'exploitation et d'énergie.

Pour satisfaire les besoins de ce marché, il faut savoir appliquer des nouvelles méthodes, technologie et produits. La gestion technique du bâtiment fait partie de ces nouvelles technologies. Dans les bâtiments fonctionnels modernes et les maisons individuelles d'un certain confort, les systèmes d'automation de bâtiments font déjà partie de l'équipement de base grâce aux technologies du bâtiment intelligent qui visent à améliorer l'environnement et la fonctionnalité du bâtiment pour les occupants et les locataires.

I-1 Le Bâtiment Intelligent :

I-1-1 Définition du bâtiment intelligent :

Un bâtiment intelligent est celui qui permet un environnement productif et rentable en misant sur l'optimisation et l'interrelation des quatre niveaux fondamentaux suivants : l'infrastructure, les systèmes, les services et la gestion.

L'immeuble intelligent aide le propriétaire, le gestionnaire et les occupants à réaliser leurs objectifs de coûts, de confort, de services, de sécurité, de flexibilité à long terme et de mise en marché.

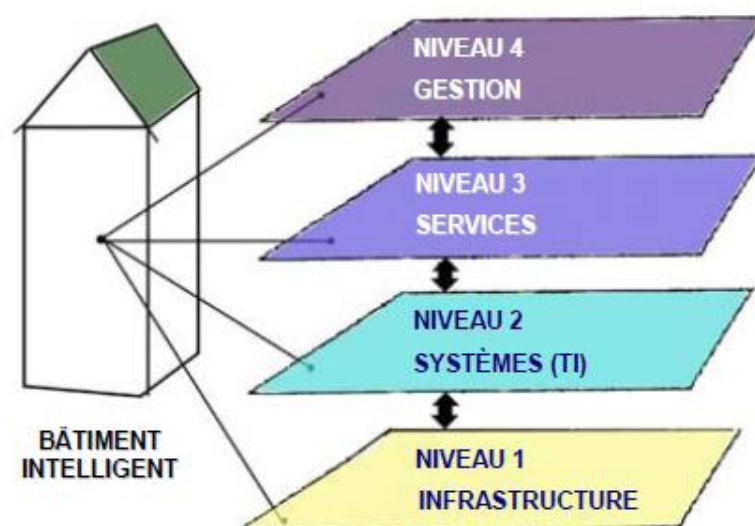


Figure I-1 : *Niveaux fondamentaux du bâtiment intelligent*

Ø L'infrastructure :

C'est le niveau de base qui favorise l'interopérabilité des systèmes. Il est couramment identifié en tant que « réseau de câblage structuré de télécommunication ». Il est formé d'éléments passifs tels des câbles en cuivre, des fibres optiques et des salles de répartition et d'équipements aménagées de façon à permettre l'établissement et le maintien des liens de communication.

Ø Les systèmes :

Le second niveau d'un bâtiment intelligent regroupe tous les éléments actifs à l'usage de ses occupants. C'est le domaine des Ti (technologies de l'information). Ordinateurs, serveurs, concentrateurs passerelles, routeurs, systèmes de téléphonie et de messagerie vocale, incluant les sans fil, font partie des appareils électroniques installés dans les salles d'équipement de l'infrastructure, ou aux postes de travail, qui procurent les moyens sophistiqués d'entrer en communication et de fournir les services. Entrent également dans la catégorie des Ti tous les logiciels d'exploitation et d'application, qui permettent l'interface essentielle entre les appareils et les usagers potentiels.

Ø Les services :

Ce niveau du bâtiment intelligent est caractérisé par l'ensemble des tâches accomplies à l'aide des systèmes, pour rencontrer les besoins des occupants. Ils concernent :

- la sécurité : accès, vidéo, incendie.
- l'automatisation d'édifice : contrôle des ascenseurs, optimisation des paramètres de confort et d'opération des systèmes électromécaniques, économie d'énergie, maintien des actifs.
- les communications : téléphonie, messagerie vocale, internet, courrier électronique.
- l'administration : tous les moyens utilisés pour l'organisation du travail et la planification financière.

Ø La gestion :

C'est le niveau supérieur du bâtiment intelligent qui établit les processus requis pour rendre les services de façon efficace et efficiente. Essentiellement, il est formé par le personnel spécialisé qui recherche constamment à obtenir les meilleurs résultats avec la meilleure technologie disponible.

I-1-2 Avantages des bâtiments intelligents [1] :

Ø Pour les occupants :

- Excellent contrôle des conditions internes de confort.
- Possibilité de contrôle individuel des pièces.
- Augmentation de la productivité des équipes de travail.
- Contrôle effectif de la consommation d'énergie.
- Améliorer la fiabilité du système (bâtiment) et la qualité de vie.

Ø Pour le propriétaire du bâtiment :

- Augmentation de la rentabilité du bâtiment.
- Facturation individuelle pour les locataires plus aisée.
- Contrôle local ou à distance du bâtiment.

Ø Pour les entreprises de maintenance :

- Disponibilité de l'information et diagnostic aisé.
- Etablissement d'un programme d'entretien informatisé
- Augmentation de l'efficacité des équipes de maintenance.
- Détection rapide des problèmes.
- Plus grande satisfaction des occupants.

Ø Du point de vue opérationnel :

- Diminution des coûts des opérations.
- Utilisation efficace des ressources et services du bâtiment.
- Grande productivité.
- Diagnostic des défauts et indications rapide des alarmes.
- Excellente schématisation et documentation du système.

Ces objectifs ne peuvent être réalisés qu'avec une combinaison bien équilibrée d'architecture de qualité et d'une bonne automation de bâtiments. Par ailleurs, les nouvelles méthodes appliquées en matière de construction industrielle (GTC /GTB) vont exactement dans cette direction. Même si les bâtiments portent toujours la griffe de l'architecte en ce qui concerne le design et l'organisation des espaces, les méthodes de planification et de réalisation n'ont pas besoin d'être redéfinies à chaque fois.

I-2 Gestion Technique Centralisée (GTC) et Gestion Technique des Bâtiments (GTB) :

Nous allons maintenant définir ce que c'est la Gestion Technique des Bâtiments (GTB) et la Gestion Technique Centralisée (GTC) [2].

I-2-1 Définition de la GTC :

La Gestion Technique Centralisée est l'outil de pilotage de l'installation, cohérent avec les ambitions du projet. Sa vocation est de mettre en œuvre, à tout instant, et avec les anticipations nécessaires, la réactivité des dispositifs proposés et leur fonctionnement combiné. Au service des hommes, elle ne se substitue pas à eux. Le principe retenu est de proposer une installation dont la tendance « naturelle », technologique, est de se rapprocher à tout moment, du fonctionnement idéal au regard des objectifs globaux du projet.

L'utilisateur peut, à tout instant, reprendre la main, de manière simple, sur les dispositifs, pour adapter le fonctionnement localisé de l'installation à son sentiment de confort.

Le fonctionnement du volet est un bon exemple : lors d'une séance de travail nocturne, l'occupant a toute latitude pour rouvrir manuellement un ou plusieurs volets dont la fermeture a été commandée par la GTC.

I-2-2 Définition de la GTB :

La gestion technique du bâtiment (GTB) désigne une partie de l'automatisation de bâtiments. Elle est composée d'un réseau de capteurs, d'actionneurs et de contrôleurs décentralisés et mis en réseau pour intégrer dans un seul niveau de communication un maximum d'installations techniques. La GTB est réalisée avec des composants munis de microprocesseurs servant à la fonction locale ainsi qu'à la communication avec les autres composants. Les fonctionnalités transversales et locales sont assurées par le paramétrage des composants. Grâce aux systèmes de bus, le câblage et l'installation deviennent nettement plus simples. Par contre, il faut maîtriser les nouveaux outils de programmation.

I-3 Principales boucles de régulation et fonctions dans les bâtiments Intelligents [1] :

I-3-1 Contrôle de l'éclairage :

aujourd'hui, un bon éclairage fait partie de notre confort: varier la lumière depuis différents endroits, enclencher ou déclencher un éclairage par des détecteurs de présence ou de mouvement, la commande et le réglage des installations en fonction de la lumière naturelle, la commande centralisée (par exemple, éteindre toutes les lampes par un seul bouton-poussoir en quittant l'habitation), commande par télécommandes multifonctions à infrarouge, la liaison avec l'installation d'alarme et la simulation de présence etc.

I-3-2 Contrôle de Chauffage, ventilation et climatisation (CVC) et qualité de l'air intérieur :

1. Contrôle de l'air ambiant :

Les circuits d'aération mixent l'air ambiant de retour et celui de l'extérieur afin de garder l'air à l'intérieur du bâtiment sein et de faire des économies sur l'énergie nécessaire au chauffage ou au refroidissement de l'air. Des capteurs analogique ou numériques de température peuvent être placés dans les pièces, dans les conduits d'air de retour et d'approvisionnement et parfois à l'extérieur du bâtiment.

Les actionneurs peuvent être placés sur les vannes d'approvisionnement ou de retour d'air, sur les vannes d'eau chaude ou froide ou bien sur les ventilateurs. Les ventilateurs d'approvisionnement et de retour sont actionnés ou arrêtés suivant les températures, le temps qu'il fait à l'extérieur, la pression à l'intérieur du bâtiment ou une combinaison de tous ces paramètres.

Nous avons opté pour les *Circuits à Volume d'Air Variable (VAV)* ; Ces unités sont très efficaces. Parmi les systèmes VAV, on trouve celui qui fournit en air pressurisé des réservoirs dont chacun est, en général, affecté à une pièce ou une aire. La pression à l'intérieur des réservoirs peut être contrôlée en agissant sur la vitesse des ventilateurs (à l'aide de variateurs de vitesse). La quantité d'air à l'intérieur des réservoirs est déterminée suivant les besoins de la pièce. Certains réservoirs VAV ont des échangeurs de chaleur internes. Les vannes d'eau froide ou chaude sont ouvertes ou fermées suivant le niveau nécessaire de chaleur demandée à l'intérieur de la pièce.

2. Circuits thermiques :

Certains systèmes de climatisation nécessitent un circuit d’approvisionnement en eau chaude et froide pour refroidir ou réchauffer l’air à pulser dans les locaux. Certains systèmes génèrent eux-mêmes de l’électricité grâce aux turbines à gaz (ou à vapeur) et utilisent l’échappement chaud de ces turbines pour chauffer l’eau du circuit thermique.

ü **Systèmes d’eau froide:** L’eau froide est souvent utilisée pour refroidir l’air des bâtiments et des équipements. Ces systèmes de refroidissement contiennent en général des pompes et des groupes frigorifiques ainsi que des capteurs de température qui mesure la température de l’eau froide.

ü **Systèmes d’eau chaude:** Cette eau chaude peut servir par exemple à l’approvisionnement des VAV à travers les serpentins de chauffage. Ces systèmes contiennent généralement des chaudières et des pompes ainsi que des capteurs de température placés dans les conduites d’approvisionnement et de retour de l’eau chaude.

1-3-3 Sécurité et Alarmes:

Dans un système d’automatisation de bâtiments, si un problème est détecté, une alarme est enclenchée pour avertir par exemple un opérateur.

Ces avertissements peuvent être envoyés par email, par téléphone ou être seulement des alarmes audibles et/ou visibles. Voici quelques exemples d’alarmes :

- Alarme pour indiquer que le filtre à air est sale.
- Alarmes de défauts d’actionneurs : vannes, pompes...
- Alarmes indiquant que le taux de monoxyde de carbone ou dioxyde de carbone est trop élevé.
- Alarmes d’intrusion.
- Alarmes d’incendies

Dans des sites contenant plusieurs bâtiments, une panne momentanée de puissance peut causer des centaines de milliers d’alarmes. Les alarmes les plus critiques sont prioritaires et peuvent être programmés pour resurgir à certains intervalles de temps jusqu’à ce que ces problèmes soient résolus.

I-4 Les systèmes CVC :

Les systèmes de CVC sont généralement contrôlés par des systèmes d'automatisation qui peuvent :

- permettre aux divers occupants de modifier la température des lieux de travail (à l'intérieur des limites prescrites).
- surveiller la température et la modifier selon le profil de l'utilisateur.
- modifier la qualité de l'air selon l'occupation des locaux et les normes du bâtiment.
- modifier l'humidité, la température et le débit de l'air.
- faire appel à des conceptions de distribution de l'air à volume d'air variable ou à volume constant. Ce premier modèle permet un plus grand contrôle individuel.

I-4-1 Système de climatisation:

La climatisation est la création et le maintien d'un air ambiant dont la température, l'humidité et la pureté sont des facteurs primordiaux. Un système de climatisation est constitué d'un dispositif centralisé, qui produit une atmosphère contrôlée à tout moment, quelles que soient les conditions climatiques.

Ü *Système tout air, à débit constant, mono-gaine :*

Le système de conditionnement d'air "tout air, à débit constant, mono-gaine" est un système où l'air est préparé (chauffé, refroidi, humidifié,...) en centrale dans un caisson de traitement d'air, puis envoyé par un réseau de gaines vers le/les locaux. On parle de **système tout air** car l'air est le fluide caloporteur de chaleur, de froid, ou d'humidité. Le terme « mono-gaine » quand à lui est du au fait qu'un seul réseau de gaine est crée pour le transport de l'air.

Dans ce type de système, l'air est traité dans une **CTA** (Centrale de Traitement d'Air).

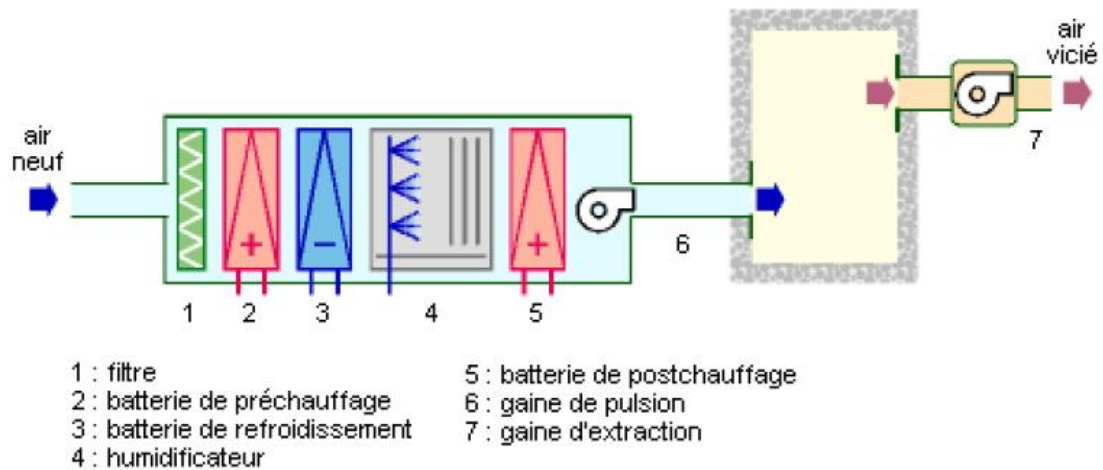


Figure I-2 : Exemple d'une CTA appliquée à un local [3]

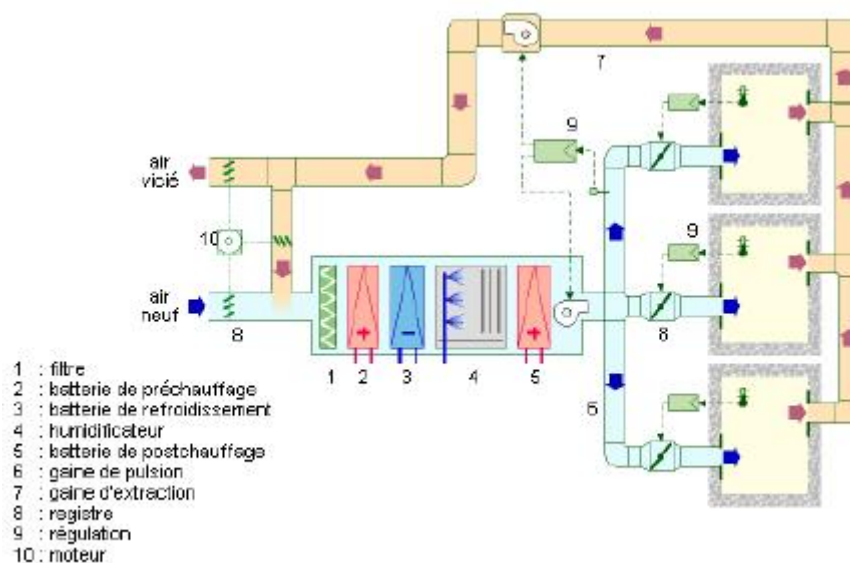


Figure I-3 : Exemple d'un système VAV mono-gaine [3]

I-4-2 Ventilation :

La ventilation est un procédé qui consiste à renouveler l'oxygène dans les locaux afin de maintenir un équilibre de la concentration du gaz carbonique. La ventilation permet également de réduire la chaleur, les odeurs désagréables et l'humidité et d'évacuer les fumées de cigarette. Dans une usine ou une raffinerie, les systèmes de ventilation évacuent les contaminants dangereux contenus dans l'air.

Etant donné que dans les applications de ventilation ou de climatisation, c'est le ventilateur centrifuge qui est le plus souvent utilisé, c'est ce dernier qui sera explicité par la suite.

a) Les courbes caractéristiques d'un ventilateur :

Les performances des ventilateurs sont répertoriées sous forme de courbes caractéristiques reprises dans la documentation des fabricants.

On retrouve dans les courbes caractéristiques :

- la hauteur manométrique totale que peut fournir un ventilateur en fonction du débit (ou point de fonctionnement).
- la vitesse du ventilateur pour chaque point de fonctionnement.
- le rendement du ventilateur pour chaque point de fonctionnement.
- la puissance absorbée à l'arbre du moteur.

Dans l'exemple ci-dessous ; pour fournir un débit de 8 000 m³/h, le ventilateur délivre une pression totale de 1 000 Pa. Pour obtenir ce point de fonctionnement le ventilateur doit tourner à 1 950 tr/min. Pour ce point de fonctionnement, son rendement sera de 81 % et la puissance à l'arbre sera proche de 2,8 kW.

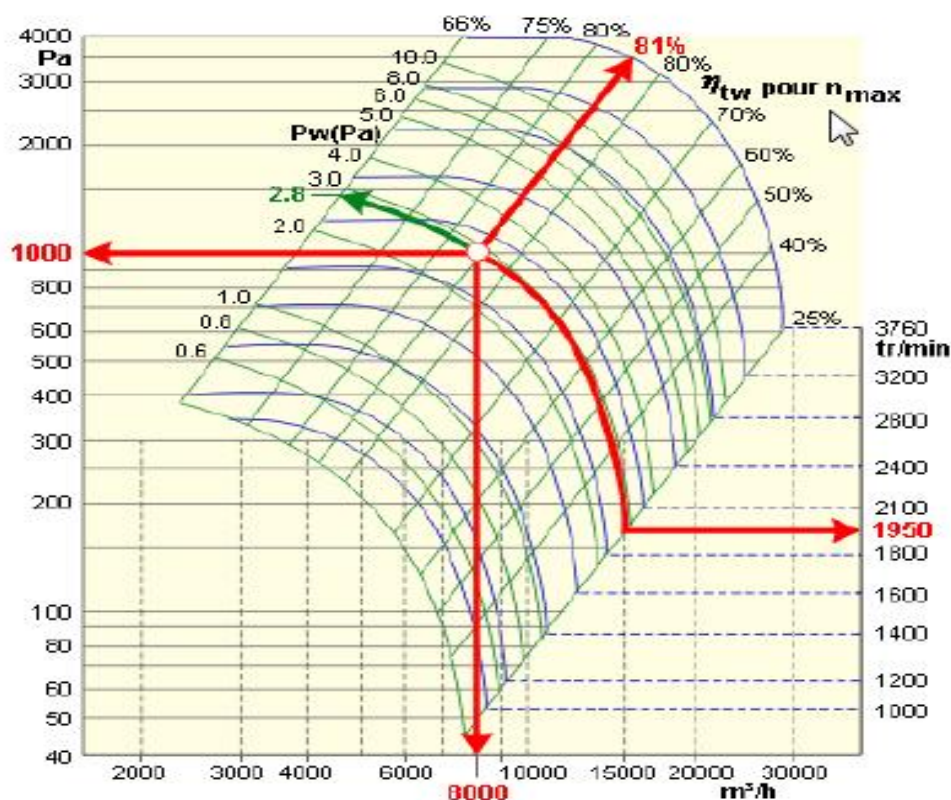


Figure I-4 : *Courbes caractéristiques d'un ventilateur centrifuge à aubes inclinées vers l'arrière*[3]

b) Point de fonctionnement :

Le ventilateur fournit donc l'énergie nécessaire à l'air en lui imposant une différence de pression totale (pression statique+dynamique) entre la prise extérieure et la bouche de pulsion (ou dans le sens inverse dans le cas d'une extraction) ; c'est-à-dire, pour mettre l'air en vitesse dans le conduit et vaincre les pertes par frottement dans celui-ci. Cette différence de pression totale est appelée "**hauteur manométrique**" du ventilateur.

Si l'on branche un ventilateur sur un circuit de ventilation, il stabilisera son débit à une valeur pour laquelle la pression qu'il fournit équivaut à la résistance du circuit. Ce point est le seul point de fonctionnement possible. Il correspond à l'intersection des **courbes caractéristiques du ventilateur** et du circuit.

Il définit la **hauteur manométrique** et le débit fournis par le ventilateur lorsque, fonctionnant à une vitesse donnée, il est raccordé au réseau considéré.

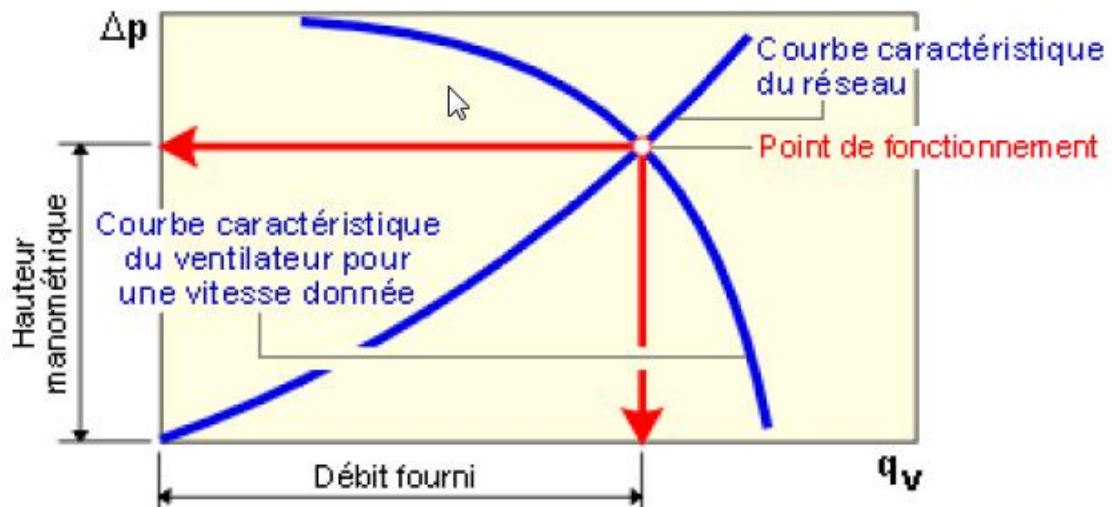


Figure I-5 : Point de fonctionnement d'un ventilateur [3]

I-5 Rôle de la GTB dans la promotion de l'efficacité énergétique :

Un des exemples illustrant le rôle que joue la GTB dans l'efficacité énergétique est l'exemple d'un système de GTB permettant une commande adaptative de l'éclairage intérieur suivant l'intensité d'éclairage naturelle du soleil.

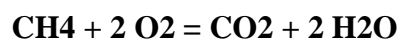
Ainsi on aura un service égal à moindre coût. Par ailleurs, l'utilisation des capteurs de présence permet d'éteindre la lumière là où aucune présence humaine n'est enregistrée, ce qui contribue à limiter le gaspillage de l'énergie.

Par ailleurs, les commandes intelligentes des systèmes CVC (Chauffage, Ventilation et Climatisation) permettent de réduire la température des locaux lorsque ceux-ci sont vides tout en garantissant une température agréable à l'arrivée des occupants. Ces commandes permettent aussi d'agir sur les stores afin de profiter de la chaleur du soleil en hiver et de s'en protéger en été, la facture de chauffage et de climatisation se trouve ainsi nettement réduite. Ceci démontre donc le rôle capital que joue le GTB dans la résolution d'un enjeu capital qui est l'efficacité énergétique.

L'efficacité énergétique (ou l'efficience énergétique) permet de réduire les consommations d'énergie à **service rendu égal** entraînant de ce fait la réduction de la facture énergétiques.

I-5-1 Avantages de l'efficacité énergétique :

Dans cette section, on donnera quelques exemples pour illustrer l'importance de l'efficacité énergétique dans la résolution des problèmes liés à la pollution et l'environnement. Nous savons bien que la consommation de l'énergie est responsable des émissions de gaz qui engendrent des problèmes écologiques tel que les pluies acides et le réchauffement climatique. Pour illustrer l'impact d'une consommation excessive de l'énergie sur l'environnement, considérons l'équation de combustion du méthane (principal composant du gaz naturel).



En faisant le bilan de masse on aura :

$$(12 + 4*1) + 2(2*16) = (12 + 2*16) + 2(2*1 + 16)$$

(Ce bilan est fait on tenant en compte le nombre de masse du carbone (12) de l'oxygène (16) et de l'hydrogène (1)). Ainsi, la combustion de 16 unité de masse du méthane donne 44 unité de masse du dioxyde de carbone, le rapport est donc de 2.75. Ceci montre donc que la réduction de combustion de méthane grâce à une utilisation efficace de l'énergie permet de réduire considérablement les émissions du dioxyde de carbone et ainsi de limiter le phénomène de réchauffement climatique.

Par ailleurs, sachant que les centrales électriques utilisant le charbon et les fuels sont les principaux responsables des émissions du dioxyde sulfurique responsable à son tour des pluies acides, on déduit que la réduction de consommation de l'énergie apportée par une utilisation efficace de cette dernière permet aussi de réduire les émissions du dioxyde sulfurique et ainsi limiter le problème de pluies acides.

En résumé, l'efficacité énergétique permet de limiter les problèmes environnementaux liés à une consommation excessive de l'énergie tout en réduisant la facture de consommation pour les utilisateurs. [4]

I-6 Equipements dans la GTB :

Dans la gestion technique du bâtiment, on distingue en gros les composants suivants:

- Appareils de système tels qu'alimentations en courant, coupleurs, routeurs, etc.
- Capteurs: boutons-poussoir, détecteurs physiques (lumière, température, pluie, etc.)
- Actionneurs: commutateurs, variateurs de lumière, actionneurs de jalousie, actionneurs pour servomoteur de vanne ou de clapet.
- Contrôleurs (API, automate programmable industriel, avec accès direct au bus), minuteriers .
- Affichages, écrans tactiles, appareils de commande.
- Appareils de communication (interface PC, coupleur téléphone, Ethernet, Internet).
- Accessoires tels que connecteurs.
- Logiciels (outils de paramétrage, visualisations).

Remarque :

Les équipements utilisés dans notre application sont mentionnés dans l'Annexe A.

I-7 Le protocole de communication :

Un protocole de communication est une spécification de plusieurs règles pour un type de communication particulier. Initialement, on nommait protocole ce qui est utilisé pour communiquer sur une même couche d'abstraction entre deux équipements différents. Par extension de langage, on utilise parfois ce mot aujourd'hui pour désigner les règles de communication entre 2 couches sur un même équipement.

Ü Définitions élémentaires :

Terrain : indique une espace délimitée géographiquement (usine, atelier, voiture...)

Bus : au sens informatique industrielle, conducteur ou ensemble de conducteurs communs à plusieurs circuits permettant l'échange de données entre eux avec :

- Liaisons communes .
- Plusieurs circuits .
- Référence à la topologie de la configuration.

Réseau : ensemble de lignes de communication qui desservent une même unité géographique
BUS / RESEAU DE TERRAIN : Terme générique d'un nouveau réseau de communication numérique, bidirectionnel, multi branche (multi drop), série reliant différents types d'équipements d'automatisme :

- E/S déportées.
- Capteur / Actionneur.
- Automate programmable.
- Calculateur.

I-7-1 Définition de Protocole MODBUS :

Le protocole MODBUS (marque déposée par MODICON) est un protocole de dialogue basé sur une structure hiérarchisée entre un maître et plusieurs esclaves.

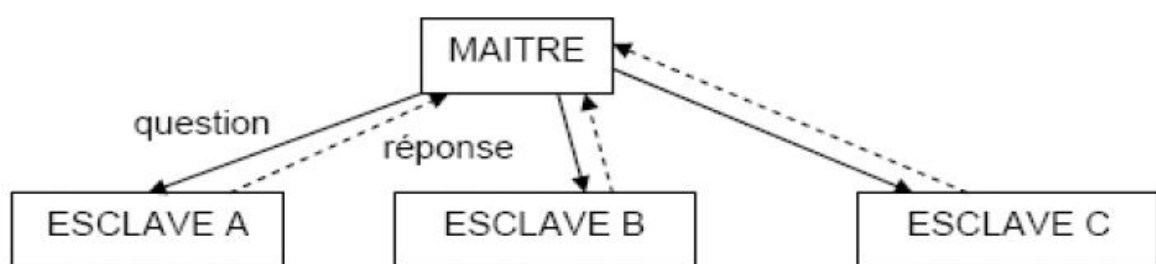


Figure I-6 : *Le principe mono maître du protocole MODBUS [5]*

Le protocole MODBUS consiste en la définition de trames d'échange. La gestion des échanges se fait de la manière suivante :

- Un seul appareil peut émettre sur le bus.
- Les échanges sont gérés par le maître. Seul le maître peut générer des échanges.

Aucun esclave ne peut de lui-même envoyer de message sans y avoir été invité.

- Le maître réitère la requête lors d'un échange erroné et décrète l'esclave interrogé absent après une non-réponse dans un temps donné.
- L'esclave émet une réponse d'exception au maître s'il ne comprend pas le message ou s'il ne peut pas le traiter. Dans ce cas, le maître peut réitérer ou non la requête.

Un contrat d'échange doit être créé, et ceci pour définir la table de réception, dédiée à l'écriture du maître dans l'esclave, et une table d'émission pour la lecture du maître dans l'esclave.

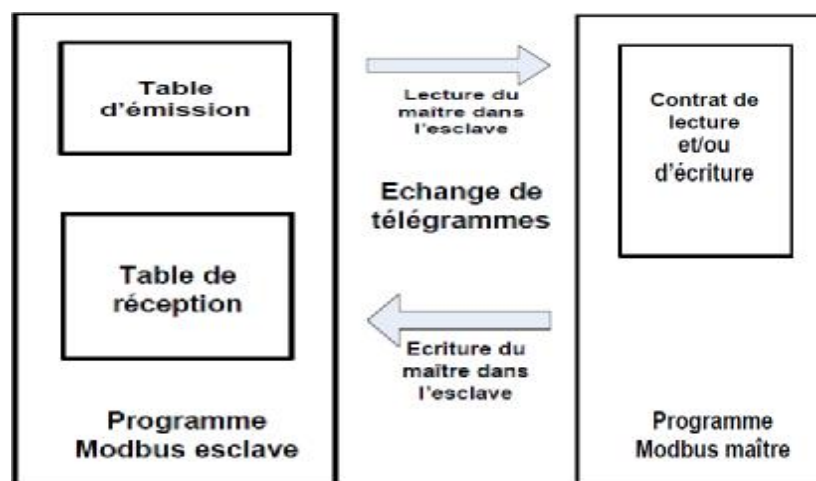


Figure I-7 : Principe des échanges MODBUS [5]

Il existe deux types de dialogue possible en MODBUS :

- Ø Echange maître vers l'esclave : le maître envoie une demande et attend une réponse (Relation maître – esclave). Le maître prend l'initiative de la transmission, puis attend, après avoir envoyé un télégramme de demande, un télégramme de réponse de l'esclave pendant la durée qui a été paramétrée comme temps de surveillance de la réponse.
- Ø Echange Maître vers tous les esclaves (Message Broadcast) : le maître utilise l'adresse 0 pour s'adresser à tous les esclaves du bus. Il diffuse un message à tous les esclaves présents sur le réseau, ceux-ci exécutent l'ordre du message sans émettre une réponse.

I-7-2 Norme OSI :

Le modèle OSI (Open System Interconnexion) a été créé par l'ISO (organisation internationale de normalisation). Il a permis d'obtenir une base commune à la description de tout réseau informatique. Ce modèle est composé de 7 couches respectant les principes suivants :

- Ø Chaque couche supporte un protocole indépendamment des autres couches.
- Ø Chaque couche procure les services à la couche immédiatement supérieure.
- Ø Chaque couche requiert les services de la couche immédiatement inférieure.

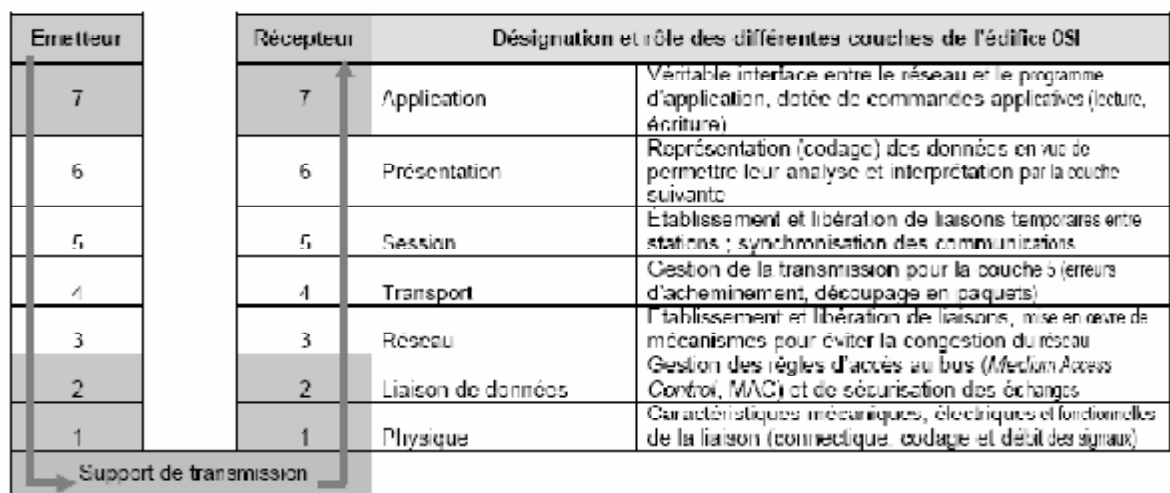


Figure I-8 : Les sept couches du modèle OSI [5]

I-7-3 Etude du réseau :

Un réseau est un ensemble de machines interconnectées dans les buts suivants :

- Ø Partager des fichiers.
- Ø Communiquer entre les machines et entre les utilisateurs.
- Ø Avoir une unicité de l'information (bases de données).
- Ø Délocaliser les utilisateurs.
- Ø Partager des équipements (imprimantes).

I-7-3-1 Ethernet:

Ethernet Spécification de câblage et de signalisation LAN (Local Area Network - Réseau local) utilisée pour connecter des équipements au sein d'un site bien précis, tel qu'un immeuble. Ethernet utilise un bus ou une topologie en étoile pour connecter différents nœuds sur un réseau.

Le réseau Ethernet a été créé aux années 1970 par Digital Intel Xerox, il est standardisé en 1985 avec la norme IEEE 802.3. En 1998 fut standardisé le « Fast Ethernet » associé à la technologie « switching technology » et au mode full duplex. De là on a commencé à l'utiliser comme réseau de terrain. Ethernet vise essentiellement les applications de:

- Ø Coordination entre automates programmables.
- Ø Supervision locale ou centralisée.
- Ø Communication avec l'informatique de gestion de production.
- Ø Communication avec des entrées/sorties distantes.

Deux profils de communication sont supportés par les coupleurs réseaux ETHERNET.

✚ Le profil ETHWAY :

Reprenant tous les mécanismes de l'architecture de communication X-WAY:

- Ø Système d'adressage X-WAY.
- Ø Messagerie UNI-TE.
- Ø Base de données distribuées (mots communs).

✚ Le profil TCP/IP sur ETHERNET permettant la communication en :

- Ø Messagerie UNI-TE avec l'ensemble de l'architecture X-WAY.
- Ø Messagerie Modbus.

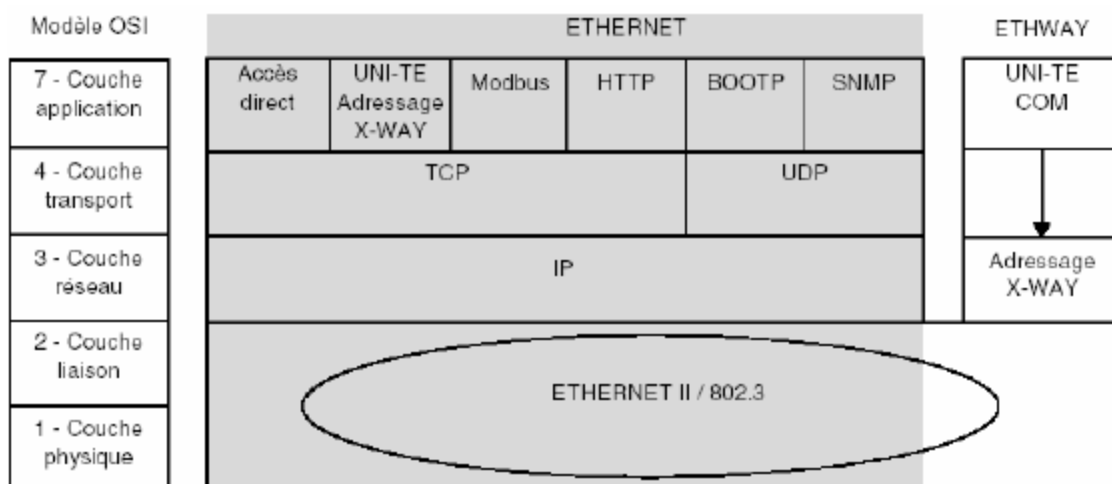


Figure I-9 : Service et protocoles associés à Ethernet

I-7-3-2 Ethernet TCP/IP:

Le mode client permet d'envoyer une requête sur un canal virtuel de communication. Chaque canal virtuel correspond à une connexion TCP sur un équipement.

Le mode serveur pour des automates clients Modbus leur permet d'accéder à la base de données et de surveiller les équipements.

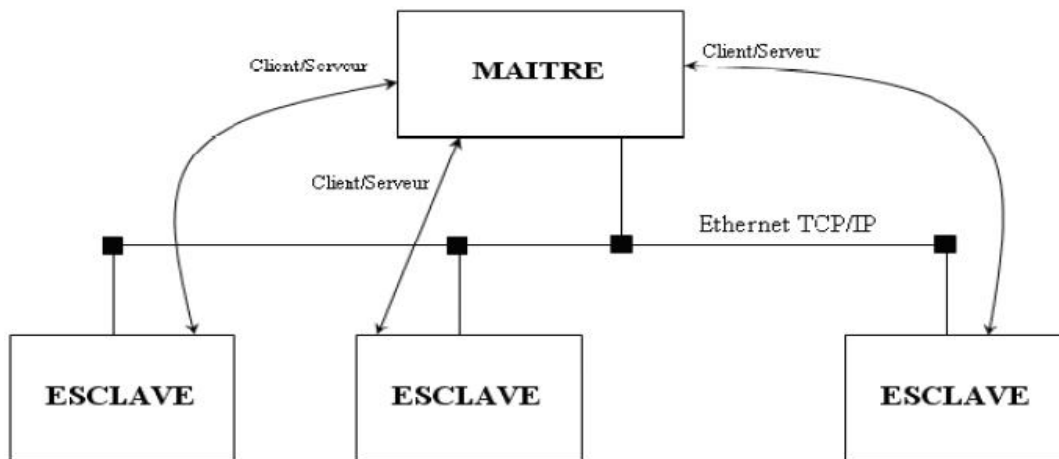


Figure I-10 : Communication avec Ethernet TCP/IP

Principe d'encapsulation :

Le protocole IP assure le transport des paquets de données numériques. Le protocole TCP assure le contrôle de la transmission des données.

Lorsque l'application Modbus envoie des données à l'aide de TCP/IP les données traversent de haut en bas chaque couche jusqu'à aboutir au support physique où elles sont alors émises sous forme de suite de bits. L'encapsulation illustré ci-dessous consiste pour chaque couche à ajouter de l'information aux données en les commençant par des en-têtes

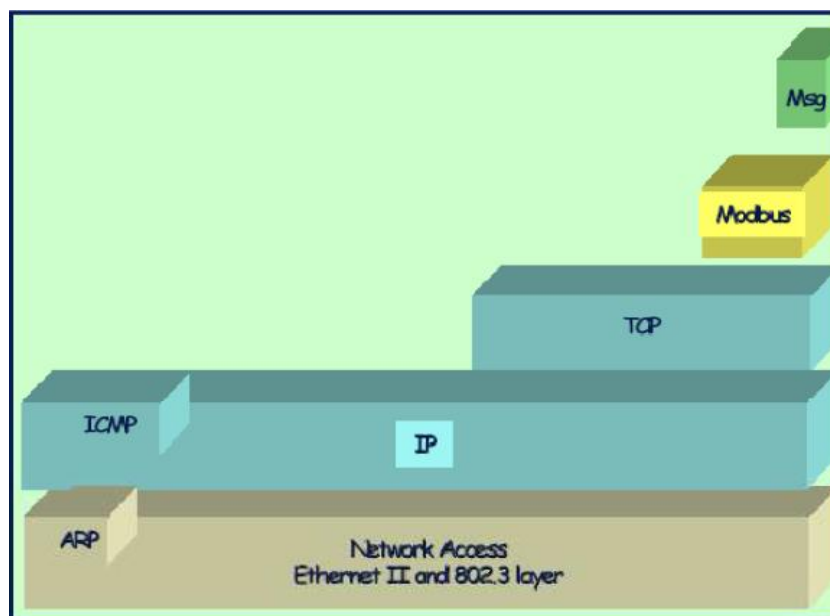
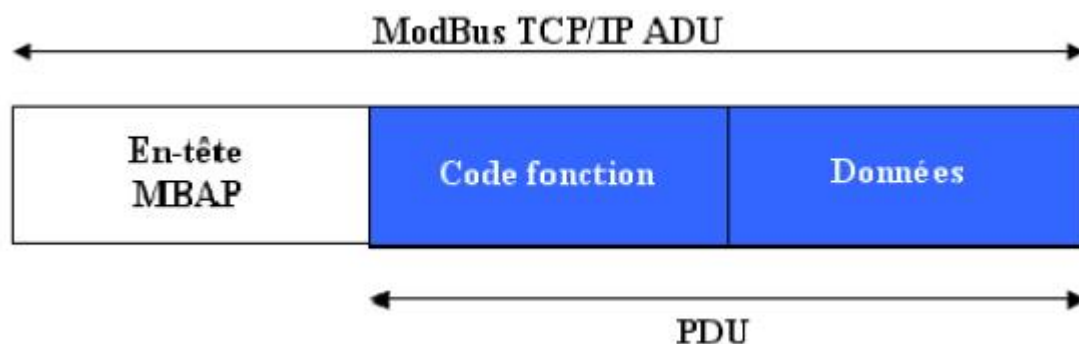


Figure I-11 : Principe d'encapsulation

I-7-3-3 Le protocole Modbus sur TCP/IP :

Le protocole Modbus sur TCP/IP est défini comme suit :



ADU : Application Data Unit

PDU : Protocol Data Unit

Toutes les requêtes et réponses de Modbus sont conçues de telle manière que le destinataire puisse vérifier qu'un message est fini.

L'information de longueur est diffusée dans l'en-tête de MBAP composé de 7 octets pour permettre au destinataire d'identifier les frontières des messages même si le message a été coupé en plusieurs paquets pour la transmission. Donc, Un même coupleur peut communiquer avec un équipement distant en mode client (par exemple un automate Modicon) et un autre équipement distant en mode serveur (par exemple un PC superviseur).

Par exemple l'automate Modicon 1 est client vis à vis de l'automate Modicon. Il ouvre la connexion TCP-IP et émet des messages Modbus vers le Modicon.

L'automate Modicon 1 est serveur vis à vis Modicon 2. Le Modicon 2 ouvre une connexion TCP/IP et émet des messages Modbus vers le Modicon1.

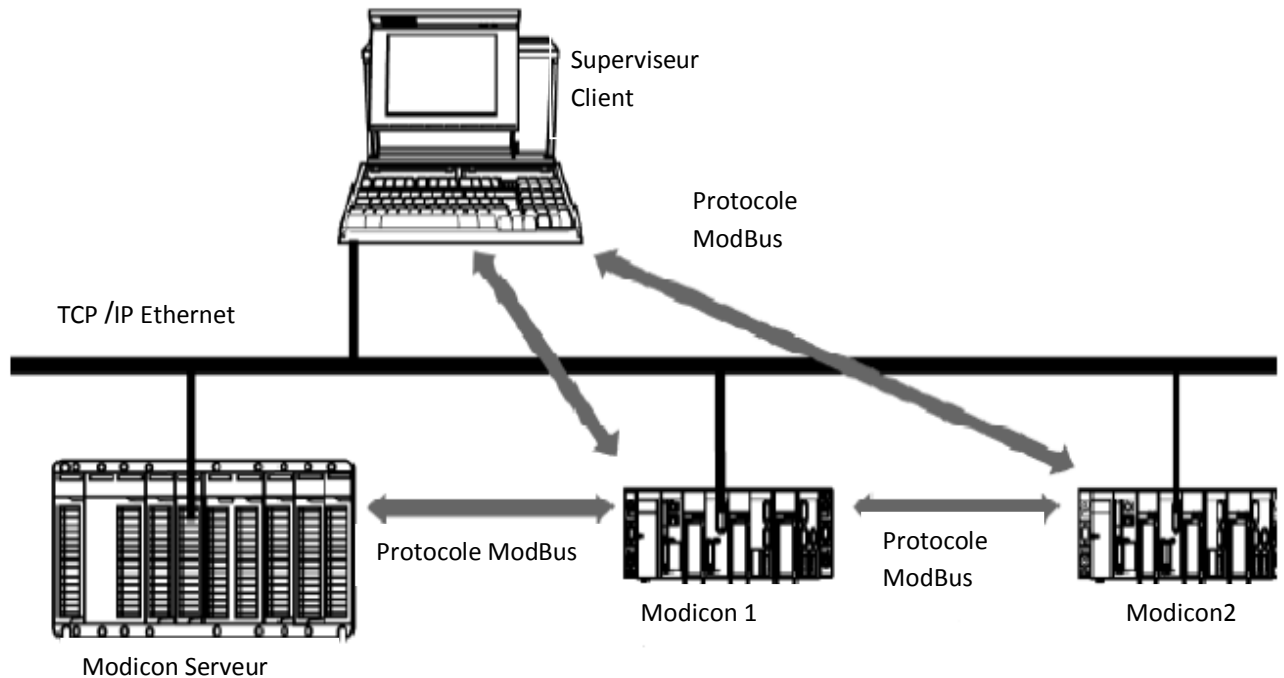


Figure I-12 : *Communication via Ethernet TCP/IP et Modbus [5]*

Conclusion :

Dans ce premier chapitre nous avons donné une idée générale des applications de gestion de bâtiment et des organes électriques communicants permettant une réalisation des architectures de communication et le bon choix des protocoles de communication.

Dans ce qui va suivre, le travail est consacré à l'étude de l'automate programmable le mieux adapté pour notre application et à son langage de programmation afin de pouvoir contrôler, d'automatiser et de programmer notre bâtiment.

Chapitre II

Présentation du Logiciel

Unity Pro

Introduction

Unity Pro est le logiciel de programmation des gammes d'automates : Modicon M340, Premium, Atrium et Quantum. Il supporte les cinq langages de la norme IEC 6113-3 qui sont : langage à blocs fonction (FBD), langage à contacts (LD), diagramme fonctionnel en séquence (SFC), liste d'instructions (IL) et littéral structuré (ST).

Dans ce chapitre, nous allons présenter ce logiciel en décrivant ses diverses fonctionnalités.

II-1 Présentation de l'interface utilisateur d'Unity Pro :

En activant le logiciel Unity Pro et après avoir configuré le matériel (chose que nous verrons par la suite), l'interface utilisateur d'Unity Pro apparaît comme ceci :

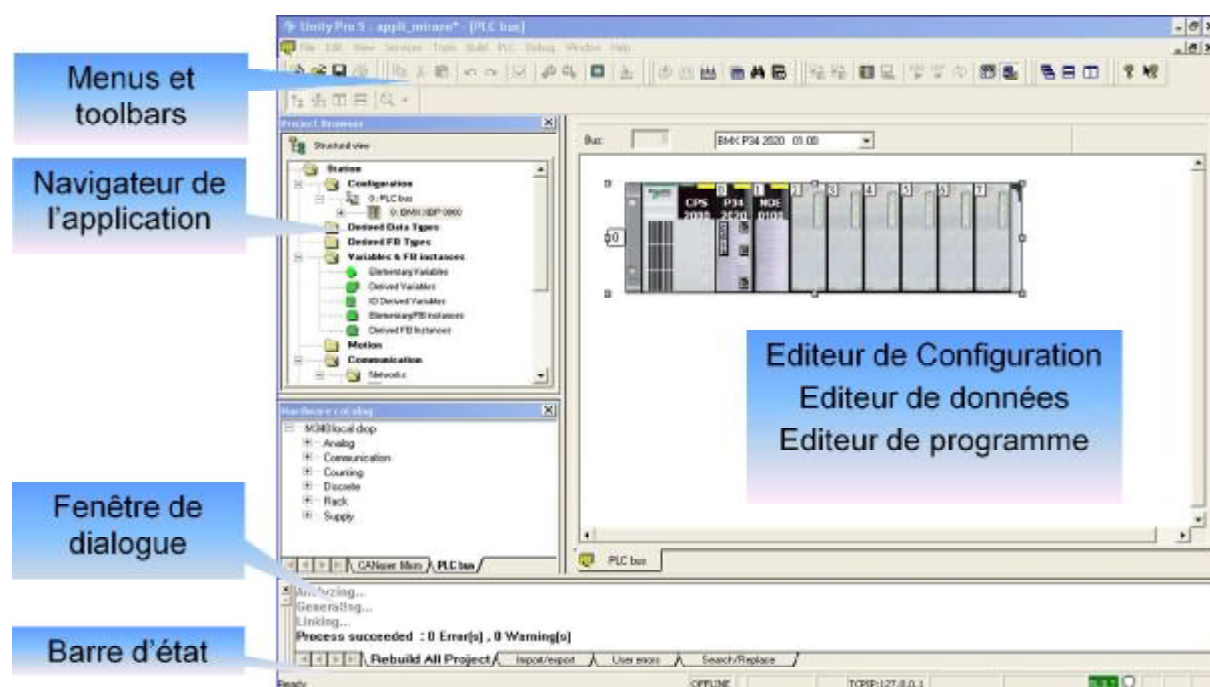


Figure II-1 : Interface utilisateur [6]

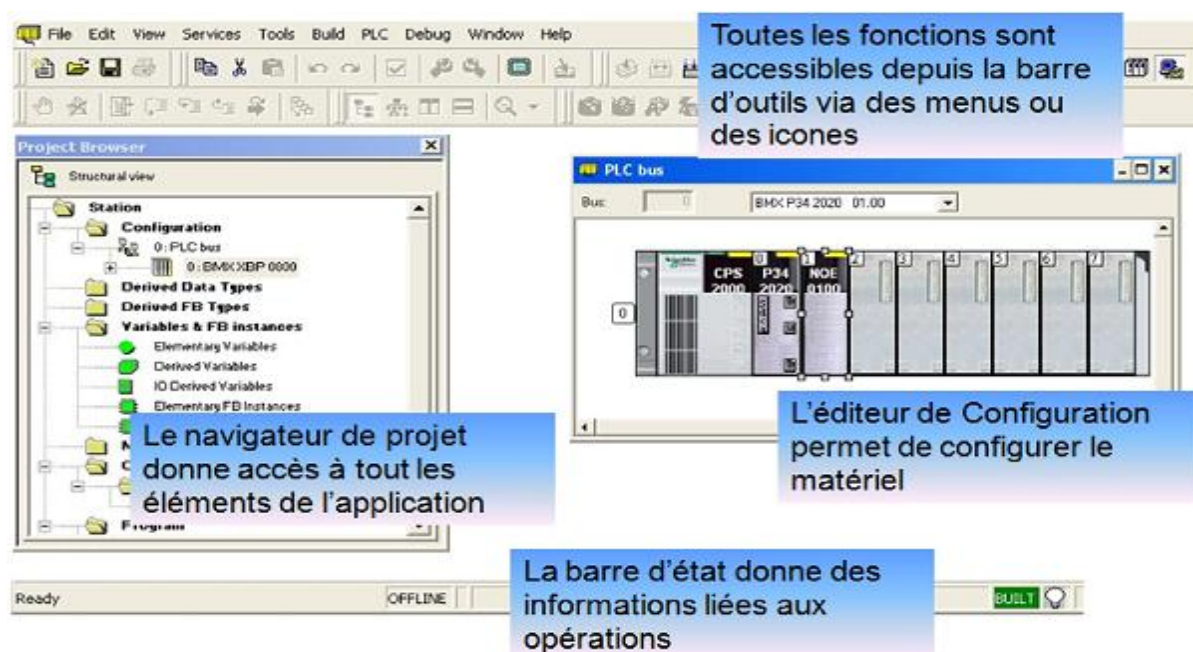


Figure II-2 : Interface utilisateur (suite) [6]

Le navigateur de projet permet de structurer ce dernier sous une forme arborescente afin de faciliter l'accès aux diverses fonctionnalités d'Unity Pro. Ce navigateur propose deux différentes vues, pour structurer et présenter le projet, qui sont : une vue structurelle et une vue fonctionnelle comme le montre la figure II-3.

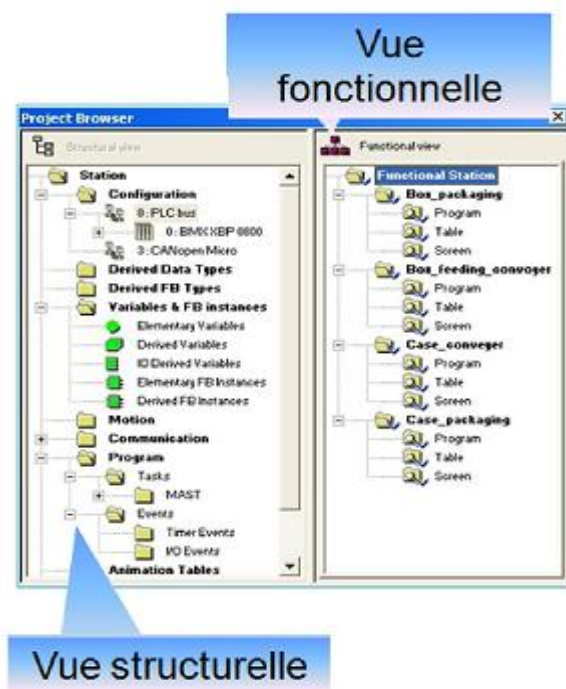


Figure II-3 : les différentes vues du navigateur de projet [6]

Avec la vue structurelle, l'utilisateur peut accéder et gérer les différents éléments de l'application (Configuration matérielle, variables, programmes...) comme le montre la figure II-4.

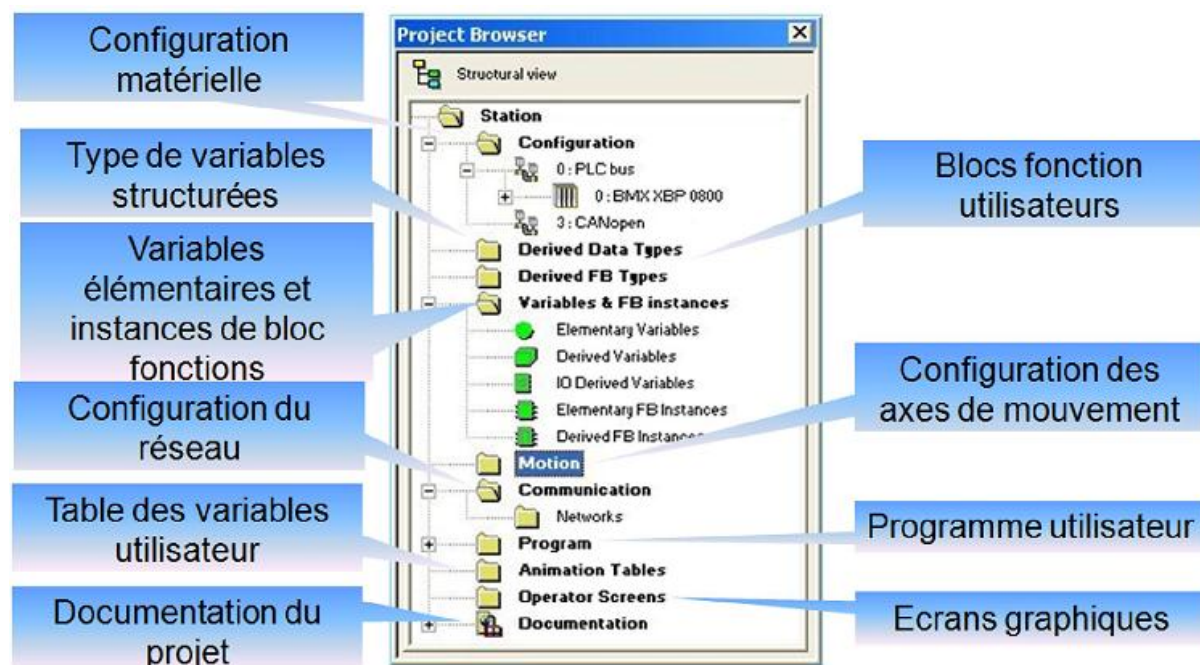


Figure II-4 : *Vue structurelle du navigateur de projet [6]*

La vue fonctionnelle quand à elle permet à l'utilisateur de structurer son projet sous forme de modules fonctionnels comme le montre la figure II-5.

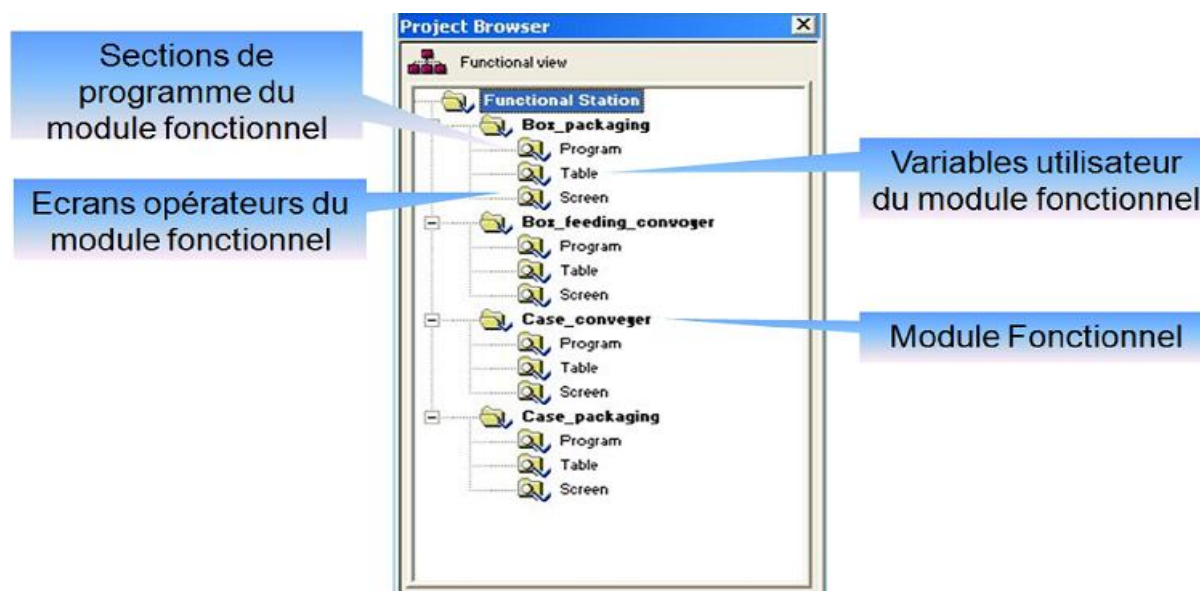


Figure II-5 : *vue fonctionnelle du navigateur de projet [6]*

II-2 Création d'un nouveau projet sous Unity Pro :

La création d'un projet sous Unity Pro passe par plusieurs étapes afin de faciliter la tâche d'automatisation. Ces étapes sont :

- 1) Configuration matérielle.
- 2) Configuration du réseau.
- 3) Définition et édition des variables (variables élémentaires, structures, tableaux...).
- 4) Structurer l'application sous forme de tâches, de sections et de modules fonctionnels.
- 5) Edition du programme dans les cinq langages de programmation mentionnés précédemment.
- 6) Compilation et simulation des programmes avant leur implémentation dans l'automate.

II-2-1 Configuration matérielle :

Dans cette étape l'utilisateur peut choisir l'unité de traitement qui correspond à ses exigences, ainsi que le rack où sera placé l'automate, et la configuration des divers modules d'entrées/sorties en fonction du nombre de capteurs et d'actionneurs et de leurs types.

II-2-1-1 Choix d'une CPU :

Dès la création d'un nouveau projet, il est demandé de choisir une station d'automatisation (Modicon M340, Premium, Quantum, Quantum Safety) suivi du choix d'une CPU comme le montre la figure I-6.

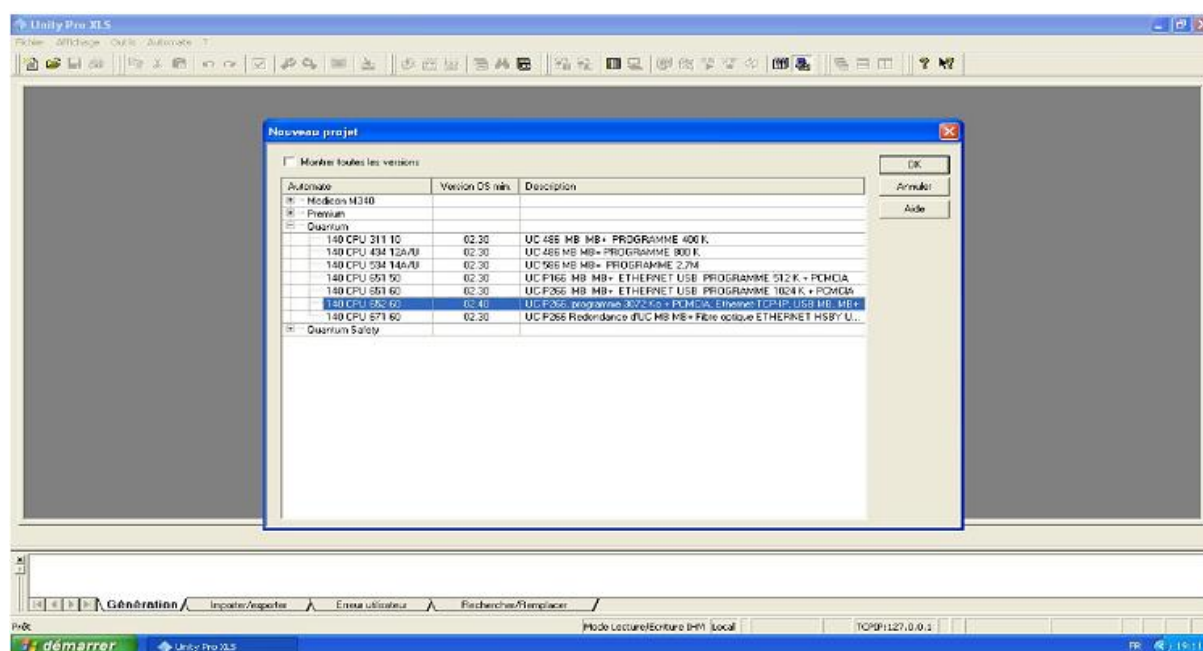


Figure II-6 : choix d'une CPU [6]

Le choix d'une CPU n'est pas irréversible, on peut ainsi remplacer une CPU déjà existante par une autre qui doit appartenir à la même station d'automate que la précédente mais cette opération ne peut se faire que si l'automate n'est pas connecté (mode local). Si l'emplacement de la CPU est incorrect ou bien certains modules d'entrée/sortie ne sont pas pris en compte par la CPU, un message d'erreur est généré.

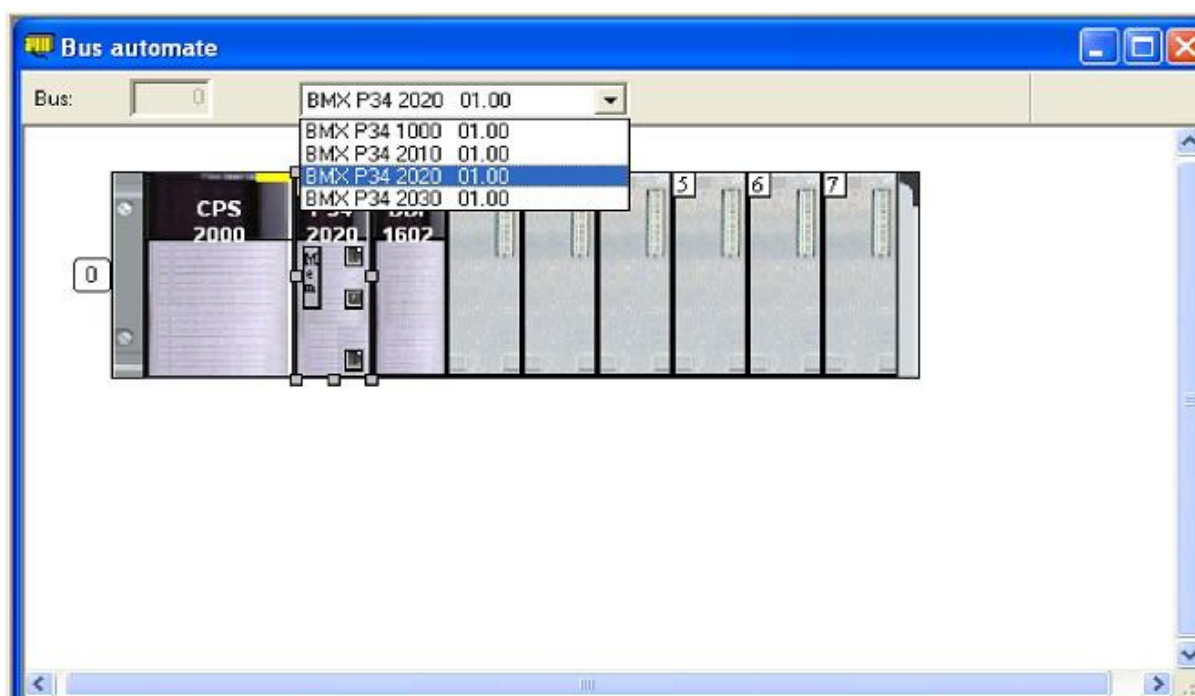


Figure II-7 : Remplacement d'une CPU [6]

II-2-1-2 Configuration d'un rack :

La configuration matérielle permet de choisir entre plusieurs types de racks différents et de les adresser. Dès la création d'un projet, un rack est sélectionné par défaut et possédant l'adresse suivante :

- 0 pour un automate de la famille Premium/Atrium ou Modicon M340.
- 1 pour un automate de la famille Quantum.

Le nombre de racks supporté par une station d'automatisation dépend de la nature de la CPU. Pour ajouter un rack, il suffit de faire glisser le rack déjà existant vers les zones 1 ou 2 comme le montre la figure II-8.

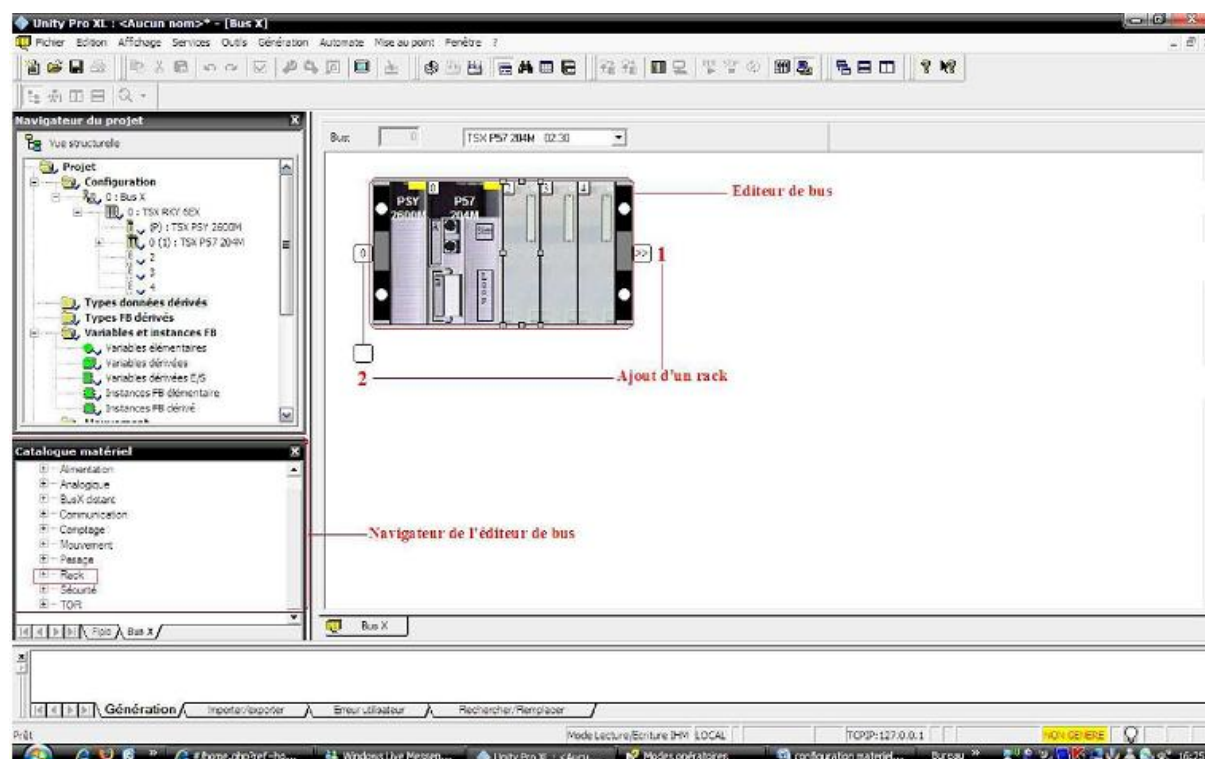


Figure II-8 : Ajout d'un rack [6]

- Si le rack est ajouté vers la direction 1, Il aura la même adresse (ici 0) et les modules seront numérotés à partir de la première puissance de 2 qui vient après le numéro du dernier module.
- Si par contre, le rack est déplacé vers la position 2, le rack aura une adresse différente et la numérotation des modules commencera à partir de l'adresse 0.

II-2-1-3 Choix du module d'alimentation :

Le choix d'un module d'alimentation dépend de plusieurs paramètres parmi lesquels figure le nombre et le type de modules d'entrée/sortie à alimenter. Le choix du module d'alimentation se fait à partir du catalogue matériel comme le montre la figure II-9.

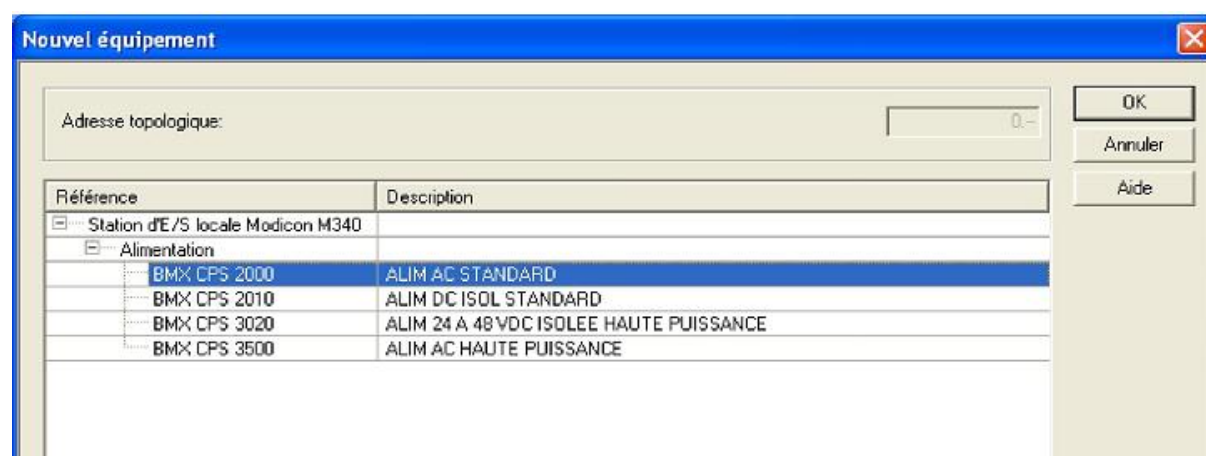


Figure II-9 : choix d'une alimentation [6]

II-2-1-4 Choix de modules d'Entrée/sortie, de communication et de modules métiers :

Le catalogue matériel permet à l'utilisateur de choisir d'autres modules à insérer dans le rack. Parmi ces modules il y a : les entrées/sorties TOR ou analogique, les modules de communication et les coupleurs ainsi que les modules métier (comptage, commande d'axes, commande pas à pas, régulation...)

Pour choisir par exemple un module analogique particulier (suivant sa référence unique) il suffit de déployer l'ensemble des références disponibles en cliquant sur le petit icône « + » puis en double cliquant sur le module recherché comme le montre la figure II-10.

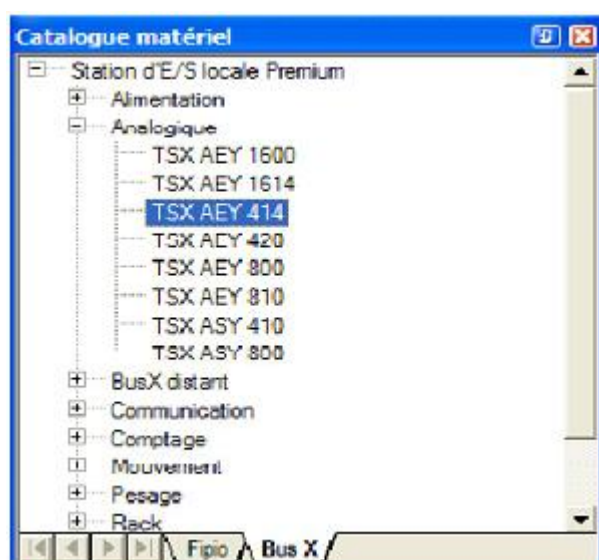


Figure II-10 : exemple de choix d'un module analogique [6]

En double cliquant sur le module choisi dans l'éditeur de configuration matérielle, une fenêtre s'ouvre donnant une description du module.

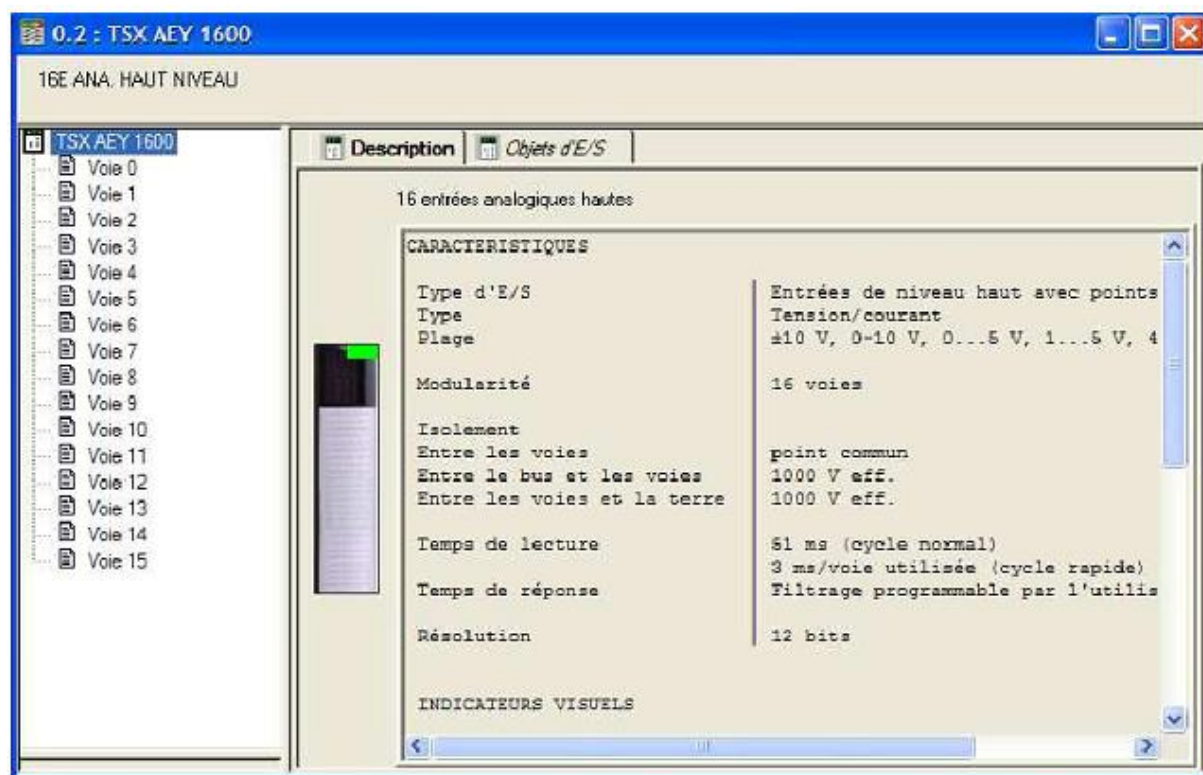


Figure II-11 : Fenêtre décrivant le module d'entrée/sortie choisi [6]

Pour un module d'entrée/sortie l'onglet « Objet d'E/S » sur la figure précédente permet de configurer le module de la façon suivante :

- Choix d'un nom pour chacune des voies ce qui implique la création automatique d'une variable élémentaire dans la table des variables.
- Choix du type de la voie : entrée/sortie tension (0-10V, 0-5V...) ou courant (4-20 mA, 0-20 mA...)
- Mise à l'échelle de la voie.
- Filtrage de la voie.

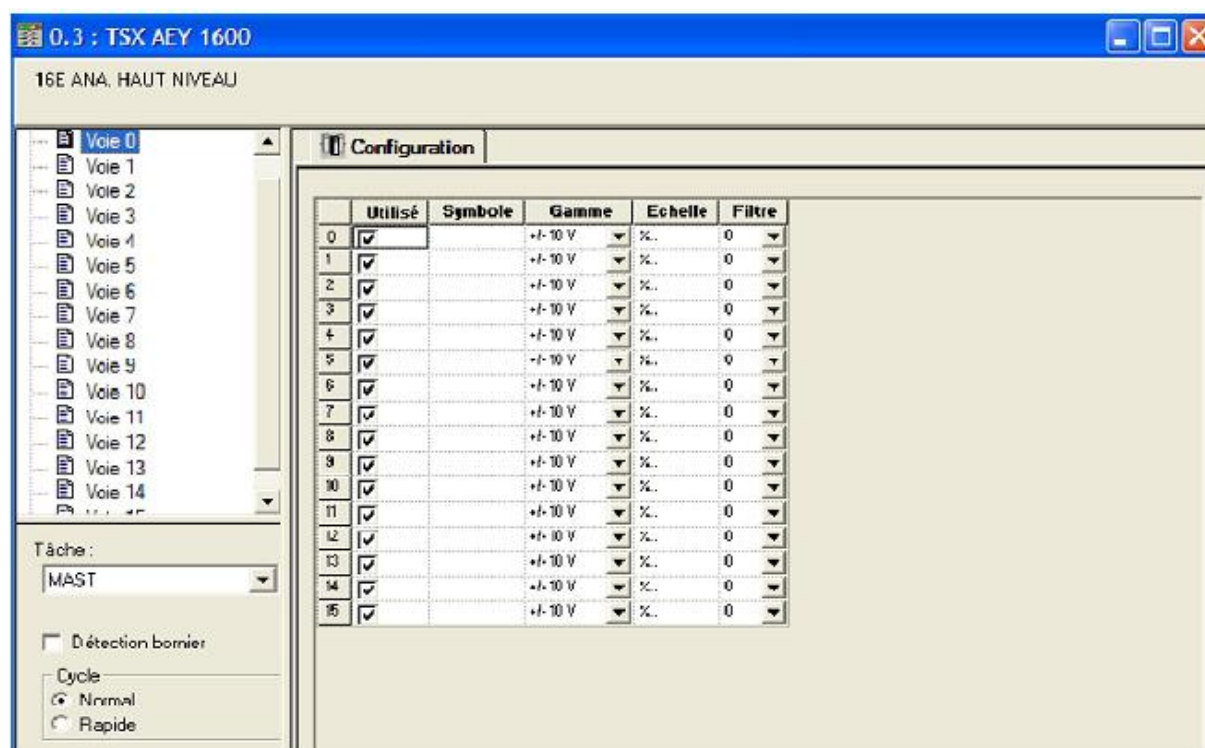


Figure II-12 : *Exemple de configuration d'une voie [6]*

II-2-2 Configuration d'un réseau :

Dans la plupart des applications d'automatisation, la définition et la configuration d'un réseau de communication devient de plus en plus inévitable du fait du développement exponentiel de la technologie des réseaux et son utilisation dans le milieu industriel ou dans notre cas pour automatiser un bâtiment.

- En premier lieu, il faudra créer un réseau logique et lui donner un nom. Cette étape s'effectue en sélectionnant « communication » dans le navigateur de projet puis en cliquant avec le bouton droit de la souris sur « réseaux » et en sélectionnant « nouveau réseau » comme le montre la figure II-13.

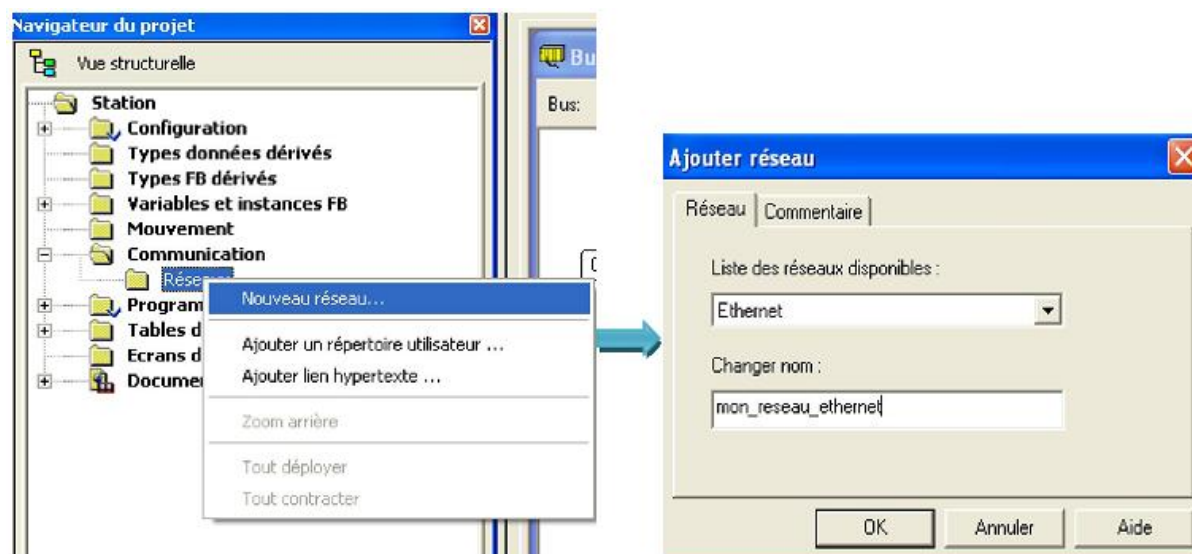


Figure II-13 : *Création d'un nouveau réseau logique [6]*

- On choisit ensuite le type de réseaux dans la liste des réseaux et on lui donne un nom significatif.
- On double clique ensuite sur le nom du réseau logique créé. Une nouvelle page apparaît qui va nous permettre de configurer ce réseau logique (adresse IP, type de module de communication...).

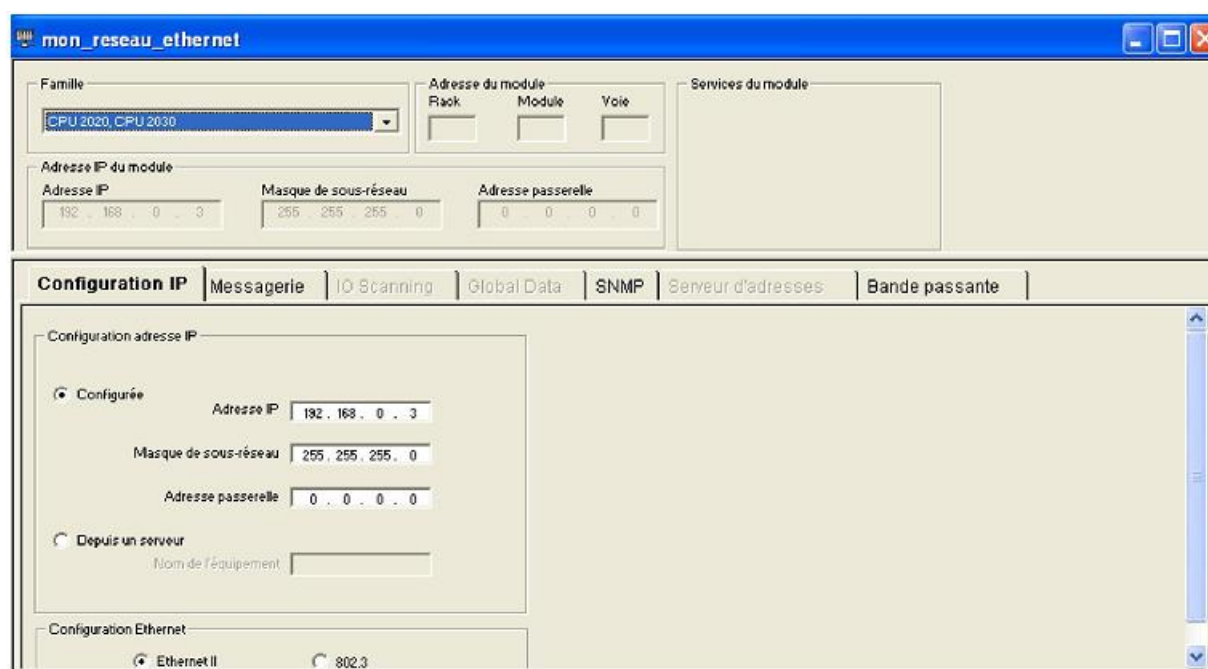


Figure II-14 : *Configuration du réseau logique [6]*

Après avoir créé le réseau logique, il faut l'associer soit à un module de communication soit à la CPU. Si le réseau est associé à un module de communication, il faut ouvrir le module puis sélectionner la voie sur laquelle sera branché le câble réseau et lui affecter le nom du réseau logique créé (dans le répertoire Fonction). Un exemple est donné par la figure II-15.

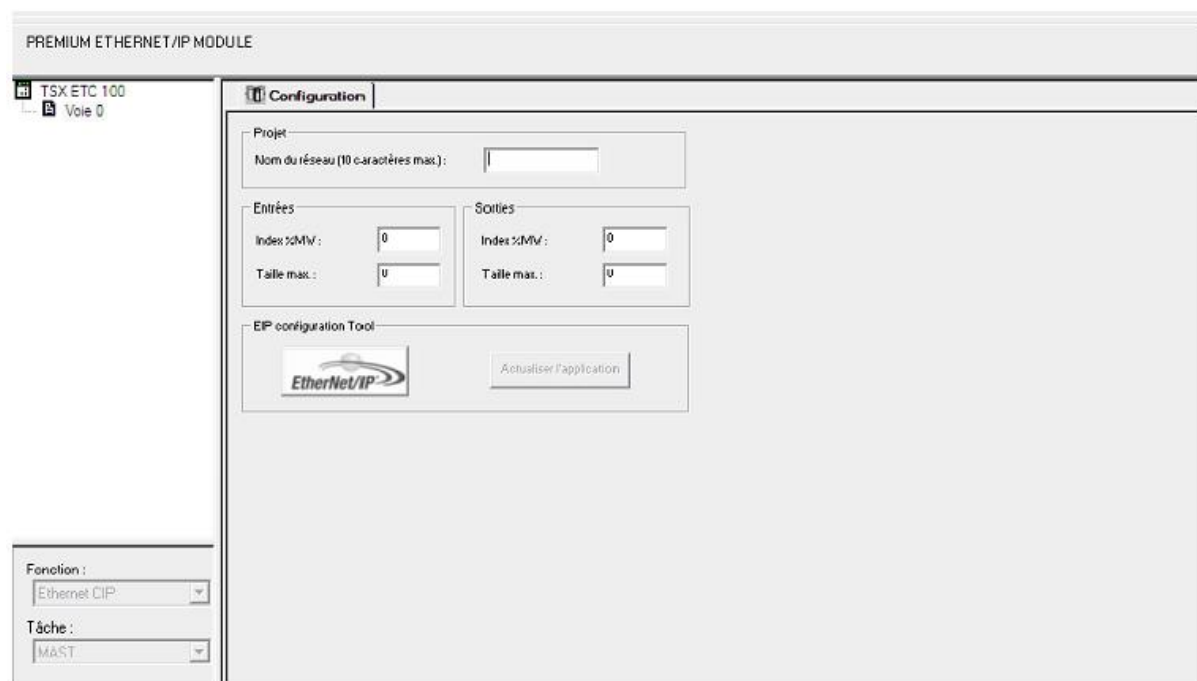


Figure II-15 : Association d'un réseau logique à un module de communication [6]

II-3-3 Définition et édition des variables :

II-3-3-1 Définitions :

Une variable est une entité mémoire qui peut être modifiée par le programme lors de son exécution. Elle peut être de différents types : BOOL (booléenne), WORD (mot de 16 bits), DWORD (mot de 32 bits)....

Une variable peut être affectée ou non. Une variable affectée possède un emplacement mémoire bien défini et connu de l'utilisateur par contre une variable non affectée ne possède pas une adresse figée, cette dernière peut changer à chaque cycle.

Une variable publique est une variable associée à certains blocs fonction et qui contient des valeurs qui ne sont pas influencées par le processus, elles peuvent par exemple être utilisées pour paramétrer le bloc fonction.

Une variable privée est une variable utilisée par le bloc fonction, lors du traitement de son algorithme, et qui est inaccessible à l'utilisateur.

II-3-3-3 Adressage des variables dans Unity Pro :

Une variable dans Unity Pro peut être adressée de différente manière selon qu'elle se trouve en mémoire interne de l'automate ou qu'elle est recueillie (ou émise) à travers un périphérique d'entrée (resp. de sortie). La syntaxe d'une instance de données (en adressage direct) est définie par le symbole % suivi d'un préfixe de localisation mémoire et dans certains cas d'informations supplémentaires. Le préfixe de localisation mémoire peut être:

- M, pour les variables internes.
- K, pour les constantes (Premium et Modicon M340).
- S, pour les variables systèmes.
- N, pour les variables réseaux.
- I, pour les variables d'entrées.
- Q, pour les variables de sorties.

%	I / Q	X / W / D	r	.	m	.	c	.	d	.	j
Symbole	Type	Format	Rack		Module		Voie		Rang		Bit
	I = Entrée Q = Sortie	X = Booléen W = Mot D = Double mot									

Tableau II-1 : Adressage physique des entrées / sorties [7]

II-4 Exemple d'application :

II-4-1 Cahier de charge :

Nous disposons d'un bureau muni de trois rangées de luminaires pour l'éclairage et de trois détecteurs de mouvement (chacun associé à une rangée) servant comme détecteur de présence. On veut que l'éclairage soit commandé par le détecteur de mouvement afin de

l'éteindre lorsqu'aucune présence n'est enregistrée pour rendre ainsi la consommation d'énergie plus rationnelle.

- Une rangée de luminaire est allumée si le détecteur de mouvement associé détecte une présence, sinon elle est éteinte au bout de 2 mn après la dernière détection de mouvement.
- De plus, le bureau dispose aussi de trois boutons poussoirs permettant l'allumage d'une rangée de luminaire même si le capteur de mouvement associé est désactivé (aucune présence n'est détectée), ces boutons peuvent être utiles lorsqu'un occupant souhaite allumer d'autres rangées que celle sous laquelle il travaille dans le but d'augmenter la luminosité.
- L'allumage d'une rangée par un bouton poussoir sera activé pour une durée déterminée (20 mn), au bout de cette durée, les luminaires de la rangée vont s'éteindre dans le cas où le bureau est vacant.
- La lumière des luminaires les plus proches de la fenêtre est modulée (régulée) en fonction de la lumière du jour et ceci en utilisant un capteur de lumière et un régulateur PI.

Le GRAFCET matérialisant ce cahier de charge est illustré à la figure II-17

Le bureau tel qu'il a été conçu sous vjeo citect (que nous verrons plus loin) est illustré à la figure II-16.

- Capt : désigne le capteur de mouvement pour chacune des rangées (1,2 et 3)
- BP : indique un bouton poussoir. • rg1 et rg2 : si l'une de ces sorties vaut 1 alors il y aura allumage de la rangée de luminaire correspondante.
- reg3 : cette sortie si elle est à 1, activera la régulation de la lumière de la 3^{ème} rangée de luminaires en fonction de la lumière du jour.
- LT1 à LT6 : lancement d'une temporisation.
- T1 à T6 : fin de la temporisation.

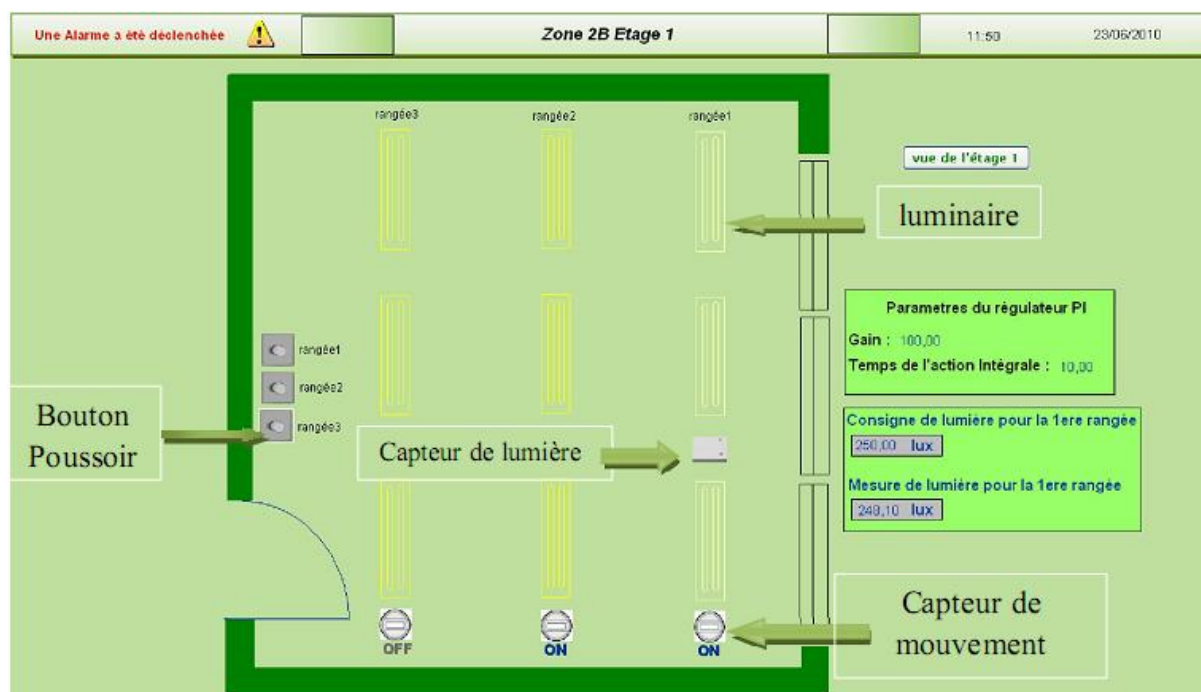


Figure II-16 : Représentation du bureau

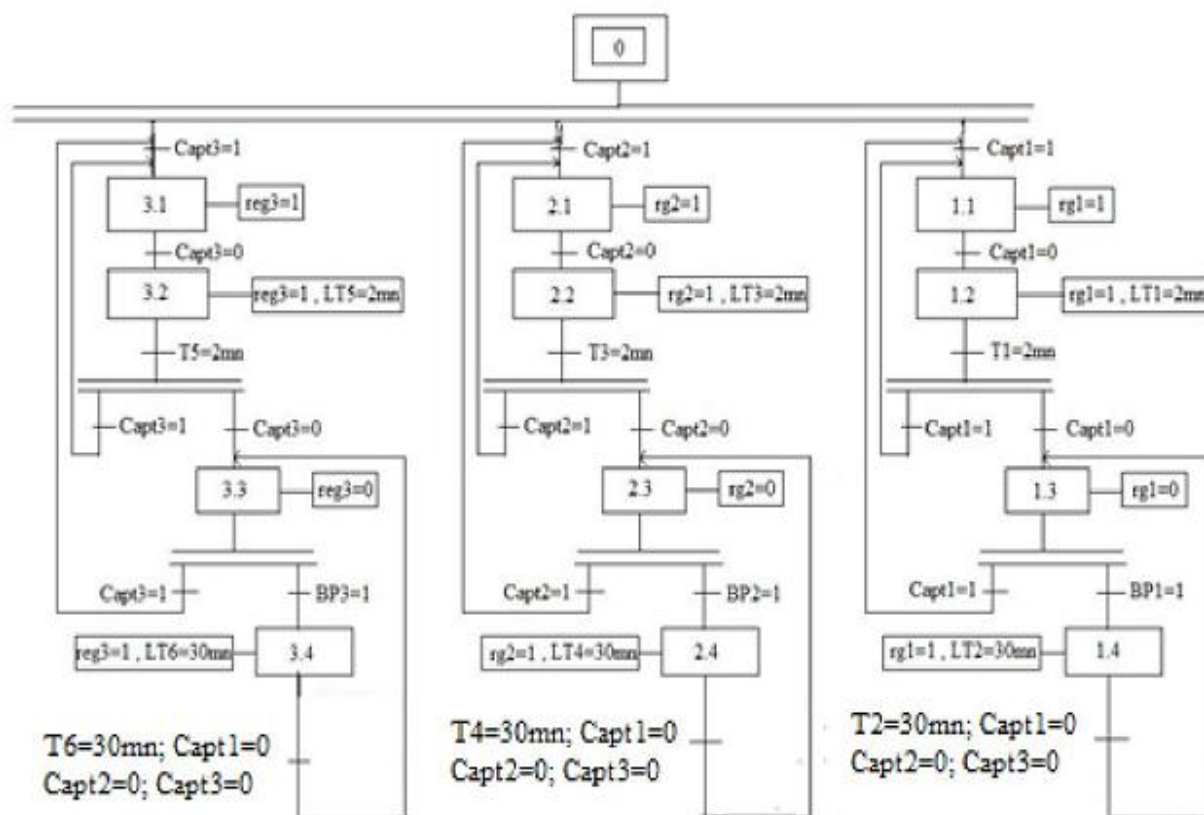


Figure II-17 : GRAFCET de l'exemple du bureau

II-4-2 Traduction du GRAFCET en équations logiques :

Nous allons programmer le GRAFCET précédent en langage FBD. Pour cela, nous allons affecter à chaque état une bascule RS ayant comme entrée S (Set) et R (Reset) dont nous allons donner les équations logiques.

Puisque le GRAFCET précédent est constitué de 3 branches semblables, nous allons uniquement donner les équations de la 1ère branche :

$$\begin{array}{ll}
 X_0 \left\{ \begin{array}{l} S_0=0 \\ R_0=(\text{Capt1}+\text{Capt2}+\text{Capt3}).X_0 \end{array} \right. & X_{1,1} \left\{ \begin{array}{l} S_{1,1}=(\text{Capt1}).X_0+(\text{Capt1})R_{1,2}+(\text{Capt1}).X_{1,3} \\ R_{1,1}=(\text{Capt1})X_{1,1} \end{array} \right. \\
 X_{1,2} \left\{ \begin{array}{l} S_{1,2}=R_{1,1} \\ R_{1,2}=(T1=2mn).X_{1,2} \end{array} \right. & X_{1,3} \left\{ \begin{array}{l} S_{1,3}=(\overline{\text{Capt1}}).R_{1,2} \\ R_{1,3}=(\text{BP1}+\text{Capt1}).X_{1,3} \end{array} \right. \\
 X_{1,4} \left\{ \begin{array}{l} S_{1,4}=(\text{BP1}).X_{1,3} \\ R_{1,4}=(T2=30mn).X_{1,4} \end{array} \right. &
 \end{array}$$

II-4-3 Programmation de l'exemple en FBD :

Pour pouvoir programmer notre exemple en FBD, nous aurons besoin de définir la **temporisation** :

La temporisation que nous avons utilisée pour ce programme se nomme dans l'Unity Pro AKF_TE qui est une temporisation à l'enclenchement.

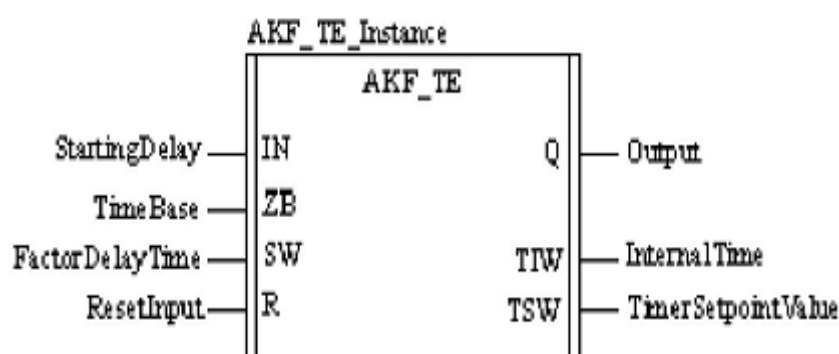
Le temps de retard est constitué d'une base temps ZB et d'un facteur SW. Le temps de retard est le résultat du produit ZB x SW.

La sortie TIW indique la valeur actuelle (mesure), qui se calcule de la manière suivante :
 TIW= temps écoulé / ZB.

Un front montant sur l'entrée IN valide le retard sur la sortie TSW, le temporisateur est lancé et l'état actuel apparaît à la sortie TIW.

Lorsque TIW est égal à TSW, la sortie Q est mise à 1. Un front montant sur l'entrée R ou front descendant sur l'entrée IN remet le temporisateur interne à zéro et la sortie Q passe à 0.

Le programme en FBD pour cet exemple est donné en Annexe C.



La description des différents paramètres ainsi que leurs types est résumée dans le tableau suivant :

Paramètre	Type de données	Description
IN	BOOL	Lancement de la temporisation
ZB	TIME	Base temps du retard
SW	INT	Facteur du retard
R	BOOL	Entrée RAZ

Description des paramètres des sorties :

Paramètre	Type de données	Signification
Q	BOOL	Sortie
TIW	INT	Temps interne (valeur actuelle de la temporisation)
TSW	INT	Consigne de la temporisation au moment du front 0 1 sur IN

Tableau II-2 : Paramètres de la temporisation AKF_TE

Conclusion :

Durant ce chapitre, nous avons vu que les automates programmables Schneider sont dotés d'un logiciel de programmation très performant qui est Unity Pro intégrant les cinq langages de programmation normalisés et possédant une interface graphique interactive et simple à utiliser.

Ce logiciel sera exploité par la suite pour la création des programmes de notre application.

Chapitre III

Introduction au Logiciel

SCADA Vijeo Citect

Introduction :

Dans un projet industriel, il est toujours nécessaire de faciliter le contrôle des différentes entités de l'usine (bâtiments) et ceci en introduisant des interfaces de communication et de supervision par le moyen de Pc ou des écrans tactiles programmés par des logiciel d'interface homme machine. Dans notre cas la supervision de gestion technique de bâtiment se fait avec logiciel Vigeo Citect.

III-1 Définition de la supervision :

La supervision industrielle consiste à surveiller l'état de fonctionnement d'un processus pour l'amener à se maintenir à son point de fonctionnement.

L'opérateur chargé de conduire une installation automatisée doit impérativement disposer en temps réel d'une visualisation de l'état et de l'évolution des paramètres du processus, qui lui permettent de prendre rapidement des décisions appropriées à ses objectifs, qui sont :

- Ø Qualité des produits.
- Ø Sécurité des biens et des personnes.

III-2 Fonctions de supervision :

Les fonctions de supervision sont les suivantes :

- Assurer la communication entre les équipements d'automatisation et les outils informatiques d'ordonnancement et de gestion de la production, pour lancer et gérer les différents programmes de production.
 - Aider l'opérateur à coordonner le fonctionnement d'un ensemble de machines enchainée constituant un îlot ou une ligne de production en assurant l'exécution d'ordre communs (marche, arrêt,...).
 - Assister l'opérateur dans leur opération de diagnostique et de maintenance préventive et curative.

III -3 Avantage de la supervision :

Le système de supervision sont des systèmes d'aide à l'opérateur ; leur but n'est pas de fonctionner en boucle fermé mais de présenter à l'opérateur des résultats expliquées et interprétées, de le guider dans les décisions qu'il doit prendre.

Les principales aides fournies par le système de supervision à l'opérateur sont :

- Le traitement des alarmes.
- La détection des défauts.
- Le diagnostic.
- La reprise en mode automatique ou en mode manuel.
- Assurer une gestion qualitative et quantitative de la production, cette tâche nécessite la collecte en temps réel de nombreuses informations, leur archivage et leur traitement immédiat ou différé.

III-4 Présentation de logiciel utilisé (Vegio Citect) :

Vijeo Citect est un logiciel pour un système d'acquisition et de contrôle des données (SCADA ou Supervisory Control And Data Acquisition), ce logiciel permet de gérer et de surveiller des processus de fabrication, de distribution et de gestion d'installations.

Il est à noter que le logiciel Vijeo Citect est un logiciel polyvalent pouvant être utilisé aussi bien dans l'industrie que pour la Gestion Technique du Bâtiment, son utilisation en GTB permet de gérer l'ensemble de l'immeuble, surveiller la consommation d'énergie et les différentes alarmes, commander l'ouverture ou la fermeture des portes, commander l'éclairage, modifier la consigne de température pour les systèmes CVC..., etc.

III - 5 Caractéristiques et avantages du logiciel Vijeo Citect :

Le logiciel Vijeo Citect dispose de plusieurs caractéristiques faisant de lui une solution de choix pour la supervision des systèmes de GTB, parmi ces caractéristiques on peut citer les suivantes :

- Analyse et présentation de tendances historiques et en temps réel sous forme de courbes et d'histogrammes (tel que la consommation d'énergie électrique des différents secteurs d'un bâtiment).

- Gestion des alarmes (alarmes relatives aux systèmes de sécurité (anti-intrusion), alarmes déclenchées par des pannes d'équipements tel que les équipements de CVC), de plus le logiciel permet aussi la génération de rapports donnant l'état des équipements ainsi que les différents événements enregistrés (alarmes, pannes,...etc.).
- Deux langages de programmation Cicode (langage évolué) et CitectVBA.
- Eléments graphiques contenant Génie, Super Génie et objets ActiveX (ces éléments vont être explicités plus loin dans ce chapitre).
- Possibilité d'utiliser des *grappes (ou Cluster)* permettant de regrouper plusieurs systèmes de contrôle en un seul système, ce qui permet de diminuer les coûts, ainsi, si on utilise plusieurs systèmes de supervision pour des sites différents, on peut utiliser un Cluster pour les regrouper en une seule station permettant de superviser les différents sites et donnant ainsi la possibilité de comparer les données et tendances des différents sites.

III-6 Outils et composantes de Vijeo Citect :

L'architecture Vijeo Citect peut être divisée en deux domaines de fonctionnalité :

1. configuration
2. exécution
3. pilotes

III-6-1 Outils de configuration :

Pour créer et configurer un projet sous Vijeo Citect, on utilise les outils suivants :

III-6-1-1 Explorateur Citect :



Application utilisée pour créer et gérer les projets Vijeo Citect. L'Explorateur Citect affiche une liste de tous les projets qui peuvent être renommé, sauvegardé ou supprimé.



III-6-1-2 Editeur de projet :

Cette application est utilisée pour créer et gérer les données de configuration relatif à un projet : tags (ou variables), les alarmes, éléments de système, éléments de communication (serveurs d'E/S, Cluster...) etc.

**III-6-1-3 Editeur graphique Citect :**

Application utilisée pour dessiner, créer et modifier les éléments graphiques d'un projet : modèles, objets graphiques, symboles, génies, super génies, etc. (la définition de ces éléments va être vue en détail plus loin), les éléments graphiques qu'on peut visualiser dans l'écran d'un superviseur sont créés grâce à cette application.

III-6-2 Outils d'exécution :

Les outils suivants permettent d'exécuter, de surveiller et de piloter les projets en phase d'exécution, Vijeo Citect dispose des outils d'exécution suivants :

**III-6-2-1 Vijeo Citect Client Web:**

Cet outil permet d'afficher un projet Vijeo Citect dans un navigateur. Cet outil permettra donc par exemple à un manager d'accéder aux ensembles de données relatives à un projet de n'importe quel poste sur le réseau ayant des droits d'accès en lecture ou en lecture/écriture et ceci sans avoir recours à un logiciel supplémentaire.

**III-6-2-2 Gestionnaire d'exécution Vijeo Citect :**

Application utilisée pour gérer et piloter la configuration de l'UC du projet et l'état d'exécution de chaque élément.

**III-6-2-3 Analyseur de processus :**

Contrôle Active X permettant de comparer et d'analyser des tendances historiques en temps réel et des données d'alarme en phase d'exécution.

III-6-3 Les pilotes :

Vijeo Citect peut communiquer avec toute une gamme de périphériques d'E/S, dont les automates, les contrôleurs de boucle et les systèmes de commande distribués.

Les pilotes permettent de communiquer avec des périphériques en utilisant divers protocoles de communication (Ethernet, TCP/IP, série, etc.) Le pilote définit les paramètres de projet requis pour que Vijeo Citect communique avec un périphérique donné. Ces paramètres concernent notamment les éléments suivants :

- Cartes
- Ports
- Périphériques
- Adressage des tags (variables)

III-7 Eléments d'un projet :

Un projet sous Vijeo Citect se construit en utilisant plusieurs éléments, cette subdivision facilite la conception et la mise en oeuvre des projets. Les éléments utilisés sont les suivants:

- Ø Éléments graphiques
- Ø Tags
- Ø Alarmes
- Ø Éléments système
- Ø Éléments de communication
- Ø Éléments des serveurs d'E/S
- Ø Cicode / CitectVBA

Par la suite on reviendra en détail sur chacun de ces éléments.

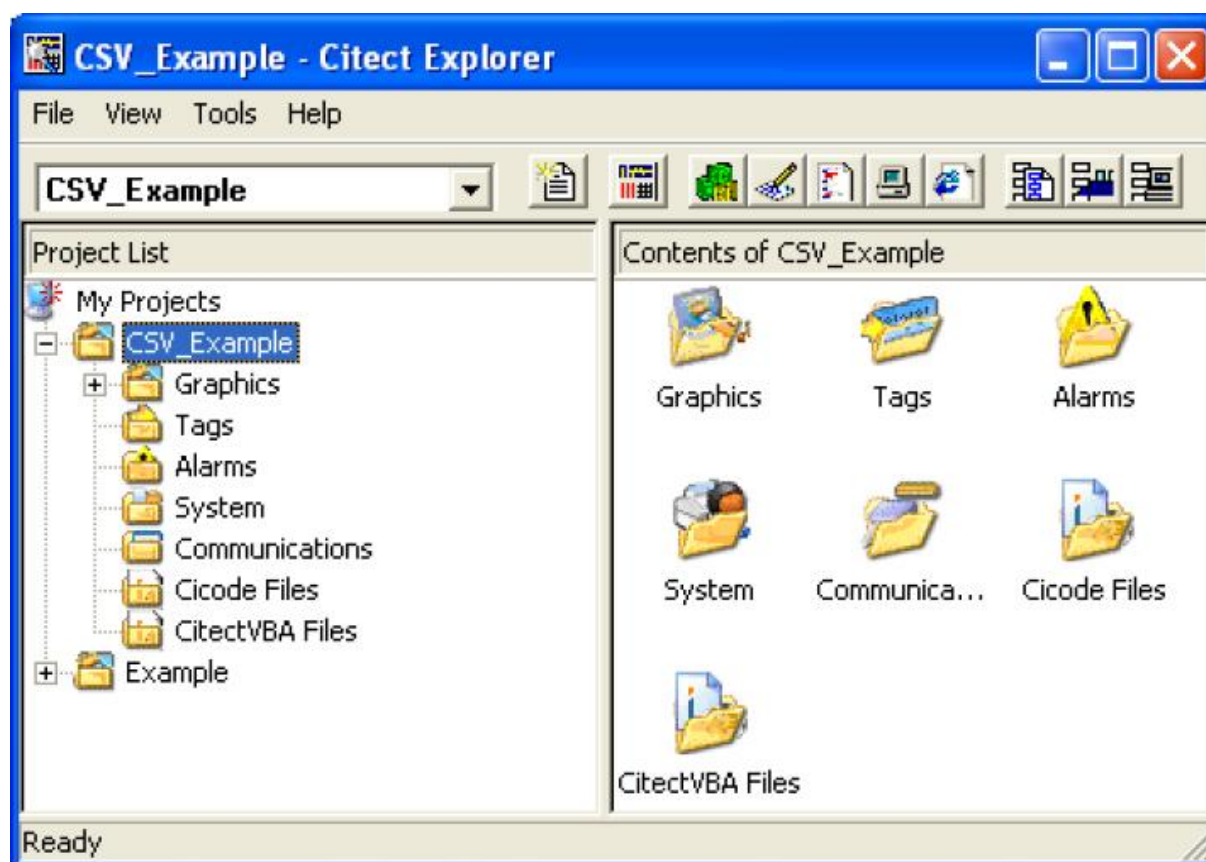


Figure III-1 *Eléments d'un projet sous Vijeo Citect [8]*

III-7-1 Eléments graphiques :

Les éléments graphiques d'un projet Vijeo Citect sont utilisés pour créer les écrans affichés sur les clients de visualisation. Ils sont décrits ci-dessous (pour le moment en se contentera par des définitions et plus loin nous allons illustrer certaines notions par des exemples)

a) *Les pages :*

C'est sur ces pages là qu'on peut créer et déposer tout les objets d'une application donnée.

b) *Les pages modèles :*

Ce sont l'ensemble de pages utilisées pour standardiser les écrans, elles permettent donc de créer les modèles de pages relatifs à un projet donné et de les personnaliser selon le gout du concepteur.

c) *Les symboles :*

Objets graphiques stockés dans une bibliothèque en vue de leur réutilisation.

d) *Les génies :*

Objets groupant plusieurs éléments graphiques et fonctionnels pour faciliter leur duplication.

e) *Les super génies :*

Génies pouvant transmettre des données spécifiques à un périphérique en phase d'exécution.

III-7-2 Les tags (Variables) :

Les tags permettent d'identifier les points de l'infrastructure à surveiller et à contrôler avec Vijeo Citect. Le nom donné à un tag devient une étiquette pour une adresse de registre, de sorte qu'il puisse être intuitivement appliqué aux pages graphiques. Un exemple de tag est la vitesse d'un ventilateur donné, la température au sein d'un bureau...etc.

III-7-3 Les Alarmes :

Les alarmes permettent d'identifier des conditions dans un système Vijeo Citect nécessitant une intervention (panne d'un équipement donné, dépassement de valeur d'un tag donné...etc.)

III-7-4 Éléments système :

Les éléments système d'un projet Vijeo Citect permettent de personnaliser, de gérer et de suivre le système superviseur. Ils comprennent :

a) *Les commandes clavier :*

Ce sont des séquences de touches associées à des instructions, ainsi, ces commandes peuvent être utilisées comme raccourci pour des instructions répétitives.

b) Les rapport :

Ils permettent de présenter l'état d'une installation et de donner des informations détaillées tel que l'état de certains équipements, la valeur de certaines variables (comme la température à l'intérieur des chambres, le niveau de consommation d'énergie dans certaines salles...etc.), ces rapports peuvent être donnés périodiquement ou suite à certain événement.

c) Les évènements :

Ce sont des commandes exécutées en réponse à des déclencheurs tels que des expressions Cicode ou des tags de variables, par exemple, un événement permet d'envoyer une commande d'ouverture de vannes de refroidissement lorsque la température dépasse une certaine valeur.

d) Les périphériques :

Ce sont des éléments pouvant transférer des données de haut niveau à d'autres éléments tels que des fichiers RTF, des fichiers ASCII et des imprimantes.

e) Les projets inclus :

Ce sont des projets prédéfinis contenant des configurations de base et qui sont automatiquement inclus dans les projets utilisateurs, l'utilisation de ces projets inclus permet au concepteur de projets de s'en passer de certains détails de configuration et d'utiliser directement les configurations prédéfinies dans les projets inclus au lieu de prendre le temps de les redéfinir.

III-7-5 Eléments de communication :

Les éléments de communication d'un projet Vijeo Citect correspondent à la représentation configurée du matériel de communication du système de supervision. Ils comprennent :

a) Les cartes :

Composant permettant divers types de communication avec des périphériques d'E/S, elle peut faire office de serveur d'E/S.

b) Les ports :

Connexion physique entre la carte et le périphérique d'E/S (comme le port COM d'un ordinateur).

c) Les périphériques d'E/S :

Ce sont les appareils communicant avec le systèmes de supervision à l'instar des API ou des contrôleurs décentralisés.

III-7-6 Eléments de serveurs d'E/S :

Les éléments des serveurs d'un projet Vijeo Citect correspondent à la représentation configurée des serveurs du système de supervision. Ils comprennent :

Ø Le cluster :

Le cluster (ou grappe) est un groupe logique de serveurs, ce regroupement permet de rendre ces serveurs accessibles à partir d'une seule station de supervision.

Ø Les serveurs d'E/S :

Serveurs de communication spécialisés échangeant des données entre les périphériques d'E/S et les clients de visualisation Vijeo Citect.

Ø Les serveurs d'alarmes :

Serveurs surveillant les alarmes et les affichant sur le(s) client(s) de visualisation approprié(s).

Ø Les serveurs de rapports :

Serveurs contrôlant le traitement des rapports.

Ø Les serveurs de tendances :

Serveurs contrôlant l'accumulation et l'enregistrement d'informations de tendances.

III-8 Création et configuration d'un nouveau projet sous Vijeo Citect :

Dans ce qui suit, nous allons illustrer par des exemples les principales notions liées à la création d'un nouveau projet, la configuration du matériel et l'utilisation des outils de VC.

III-8-1 Création d'un nouveau projet :

Pour créer un nouveau projet sous Vijeo Citect, on doit suivre les étapes suivantes :

1. Lancez l'Explorateur Citect, l'éditeur graphique et l'éditeur de projet se lancent automatiquement.
2. Cliquez sur le bouton **Nouveau projet** ou, dans le menu **Fichier**
3. Sélectionnez **Nouveau projet**.
4. Entrer un **Nom** pour le projet que vous désirez créer (obligatoire).
5. Entrer une **Description** ainsi que l'**Emplacement** où les fichiers du nouveau projet doivent être stockés.
6. Sélectionnez un **Style de modèles** et une **Résolution de modèles** pour définir l'apparence des pages graphiques.
7. Cliquez sur **OK** pour créer le projet ou sur **Annuler**.
Le nouveau projet est ainsi créé.

III-8-2 Configuration du nouveau projet :

Notre projet étant créé, on doit maintenant le configurer en définissant le cluster, les différents serveurs et en exécutant l'assistant de configuration de l'ordinateur. Ces configurations permettent au superviseur de communiquer avec les périphériques, elles permettent aussi de définir des réglages de base tel que l'activation ou la désactivation des rapports, événements, tendances...etc.

III-8-2-1 Définition de cluster et des serveurs :

Comme on l'a déjà mentionné, un cluster correspond au regroupement d'un ensemble de serveurs les rendant accessibles à partir d'une seule station. Pour définir un cluster, on doit suivre les étapes suivantes :

1. Dans l'Éditeur de projets, Sélectionnez **Serveurs | Clusters**.
2. Entrez le nom du cluster et éventuellement un commentaire dans la boîte de dialogue Cluster comme indiqué dans la figure ci-dessous.
3. Cliquez sur le bouton **Ajouter** pour ajouter un nouvel enregistrement ou sur **Remplacer** si vous avez modifié un enregistrement.



Figure III-2 : Définition d'un nouveau cluster [8]

Le cluster étant défini, on peut à présent définir nos serveurs, pour cela, on doit suivre les étapes suivantes :

1. Sur l'éditeur de projet Citect, on sélectionne **Serveurs**
2. On choisit le serveur à définir (serveur d'E/S, d'alarmes, de tendances ou de rapport)
3. On entre les paramètres du serveur (le cluster associé, le nom du serveur, l'adresse sur le réseau (à laisser vide si le serveur est logique, i.e. défini au sein de la station de supervision)).
4. On clique sur le bouton **Ajouter** pour ajouter un nouveau serveur ou sur **Remplacer** pour effectuer une modification.

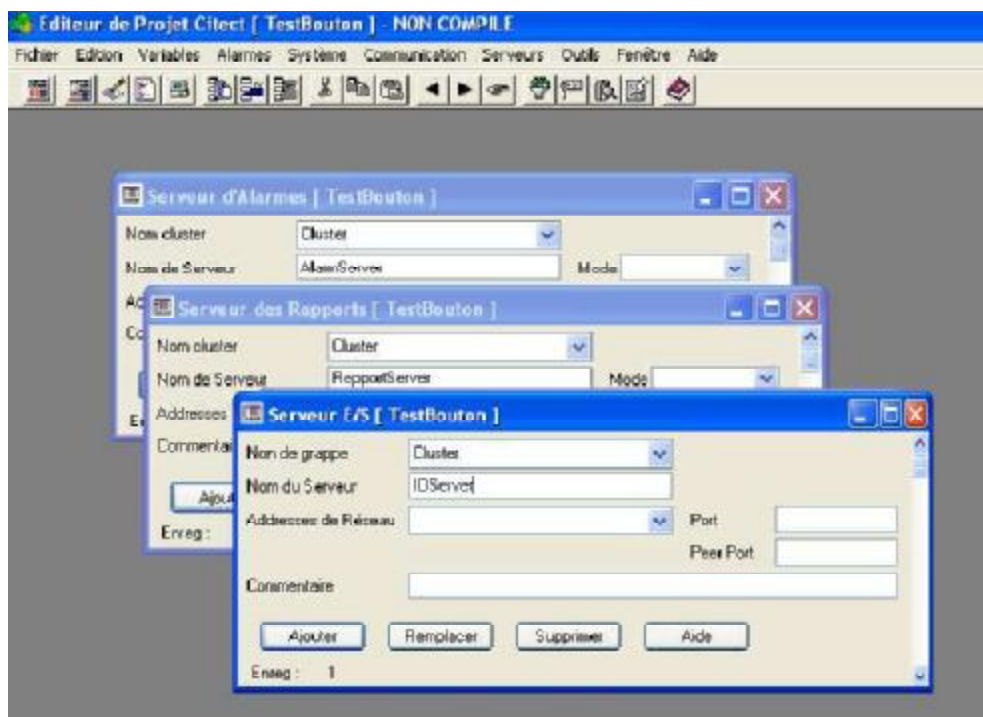


Figure III-3 : Définition des serveurs sur Vijeo Citect [8]

III-8-2-2 Exécution de l'Assistant de Configuration de Poste :

L'Assistant de Configuration de l'ordinateur permet de définir certains réglages de base tel que le rôle assigné à l'ordinateur, on effet, le poste de supervision peut être un serveur et un poste client à la fois pour les petits projets ne possédant pas des serveurs physiques, ou seulement un poste client (poste de visualisation) pour les projets où des serveurs dédiés (d'E/S, d'alarmes, de rapports,...etc.) ont été prévus.

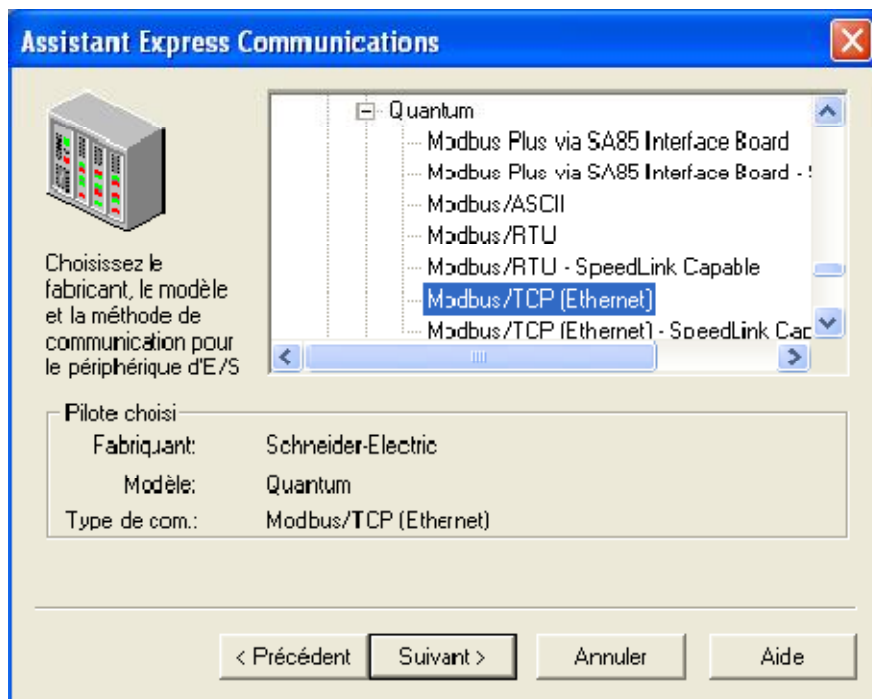


Figure III-4 : Choix du périphérique d'E/S [8]

Remarque :

Afin d'établir une communication entre le Vijeo Citect et le simulateur de l'automate disponible au niveau du logiciel Unity Pro, on doit saisir l'adresse ip statique du simulateur (adresse par défaut et figée) qui est : **127.0.0.1** comme indiqué par la figure ci-dessous.

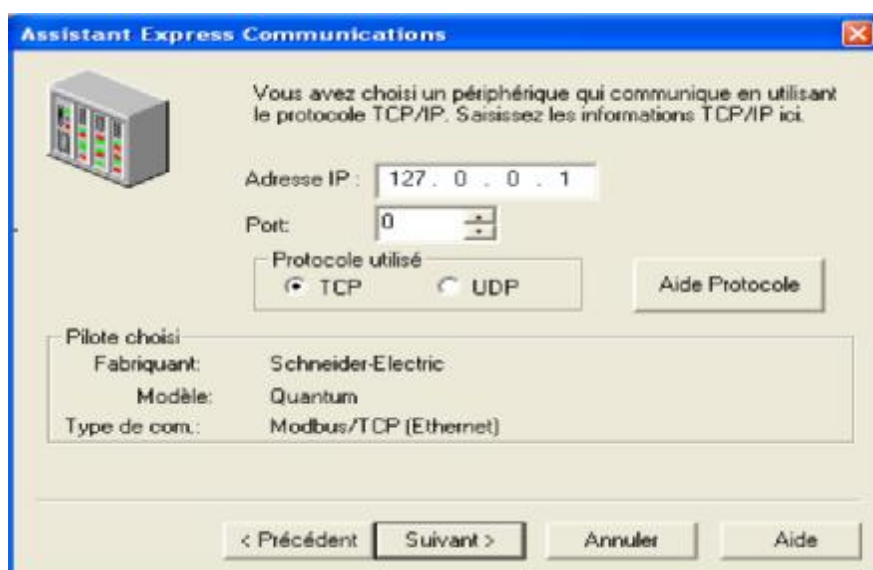


Figure III-5 : Configuration de l'adresse et du protocole de communication [8]

III-9 Les graphiques :

Nous allons maintenant entrer dans le vif du sujet et montrer comment créer les graphiques constituant un écran de supervision, pour cela, on considère une petite application à la base de laquelle on va illustrer tous les aspects en liaison avec la création des graphiques.

Cahier de charge :

Nous disposons d'un bureau muni de trois rangées de luminaires pour l'éclairage et de trois détecteurs de mouvement (chacun associé à une rangée) servant comme détecteur de présence, on veut que l'éclairage soit commandé par le détecteur de mouvement afin de l'éteindre lorsqu'aucune présence n'est enregistrée pour rendre ainsi la consommation d'énergie plus rationnelle.

- Une rangée de luminaire est allumée si le détecteur de mouvement associé détecte une présence, sinon elle est éteinte.
- De plus, le bureau dispose aussi de trois boutons poussoirs permettant l'allumage d'une rangée de luminaire même si le capteur de mouvement est désactivé (aucune présence n'est détectée), ces boutons peuvent être utiles lorsqu'un occupant souhaite allumer d'autres rangées que celle sous laquelle il travaille dans le but d'augmenter la luminosité.
- L'allumage d'une rangée par un bouton poussoir sera activé pour une durée déterminée (20 mn), au bout de cette durée, les luminaires de la rangée vont s'éteindre dans le cas où le capteur de mouvement indique que les bureaux sous la rangée sont toujours vacants.

III-9-1 Création de la page principale :

La figure suivante donne un aperçu de l'écran de visualisation de notre application sur la station de supervision.

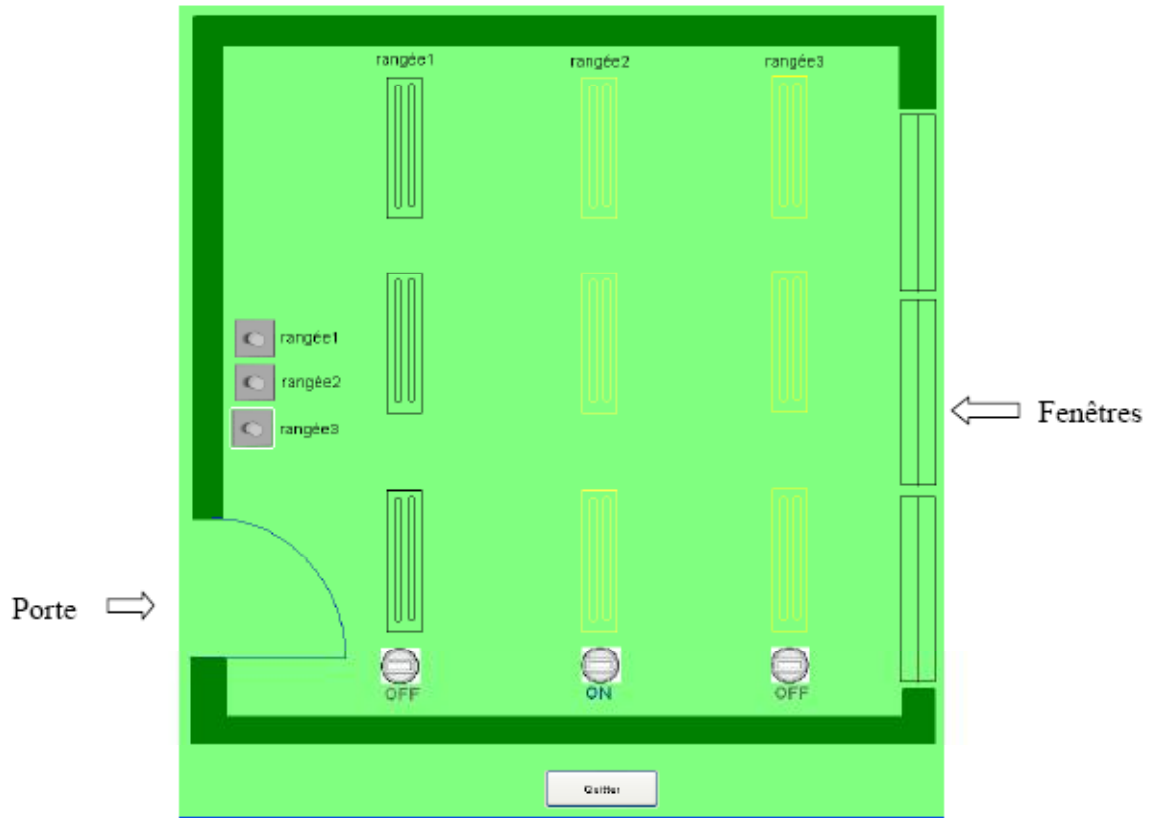


Figure III-6 : *Ecran de visualisation de l'application*

Sur cette figure, nous remarquons que la deuxième rangée est allumée puisque le détecteur de mouvement est activé, par ailleurs, quoique le détecteur de mouvement de la troisième rangée est désactivé, les luminaires de cette rangée sont allumés et ceci suite à une action sur le bouton poussoir de la rangée 3.

Après avoir créé et configuré le projet en suivant toutes les étapes listées dans la section précédente, nous ouvrons une nouvelle page dans l'éditeur graphique Citect, cette page comportera tout les éléments graphiques de notre application.

A présent, nous commencerons par concevoir les éléments de base de notre application (le contour, les fenêtres, la porte et les luminaires), ces éléments sont à concevoir en utilisant les outils de dessin de Vijeo Citect tel que les polygones, le cercle, le rectangle...etc. cette étape étant simple et ne nécessite pas plus d'étalement.

La deuxième étape consiste à définir tout les tags auxquels nous aurons besoin, ces variables permettent le transfert de données entre la station Vijeo Citect et le périphérique d'E/S.

Définition de tags sous Vijeo Citect :

Chaque tag (variable) est défini par son nom, son type (digital, real, ...etc.), le périphérique associé (qui doit être déjà défini lors de l'exécution de l'étape 'Assistant Express Communications') ainsi que son adresse sur le périphérique.

Nous allons à présent établir la liste de tous les tags à utiliser dans cette application. Tout d'abord, nous aurons besoin de tags permettant de déterminer l'état d'un capteur de mouvement, l'information qui détermine l'état du tag provient évidemment de l'automate auquel est connecté le capteur sur une carte d'entrée. Ces tags vont servir après pour déterminer l'expression à afficher au dessous du symbole du capteur ('ON' en bleu ou 'OFF' grisé). De plus, on aura aussi besoin de trois tags servant à transmettre l'action d'appui sur les boutons poussoirs pour allumer les luminaires. Ces six entrées vont servir à déterminer les trois sorties donnant l'état de chacune des rangées de luminaires, les sorties seront calculées grâce à un programme sur Unity Pro obtenu après traduction du cahier de charge en un grafcet.

Procédure pour définir un tag :

Sur l'éditeur de projet Citect, nous cliquons sur le menu tag, nous choisissons ensuite les variables élémentaires, la fenêtre suivantes s'affichera alors :

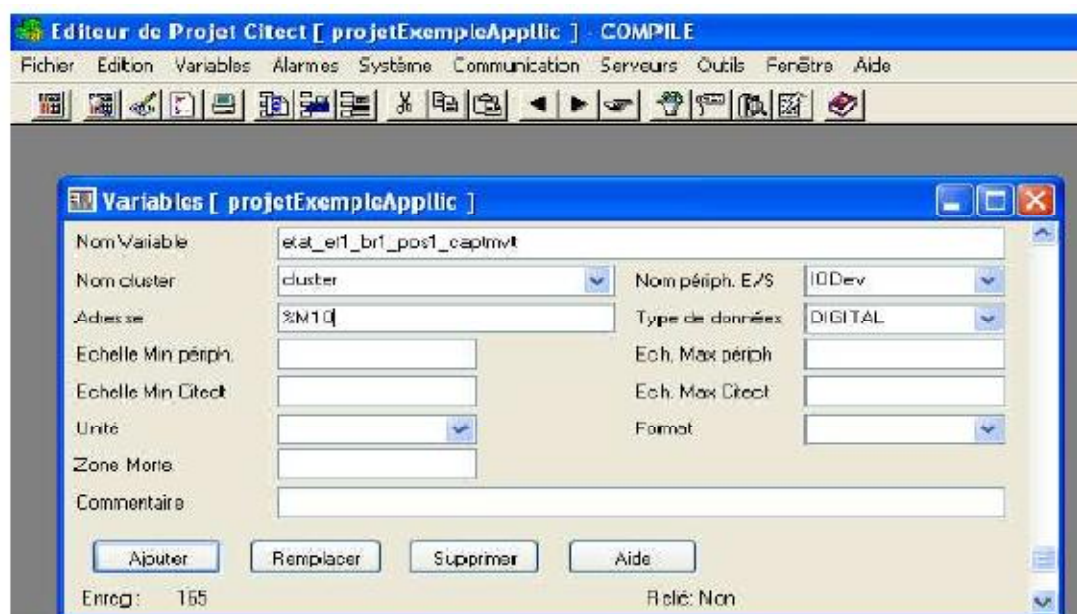


Figure III-7 : Définition de tags

La fenêtre qui s'affiche permet de saisir toutes les informations requise sur le tag :

- Le nom de la variable : ce nom doit être choisis de manière judicieuse et bien structuré, il est préférable qu'il comporte : le type de la variable (**état** d'un capteur (activé ou désactivé) ou bien **action** sur une sortie), l'emplacement et le type du périphérique relative à la variable.

Le cluster associé au projet.

- Le nom du périphérique (l'automate).
- L'adresse de la variable sur le périphérique.
- Le type de données transférées par le tag (digitale, real...etc.)

Le tableau suivant donne tout les tags auquel on aura besoin pour cette application :

Nom du tag	Type de données	Commentaire
etat_et1_br1_pos2_captmvt	DIGITAL	Tag donnant l'état du capteur de mouvement de la 1 rangée (position 1)
etat_et1_br1_pos2_captmvt	DIGITAL	Tag donnant l'état du capteur de mouvement de la 2 rangée (position 2)
etat_et1_br1_pos2_captmvt	DIGITAL	Tag donnant l'état du capteur de mouvement de la 3 rangée (position 3)
action_et1_br1_pos1_BP	DIGITAL	Tag servant à transmettre l'action sur le Bouton Poussoir 1 de la station de supervision vers le périphérique
action_et1_br1_pos1_BP	DIGITAL	Tag servant à transmettre l'action sur le Bouton Poussoir 2 de la station de supervision vers le périphérique

action_et1_br1_pos1_BP	DIGITAL	Tag servant à transmettre l'action sur le Bouton Poussoir 3 de la station de supervision vers le périphérique
etat_et1_br1_rg1	DIGITAL	Tag donnant l'état de la première rangée de luminaire (ON ou OFF)
etat_et1_br1_rg1	DIGITAL	Tag donnant l'état de la deuxième rangée de luminaire (ON ou OFF)
etat_et1_br1_rg1	DIGITAL	Tag donnant l'état de la troisième rangée de luminaire (ON ou OFF)

Tableau III-1 : liste des tags utilisés

Remarque : concernant les noms donnés aux tags, et1 signifie étage 1, et br1 signifie bureau 1, en effet, on suppose que le bureau de notre application est le premier bureau localisé dans le premier étage d'un bâtiment, ainsi, notre petite application peut être insérée dans un projet contenant plusieurs étages et plusieurs bureaux et l'attribution de noms structurés aux tags permettra une extension plus facile.

Jusqu'à présent, nous avons créé les graphiques simples (la carcasse de notre projet : frontière du bureau, fenêtre, porte...etc.) ainsi que les tags auquel on aura besoin, la prochaine étape consistera à créer les génies relatifs capteurs de mouvement et aux boutons poussoir.

III-9-2 Création des génies et des super génies:

Un génie est un élément graphique permettant de combiner plusieurs éléments entre eux, de les associer à des tags, et d'ajouter le tout ensuite (le génie) à la bibliothèque, dans ce qui suit, nous allons illustrer les génies par des exemples.

Pour créer un nouveau génie, on doit aller sur l'éditeur graphique Citect, sélectionner le menu **Fichier** puis cliquer sur nouveau, la fenêtre ci-dessous s'affiche alors et on doit cliquer sur le bouton **Génie**, la fenêtre permettant la création du génie s'affiche alors.

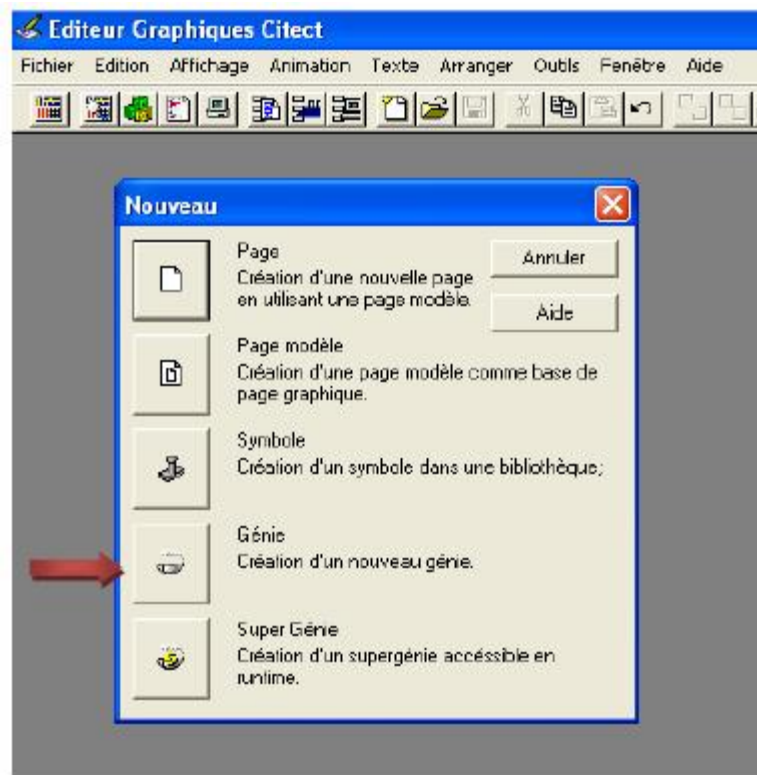


Figure III-8: *Création d'un nouveau génie*

III-9-2-1 Création des génies des capteurs de mouvement :

Après Avoir ouvert la page d'un nouveau génie en suivant les étapes décrit dans la section précédente, nous commencerons par éditer le symbole du capteur, et si nous disposons déjà d'un symbole à partir d'un fichier externe, il suffit alors de coller l'image en faisant un clique droit sur la page du génie en en choisissons **coller**.

La prochaine étape maintenant (et la plus intéressante) consiste à programmer les deux symboles **ON** et **OFF** pour qu'ils s'affichent suivant l'état du tag '*etat_et1_br1_posI_captmvt*' où $I = \{1, 2, 3\}$

Pour éditer les deux textes **ON** et **OFF**, on doit cliquer sur l'outil TEXT (symbole **A**) dans la barre d'outils sur la page du nouveau génie et ceci comme indiqué par la figure III-9, on positionne ensuite le curseur à l'endroit où le texte doit s'afficher, **on appui sur la touche espace**, ensuite on fait un double clique, la fenêtre des propriétés du texte s'affiche alors, on commencera par l'édition de l'apparence du texte comme indiqué sur la figure III-10, pour cela, on clique sur l'onglet **Affichage Valeur** indiqué par la flèche, on suite on édite le texte

VRAI et le texte FAUX et on fait entré le tag déterminant quel valeur le texte va prendre (le texte VRAI s'affiche si le tag choisi possède la valeur1).

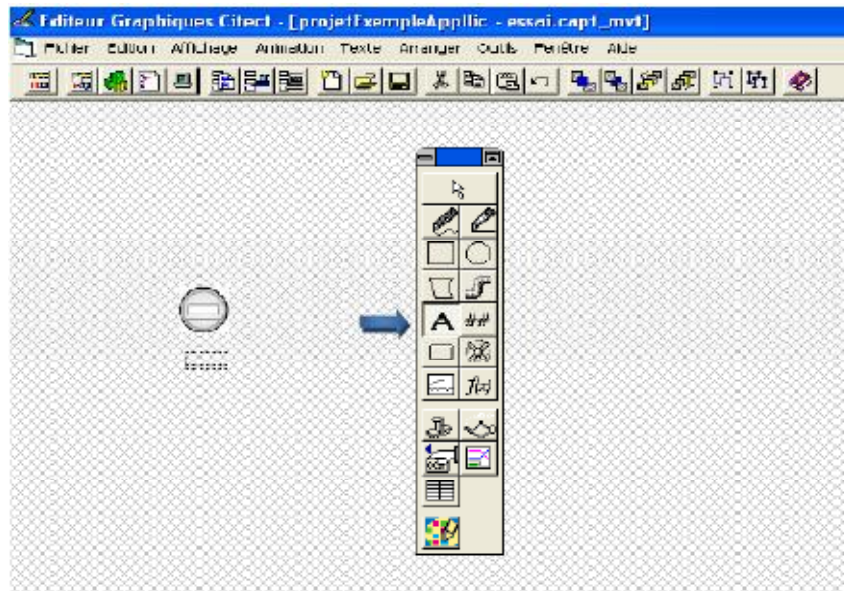


Figure III-9: Ajout d'un texte sous le symbole du capteur

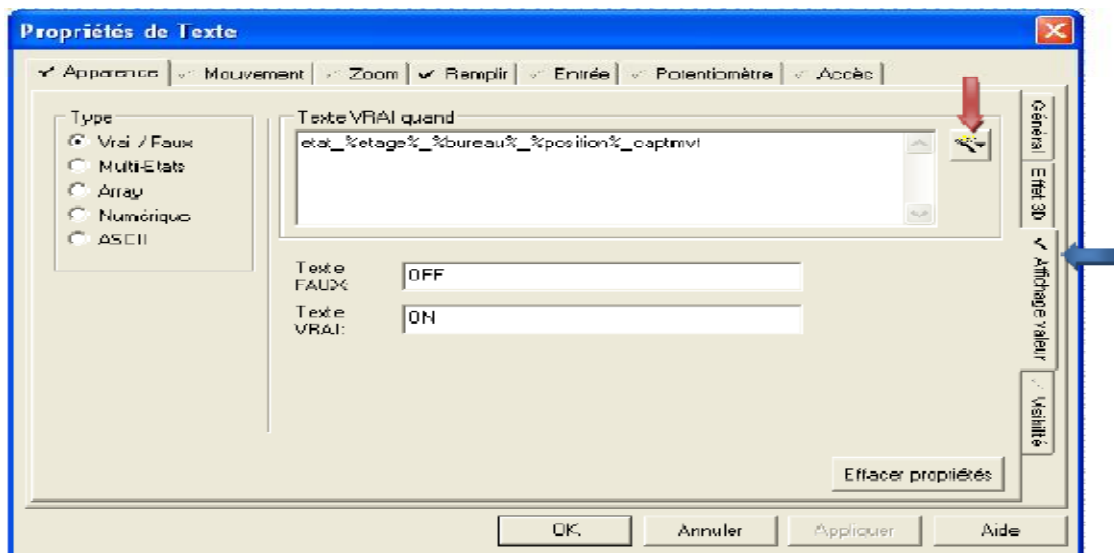


Figure III-10: Edition de l'apparence du texte

Choix du tag associé au texte : deux possibilités existent

1. Soit on choisit un tag en cliquant sur le bouton indiqué par la flèche rouge descendante, la liste de tous les tags créés s'affiche alors et on peut sélectionner le tag désiré. Cette option n'est pas intéressante car on impose ainsi des tags *statiques* et le génie crée ne peut alors être utilisé avec d'autres tag rendant ainsi son option de duplication inutilisable.

2. Soit utiliser des chaînes de substitution “%” comme indiqué dans la figure III-10, cela permet une attribution *dynamique* de tags et ceci lors de son appel dans la page principale, plus concrètement, si on fait comme dans l'exemple en écrivant “etat_%etage%_%bureau%_%position%_captmvt” dans la case *Texte Vrai quand*, et après enregistrement du génie dans la bibliothèque, si on colle le génie dans la page principale (on expliquera plus loin comment sauvegarder un génie et comment l'appeler de la page principale), un **popup** s'affiche et on peut alors entrer le tag associé au génie, la figure III-11 illustre le popup et les champs permettant d'entrer le tag, on voit donc que tout ce qui est entre les délimiteurs de substitution “%” sera affiché dans le popup et prendra la valeur entrée (etat_%etage%_%bureau%_%position%_captmvt sera donc remplacé par état_et1_br_1_pos2_captmvt).

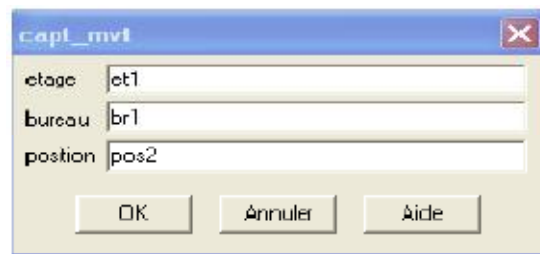


Figure III-11 : *Popup permettant d'associer un tag à un génie*

Jusqu'à présent, nous avons créé un génie comportant le symbole d'un capteur et un texte ayant deux valeurs (ON et OFF) et nous avons associé des tags *dynamiques* permettant de déterminer quelle valeur le texte prendra, Nous allons maintenant développer un peu plus notre génie en associant des couleurs au texte qui s'affiche ; pour cela, on clique sur l'onglet remplir puis sur couleur comme indiqué par les flèches sur la figure III-12 :

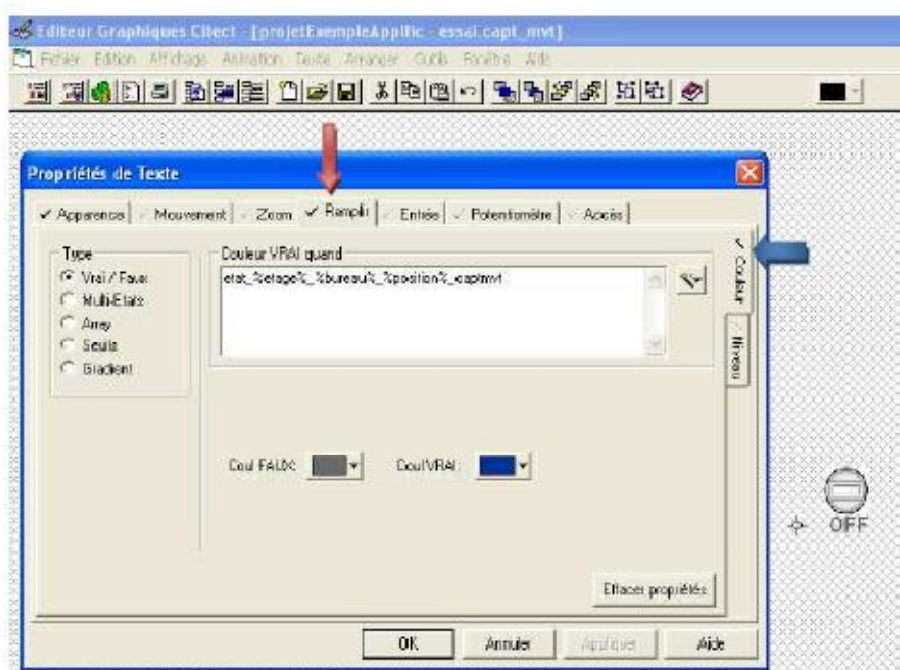


Figure III-12 : *Programmation de la couleur du texte*

Ainsi donc, suivant la valeur du tag associé au génie, le texte prend la couleur bleu pour ON ou gris pour OFF. Ceci termine donc la conception de notre génie, reste à montrer comment le sauvegarder et l'appeler ensuite de la page principale. Pour sauvegarder un génie, il suffit d'aller sur le menu **Fichier| Sauvegarder sous**, la fenêtre ci-dessous s'affiche alors, on peut choisir le nom du génie, la librairie (bibliothèque) dans laquelle il va être sauvegarder, et le projet auquel il appartient.

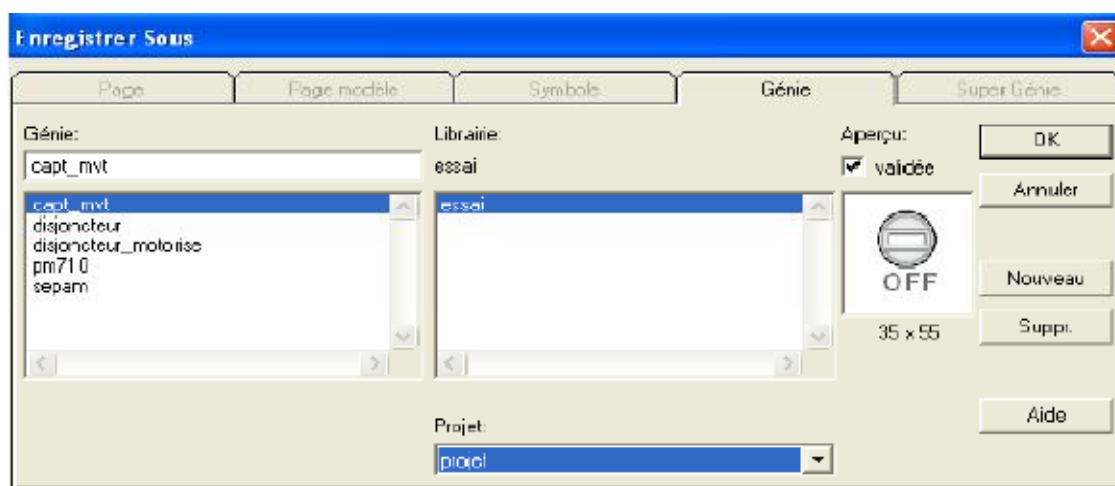


Figure III-13: *Sauvegarde d'un génie, association à une librairie et à un projet*

Coller le génie dans la page principale : pour ce faire, on sélectionne l'outil **génie** (indiqué par la flèche bleue) dans la barre d'outils de la page principale, on sélectionne ensuite la librairie à laquelle le génie appartient (essai) on choisit ensuite le génie désiré parmi les génies appartenant à cette librairie, cela étant fait, le popup déjà discuté s'affichera et nous permettra de lier des tags à notre génie.

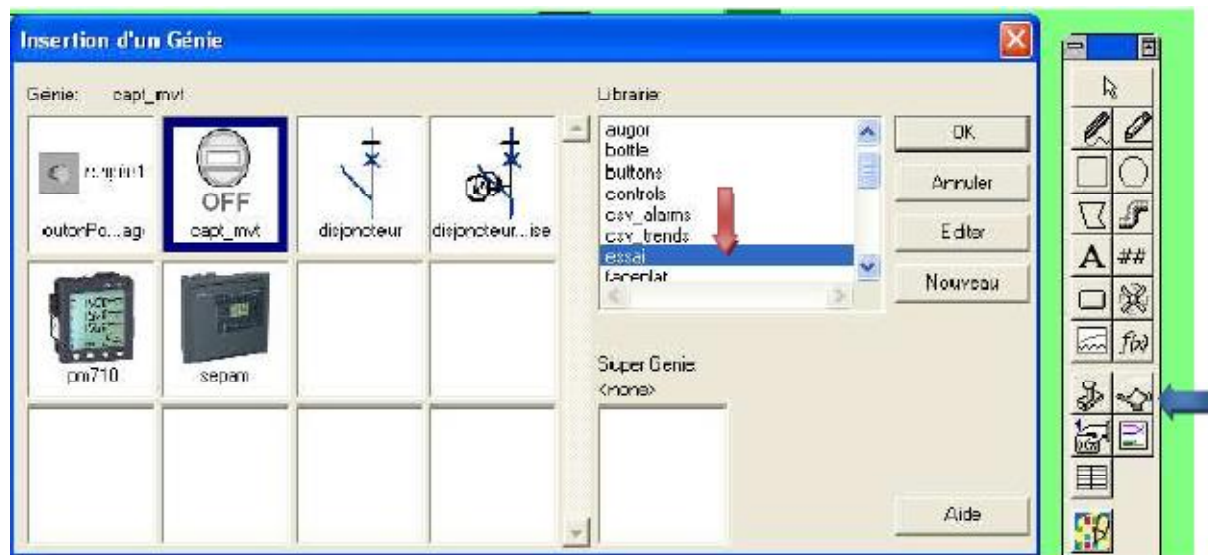


Figure III-14: Coller un génie dans la page principale

III-9-2-2 Création des génies et des supers génies des boutons poussoirs :

Comme nous l'avons déjà mentionné, les boutons poussoirs permettent de commander manuellement les luminaires et de les allumer (pour un temps déterminé) même en cas de non présence de personnes. Nous allons maintenant créer un génie comportant le symbole du bouton poussoir, en cliquant sur ce génie, on veut faire apparaître un popup menu (le super génie) comportant un bouton qui permet l'allumage d'une rangée de luminaire. Tout cela va s'éclaircir étape par étape.

Commençons par créer un nouveau génie (comme indiqué sur la figure III-8), sur la page du génie, on insère le symbole du bouton poussoir (image bitmap) ramené d'un fichier externe, pour configurer le génie de façon à faire apparaître le super génie après un clique, on utilise la fonction AssPopUP, comme indiqué sur la figure III-15, on clique donc d'abord sur **Entrée** puis **Clique** (fenêtre Propriétés du bouton, comme indiqué par els flèches), on active la case **Haut**, puis on introduit la fonction AssPopUp ainsi que ses arguments.

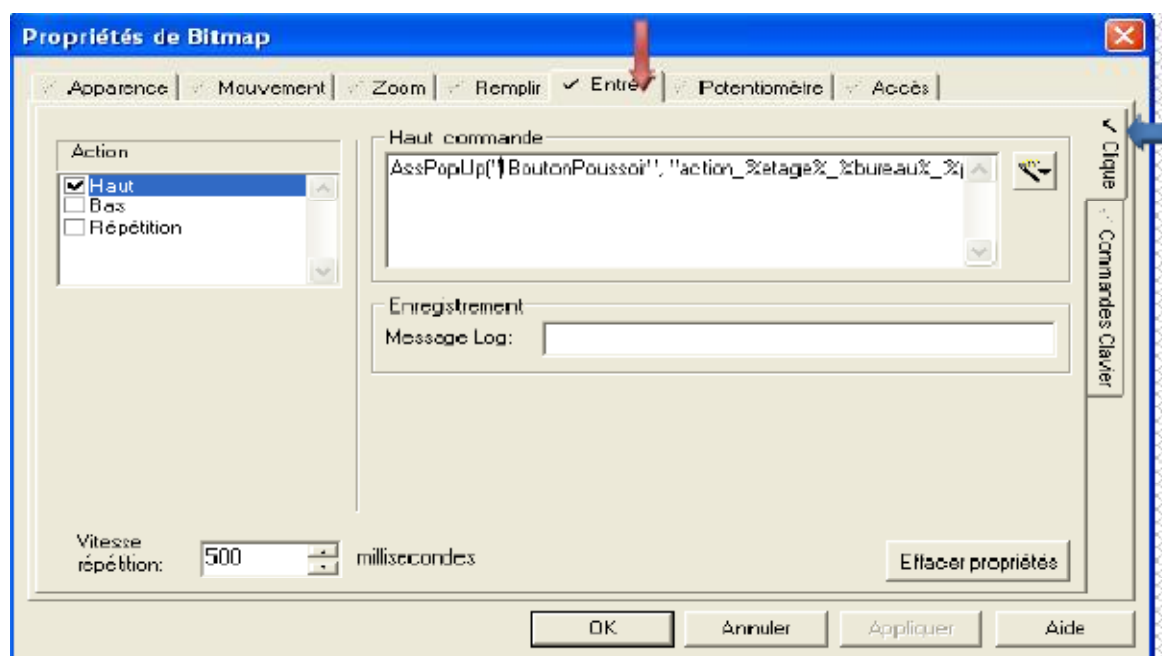


Figure III-15: Configuration du génie

La fonction AssPopUp admet comme arguments le nom du super génie à associé au génie ainsi que tous les tags que le génie doit transmettre au super génie. Sur le champ de commande, on doit introduire la fonction AssPopUp avec les arguments suivants :

AssPopUp("!BoutonPoussoir", "action_%etage%_%bureau%_%position%_BP"), où :

- !BoutonPoussoir représente le nom du super génie qu'on va créer après.
- *action_%etage%_%bureau%_%position%_BP* sont les tags que le génie doit transmettre au super génie.

III-9-2-3Création du super génie :

le super génie est un pop menu comportant en général des boutons de commande et qu'on associe à un génie. Pour créer un nouveau super génie, on doit aller sur l'éditeur graphique Citect, sélectionner **nouveau** dans le menu **Fichier**, puis sélectionner **super génie**, ceci étant fait, la page du super génie s'ouvre et on peut commencer sa configuration. On commence d'abord par ajouter un bouton qui permettra l'allumage de la rangée de luminaire (le bouton doit être ajouté dans la zone délimitée par un trait discontinu), ceci étant fait, on configure le bouton maintenant afin qu'il puisse changer l'état du tag passé par le génie et par suite allumer le luminaire.

Pour configurer le bouton, on édite ses propriétés et on clique sur **Entrée** puis **Clique** comme indiqué par les flèches, ensuite dans le champ de commande, on écrit le code illustré sur la figure ; à savoir : **?1 ?=1 ; Sleep(10) ; ?1 ?=0.**

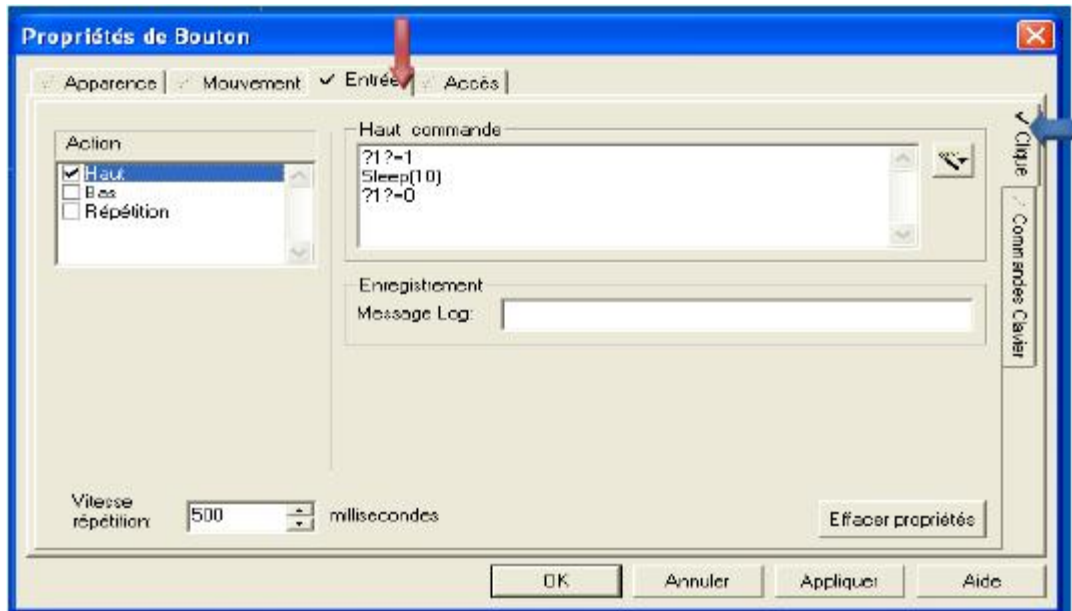


Figure III-16: Configuration du bouton d'allumage dans le super génie

Explication du code : ?1?=1 signifie que le premier tag passé par le génie à son super génie grâce à la fonction AssPopUp (dans notre cas le tag est : *action_%etage%_%bureau%_%position%_BP*) va être mis à 1, ainsi donc, le numéro qui est entre points d'interrogation indique le numéro du tag passé par le génie à son super génie.

Dans le cahier de charge, nous avons mentionné que l'allumage des luminaires par un bouton poussoir doit être temporisé afin d'éviter le risque d'oublier d'un luminaire allumé, ceci justifie donc l'ajout de la fonction **Sleep**, qui doit permettre un allumage de 20mn comme le cahier de charge l'indique (ici on se contente à 10s pour des besoins de simulation) cette temporisation passée, on remet le tag à 0 pour éteindre le luminaire en écrivant **?1 ?=0 ;** La configuration du bouton d'allumage est ainsi terminée, on doit à présent ajouter un autre bouton permettant de quitter la page du super génie lorsque celle-ci est appelée, après ajout du bouton, on édite ses propriétés, on clique sur **Apparence**, puis **Symbole**, puis sur **Config**, ensuite on choisit le symbole approprié de la liste comme indiqué sur la figure.

L'étape suivante est d'ajouter la fonction permettant de quitter le super génie, pour cela, dans les propriétés du bouton, on va sur **Entrée** ensuite on ajoute la fonction **WinFree()** dans le champ de commande comme indiqué par la **figure III-17**

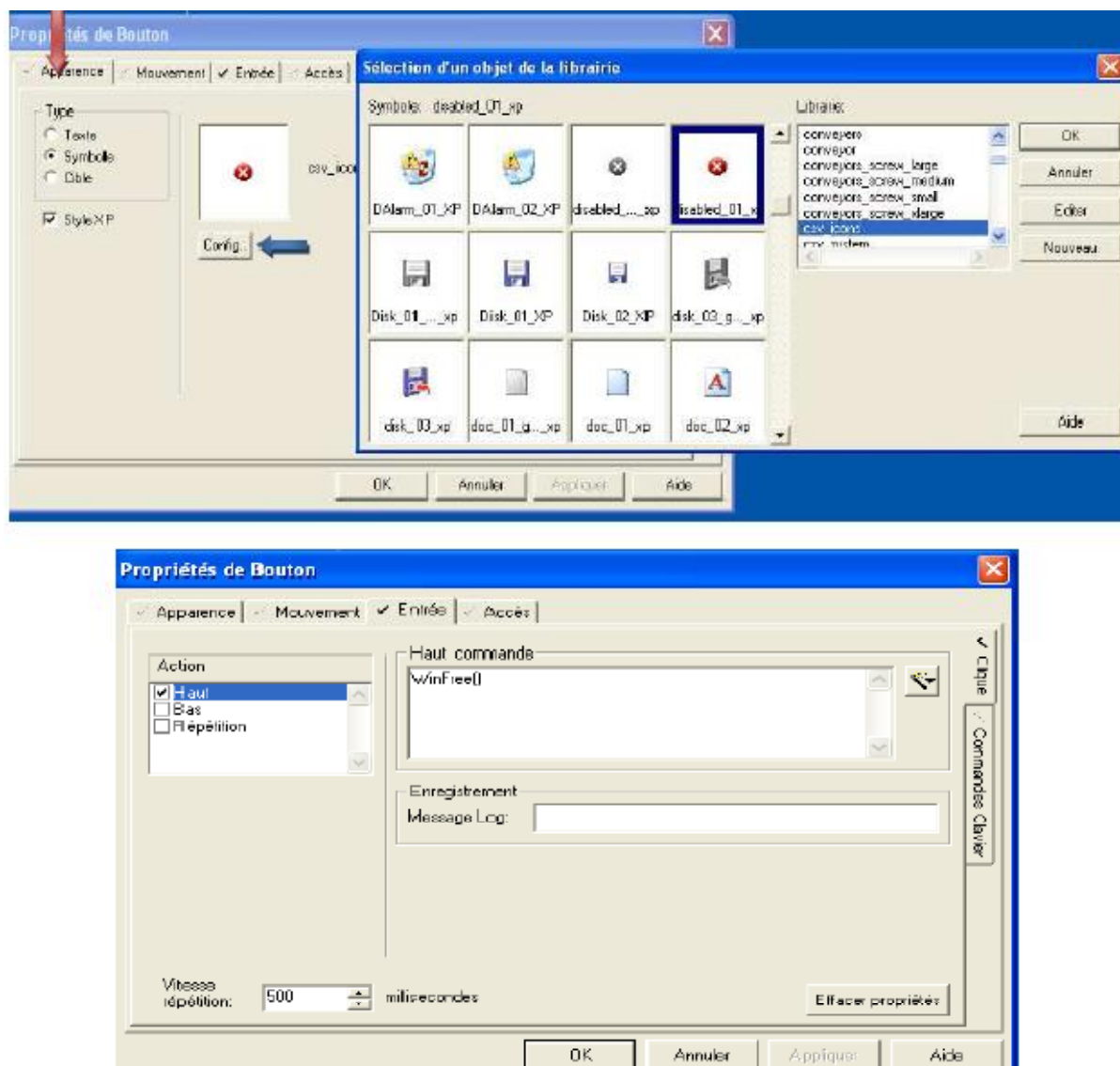


Figure III-17 : Configuration du bouton quitter dans le super génie

Ceci termine donc la configuration de notre super génie, reste à le sauvegarder sous le nom !BoutonPoussoir. Notre génie étant créé, on doit maintenant lier le génie du bouton poussoir au super génie, pour cela on ouvre le génie, on clique sur **Edition, Lier des super génies**, on ajoute ensuite le super génie comme indiqué sur la **figure III-18**.

La **figure III-19** donne un aperçu de l'exécution de notre application où le super génie apparaît suite à un clique sur le génie correspondant ; le bouton poussoir. Le clique sur le

bouton **Allumer** du super génie permet d'allumer la deuxième rangée malgré le fait que le détecteur de mouvement est désactivé.



Figure III-18: Lier un super génie à un génie

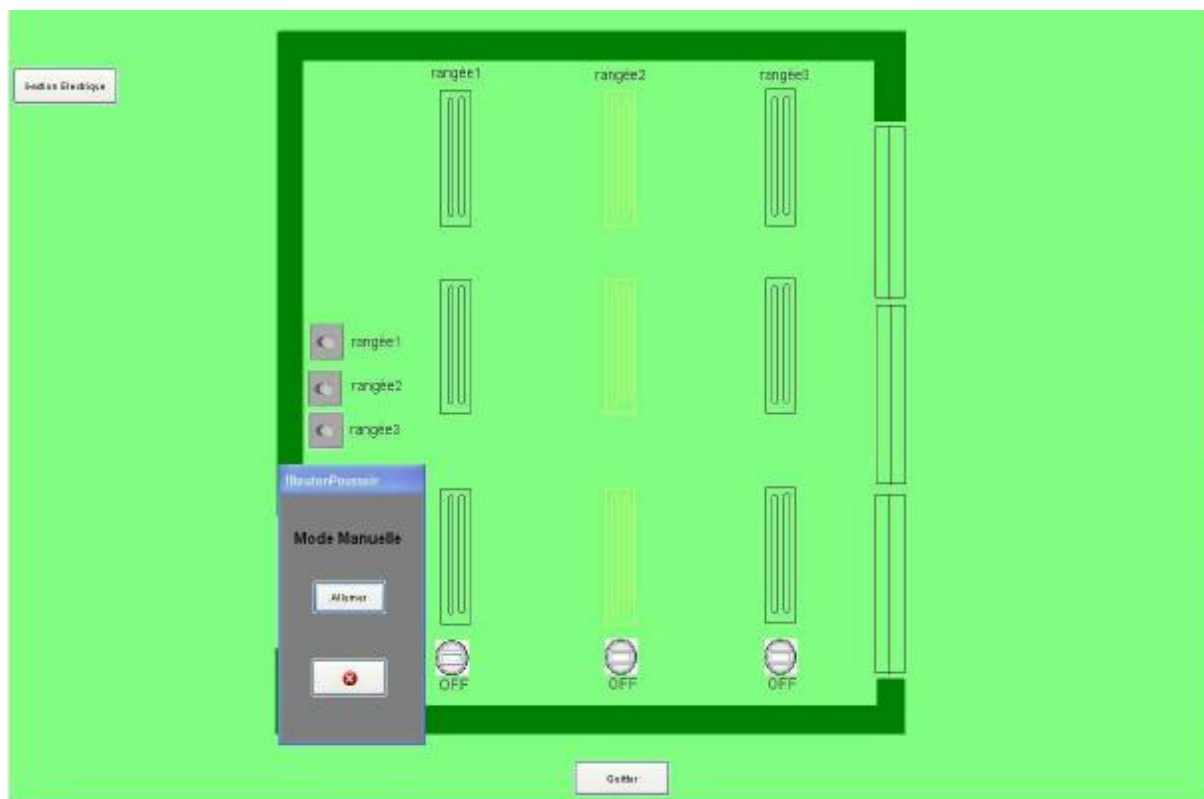


Figure III-19: Aperçu de l'exécution de l'application, appel d'un super génie

Remarque : pour configurer le luminaire afin qu'il change de couleur lorsqu'il est allumé, il suffit d'éditer ses propriétés, de cliquer sur **Remplir** puis d'ajouter le tag approprié comme indiqué sur la figure, le luminaire devient ainsi jaune lorsque le tag prend la valeur 1.

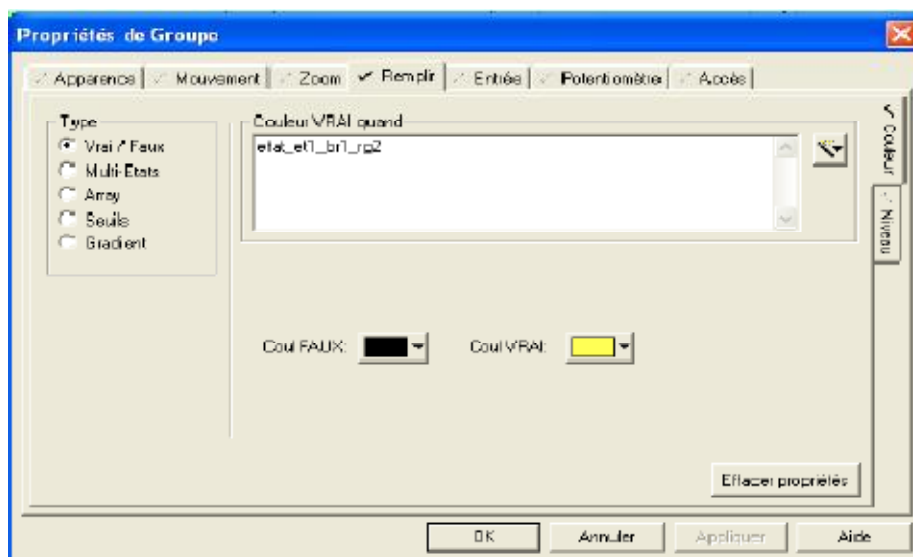


Figure III-19 : Configuration d'une rangée de luminaire

Conclusion :

Durant ce chapitre, nous avons présenté de façon succincte le logiciel Vijeo Citect ainsi que certaines de ces caractéristiques et fonctionnalités, certaines options du logiciel ont été illustrés à l'aide d'un exemple d'application de façon à simplifier l'apprentissage.

Nous avons montré que grâce à ces fonctionnalités avancées, Vijeo Citect constitue une solution SCADA complète et homogène.

A noter que ce chapitre ne se prétends nullement d'être un manuel d'apprentissage du logiciel mais plus tôt un petit guide présentant quelques unes des principales fonctionnalités du logiciel et permettant une prise en main rapide par l'apprenti. Pour une utilisation plus avancée, la consultation d'un manuel et du Help du logiciel sont nécessaires.

Chapitre IV

Présentation de l'Application

IV-1 : Introduction et descriptif général de l'application :

Au cours de ce chapitre, nous allons présenter l'application mise en œuvre dans le cadre de notre Projet de Fin d'étude. En effet, en utilisant les concepts et les outils décrits lors des chapitres précédents (outils tel que le logiciel Unity Pro ou encore le logiciel SCADA Vijeo Citect), nous allons créer notre propre application SCADA permettant le contrôle et la supervision d'une tour de bureaux, cette application va permettre le contrôle et la gestion de divers systèmes à savoir :

1. La gestion technique d'électricité, (la gestion de l'ensemble de l'installation électrique alimentant notre bâtiment).
2. La gestion du système d'éclairage (intérieur et extérieur).
3. La Gestion du système HVAC (Heating, Ventilating and Air Conditioning) à travers le contrôle des **CTA (Centrale de Traitement d'Air)**.

L'application développée sera dotée de toutes les fonctionnalités d'une application SCADA complète, elle permettra entre autres d'effectuer les tâches suivantes :

1. Superviser toute l'installation et déterminer l'état de chacun des ses sous systèmes : état de tous les disjoncteurs de protection de notre installation électrique, état de l'éclairage dans chacune des zones de la tour, état des différentes CTA permettant le contrôle de la température...etc.
2. Commander les différents sous systèmes de notre installation : ouverture des disjoncteurs des différents systèmes électrique, allumage ou extinction de l'éclairage directement du superviseur (on reviendra sur tous les modes de gestion de l'éclairage plus loin), modification des différentes consignes tel que la consigne de température d'air repris de la CTA...etc
3. Signaler à l'opérateur les différentes alarmes qui surviennent lors du fonctionnement du système : alarme signalant la coupure du réseau électrique, alarme signalant le déclenchement d'un disjoncteur donné...etc.

IV-1-1 Description de la tour :

Notre tour dispose d'un sous sol, d'un rez-de-chaussée et de six niveaux ainsi que de zones extérieures dont on peut commander l'éclairage :

- Le sous sol comporte les différents systèmes électrique (les TGBT : Tableau Général Basse Tension, Groupe Electrogène, Onduleur...etc.), il est subdivisé en 10 zones d'éclairage (zone 1A jusqu'à 6A et zone 1B à 4B).
- Le rez de chaussée est subdivisé à son tour en 14 zones d'éclairage (de 1A à 14A).
- Chacun des six niveaux est subdivisé en 17 zones d'éclairage (de 1A à 10A et de 1B à 7B), un niveau comporte donc deux tableau divisionnaires TDA et TDB relatifs aux zones A et B, ces tableaux comporte les télérupteurs de commande d'éclairage ainsi que les disjoncteurs de protection. Les niveaux disposent aussi des CTA permettant la climatisation.



Figure IV-1: Page d'Accueil du projet

La figure ci-dessus illustre la page d'accueil de notre projet, elle comporte des raccourcis vers tous les niveaux de notre tour (le sous sol, le rez-de-chaussée ainsi que les six niveaux), elle comporte aussi des raccourcis vers les principaux systèmes composant notre tour : l'installation électrique, les centrales de traitement d'air (CTA) ainsi que le tableau divisionnaire de l'éclairage extérieur. Les boutons de raccourcis des niveaux permettent d'accéder au système de commande de l'éclairage du niveau correspondant.

IV-2 Description de l'installation électrique :

La **figure IV.2** donne le synoptique général de l'installation électrique de notre bâtiment, on y voit les éléments suivants : un poste de livraison, deux TGBT, des Tableaux Divisionnaires (TD) pour l'éclairage, deux groupes électrogènes de secours avec un TGBT Groupe, un onduleur avec son TGCO (Tableau Général du Courant Ondulé). En cliquant sur un élément dans le synoptique, on va directement vers la page correspondante.

Dans ce qui suit, on donnera un descriptif général des éléments de notre installation électrique.

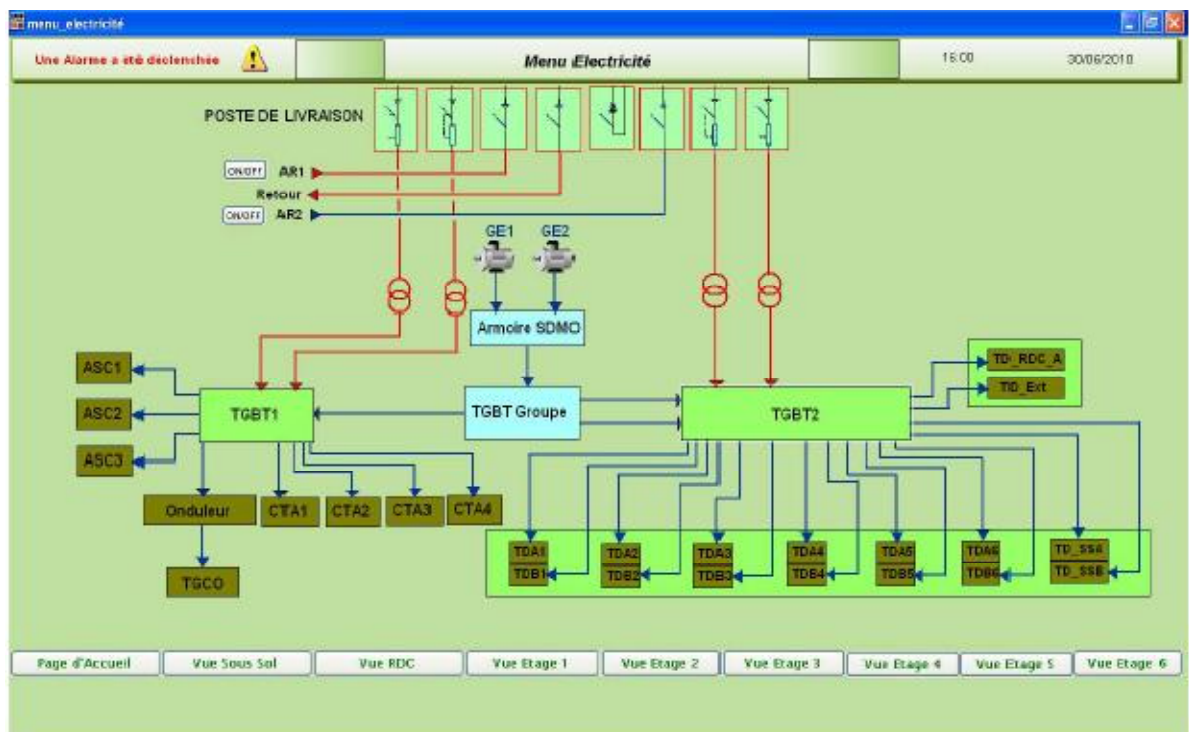


Figure IV-2 : Synoptique général de l'installation électrique

IV-2-1 Le Poste de livraison :

Alimenté par deux arrivées de 10KV, dont une seule est fonctionnelle à la fois, la deuxième arrivée est considérée comme une arrivée de secours en cas de coupure électrique dans la première. Le schéma du poste de livraison est illustré sur la figure IV-3.

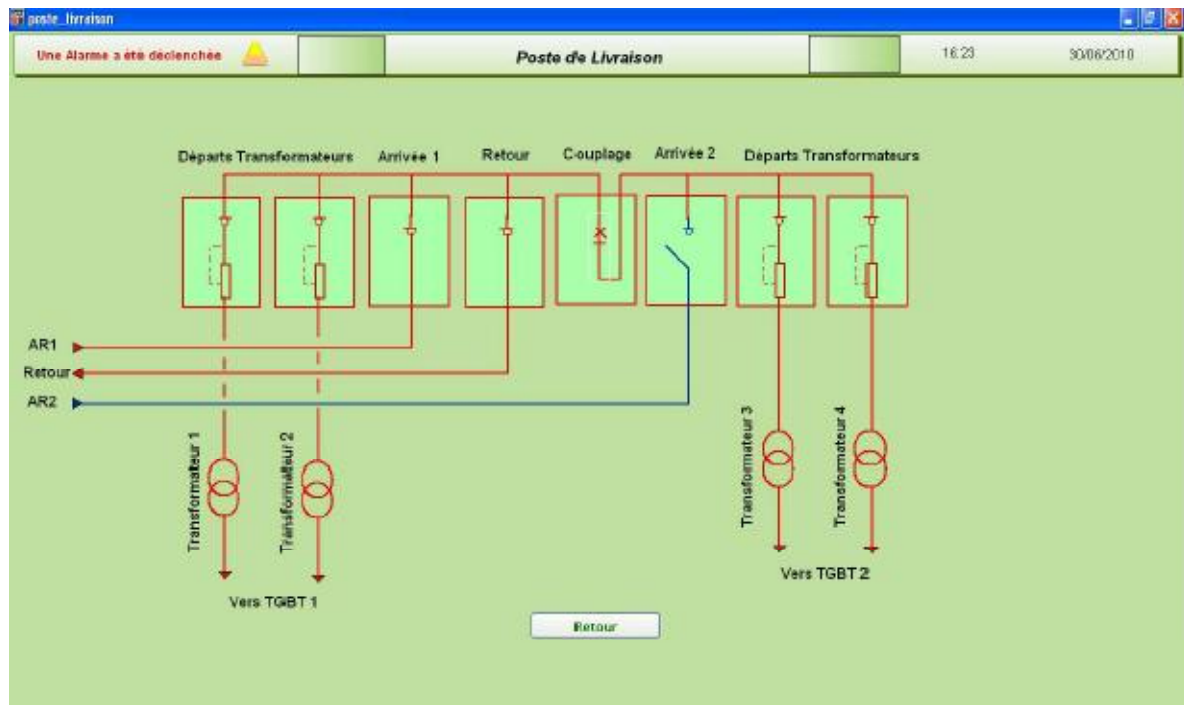


Figure IV-3 : Schéma électrique du poste de livraison

Le poste de livraison est constitué de huit cellules remplissant divers fonctions, on distingue :

- Deux cellules de raccordement au réseau contenant chacune un interrupteur sectionneur 10KV (sur la figure schématisé par Arrivée 1 et Arrivée 2), ces deux interrupteurs ne sont jamais fermés simultanément.
- Une cellule de départ 'Retour' permettant d'alimenter un autre client.
- Une cellule de couplage contenant un disjoncteur de couplage, ce disjoncteur est toujours fermé sauf lorsqu'on veut isoler un TGBT (pour des raisons de maintenance par exemple).
- Deux cellules de départ vers transformateurs (moyenne puissance) contenant des interrupteurs fusibles.
- Deux cellules de départ vers transformateurs (grande puissance) contenant des interrupteurs fusibles.

Il est à signaler que mis à part l'interrupteur de couplage, aucun des six autres interrupteurs du poste de livraison n'est commandable par l'opérateur à partir du superviseur.

Le fonctionnement du système est décrit comme suit :

- Par défaut, l'interrupteur de la cellule Arrivée 1 est fermé, celui de l'Arrivée 2 est ouvert et le coupleur est fermé, les deux TGBT sont ainsi alimentés par cette arrivée 1.
- Dès qu'il y'a coupure sur l'arrivée 1, l'interrupteur de cette arrivée s'ouvre, celui de l'Arrivée 2 se ferme et l'installation est ainsi alimentée par la deuxième arrivée.
- Lorsque le réseau sera rétabli sur l'arrivée 1, l'interrupteur de l'arrivée 2 s'ouvre, et on rebascule sur l'arrivée 1.

IV-2-2 Les transformateurs :

Comme illustré sur la **figure IV-3** notre installation est dotée de 4 transformateurs assurant la liaison entre le poste de livraison et les TGBT.

On dispose de deux transformateurs de 1250KVA assurant la liaison entre le poste de livraison et le TGBT 1 et de deux autres transformateurs de 630KVA assurant la liaison avec le TGBT 2.

IV-2-3 Le TGBT 1 :

Le premier Tableau Général Basse Tension (TGBT) est utilisé pour l'alimentation des installations énergivores, à savoir les CTA (Centrales de Traitement d'Air), le GEG (Groupe d'Eau Glacée), les ascenseurs, l'onduleur et la pompe de relevage. C'est pourquoi ce tableau est alimenté par les transformateurs ayant une plus grande puissance (les deux transformateurs de 1250KVA).

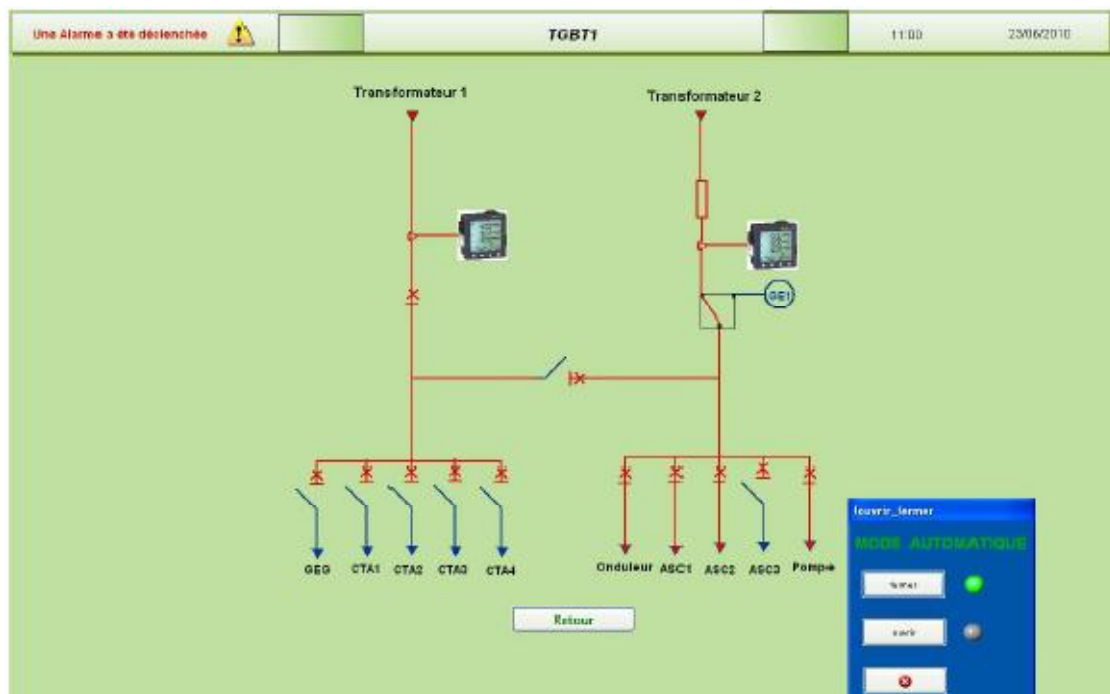


Figure IV-4: Schéma électrique du tableau TGBT 1

La **figure IV-4** illustre le schéma électrique du TGBT 1 où on peut visualiser les deux arrivées. Un inverseur de source est prévu sur l'arrivée venant du transformateur 2 alimentant l'onduleur, les ascenseurs et la pompe de relevage. Cet inverseur de source permet le basculement vers le générateur en cas de coupure de réseau, ainsi, on voit que les installations les plus énergivores (les CTA et le GEG) ne sont pas secourus afin de ne pas surcharger le générateur (de plus, ces installations ne sont pas d'une importance capitale comme le sont les ascenseurs, l'onduleur et la pompe de relevage, et le délestage effectué ne sera donc pas préjudiciable pour les occupants de la tour).

Remarque :

1. La couleur d'un câble a été programmée de façon à ce qu'elle devienne rouge lorsque ce câble est sous tension électrique et bleu dans le cas inverse. L'état d'un câble est obtenu en exploitant les informations relatives aux états des disjoncteurs en amont (l'état d'un disjoncteur est obtenu grâce à ses contacts auxiliaires) ainsi que des états des câbles les traversant.

2. On a programmé des supers génies pour la commande des disjoncteurs, en effet, en cliquant sur un disjoncteur, une fenêtre apparaît et permet d'avoir le mode du disjoncteur (local ou automatique) et permet aussi de changer son état (l'ouvrir ou le fermer s'il est en mode automatique).

IV-2-4 Le TGBT 2 :

Le TGBT 2 est utilisé pour l'alimentation des tableaux divisionnaires (TD) de l'éclairage. Ce TGBT est alimenté donc par les deux transformateurs de 630 KVA, il est doté de deux inverseurs de source permettant de basculer sur les groupes électrogènes lorsqu'il y a coupure du réseau. La liaison avec le groupe électrogène se fait à travers le Tableau Général du Groupe Electrogène (TGGE) qu'on verra par la suite.

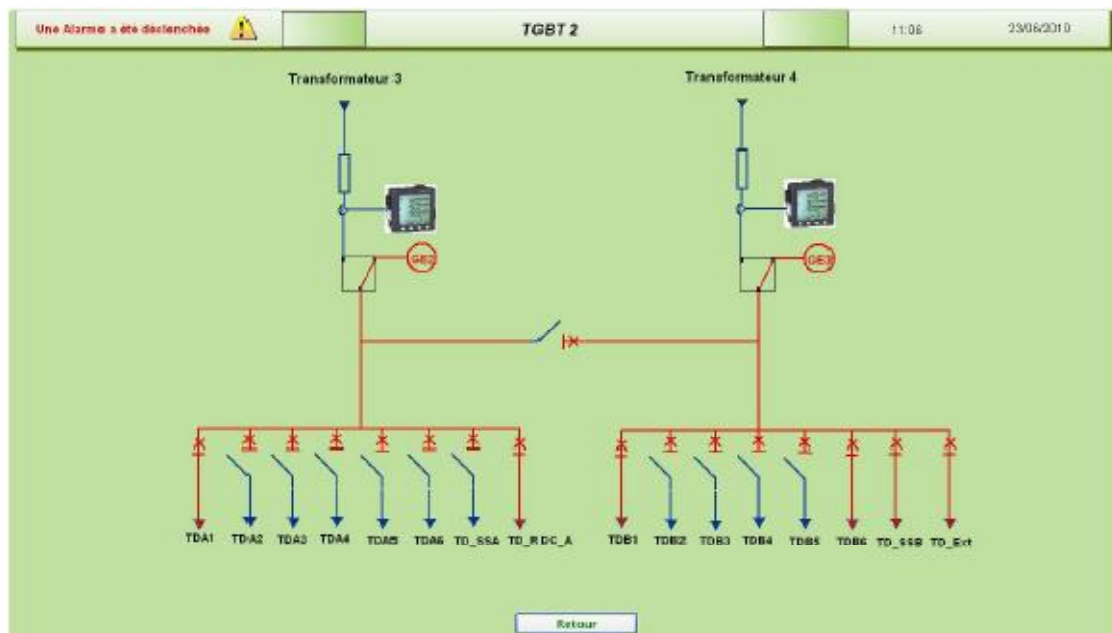


Figure IV-5 : Schéma électrique du TGBT 2

IV-2-5 Le Tableau divisionnaire d'éclairage :

On a vu auparavant que le TGBT 2 comporte des disjoncteurs de départ pour l'alimentation de neuf Tableaux Divisionnaires (TD) d'éclairage : les six TD correspondant aux six niveaux, ainsi que celui du rez-de-chaussée, du sous sol, et de l'éclairage extérieur.

La **figure IV-6** illustre le schéma électrique d'un tableau divisionnaire d'éclairage (ici le TD du 1^{er} niveau). Il comporte deux disjoncteurs de tête pour chacune des deux zones A et B (le Compact NSX 160) ainsi que des disjoncteurs de départ pour la protection (des disjoncteurs de la série C60N ayant comme courant assigné 16A et comme courant de déclenchement 10 Ka. Seuls les deux disjoncteurs de tête sont commandables, les disjoncteurs de départ ne sont prévus que pour la protection.

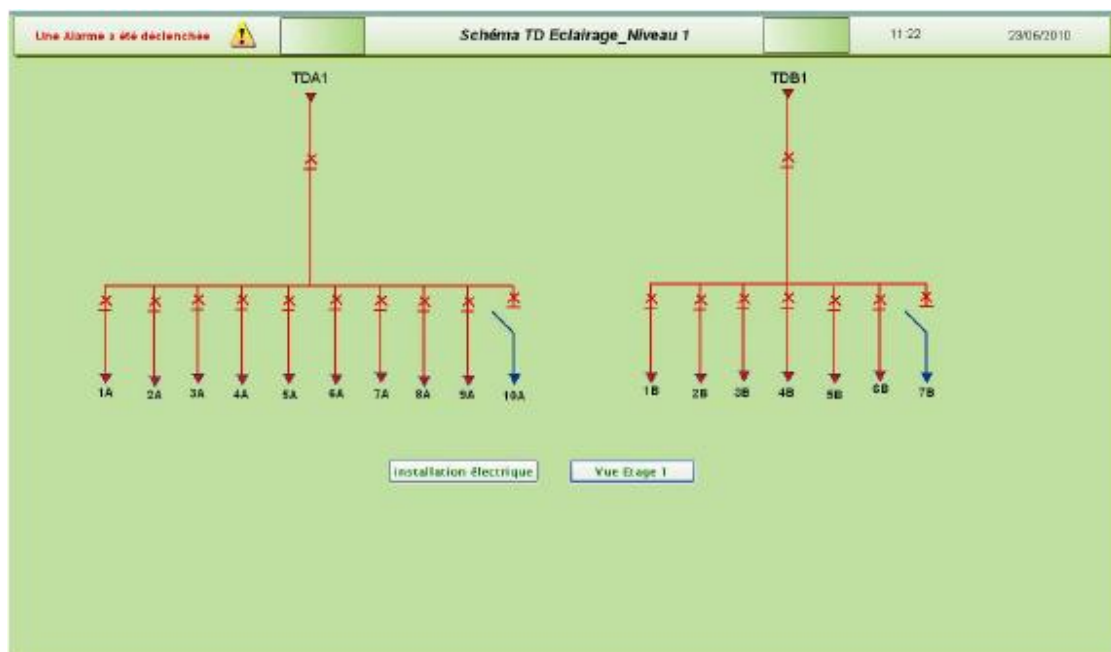


Figure IV-6 : *Le tableau divisionnaire pour l'éclairage du premier étage*

IV-2-6 Le Tableau Général du Groupe Electrogène TGGE :

Notre installation dispose de deux groupes électrogènes à utiliser en cas de coupure du réseau sur les deux arrivées, la puissance des deux groupes étant couplée au niveau de l'armoire SDMO, la sortie de cette armoire alimente le tableau TGGE qui dispose d'un disjoncteur de tête et de trois disjoncteurs de départ vers les TGBT. Tous les disjoncteurs de ce tableau sont commandables par l'opérateur.

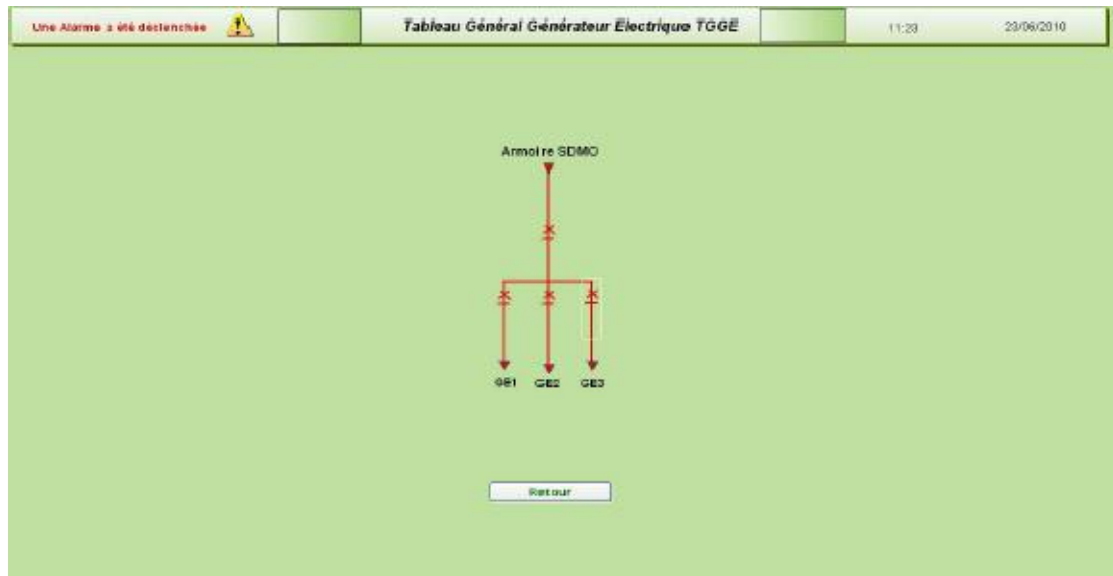


Figure IV-7: Schéma électrique du Tableau Général du Groupe Electrogène

IV-2-6 Les alarmes :

Pour tous projets de GTB, la configuration et l'affichage des alarmes est une étape primordiale qui permet à l'opérateur de savoir à tout moment si une panne ou une anomalie est survenue dans le système, ce qui permettra de prendre les mesures nécessaires dans les plus brefs délais afin de palier au problème. Aussi, cette étape importante n'a pas été omise dans notre projet et plusieurs alarmes ont été prévues et configurées. En effet dès qu'une alarme survient, un symbole d'alarme (le triangle en haut à gauche de l'écran) commence à clignoter et une écriture rouge apparaît indiquant qu'une alarme a été déclenchée. En cliquant sur le symbole d'alarme, on peut aller sur la page d'alarme (**figure IV-8**) pour visualiser toutes les alarmes, cette page donne des informations sur la date et l'heure du déclenchement de l'alarme, le nom et la description de cette dernière, ainsi que son état (lorsque la condition déclenchant l'alarme n'est plus valide, l'état est alors faux). Cette page permet à l'opérateur d'acquitter une alarme en cliquant dessus.

En utilisant le bouton de navigation sur la page d'alarme (indiqué par la flèche bleue), on peut aussi se rendre vers une autre page donnant l'historique et le sommaire de toutes les alarmes déclenchées avec plus de détails (tel que la catégorie de l'alarme, la date et l'heure de l'acquittement, la durée, l'état...etc.)

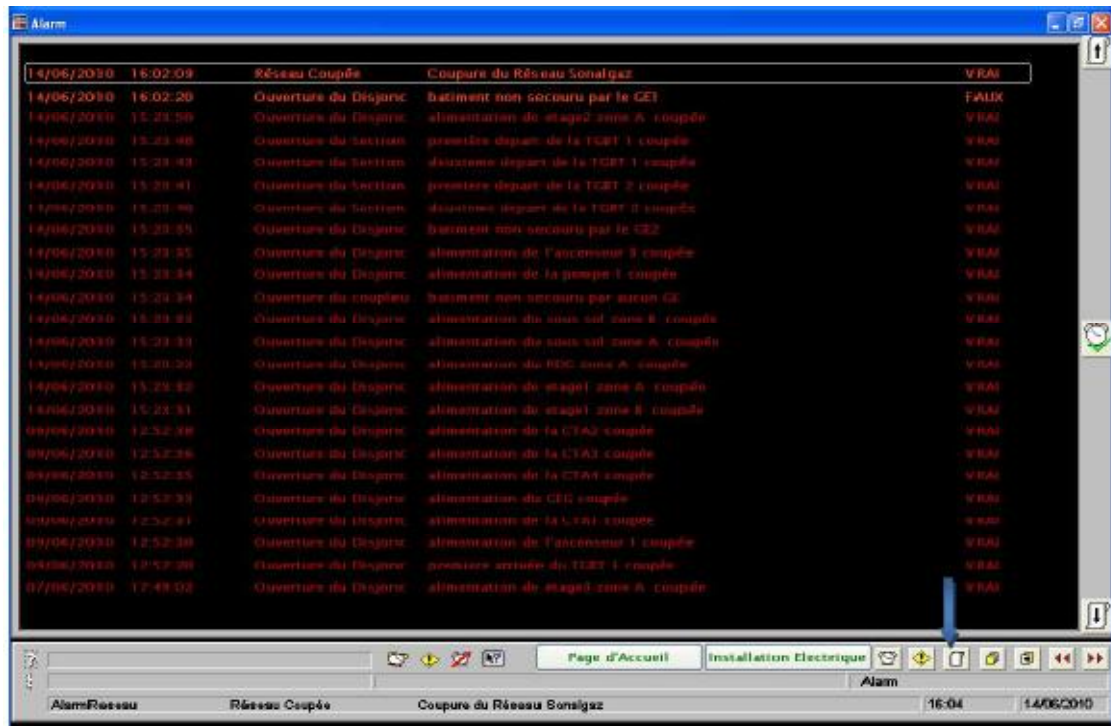


Figure IV-8 Page d'alarme

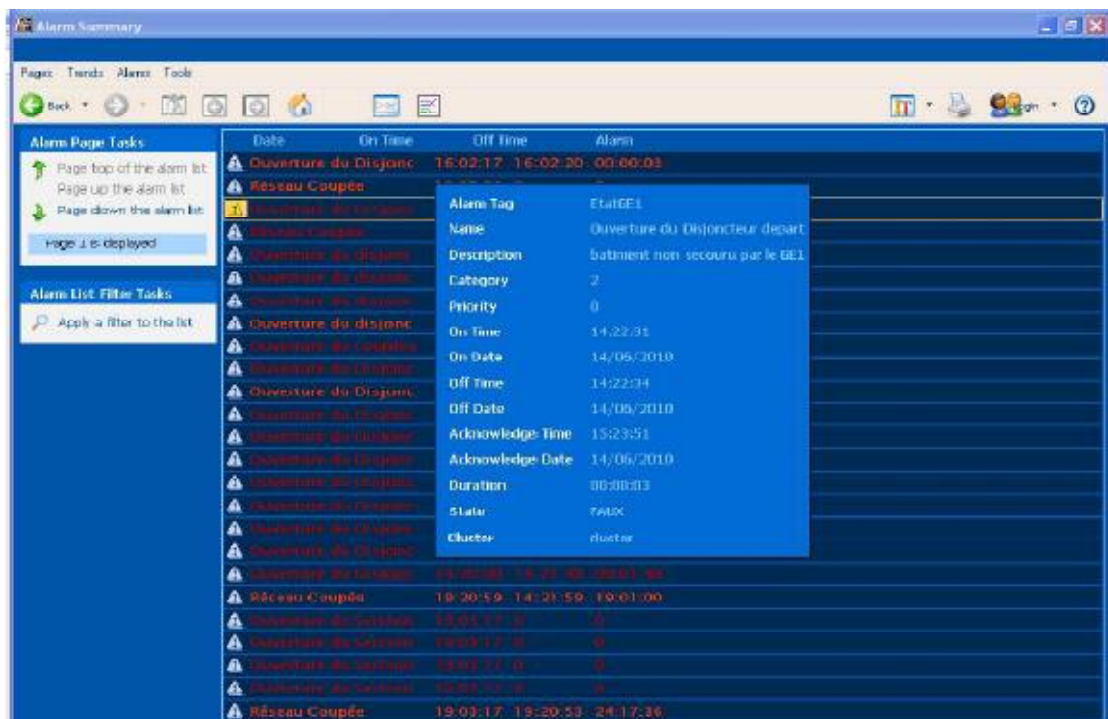


Figure IV-9 : Page sommaire des alarmes

IV-3 description du système d'éclairage :

Pour la commande du système d'éclairage de notre tour, les modes de gestion suivants ont été prévus :

1. Mode Manuel : où l'opérateur peut allumer ou éteindre l'éclairage dans une zone en appuyant sur un bouton à partir de la station de supervision.

2. Mode de gestion horaire (Automatique) : Ce mode permet à l'opérateur de saisir l'heure d'allumage et d'extinction d'une zone à partir de la station de supervision.

3. Mode de gestion en fonction de la présence et de l'apport en éclairage naturel :

Ce mode, implémenté uniquement dans les endroits enregistrant des périodes d'absence assez prolongée permet de gérer l'éclairage tout en tenant compte de la présence des personnes ainsi que de l'éclairage naturel, ce qui permet d'avoir une gestion énergétiquement plus efficace.

4. Mode crépusculaire : ce mode, prévu exclusivement pour l'éclairage extérieur, permet d'allumer ou d'éteindre la lumière extérieure selon l'état de la lumière du jour. Pour cela on utilise un capteur crépusculaire ayant une sortie logique, cette sortie est mise à 1 dès que le niveau d'éclairage descend au dessous d'un seuil réglable donnée.

Dans ce qui suit, on donnera un descriptif des systèmes d'éclairage intérieur et extérieur respectivement et où l'opérateur peut choisir entre l'un des modes de gestion décrits précédemment.

IV-3-1 Description du système d'éclairage intérieur :

Pour la gestion du mode d'éclairage intérieur, on dispose d'un tableau permettant le choix du mode de gestion pour chacune des zones d'un étage donnée (**figure IV-10**).

Pour le mode manuel, l'opérateur dispose d'un bouton permettant l'allumage ou l'extinction d'une zone donnée. Un voyant permet de déterminer l'état de l'éclairage de la zone. Si l'opérateur choisit le mode automatique (ou mode de gestion horaire), le premier mode sera automatiquement désactivé, le bouton d'allumage et d'extinction sera à son tour grisé (désactivé) et l'opérateur pourra alors se rendre vers la page de Gestion Horaire pour introduire les heures d'allumage et d'extinction d'une zone donnée.

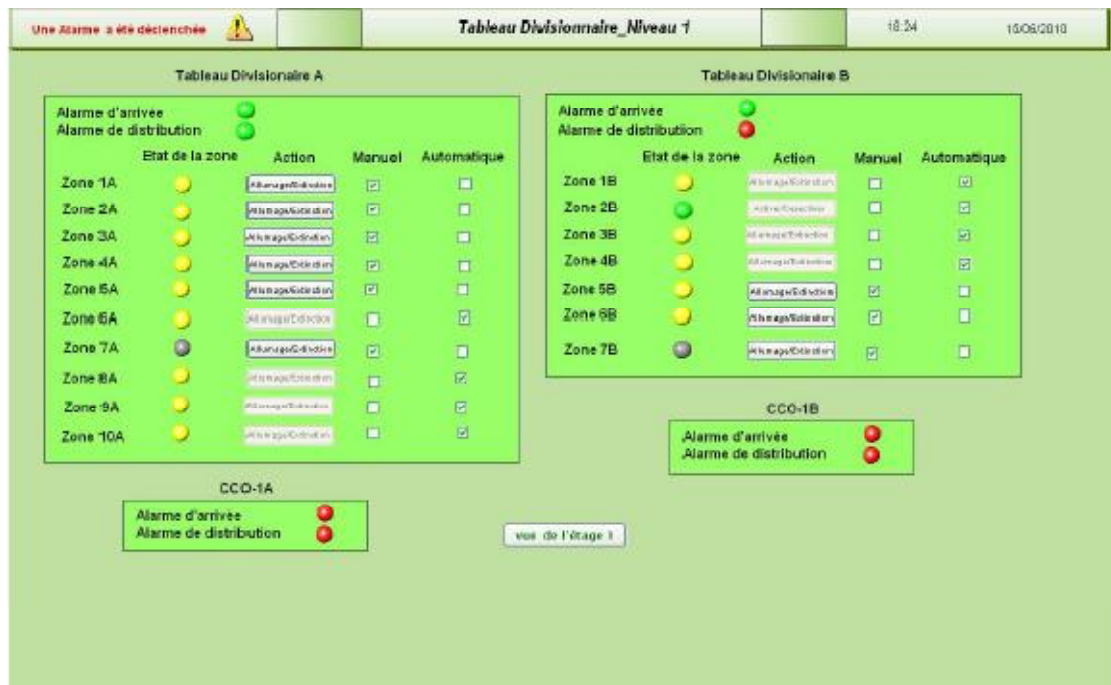


Figure IV-10 Page de commande d'éclairage du premier étage

Remarque :

- En plus du symbole d'alarme disposé en haut à gauche de l'écran, la page dispose aussi de deux indicateurs d'alarmes ; **Alarme d'arrivée** et **alarme de distribution**. La première indique que le disjoncteur de tête du tableau divisionnaire est ouvert, tandis que l'**Alarme d'arrivée** indique que l'un des disjoncteurs de protection est ouvert, l'opérateur pourra alors vérifier quel disjoncteur a été déclenché en se rendant vers la page du Tableau divisionnaire correspondant
- Il est évident de noter que l'éclairage d'une zone sera désactivé dès que cette zone n'est plus alimentée en électricité.

La figure ci-dessous illustre la vue de l'étage 1, on voit que les zones allumées deviennent éclairées en jaune, les zones non éclairés sont sombres (grises), la page comporte aussi des boutons de navigation qui permettent de se rendre vers la page de commande d'éclairage (choix du mode et commande), la page de gestion horaire, le tableau divisionnaire ainsi que la page d'accueil et celle de l'installation électrique.



Figure IV-11 : Vue de l'étage 1

IV-3-1-1 Gestion de l'éclairage en fonction de la présence et de l'apport en éclairage naturel

Les deux modes vus précédemment (mode manuel et mode de gestion horaire) ne sont appliqués généralement que dans les espaces ouverts (*Open Space*) qui enregistrent un mouvement continu du personnel et où il sera gênant et non efficace de prévoir un mode de gestion en fonction de la présence des occupants en raison des allumages-extinctions fréquents. Cependant, pour les espaces fermés qui enregistrent en général des périodes de non occupation assez longues (bureau, salle de réunion...etc.), adopter une politique de gestion horaire de l'éclairage sera irrationnel et ne sera pas en concordance avec l'efficacité énergétique (un des objectifs de la GTB), et il sera donc judicieux et économiquement rentable de prévoir des capteurs de mouvement afin d'éteindre l'éclairage lorsque le local n'est pas occupé.

Tenant compte de ce qui a été dit et afin de réduire la facture de la consommation de l'éclairage, nous avons implémenté dans notre application (au niveau de la zone 2B) une gestion de l'éclairage en fonction de la présence des occupants et de l'éclairage naturel, le cahier de charge est donné ci-dessous.

Cahier de charge :

Nous disposons d'un bureau muni de trois rangées de luminaires pour l'éclairage et de trois détecteurs de mouvement (chacun associé à une rangée) servant comme détecteur de présence. On veut que l'éclairage soit commandé par le détecteur de mouvement afin de l'éteindre lorsqu'aucune présence n'est enregistrée pour rendre ainsi la consommation d'énergie plus rationnelle.

- Une rangée de luminaire est allumée si le détecteur de mouvement associé détecte une présence, sinon elle est éteinte au bout de 2 mn après la dernière détection de mouvement.
- De plus, le bureau dispose aussi de trois boutons poussoirs permettant l'allumage d'une rangée de luminaire même si le capteur de mouvement associé est désactivé (aucune présence n'est détectée), ces boutons peuvent être utiles lorsqu'un occupant souhaite allumer d'autres rangées que celle sous laquelle il travaille dans le but d'augmenter la luminosité.
- L'allumage d'une rangée par un bouton poussoir sera activé pour une durée déterminée (20 mn), au bout de cette durée, les luminaires de la rangée vont s'éteindre dans le cas où le bureau est vacant.
- La lumière des luminaires les plus proches de la fenêtre est modulée (régulée) en fonction de la lumière du jour et ceci en utilisant un capteur de lumière et un régulateur PI.

Le grafcet matérialisant ce cahier de charge est illustré ci-dessous :

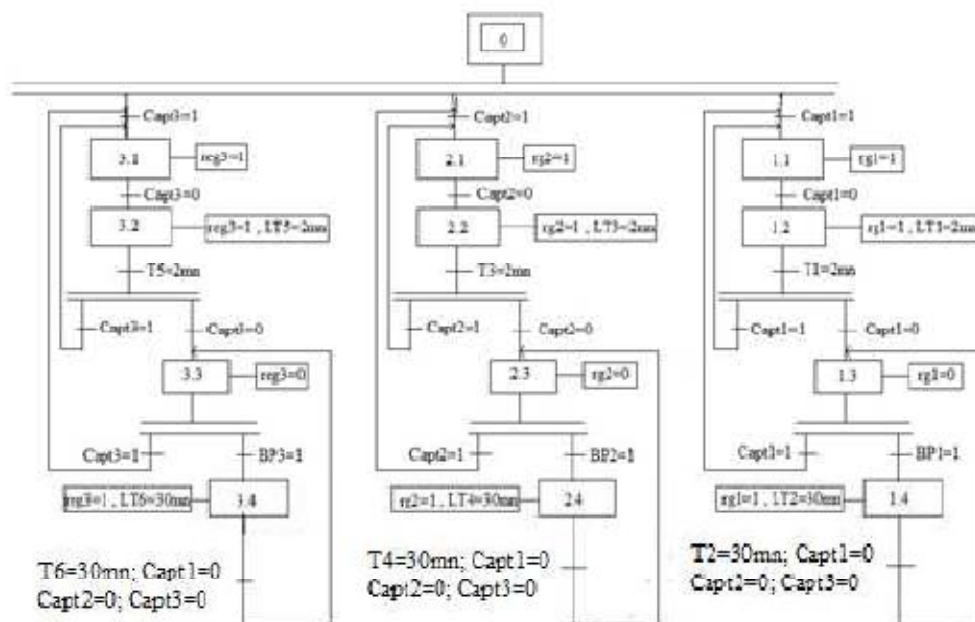


Figure IV-12 : Grafcet du cahier de charge de l'éclairage de la zone 2B

La **figure IV-13** donne une schématisation de notre bureau, on peut visualiser les différents luminaires, les trois boutons poussoirs, le capteur de lumière ainsi que les capteurs de mouvement. On dispose aussi d'un tableau permettant de saisir les paramètres du régulateur PI ainsi que d'un autre permettant de saisir la consigne d'éclairage sur la première rangée en lux, ce tableau permet aussi de visualiser la mesure donnée par le capteur de luminosité convertie en lux.

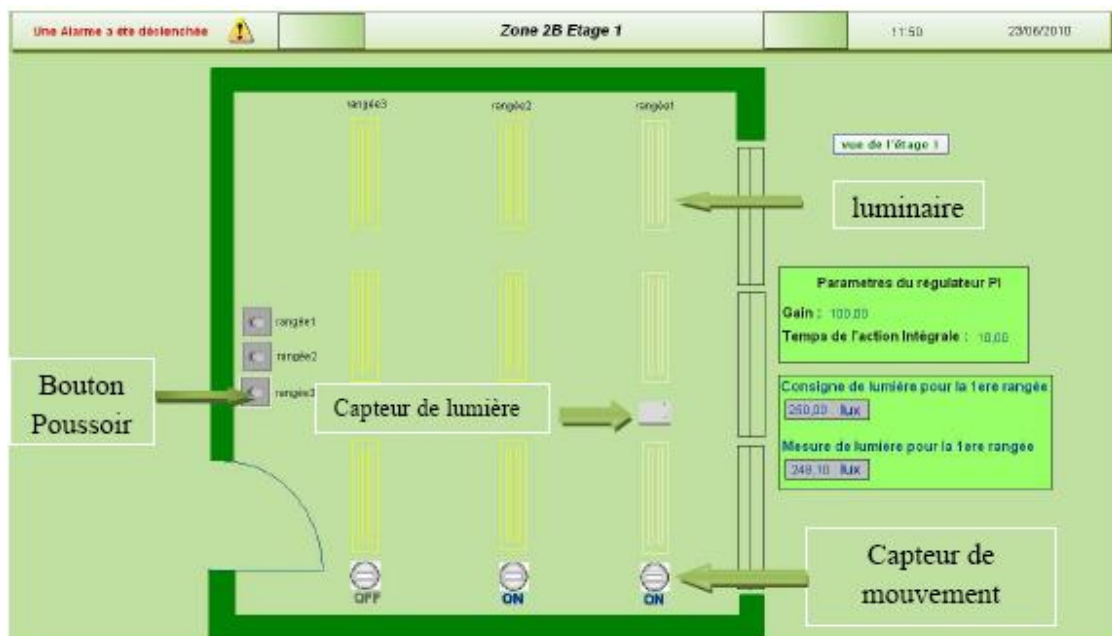


Figure IV-13 : Gestion de l'éclairage de la zone 2B en fonction de la présence et de l'éclairage naturel

- Sur la figure précédente, on voit que les deux premières rangées de luminaires sont allumées du fait que les capteurs de mouvement correspondant sont actifs.
- Bien que le capteur de mouvement relatif à la troisième rangée indique qu'il n'y a aucune présence, on voit que le troisième groupe de luminaires est allumé et ceci suite à une action sur le bouton poussoir.
- Les luminaires de la première rangée sont moins éclairées ; en effet, on utilise un capteur de lumière pour mesurer le niveau d'éclairement et adapter l'intensité d'éclairage des luminaires en fonction de l'apport de la lumière du jour pour avoir finalement le niveau d'éclairement requis. De ce fait, il est normal de voir que les luminaires de la première rangée sont moins éclairées que celles des deux autres rangées.

IV-3-2 Descriptif du système d'éclairage extérieur :

Le système d'éclairage extérieur est commandé avec les même modes de gestion que celui de l'éclairage intérieur à une différence près, il comporte un nouveau mode de gestion, le mode *crépusculaire*.

Ce mode permet l'allumage de la lumière extérieur le matin dès que le niveau d'éclairement dépasse un certain seuil et l'extinction le soir dès que le niveau d'éclairement descend en dessous d'un certain seuil, pour cela on utilise un capteur crépusculaire dont le seuil est réglable.

La **figure IV-14** illustre le schéma électrique du tableau divisionnaire de l'éclairage extérieur, il comporte un disjoncteur de tête commandable ainsi que des disjoncteurs de départ pour la protection (ces disjoncteurs ne sont pas commandables, la commande se fait par action sur des télérupteurs ; non schématisés sur la figure).

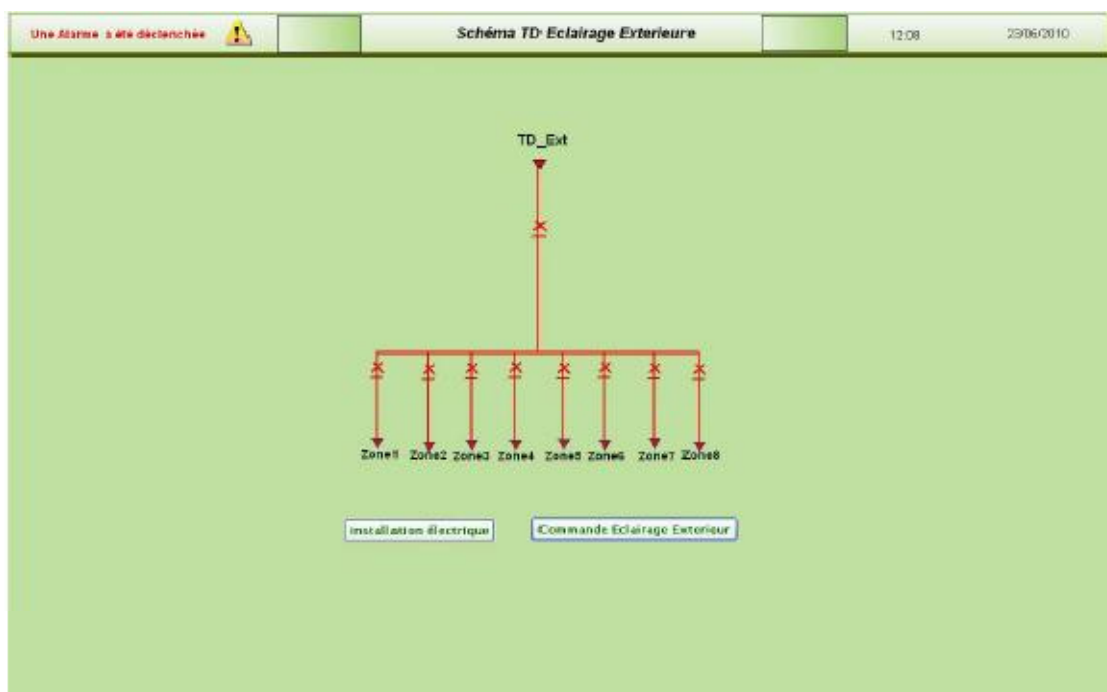


Figure IV-14 : *Tableau divisionnaire de l'éclairage extérieur*

La **figure IV-15** illustre la page de gestion de l'éclairage extérieur, on trouve les cases permettant de choisir l'un des trois modes de gestion, des boutons pour l'allumage et l'extinction des zones qui ne sont actives que dans le mode manuel, des voyant donnant l'état de l'éclairage des zones (allumées ou

éteintes). La page dispose aussi d'un espace permettant à l'opérateur de saisir l'heure d'allumage et d'extinction des zones configurées en mode horaire.

Par ailleurs, on a prévu des voyants pour indiquer l'état du disjoncteur de tête du tableau divisionnaire (alarmes arrivée), un autre voyant pour indiquée que l'un des disjoncteurs de protection a été déclenché (l'opérateur pourra déterminer le disjoncteur ouvert à partir des voyants indiquant l'état d'éclairage dans la page de gestion), finalement, on dispose aussi d'un voyant pour indiquer l'état du capteur crépusculaire, ce capteur (ayant une sortie logique) se met à 1 lorsque le niveau d'éclairement descend au dessous d'un seuil donné.

Les boutons de navigation permettent de se rendre soit vers la page d'accueil, soit vers le schéma d'installation électrique, ou encore vers le tableau divisionnaire de l'éclairage extérieur.

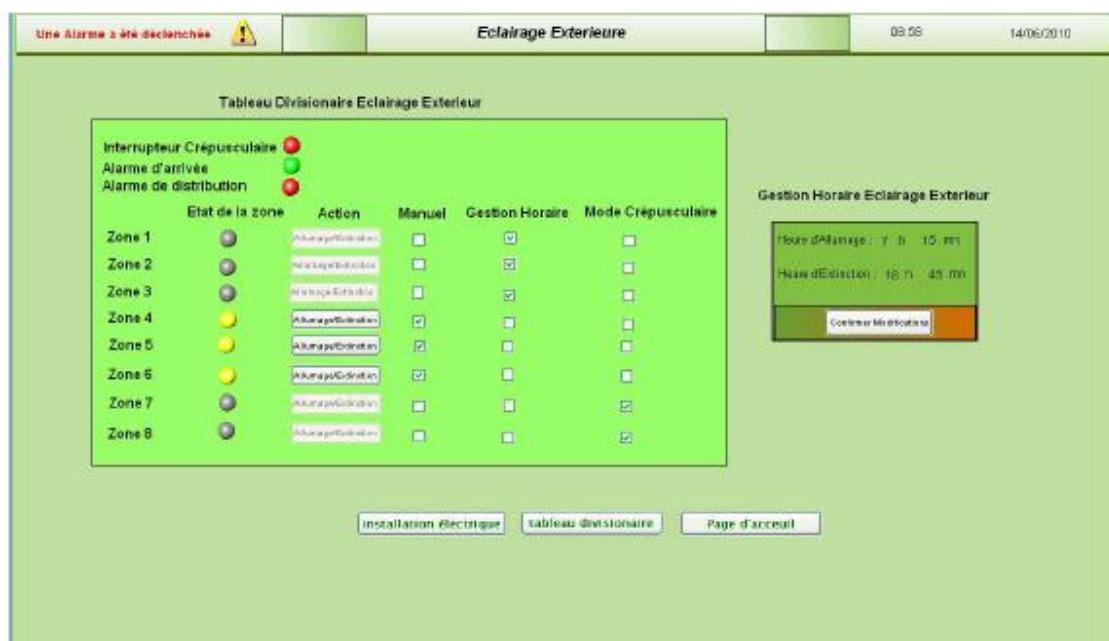


Figure IV-15 : Page de gestion de l'éclairage extérieur

IV-4 Centrales de Traitement d'Air (CTA) :

IV-4-1 Présentation :

Pour notre application, nous allons utiliser une CTA pour chaque deux niveaux, nous aurons donc besoin de quatre CTA puisqu'il y a huit niveaux (Sous Sol, Rez de chaussée et les six étages). Pour accéder à la page des CTA, il faut cliquer sur le bouton « Centrales de Traitement d'Air » comme le montre la figure suivante :



Figure IV-16 : Accès à la page des CTA

La page des CTA nous permet d'accéder à l'une des quatre CTA comme le montre la **figure IV-17**.



Figure IV-17 : Choix d'une CTA

En cliquant sur l'une des quatre CTA, on accède à une page représentant le synoptique de la CTA choisie afin de la configurer, d'entrer des consignes (débit du ventilateur, température d'air repris) et de visualiser certaines grandeurs. En cliquant par exemple sur la figure « CTA1 », la page de la **figure IV-18** apparaît :

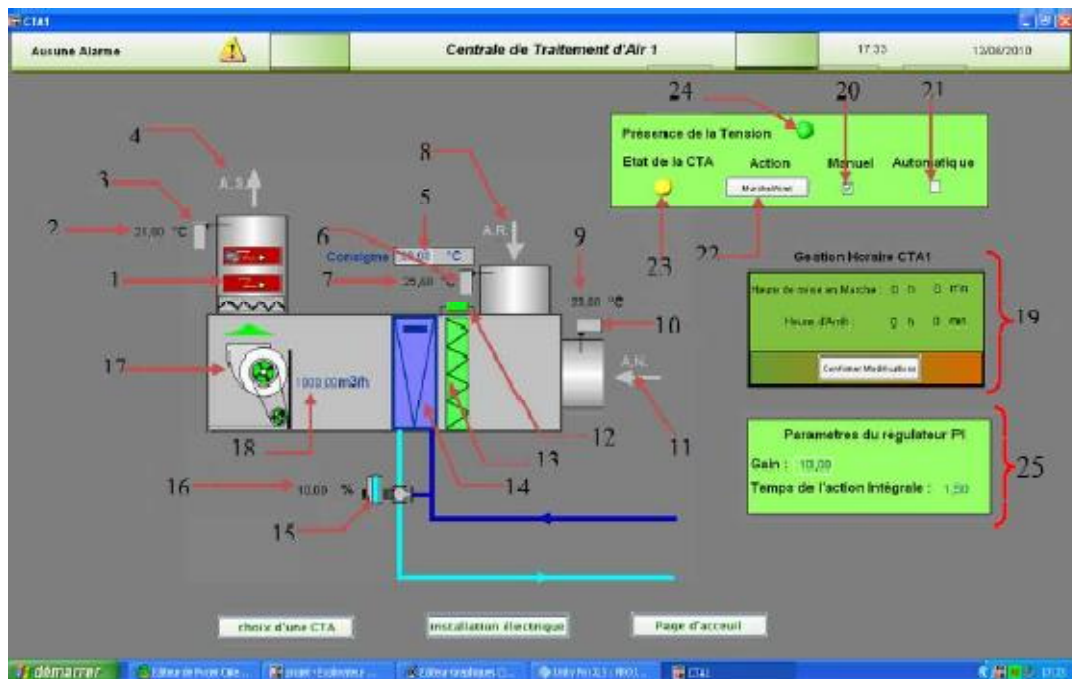


Figure IV-18: *Synoptique d'une CTA*

- 1) Batterie chaude (électrique).
- 2) Affichage de la température d'air soufflé.
- 3) Capteur de température d'air soufflé.
- 4) Sortie d'air soufflé.
- 5) Entrée de la consigne « température d'air repris ».
- 6) Capteur de la température d'air repris.
- 7) Affichage de la température d'air repris.
- 8) Entrée de l'air repris.
- 9) Affichage de la température d'air neuf (extérieur).
- 10) Capteur de la température d'air neuf (extérieur).
- 11) Entrée de l'air neuf.
- 12) Capteur de pression différentiel servant à détecter l'encrassement du filtre.
- 13) Filtre à air.
- 14) Batterie d'eau froide.
- 15) Vanne 3 voies.
- 16) Taux d'ouverture de la vanne 3 voies (en %)

- 17) Ventilateur centrifuge.
- 18) Consigne de débit d'air soufflé par le ventilateur.
- 19) Entrée de l'heure de mise en marche et de l'arrêt de la CTA lorsqu'on est en mode « automatique » c.à.d. en mode gestion horaire de marche et d'extinction.
- 20) Mode manuel : mise en marche et extinction de la CTA grâce au bouton poussoir « Marche/Arrêt ».
- 21) Mode Automatique : mode de la gestion horaire.
- 22) Bouton « Marche/Arrêt » de la CTA
- 23) Etat de la CTA : en marche (couleur jaune) ou en arrêt (couleur grise).
- 24) Indique si la CTA est alimentée ou non.
- 25) Paramètres du régulateur PI.

Conclusion :

Durant ce chapitre, nous avons exposé l'application conçue sous Vijeo Citect pour la supervision de l'installation électrique, de l'éclairage et de climatisation pour un bâtiment à six niveaux avec rez-de-chaussée et sous sol. Pour se rapprocher au mieux de la réalité du terrain, nous avons pris soin de chercher dans diverses fiches techniques des équipements adéquats pour concevoir notre système de supervision.

Conclusion Générale

Conclusion Générale

Conclusion Générale

Etant donné la place qu'occupe actuellement les bâtiments intelligents, leur consacrer un projet de fin d'étude nous a semblé une excellente manière d'appliquer nos connaissances acquises durant notre cycle d'ingénieur en automatique.

En effet, durant ce projet, nous avons conçu une application de supervision de l'installation électrique, d'éclairage et de climatisation d'un bâtiment et ceci en utilisant nos connaissances en automatique, électrotechnique et informatique industrielle. Nous avons de plus exploré divers domaines tels que le génie climatique qui traitent entre autre des théories du confort visuel et thermique. Ces domaines qui ne faisaient pas partie de notre formation mais étaient indispensables à la conception de notre application.

Perspectives

Ce mémoire constitue l'aboutissement de notre cursus d'ingénieur mais également un point de départ pour d'autres projets de fin d'études traitant des bâtiments intelligents. En effet de nombreuses perspectives s'offrent à nous :

- Dimensionnement des équipements de l'installation électriques (disjoncteurs, fusibles...) et de la Centrale de Traitement d'Air (les différents capteurs et actionneurs, diamètres des conduits, type et puissance des ventilateurs...) en vue de l'implémentation de la station de supervision sur un bâtiment réel.
- Dimensionnement des paramètres des régulateurs utilisés et ceci soit par des méthodes expérimentales sur terrain ou par une modélisation analytique.
- Utilisation d'équipements intelligents et communiquant sur des réseaux orientés bâtiment.
- Intégration du GEG (Groupe d'Eau Glacée) dans le système de supervision

Bibiographie

Bibliographique

[1] [http://conseils.xpair.com/gestion technique batiment/conseilconsulter savoir fairedomoti que_tc_gtb-934.html](http://conseils.xpair.com/gestion_technique_batiment/conseilconsulter_savoir_fairedomoti_que_tc_gtb-934.html)

[2] <http://www.climatechange.org>

[3] www.energieplus-lesite.be.

[4] W.C. Turner, S. Doty, « Energy Management Handbook », The Fairmont Press, 2006.

[5] FEKNOUS Baya, HAMI Elias, « Development de blocs fonction sous UNITY PRO, Application aux Régulateurs P.I.D Auto-Ajustables », Projet de Fin d'Etude, Automatique, ENP 2009.

[6] logiciel Unity Pro.

[7] Formation Unity Pro (document Schneider Electric).

[8] logiciel Vegio Vitect.

Fiches Techniques

[9] Karno, « Vannes de régulation trois voies», 2002

[10] TAC Automation, « Automatisme du Bâtiment », 2007

[11] Schneider Electric, « Efficacité Energétique des Bâtiments »,2009

[12] Schneider Electric, « Catalogue résidentiel et petit tertiaire »,2010

Annexes

Equipements dans la GTB :

A-1 Equipement MT :

1. Le câble MT : le câble MT fait la liaison entre la cellule (déjà existante sur le site) et le transformateur. Sa fonction proprement dit est le transport d'énergie électrique d'un endroit vers un autre. Ce câble sera placé dans la cellule existante jusqu'au transformateur qui lui sera à proximité (c'est-à-dire dans le même local, très peu distant l'un de l'autre).

2. Le transformateur : le transformateur a pour rôle essentiel d'abaisser ou l'augmenter la tension, ici dans notre installation le transformateur sera d'abaisser la tension donc généralement on dit que c'est un transformateur abaisseur. Il aura du côté primaire (entrée du transfo) une tension de 10KV et du côté secondaire (en sorti du transfo) il aura comme valeur une tension de 400V.

A-2 Equipement Electrique BT :

1. Tableau Electrique : Le tableau électrique recouvre les divers organes de commande et de puissance. Il y aura placé sur ses armoires des étiquettes afin de pouvoir les identifier.

Les armoires sont juxtaposable (extensible), fabriquées à partir d'une feuille d'acier, protégée à l'intérieur et à l'extérieur par une peinture résine polyester époxy et grise.



Figure A-1 : Exemple de disposition d'armoire dans un TGBT

A-3 Relais de protection et de contrôle S80 :

Le S80 est un relais numérique de protection et de contrôle de la gamme Sepam série 80, ayant pour rôle de surveiller en permanence l'état du réseau d'énergie et de provoquer sa mise en sécurité lors d'une perturbation indésirable : surcharge, court-circuit. Spécialement étudié pour répondre aux clients exigeants des grands sites industriels. Il apporte des solutions éprouvées pour la distribution électrique et pour la protection des machines.

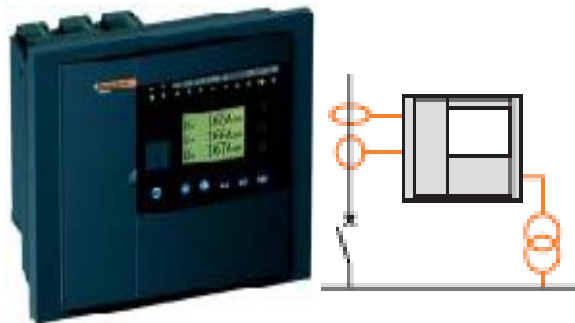


Figure A-2 : Relais de protection S80

Il vise principalement les fonctions suivantes :

- protection complète des transformateurs et groupes-blocs.
- protection complète des moteurs et des générateurs.
- Contrôle du synchronisme entre 2 réseaux à coupler.
- Mesure du taux de distorsion harmonique sur le courant et la tension, pour évaluer la qualité de l'énergie du réseau.
- 42 entrées /23sorties pour assurer la commande intégrale de l'équipement.
- Interface Home Machine synoptique pour la commande locale de l'appareillage.
- Logiciel de programmation (Logipam) en option, pour programmer des fonctions de commande et de surveillance spécifiques.
- 2ports de communication Modbus, pour intégration de Sepam dans 2 réseaux distincts, ou dans des architectures redondantes.
- Cartouche mémoire amovible pour remise en service rapide après remplacement d'une unité de base défectueuse
- Pile de sauvegarde pour conservation des données historiques et des enregistrements d'oscilloperturbographie.

A-4 Les actionneurs analogiques :

Vanne mélangeuse 3 voies :

Pour alimenter la batterie froide avec de l'eau à basse température (température modulable), on réalise un mélange entre l'eau froide qui arrive du groupe frigorifique (**GEG** : Groupe Eau Glacée) et l'eau qui arrive de la batterie froide et qui retourne vers le GEG.

C'est le rôle de la vanne, appelée "vanne trois voies mélangeuse", placée entre l'aller et le retour de l'installation qui permet d'effectuer ce mélange. Son principe de fonctionnement est basé sur la rotation d'un secteur entre les 3 voies d'eau comme le montre la figure suivante :



Figure A-3 : Principe de fonctionnement de la vanne 3 voies mélangeuse

[3]

Nous avons pris comme exemple les vannes 3 voies de la gamme **karno** dont les références sont résumées dans le tableau suivant :

Référence	Désignation
V3D20-M23	Vanne 3 voies DN 20 Moteur 230V 3 Points
V3D20-M49	Vanne 3 voies DN 20 Moteur 0...10V
V3D25-M23	Vanne 3 voies DN 25 Moteur 230V 3 Points
V3D25-M49	Vanne 3 voies DN 25 Moteur 0...10V
V3D32-M23	Vanne 3 voies DN 32 Moteur 230V 3 Points
V3D32-M49	Vanne 3 voies DN 32 Moteur 0...10V
V3D40-M23	Vanne 3 voies DN 40 Moteur 230V 3 Points
V3D40-M49	Vanne 3 voies DN 40 Moteur 0...10V
V3D50-M23	Vanne 3 voies DN 50 Moteur 230V 3 Points
V3D50-M49	Vanne 3 voies DN 50 Moteur 0...10V

Tableau A-1 : Les différentes gammes de vannes 3 voies « karno » [9]

Ces vannes sont actionnées par des servomoteurs dont voici les principales caractéristiques :

	M23	M49
Tension d'alimentation	230 VAC +6% -10%	24 VAC \pm 10%
Consommation	7 VA	3 VA
Signal de télécommande	3-points	0 .. 10 Vdc (70 kOhm)

Tableau A-2 Les principales caractéristiques des servomoteurs pilotant les vannes 3 voies « karno » [9]

Pour notre application nous avons choisi de prendre une vanne 3 voies pilotée par un moteur 0-10V (2fils).

A-5 Centrale de mesure Power Meter PM710 :

Le PM710 est un appareil de mesure permet à la fois d'économiser l'énergie et de réduire le coût de l'énergie consommée, grâce à ses fonctions de sous-comptage et d'optimisation du contrat, et maîtriser le réseau électrique en temps réel et donc d'augmenter sa fiabilité. Il vise principalement les applications :

- Ø Mesure locale.
- Ø Sous-comptage / allocation des coûts.
- Ø Surveillance à distance de l'installation électrique.
- Ø Surveillance des harmoniques (THD).
- Ø Alarmes sur seuils maxi/mini et sur l'état des entrées/sorties.



Figure A-4 : Centrale de mesure PM710

A-6 Les capteurs :

A-6-1 Détecteur de présence :

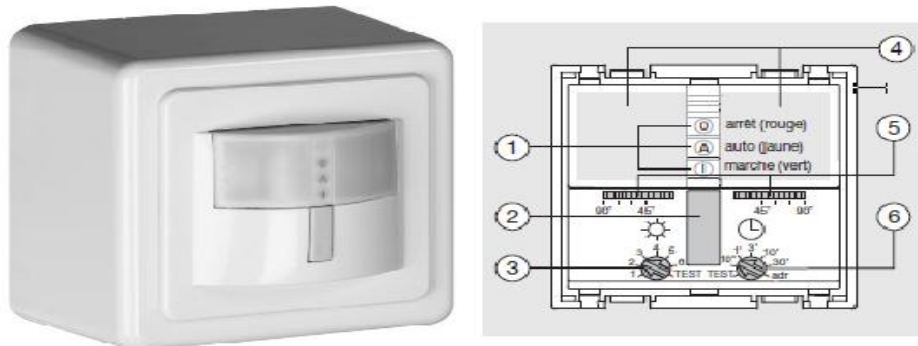


Figure A-5: Le détecteur de présence

1 -Voyants LED 2 -Touche de commande 3 -Réglage du seuil de luminosité 4 -Lentilles de détection 5 -Réglage de l'angle de détection 6 -Temporisation

Les caractéristiques de détecteur de présence sont illustrées dans le tableau suivant :

La tension U (V)	230 V \pm 10%
La fréquence F (Hz)	50 Hz
L'adresse IP	20
Protection	Electronique
Angle de détection	90° -> 180°
Temporisation	2 s -> 30 min 1 s
Seuil de luminosité	5 -> 800 Lux
Charge minimum (P min)	60 VA
Utilisation en Maître	Oui
Utilisation en Esclave	Non

A-6-2 Le capteur de luminosité de jour :

Ce capteur est à utilisé pour le mode de la gestion d'éclairage en fonction de la luminosité du jour. Le capteur SLR310 fabriqué par la firme TAC (filère de Schneider) constitue un choix pour notre application.

Caractéristique du capteur de luminosité SLR 310 :

- température de fonctionnement -20... +70 °C.
- précision ± 5 à 25 °C.
- plage de mesure (choix par cavaliers) 0 - 400 lx ou 0 - 20 klx.
- signal de sortie 0 - 10 V (3 fils).
- dimensions 75 x 75 x 25 mm.
- alimentation 24 V CA ou 15 V CC.
- montage mural.



Figure A-6: *Le capteur de luminosité SLR 310 [10]*

A-6-3 Détecteur de mouvement Kallysto : le détecteur de mouvement Kallysto est utilisé pour une commande d'éclairage moderne. Il assure un éclairage confortable et adapté aux besoins, par exemple dans les couloirs, cages d'escaliers, halles. Monté à hauteur d'interrupteur, un détecteur de mouvement remplace pour ainsi dire un interrupteur d'éclairage. Ainsi, le détecteur de mouvement procure non seulement une meilleure sécurité, mais contribue également à économiser de l'énergie car l'éclairage ne sera activé qu'en présence d'une personne.



Figure A-8 : *capteur de lumière*

A-7 Les disjoncteurs :

A-7 -1 Le disjoncteur Compact série NSX :

Cette gamme propose des disjoncteurs ayant un courant allant de 100 à 630A ; pour notre application, nous avons utilisé des disjoncteurs de 100A, 250A et 400A. Le pouvoir de coupure de ces disjoncteurs varie de 36KA à 150KA.



Figure A-9: *Le disjoncteur Compact NSX [11]*

A-7-1-1 Les auxiliaires électriques :

Le compact NSX possède un seul modèle d'auxiliaire utilisable sur tous les appareils Compact NSX100 à NSX630, ce modèle réalise 3 fonctions différentes selon l'emplacement dans lequel il est logé (OF, SD, SDE).

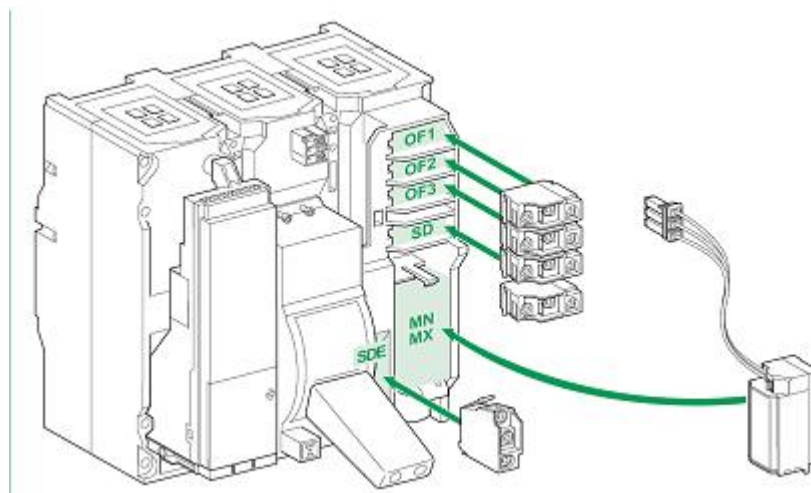


Figure A-10 : *Emplacement des auxiliaires dans un Compact NSX [11]*

A-7-1-2 Télécommande :

Les Compact NSX100 à 630 peuvent recevoir un bloc de télécommande adaptable en face avant (la référence est LV432641). Ces blocs sont commandés par une tension alternative de 230 V. Ils restent toutefois actionnables manuellement en face avant.

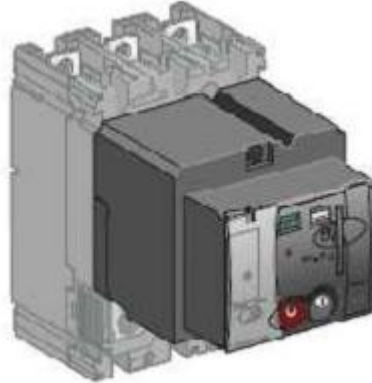


Figure A-11: *Bloc de télécommande montée sur le Compact NSX [11]*

A-7-2 L'inverseur de source Masterpact 2500 et Masterpact 1000 :

L'inverseur de source est un élément essentiel pour la continuité de service et la gestion de l'énergie. Le système réalise la permutation entre :

- une source N qui alimente normalement l'installation.
- une source R de remplacement qui peut être une arrivée de réseau supplémentaire ou un groupe électrogène. L'inverseur de source est réalisé autour de deux appareils interverrouillés mécaniquement. Les interverrouillages empêchent toute mise en parallèle même transitoire des deux sources. Pour notre application, trois inverseurs de sources ont été prévu pour secourir l'ensemble de l'installation électrique de notre bâtiment (sauf le système de climatisation). Ces inverseurs de sources sont disposés au niveau des TGBT 1 et 2.

A-7-2-1 L'inverseur de source Masterpact 2500 :

L'inverseur de source Masterpact 2500 constitue un choix pour le TGBT 1. Ce dispositif est réalisé autour de deux disjoncteurs, l'un associé à la source normal et l'autre à la source de secours (le groupe électrogène à travers son TGGE). Les deux disjoncteurs sont interverrouillés mécaniquement afin d'éliminer tous risque de fermeture simultanée même transitoire.

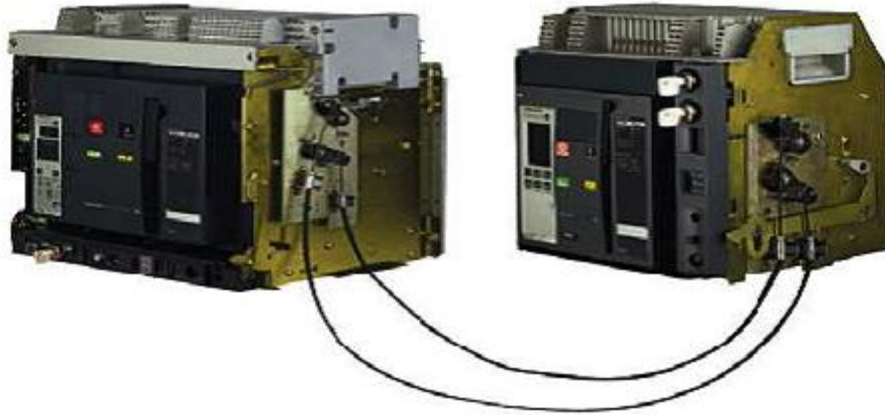


Figure A-12 : *Interverrouillage par câble de deux disjoncteurs Masterpact constituant un inverseur de source [11]*

A-7-2-2 L'inverseur de source Masterpact 1000 :

Cet inverseur de source constitue un choix pour secourir les deux cotés de TGB2, il possède toutes les caractéristiques du masterpact 2500.

A-7-3 Le disjoncteur série C60N :

Les tableaux divisionnaires d'éclairage de notre application contiennent des disjoncteurs de départ permettant la protection.

Le disjoncteur série C60N ayant comme référence 24202 constitue un choix pour notre application. Ce disjoncteur possède un calibre de courant de 16A et un courant de déclenchement de 10 kA.



Figure A-13: *Disjoncteur C60N avec bloc différentiel Vigi [11]*

A-7-4 Les télérupteurs :

L'alimentation des lampes constituant le système d'éclairage de notre application se fait à travers des télérupteurs. Les télérupteurs sont utilisés pour la commande par boutons poussoirs de circuits d'éclairage constitués de :

- lampes à incandescence, lampes halogènes basse tension, etc. (récepteurs résistifs).
- tubes fluorescents, lampes à décharge, etc. (récepteurs inductifs).

Le télérupteur ayant comme référence 15032 de Schneider constitue un choix pour notre application. Il possède un calibre de courant de 16A, la tension de commande est de 230 V CA.

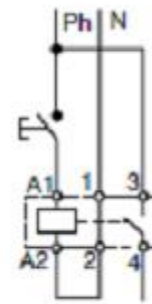


Figure A-14: *Le télérupteur 15032 de Schneider [12]*

La figure ci-dessus illustre le fonctionnement du télérupteur, lorsque celui-ci reçoit une impulsion de commande de l'automate (schématiser par un bouton poussoir dans la figure), la bobine du télérupteur se trouvera branchée entre la phase et le neutre et elle sera donc excitée, ceci engendrera la fermeture du contact alimentant le circuit d'éclairage et donc l'allumage de ce dernier.

A-8 Les contrôleurs :

A-8-1 Les automates programmables Quantum :

A-8-1-1 Présentation de la Plate-forme Quantum :

Le système Quantum est un calculateur spécifique offrant des fonctions de traitement numérique. Quantum est conçu pour fournir un contrôle en temps réel aux applications industrielles et de production dans une architecture modulaire et évolutive utilisant les modules suivants :

- Modules de l'automate (CPU).
- Modules d'alimentation (CPS).

- Modules d'E/S (Dxx, Axx).
- Modules d'interface réseau (y compris les modules bus de terrain).
- Modules intelligents/spécifiques.



Figure A-10 : Automate Quantum

Il offre ainsi de nombreuses fonctionnalités dont on peut citer :

- Temps de cycle réduit avec acquisition rapide des entrées/sorties.
- Traitement d'interruption sur événement de temps ou en provenance d'entrées.
- Traitement possible en tâche rapide comme en tâche maître.
- Extension des capacités mémoire par cartes mémoire PCMCIA.
- Multiples ports de communication intégrés au processeur.
- Diagnostic et maintenance aisés grâce au bloc de visualisation LCD en face avant des processeurs haut de gamme.

a) L'Unité Centrale (CPU) :

Le module est installé sur un rack d'E/S locales Quantum. Il s'agit d'un système électronique numérique qui utilise une mémoire programmable pour le stockage interne des instructions de l'utilisateur. Ces instructions servent à implémenter des fonctions spécifiques, telles que: l'arithmétique, la logique, le couplage...etc.

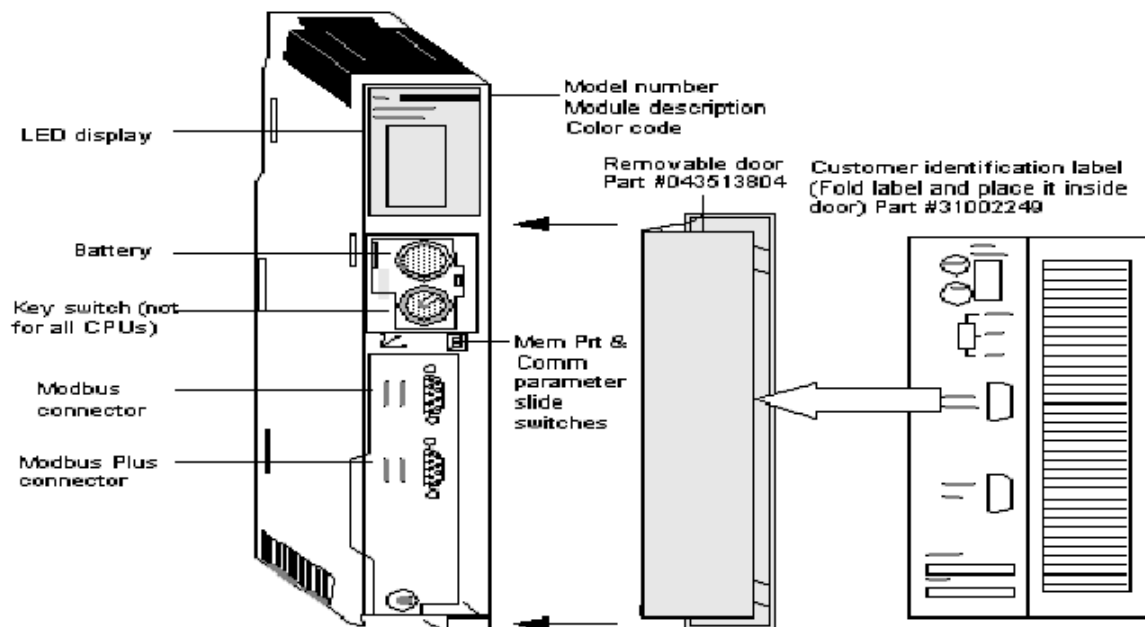


Figure A-11: Processeur de base

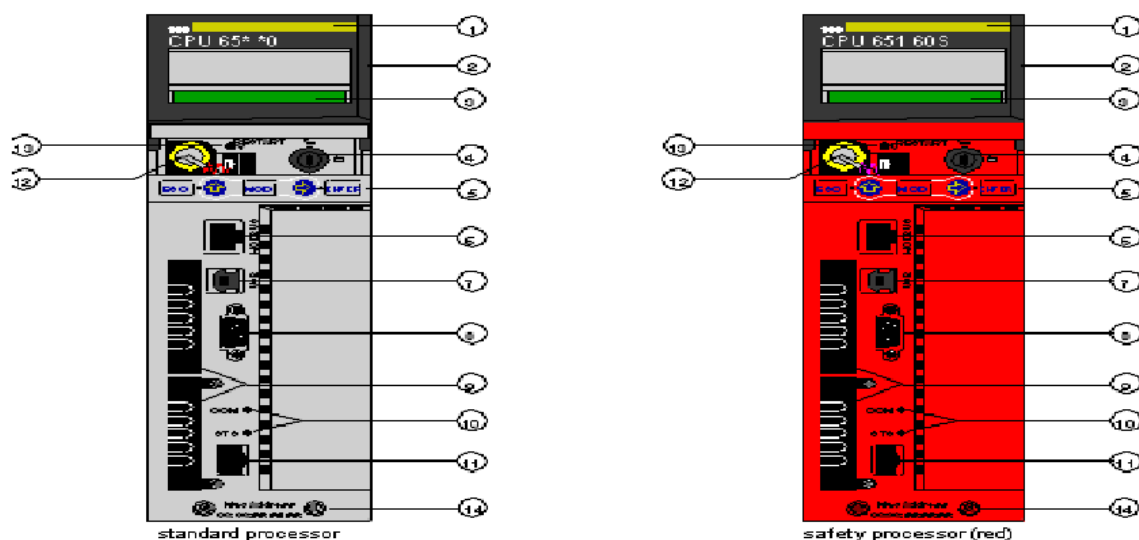


Figure A-12: Processeurs avancés

1. Numéro du modèle, description du module, code couleur. 2. Couvre-objectif (ouvert).
3. Ecran LCD (recouvert ici par le couvre-objectif). 4. Interrupteur à clé.
5. Clavier (comportant 2 voyants rouges). 6. Port Modbus (RS-232) (RS-485).
7. Port USB. 8. Port Modbus Plus. 9. Emplacements PCMCIA (A et B).
10. Voyants (jaunes) pour la communication Ethernet. 11. Port Ethernet.
12. Pile (installée par l'utilisateur). 13. Bouton de réinitialisation. 14. 2 vis.

b) Modules d'alimentation (CPS) :

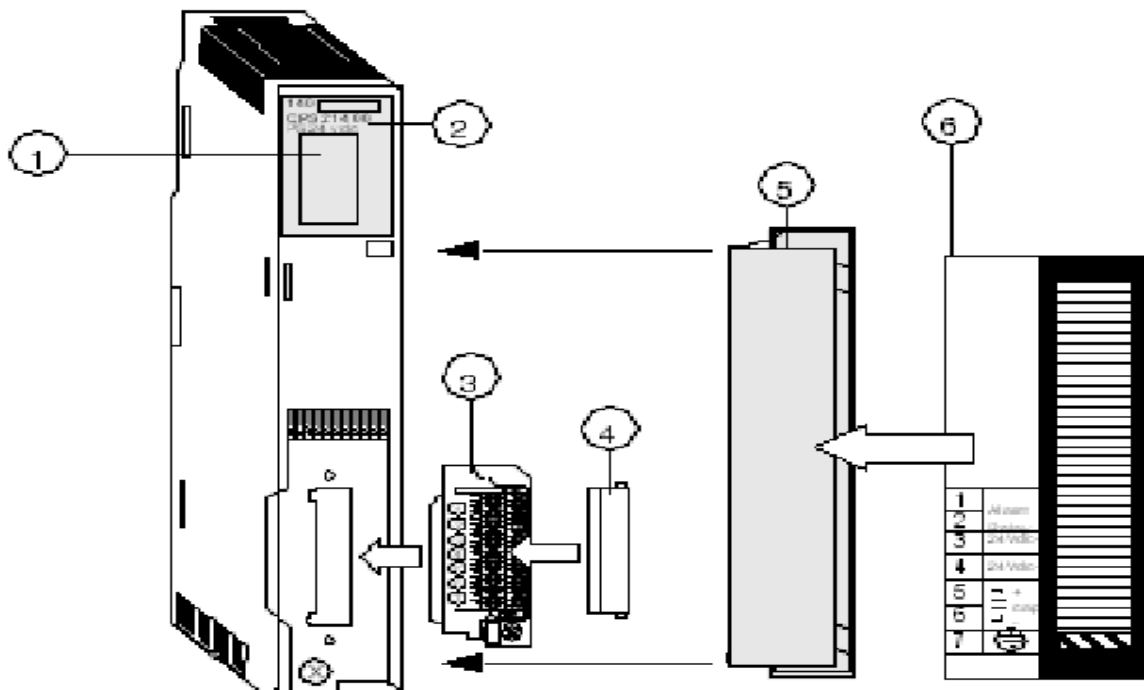


Figure A-13: Vue d'un module d'alimentation

1. Voyants 2. Numéro du modèle, description du module, code couleur 3. Connecteur de câblage 4. Capot de protection du connecteur de câblage 5. Capot amovible 6. Étiquette d'identification.

A-8-2 Les contrôleurs spécialisés :

En plus des automates programmables classiques, d'autres contrôleurs spécialisés et dédiés au domaine de l'automatisation des bâtiments sont aussi utilisés. Le contrôleur IHC (pour Intelligent Home Control) est l'un des contrôleurs les plus répandus qui sont dédiés au domaine du bâtiment, c'est un **système centralisé** pour la gestion d'une installation électrique, toutes les entrées (bouton-poussoir, détecteurs de mouvement et autres capteurs) et toutes les sorties (lampes, volets-roulants, chauffage, ventilation...) sont ramenées au **coffret électrique**. Le système est construit autour **d'un petit automate** (le contrôleur IHC) qui les ordres d'exécution (aux sorties) selon la programmation réalisée et en fonction des entrées. Ce contrôleur possède un large choix de fonctions : commandes centralisées, scénarios d'éclairage, contrôle de volets roulants ...etc.



Figure I-13 : Contrôleur IHC de Schneider

A-9 Ventilateurs utilisés :

A-9-1 Les **ventilateurs centrifuges** : l'air est aspiré parallèlement à l'axe de rotation et propulsé par force centrifuge perpendiculairement à ce même axe.



Ventilateur centrifuge.

Figure A-14 : Le ventilateur centrifuge

Il existe deux principaux types de ventilateurs centrifuges qui sont :

- **A aubes inclinées vers l'avant** : appelés aussi "**à action**" ou "**en cage d'écureuil**".

La roue de ces ventilateurs comprend un nombre important d'aubes de faible hauteur.

Elles sont inclinées dans le sens de rotation de la roue.

- **A aubes inclinées vers l'arrière** : appelés aussi "**à réaction**". La roue de ces ventilateurs comprend un nombre réduit d'aubes de plus grande hauteur. Elles sont inclinées dans le sens inverse de la rotation de la roue.

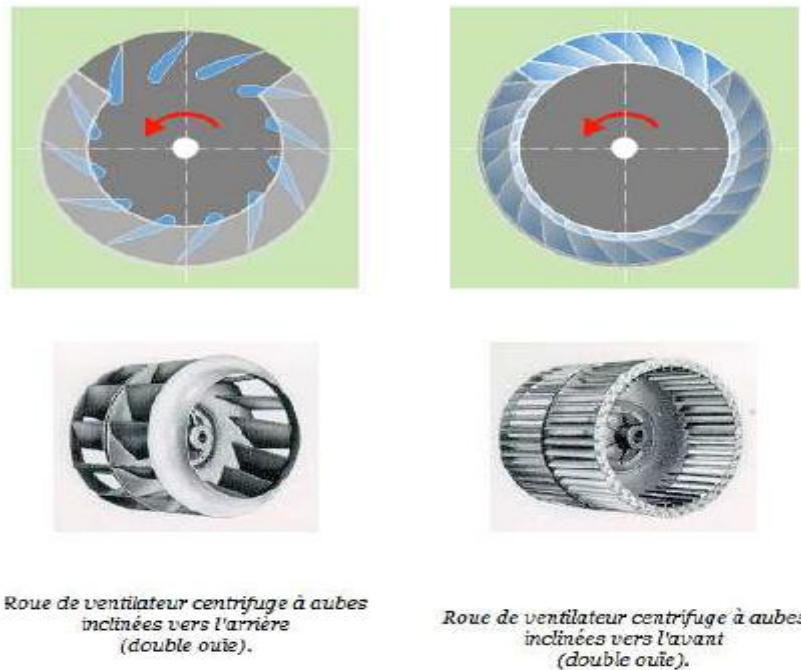


Figure A-15 : Les deux principaux types de ventilateurs centrifuges

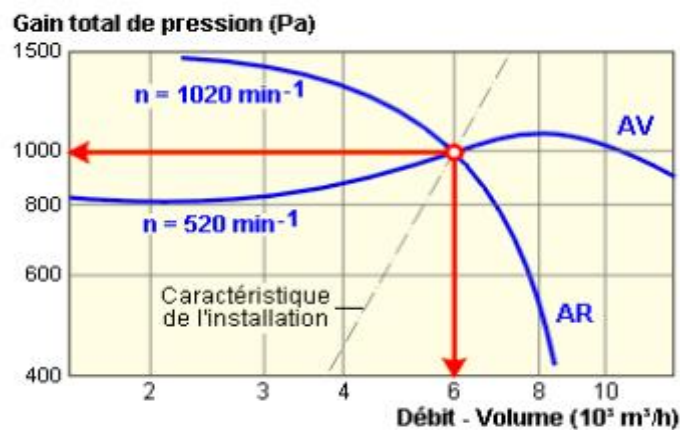


Figure A-16: Courbe caractéristique d'un ventilateur à aubes recourbées vers l'avant (AV) et à aubes recourbées vers l'arrière (AR), pour un même point de fonctionnement.

Pour un même point de fonctionnement, la roue avec aubes vers l'avant nécessite toujours une plus faible vitesse de rotation. Le bruit produit par la roue avec aubes vers l'avant sera donc moins intense. Par contre, lorsque la perte de charge varie dans le circuit aéraulique, la variation de débit qui en résulte est nettement plus importante pour les roues inclinées vers l'avant. Leur point théorique de fonctionnement doit donc être calculé avec beaucoup de rigueur.

En effet, si la résistance réelle est inférieure à la résistance calculée, le ventilateur débitera nettement plus que prévu et sa consommation sera supérieure aux prévisions. A l'inverse des roues à aubes inclinées vers l'arrière, la roue avec aubes vers l'avant ne demande pratiquement pas de variation de vitesse pour obtenir une augmentation importante de débit.

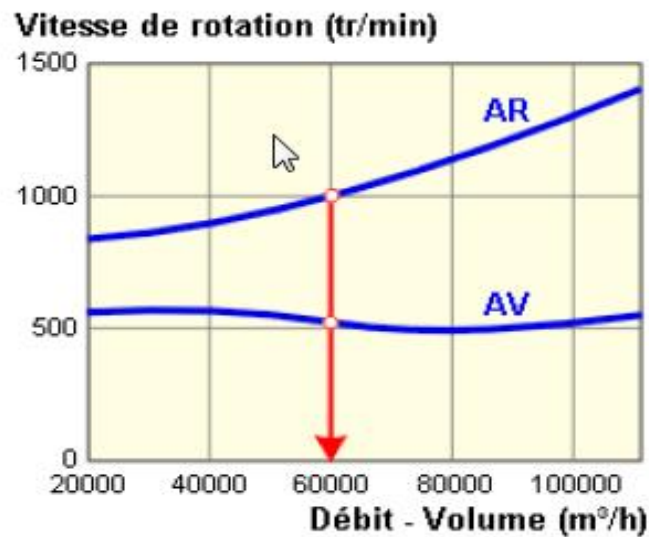


Figure A-17 : Variation du débit d'air en fonction de la vitesse de rotation du ventilateur

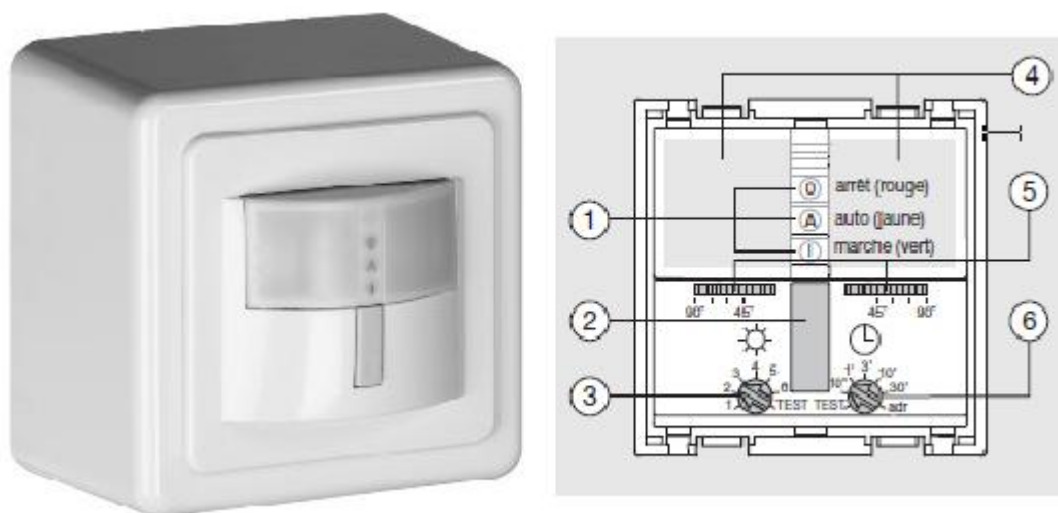


Figure A-18 : Le détecteur de présense

1 -Voyants LED **2** -Touche de commande **3** -Réglage du seuil de luminosité **4** -Lentilles de détection **5** -Réglage de l'angle de détection **6** -Temporisation

Caractéristiques techniques :

Les caractéristiques de détecteur de présence sont illustrées dans le tableau suivant.

La tension U (V)	230 V \pm 10%
La fréquence F (Hz)	50 Hz
L'adresse IP	20
Protection	Electronique
Angle de détection	90° -> 180°
Temporisation	2 s -> 30 min 1 s
Seuil de luminosité	5 -> 800 Lux
Charge minimum (P min)	60 VA
Utilisation en Maître	Oui
Utilisation en Esclave	Non

B-1 Les évènements :

Les événements servent à déclencher une action, comme une commande ou une série de commandes. Ils permettent par exemple d'indiquer l'achèvement d'un processus à un opérateur ou de déclencher l'exécution d'une série d'instructions lorsqu'un processus franchit une étape spécifique.

L'activation d'un événement peut se faire par trois méthodes :

1. Automatiquement ; à un temps donné et avec une périodicité définie.
2. Automatiquement ; lorsqu'un déclencheur prend la valeur vraie.
3. Automatiquement ; lorsqu'un déclencheur prend la valeur vraie et un temps donné avec une périodicité définie.

Exemple d'évènement : Nous allons maintenant illustrer les événements par un exemple, en se référant à notre exemple d'application précédemment défini, on considère maintenant un autre scénario, on veut maintenant adopter une gestion horaire de l'éclairage de notre bureau ; les luminaires s'allument à une heure donnée (8 :00h par exemple) et s'éteignent à une heure donnée (18 :00h par exemple) pour cela il suffit de définir deux événements et de leurs associer les actions adéquates. La définition de ces deux événements se fait en suivant les étapes suivantes :

Etape 1 : Sur l'éditeur de projet Citect, on clique sur **Systèmes**, on sélectionne ensuite **événements**, une fenêtre apparaît alors permettant de configurer l'évènement.

Etape 2 : on édite les propriétés de l'évènement comme indiqué sur la figure III-19, on commence par choisir le nom de l'évènement, **Heure** permet de définir l'heure de déclenchement de l'évènement, et **Période** définit la périodicité de l'évènement (dans notre cas 24h, puisque on veut que l'allumage soit quotidien), dans le champs **Déclenchement**, on choisit (éventuellement) le tag permettant de déclencher l'évènement (notre évènement sera déclenché indépendamment d'une condition, donc pas besoin de tag déclencheur), dans le champs **Action**, on écrit le code définissant toutes les actions relatives à notre évènement (ici c'est la mise à 1 des tags relatifs à l'éclairage afin d'allumer les luminaires des rangées).

The screenshot shows a software window titled "Evénements [projetExempleAppllic]". It contains several input fields and buttons. The "Nom" field is set to "Global". The "Heure" field is set to "08:00:00". The "Nom Cluster" dropdown is set to "cluster". The "Période" dropdown is set to "24:00:00". The "Déclenchement" field is empty. The "Action" field contains the text "etat_et1_br1_rg1=1; etat_et1_br1_rg2=1; etat_et1_br1_rg3=1;". The "Commentaire" field contains the text "Allumage des Luminaires du premier etage". Below the fields are four buttons: "Ajouter", "Remplacer", "Supprimer", and "Aide". At the bottom left, it says "Enreg : 1".

Figure B-1: Configuration d'un évènement

Etape 3 : notre évènement étant maintenant défini, reste à l'activer en utilisant l'assistant de configuration de poste (mode avancé), cette étape n'est pas nécessaire si on choisit Global comme nom de l'évènement puisque tous les évènements ayant comme nom global seront activés par défaut.

B-2 Les alarmes :

Les alarmes permettent d'alerter d'opérateur d'évènements spécifiques, il existe deux types d'alarmes :

1. **Les alarmes matérielles :** Vijeo Citect exécutent automatiquement des programmes de diagnostic pour vérifier l'état des périphériques, les défauts éventuelles sont reporter automatiquement à l'opérateur, cette option est intégrée automatiquement à Vijeo Citect et ne nécessite pas de configuration.
2. **Les alarmes configurées :** au contraste des alarmes matérielles, ces alarmes sont à configurer par le développeur pour permettre de **signaler** des évènements spécifiques.

Nous allons maintenant illustrer comment configurer des alarmes de base en se basant sur un petit exemple d'application.

Exemple d'application :

En considérant toujours notre application précédente, on veut maintenant configurer une alarme pour signaler l'intrusion d'une personne, cette personne sera active si l'un des détecteur de mouvement détecte la présence d'une personne entre 20 :00 h du soir et 8 :00h du matin, en cas de déclenchement d'alarme, l'opérateur pourra alors vérifier (à l'aide d'une camera de surveillance par exemple) qu'il s'agit bien d'un intrus et non pas de quelqu'un du personnel.

a) Etape 1 : vérification de la plage horaire :

Pour configurer notre évènement, on a besoin d'abord de vérifier qu'on est entre 20 :00h et 8 :00h, pour cela la méthode la plus simple est de mettre un nouveau tag à 1 en utilisant un évènement et ceci comme décrit dans la section **III-10**. La figure C-2 illustre les deux évènements permettant de marquer l'intervalle dans lequel on veut signaler une intrusion.

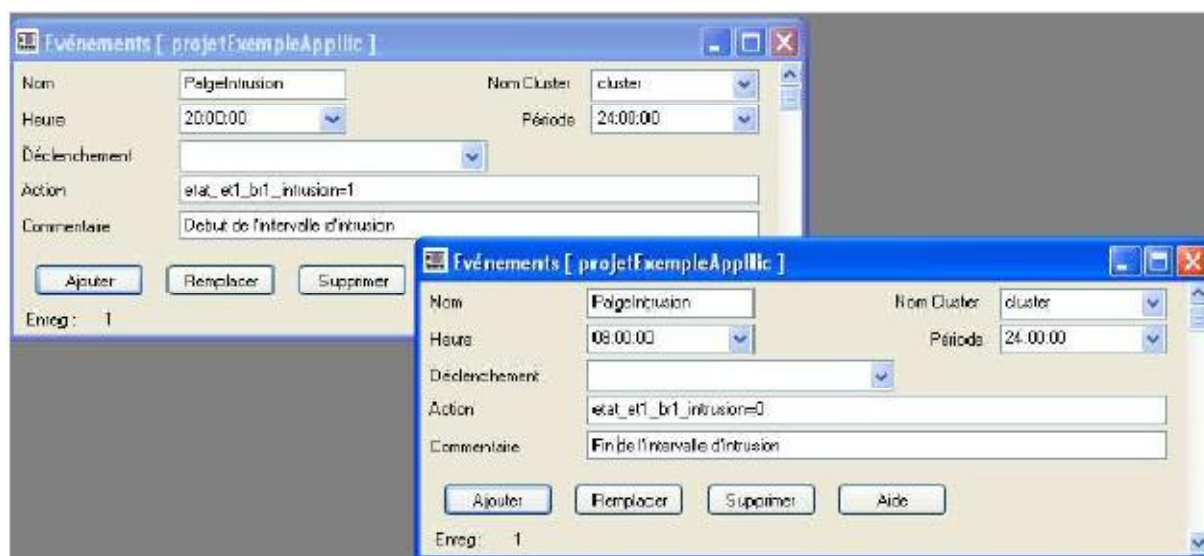


Figure B-2 : Configuration de deux évènements afin de marquer l'intervalle de détection d'intrusion

b) Etape 2 : Configuration de l'alarme

Nous allons maintenant configurer notre alarme, pour cela, on commencera d'abord par définir un serveur d'alarme (s'il n'est pas déjà défini), pour ce faire, il suffit de suivre les étapes expliquées dans la section **III-8-2-1**. L'étape suivante consiste à configurer l'alarme proprement dite, sur l'éditeur de projet Citect on clique sur le menu alarme, on choisit alors **Alarmes Digitales**, la fenêtre illustrée sur la figure C-3 apparaît alors et permet d'éditer la configuration de l'alarme.

Sur la fenêtre on voit donc les propriétés suivantes :

Tag de l'alarme : Nom de l'alarme. Le nom doit être unique au cluster.

Nom du cluster : Nom du cluster qui exécute cette alarme. Si le nom du cluster n'est pas défini, Vijeo Citect estime que l'alarme est exécutée sur tous les clusters définis.

Nom de l'alarme : Nom du périphérique physique associé à l'alarme. Cette propriété est facultative. Vijeo Citect ne l'utilise que pour afficher les détails d'une alarme à l'écran ou pour les enregistrer dans un périphérique.

Description de l'Alarme : Elle peut inclure des données variables. Cette propriété est facultative. Vijeo Citect ne l'utilise que pour afficher les détails d'une alarme à l'écran ou pour les enregistrer dans un périphérique.

Tag A / Tag B : Les variables numériques (tags) déclenchant l'alarme, l'alarme devient active lorsque les deux tags (ou seulement le tag A si le champ du tag B est laissé vide) prennent la valeur 1.

Catégorie : Numéro ou étiquette de la catégorie d'alarme. Cette propriété est facultative. Si la catégorie n'est pas spécifiée, l'alarme utilise par défaut la catégorie 0.

Délai (hh:mm:ss) : Délai de l'alarme ; Une alarme horodatée devient active lorsque l'état de la condition de déclenchement reste vrai pendant l'intervalle de temps défini par ce paramètre.

Figure B-4: Configuration d'une alarme digitale

Sur la figure C-4, on a configuré une alarme digitale relative au capteur de mouvement de la deuxième rangée, on refait la même chose ensuite avec les deux autres capteurs (en entrant comme variable A les deux tags `etat_et1_br1_pos1_captmvt` et `etat_et1_br1_pos3_captmvt` successivement. Sur la figure, nous remarquons que nous avons utilisé la catégorie 2, l'étape suivante consistera à éditer ses propriétés.

c) Etape 3 : Configuration de la catégorie d'alarme :

Une catégorie permet de définir les caractéristiques de l'affichage (police, type de page), de l'enregistrement (imprimante ou fichier), et l'action à entreprendre lors du déclenchement d'une alarme (par exemple, activer une alarme sonore). Pour configurer une catégorie, on sélectionne **Alarmes** puis **Catégorie d'Alarmes**. Une fenêtre apparaît alors (Figure C-5) permettant d'éditer les propriétés de cette catégorie.

Figure B-5: Configuration d'une catégorie d'alarme

Les propriétés d'une catégorie sont expliquées ci-dessous :

Numéro de catégorie : Catégorie de l'alarme (0-16 375) la catégorie 254 est réservée aux entrées de l'historique d'alarmes créées par l'utilisateur. La catégorie 255 est réservée aux alarmes matérielles.

Affichage sur la page d'alarmes : Détermine si les alarmes de cette catégorie sont affichées sur la page d'alarmes. La valeur par défaut de ce champ est VRAI.

Affichage sur la page d'historique : Détermine si les alarmes de cette catégorie sont affichées sur la page d'historique. La valeur par défaut de ce champ est VRAI.

Non acquittées : police des alarmes désactivées : Police utilisée pour l'affichage des alarmes qui ne sont plus actives (mais n'ont pas été acquittées). Cette propriété est facultative.

Acquittées : police des alarmes désactivées : Police utilisée pour l'affichage des alarmes qui ne sont plus actives et qui ont été acquittées. Cette propriété est facultative.

Non acquittées : police des polices actives : Police utilisée pour l'affichage d'une Alarme active non acquittée. Cette propriété est facultative. Si vous ne précisez pas de police, la valeur par défaut est 10 point JAUNE.

Acquittées : police des alarmes actives : Police utilisée pour l'affichage des alarmes actives acquittées. Cette propriété est facultative. Si vous ne précisez pas de police, la valeur par défaut est 10 point CYAN.

Action ON : Commande Cicode exécutée lorsqu'une alarme de cette catégorie est acquittée.

Exemple : **Action Alarme ON :** STOP_PROCESS = 1; La variable numérique STOP_PROCESS est activée (ON) lorsqu'une alarme de cette catégorie est déclenchée.

Action OFF : Commande Cicode exécutée lorsqu'une alarme de cette catégorie est réinitialisée (254 caractères maximum). Exemple : **Action Alarme OFF:**

ENABLE_PROCESS = 1;

La variable numérique ENABLE_PROCESS est activée (ON) lorsqu'une alarme de cette catégorie est réinitialisée.

Action ACK : Commande Cicode exécutée lorsqu'une alarme de cette catégorie est acquittée.

Périphérique d'historique : Périphérique auquel l'historique d'alarme est envoyé. Une alarme est enregistrée dans le périphérique d'historique lorsqu'elle est désactivée après avoir été affichée pendant une période supérieure à celle spécifiée par le paramètre SummaryTimeout. Une alarme enregistrée dans ce périphérique est supprimée de la page d'historique. Cette propriété est facultative. Si vous ne la spécifiez pas, les historiques d'alarmes ne sont pas enregistrés.

Périphérique d'enregistrement : Périphérique auquel les changements d'état de l'alarme sont envoyés. Une alarme est écrite dans ce périphérique à chaque fois qu'elle change d'état (ex. on, off, acquittée, activée, désactivée). Lorsque l'alarme est imprimée ou écrite dans un fichier ou un périphérique, le format spécifié dans ce périphérique prime sur le format d'affichage. Cette propriété est facultative. Si vous ne la spécifiez pas, les changements d'états ne sont pas enregistrés.

Enregistrement des transitions d'alarmes ON : Enregistre les détails des alarmes lorsqu'elles deviennent actives. La valeur par défaut de ce champ est VRAI.

Enregistrement des transitions d'alarmes OFF : Enregistre les détails des alarmes lorsqu'elles deviennent inactives. La valeur par défaut de ce champ est VRAI.

Enregistrement des transitions d'alarmes ACK : Enregistre les détails des alarmes lorsqu'elles sont acquittées (6 caractères maximum). La valeur par défaut de ce champ est VRAI.

III-11-2 Exécution de l'exemple :

Afin d'afficher une alarme lors de l'exécution de notre application, on change le modèle de page de notre application, on adopte un modèle préconfiguré contenant déjà le raccourci permettant d'aller vers les pages d'alarmes, pour cela on sélectionne **Fichier** puis **Nouveau**, on sélectionne après le modèle de page **XpStyle ; Normale**

Ceci étant fait, on compile ensuite le projet et on l'exécute, la page illustrée sur la figure C-6 s'affiche alors. En allant sur Unity Pro, on peut mettre les tags relatifs aux capteurs de présence à 1 pour simuler une présence. Si on se trouve dans la plage d'heure où le programme de detection d'intrusion est activé, on entendra alors un Bip et le symbole d'alarme (flèche rouge) se mettra à clignoter alors. Dans le bas de la page s'affichera aussi le nom de l'alarme et l'heure de son déclenchement.

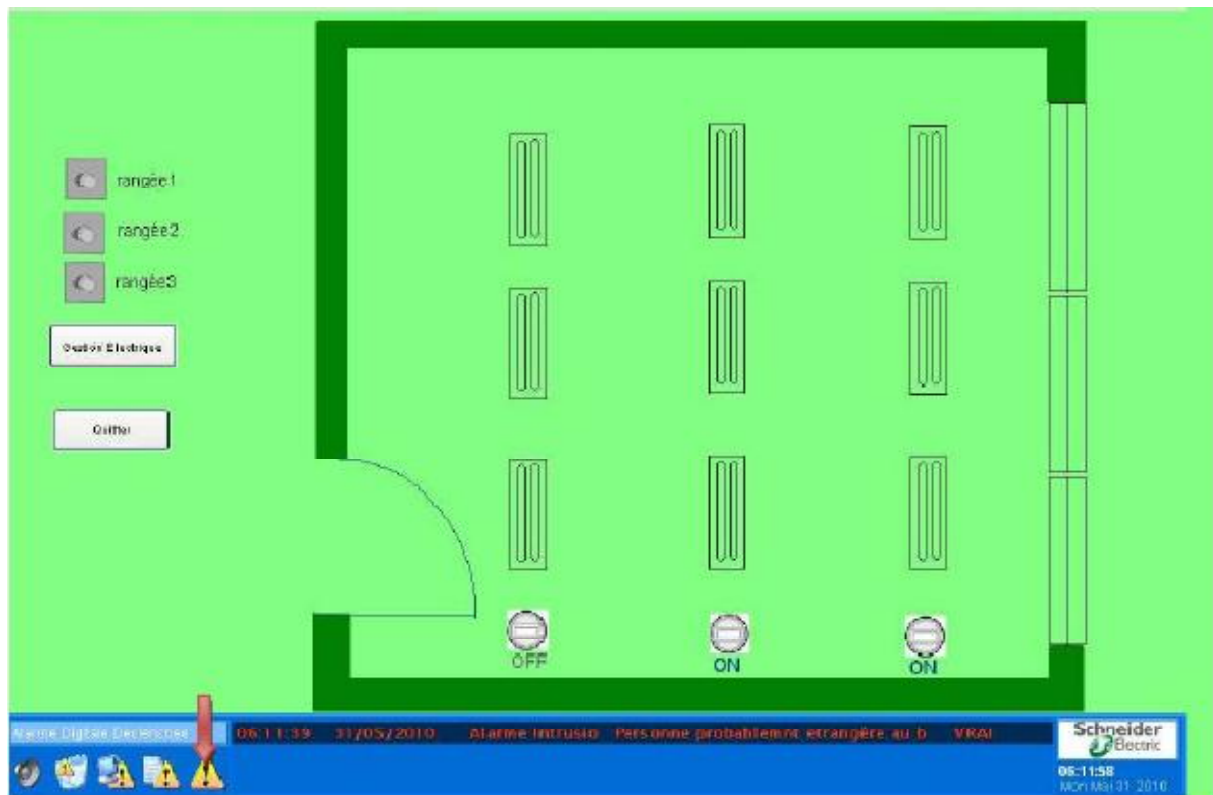


Figure B-6 : Exécution de l'application et affichage de l'alarme

En sélectionnant le symbole de la page d'alarme (flèche rouge), la page d'alarme s'affichera (Figure C-7), et on peut alors visualiser toutes les informations relatives à l'alarme. Noter que certaines informations (On Time,...etc.) ne seront affichées que si les périphériques d'historique et d'enregistrement ont été définis.

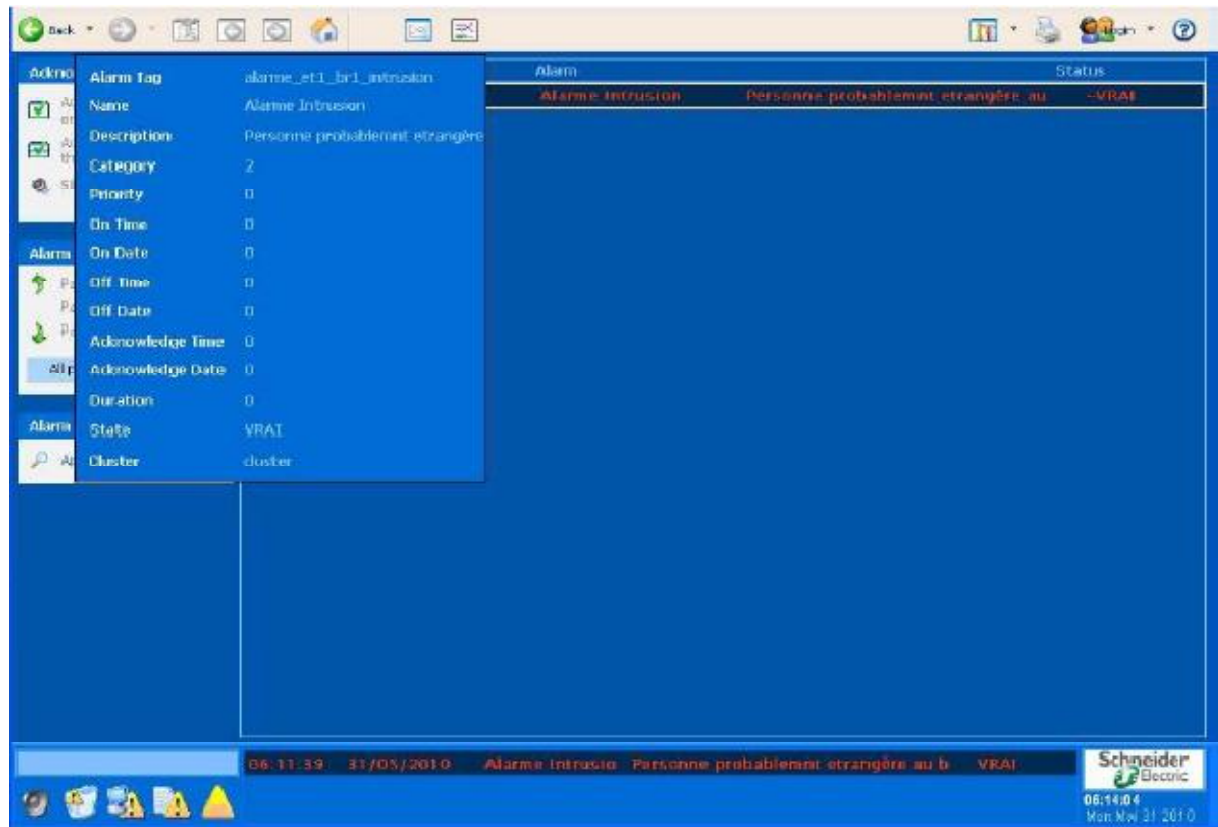
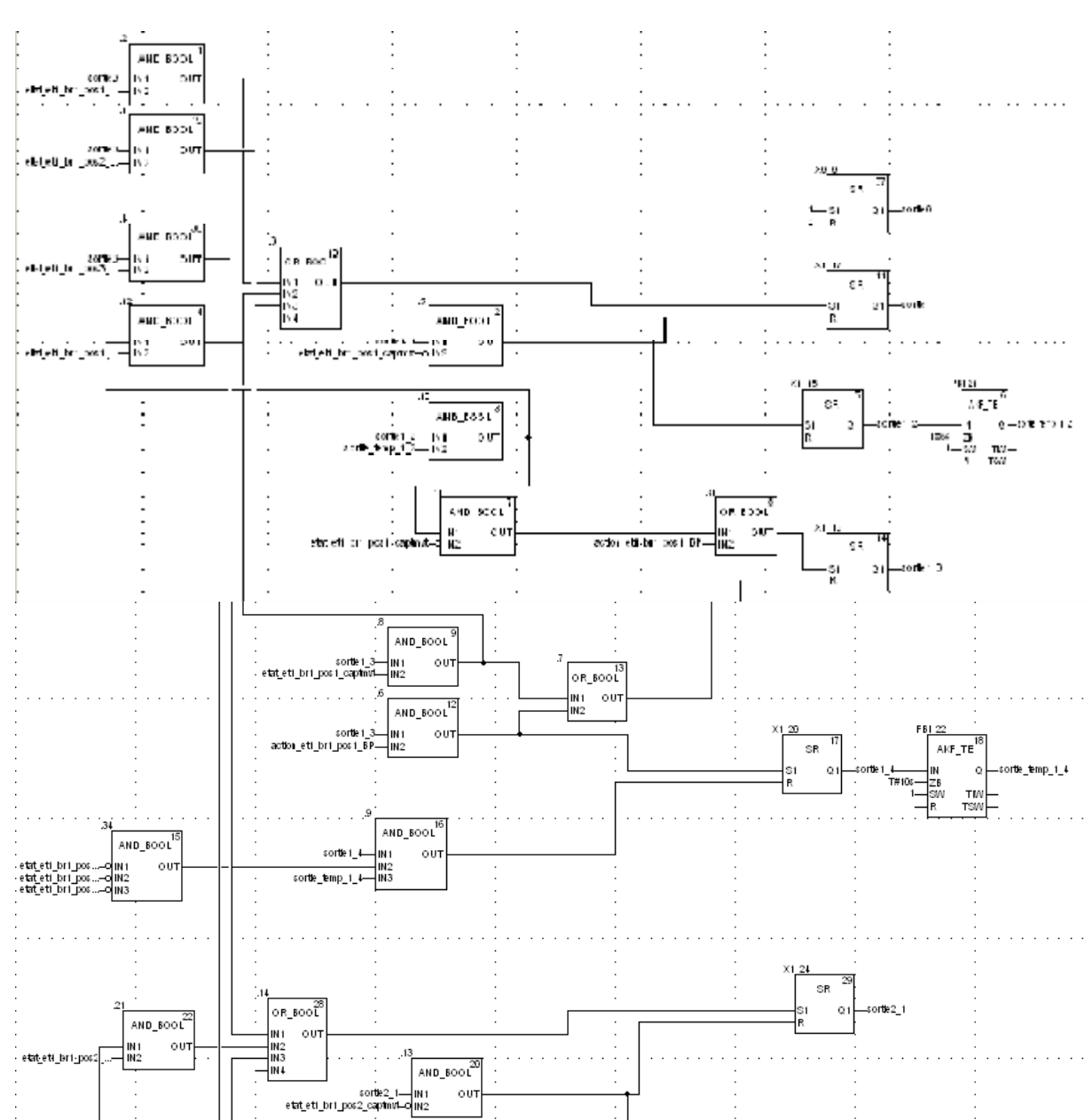
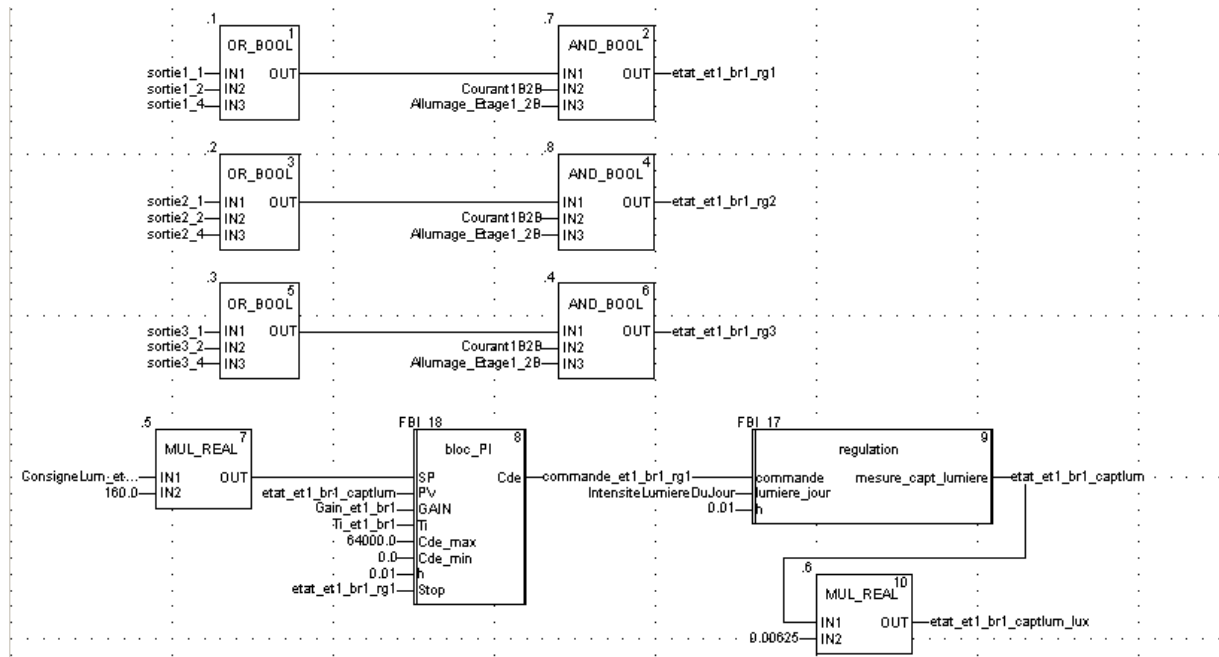


Figure B-7: Page d'alarme

1-1 Programme de l'exemple de l'allumage des luminaires dans un bureau



1



1.2 Programme de gestion et de simulation de l'installation électrique :

(*Actualisation des états des disjoncteurs*)

```

EtatGEG_Disjoncteur:=ActionGEG_Disjoncteur;
EtatCTA1_Disjoncteur:=ActionCTA1_Disjoncteur;
EtatCTA2_Disjoncteur:=ActionCTA2_Disjoncteur;
EtatCTA3_Disjoncteur:=ActionCTA3_Disjoncteur;
EtatCTA4_Disjoncteur:=ActionCTA4_Disjoncteur;
EtatOnduleur_Disjoncteur:=ActionOnduleur_Disjoncteur;
EtatASC1_Disjoncteur:=ActionASC1_Disjoncteur;
EtatASC2_Disjoncteur:=ActionASC2_Disjoncteur;
EtatASC3_Disjoncteur:=ActionASC3_Disjoncteur;
EtatPompe1_Disjoncteur:=ActionPompe1_Disjoncteur;
EtatTDA1_Disjoncteur:=ActionTDA1_Disjoncteur;
EtatTDA2_Disjoncteur:=ActionTDA2_Disjoncteur;
EtatTDA3_Disjoncteur:=ActionTDA3_Disjoncteur;
EtatTDA4_Disjoncteur:=ActionTDA4_Disjoncteur;
EtatTDA5_Disjoncteur:=ActionTDA5_Disjoncteur;
EtatTDA6_Disjoncteur:=ActionTDA6_Disjoncteur;
EtatTD_SSA_Disjoncteur:=ActionTD_SSA_Disjoncteur;
EtatTD_RDCA_Disjoncteur:=ActionTD_RDCA_Disjoncteur;
EtatTDB1_Disjoncteur:=ActionTDB1_Disjoncteur;
EtatTDB2_Disjoncteur:=ActionTDB2_Disjoncteur;
EtatTDB3_Disjoncteur:=ActionTDB3_Disjoncteur;
EtatTDB4_Disjoncteur:=ActionTDB4_Disjoncteur;
EtatTDB5_Disjoncteur:=ActionTDB5_Disjoncteur;
EtatTDB6_Disjoncteur:=ActionTDB6_Disjoncteur;
EtatTD_SSB_Disjoncteur:=ActionTD_SSB_Disjoncteur;
EtatCCOA1_Disjoncteur:=ActionCCOA1_Disjoncteur;
EtatCCOA2_Disjoncteur:=ActionCCOA2_Disjoncteur;

```


Annexe C

```
EtatCCOA3_Disjoncteur:=ActionCCOA3_Disjoncteur;
EtatCCOA4_Disjoncteur:=ActionCCOA4_Disjoncteur;
EtatCCOA5_Disjoncteur:=ActionCCOA5_Disjoncteur;
EtatCCOA6_Disjoncteur:=ActionCCOA6_Disjoncteur;
EtatCCOB1_Disjoncteur:=ActionCCOB1_Disjoncteur;
EtatCCOB2_Disjoncteur:=ActionCCOB2_Disjoncteur;
EtatCCOB3_Disjoncteur:=ActionCCOB3_Disjoncteur;
EtatCCOB4_Disjoncteur:=ActionCCOB4_Disjoncteur;
EtatCCOB5_Disjoncteur:=ActionCCOB5_Disjoncteur;
EtatCCOB6_Disjoncteur:=ActionCCOB6_Disjoncteur;
Etat1A_Disjoncteur:=Action1A_Disjoncteur;
Etat1B_Disjoncteur:=Action1B_Disjoncteur;
EtatGE1_Disjoncteur:=ActionGE1_Disjoncteur;
EtatGE4_Disjoncteur:=ActionGE4_Disjoncteur;
EtatGE5_Disjoncteur:=ActionGE5_Disjoncteur;
EtatGE6_Disjoncteur:=ActionGE6_Disjoncteur;
EtatCouplage1_Disjoncteur:=ActionCouplage1_Disjoncteur;
EtatExt_Disjoncteur:=ActionExt_Disjoncteur;
EtatTD_Ext_Disjoncteur:=ActionTD_Ext_Disjoncteur;
Etat2A_Disjoncteur:=Action2A_Disjoncteur;
Etat2B_Disjoncteur:=Action2B_Disjoncteur;
Etat3A_Disjoncteur:=Action3A_Disjoncteur;
Etat3B_Disjoncteur:=Action3B_Disjoncteur;
Etat4A_Disjoncteur:=Action4A_Disjoncteur;
Etat4B_Disjoncteur:=Action4B_Disjoncteur;
Etat5A_Disjoncteur:=Action5A_Disjoncteur;
Etat5B_Disjoncteur:=Action5B_Disjoncteur;
Etat6A_Disjoncteur:=Action6A_Disjoncteur;
Etat6B_Disjoncteur:=Action6B_Disjoncteur;
EtatRDCA_Disjoncteur:=ActionRDCA_Disjoncteur;
EtatSSA_Disjoncteur:=ActionSSA_Disjoncteur;
EtatSSB_Disjoncteur:=ActionSSB_Disjoncteur;
EtatTGBT1_Disjoncteur:=ActionTGBT1_Disjoncteur;
EtatTGCO1_Disjoncteur:=ActionTGCO1_Disjoncteur;
(*poste_livraison*)
CouleurAR1_Disjoncteur:=etat_reseau1;
CouleurAR2_Disjoncteur:=etat_reseau2;
EtatAR1_Disjoncteur:=etat_reseau1;
EtatAR2_Disjoncteur:=AND(etat_reseau2,NOT(etat_reseau1));
IF EtatCouplage1_Disjoncteur=0 THEN
CouleurTransfo1_Disjoncteur:=AND(EtatAR1_Disjoncteur,CouleurAR1_Disjoncteur);
CouleurTransfo3_Disjoncteur:=AND(EtatAR2_Disjoncteur,CouleurAR2_Disjoncteur);
ELSIF EtatCouplage1_Disjoncteur=1 THEN
    IF etat_reseau1=0 THEN
        CouleurTransfo3_Disjoncteur:=AND(EtatAR2_Disjoncteur,CouleurAR2_Disjoncteur);
        CouleurTransfo1_Disjoncteur:=CouleurTransfo3_Disjoncteur;
    ELSIF etat_reseau1=1 THEN
        CouleurTransfo1_Disjoncteur:=AND(EtatAR1_Disjoncteur,CouleurAR1_Disjoncteur);
        CouleurTransfo3_Disjoncteur:=CouleurTransfo1_Disjoncteur;
END_IF
```

Annexe C

```
END_IF;
CouleurTransfo2_Disjoncteur:=CouleurTransfo1_Disjoncteur;
CouleurTransfo4_Disjoncteur:=CouleurTransfo3_Disjoncteur;
CouleurCouplage1_Disjoncteur:=CouleurTransfo1_Disjoncteur;

(*TGBT1*)
CouleurTGBT1_Disjoncteur:=AND(EtatTransfo1_Disjoncteur,CouleurTransfo1_Disjoncteur);
Couleur1Inv1_Selecteur:=AND(EtatTransfo2_Disjoncteur,CouleurTransfo2_Disjoncteur);
Couleur2Inv1_Selecteur:=AND(EtatGE4_Disjoncteur,CouleurGE4_Disjoncteur);
EtatInv1_Selecteur:=OR(etat_reseau1,etat_reseau2);
intermediaire1:=AND(EtatTGBT1_Disjoncteur,CouleurTGBT1_Disjoncteur);
intermediaire2:=OR(Couleur1Inv1_Selecteur,Couleur2Inv1_Selecteur);
intermediaire3:=OR(etat_reseau1,etat_reseau2);
IF (intermediaire3=1)AND((intermediaire1=0)OR(intermediaire2=0)) THEN
    EtatCouplage2_Disjoncteur:=1;
ELSE
    EtatCouplage2_Disjoncteur:=0;
END_IF;
IF EtatCouplage2_Disjoncteur=1 THEN
    CouleurGEG_Disjoncteur:=OR(intermediaire1,intermediaire2);
    CouleurOnduleur_Disjoncteur:=CouleurGEG_Disjoncteur;
    ELSIF EtatCouplage2_Disjoncteur=0 THEN
        CouleurGEG_Disjoncteur:=intermediaire1;
        CouleurOnduleur_Disjoncteur:=intermediaire2;
END_IF;
CouleurCTA1_Disjoncteur:=CouleurGEG_Disjoncteur;
CouleurCTA2_Disjoncteur:=CouleurGEG_Disjoncteur;
CouleurCTA3_Disjoncteur:=CouleurGEG_Disjoncteur;
CouleurCTA4_Disjoncteur:=CouleurGEG_Disjoncteur;
CouleurASC1_Disjoncteur:=CouleurOnduleur_Disjoncteur;
CouleurASC2_Disjoncteur:=CouleurOnduleur_Disjoncteur;
CouleurASC3_Disjoncteur:=CouleurOnduleur_Disjoncteur;
CouleurPompe1_Disjoncteur:=CouleurOnduleur_Disjoncteur;
CouleurCouplage2_Disjoncteur:=CouleurOnduleur_Disjoncteur;

(*TGBT2*)
Couleur1Inv2_Selecteur:=AND(EtatTransfo3_Disjoncteur,CouleurTransfo3_Disjoncteur);
Couleur2Inv2_Selecteur:=AND(EtatGE5_Disjoncteur,CouleurGE5_Disjoncteur);
EtatInv2_Selecteur:=OR(etat_reseau1,etat_reseau2);
Couleur1Inv3_Selecteur:=AND(EtatTransfo4_Disjoncteur,CouleurTransfo4_Disjoncteur);
Couleur2Inv3_Selecteur:=AND(EtatGE6_Disjoncteur,CouleurGE6_Disjoncteur);
EtatInv3_Selecteur:=EtatInv2_Selecteur;
intermediaire1:=OR(Couleur1Inv2_Selecteur,Couleur2Inv2_Selecteur);
intermediaire2:=OR(Couleur1Inv3_Selecteur,Couleur2Inv3_Selecteur);
IF (intermediaire1=0)OR(intermediaire2=0) THEN
    EtatCouplage3_Disjoncteur:=1;
ELSE
    EtatCouplage3_Disjoncteur:=0;
END_IF;
CouleurTDA1_Disjoncteur:=OR(intermediaire1,intermediaire2);
CouleurTDA2_Disjoncteur:=CouleurTDA1_Disjoncteur;
CouleurTDA3_Disjoncteur:=CouleurTDA1_Disjoncteur;
```

Annexe C

CouleurTDA4_Disjoncteur:=CouleurTDA1_Disjoncteur;
CouleurTDA5_Disjoncteur:=CouleurTDA1_Disjoncteur;
CouleurTDA6_Disjoncteur:=CouleurTDA1_Disjoncteur;
CouleurTD_SSA_Disjoncteur:=CouleurTDA1_Disjoncteur;
CouleurTD_RDCA_Disjoncteur:=CouleurTDA1_Disjoncteur;
CouleurTDB1_Disjoncteur:=CouleurTDA1_Disjoncteur;
CouleurTDB2_Disjoncteur:=CouleurTDA1_Disjoncteur;
CouleurTDB3_Disjoncteur:=CouleurTDA1_Disjoncteur;
CouleurTDB4_Disjoncteur:=CouleurTDA1_Disjoncteur;
CouleurTDB5_Disjoncteur:=CouleurTDA1_Disjoncteur;
CouleurTDB6_Disjoncteur:=CouleurTDA1_Disjoncteur;
CouleurTD_SSB_Disjoncteur:=CouleurTDA1_Disjoncteur;
CouleurTD_Ext_Disjoncteur:=CouleurTDB1_Disjoncteur;
CouleurCouplage3_Disjoncteur:=CouleurTDA1_Disjoncteur;

(*TGGE*)

FBI_27 (IN:=NOT(OR(etat_reseau1,etat_reseau2)), ZB:=T#3s,
SW:=1,Q=>intermediaire1_tempo);
etat_GE1:=intermediaire1_tempo;
etat_GE2:=etat_GE1;
CouleurGE1_Disjoncteur:=OR(etat_GE1,etat_GE2);
CouleurGE4_Disjoncteur:=AND(EtatGE1_Disjoncteur,CouleurGE1_Disjoncteur);
CouleurGE5_Disjoncteur:=CouleurGE4_Disjoncteur;
CouleurGE6_Disjoncteur:=CouleurGE4_Disjoncteur;

(*TGCO*)

CouleurTGCO1_Disjoncteur:=Etat_Onduleur;
CouleurCCOA1_Disjoncteur:=AND(CouleurTGCO1_Disjoncteur,EtatTGCO1_Disjoncteur);
CouleurCCOA2_Disjoncteur:=CouleurCCOA1_Disjoncteur;
CouleurCCOA3_Disjoncteur:=CouleurCCOA1_Disjoncteur;
CouleurCCOA4_Disjoncteur:=CouleurCCOA1_Disjoncteur;
CouleurCCOA5_Disjoncteur:=CouleurCCOA1_Disjoncteur;
CouleurCCOA6_Disjoncteur:=CouleurCCOA1_Disjoncteur;
CouleurCCOB1_Disjoncteur:=CouleurCCOA1_Disjoncteur;
CouleurCCOB2_Disjoncteur:=CouleurCCOA1_Disjoncteur;
CouleurCCOB3_Disjoncteur:=CouleurCCOA1_Disjoncteur;
CouleurCCOB4_Disjoncteur:=CouleurCCOA1_Disjoncteur;
CouleurCCOB5_Disjoncteur:=CouleurCCOA1_Disjoncteur;
CouleurCCOB6_Disjoncteur:=CouleurCCOA1_Disjoncteur;

(*TGBT2_Suite*)

Couleur1A_Disjoncteur:=AND(CouleurTDA1_Disjoncteur,EtatTDA1_Disjoncteur);
Couleur2A_Disjoncteur:=AND(CouleurTDA2_Disjoncteur,EtatTDA2_Disjoncteur);
Couleur3A_Disjoncteur:=AND(CouleurTDA3_Disjoncteur,EtatTDA3_Disjoncteur);
Couleur4A_Disjoncteur:=AND(CouleurTDA4_Disjoncteur,EtatTDA4_Disjoncteur);
Couleur5A_Disjoncteur:=AND(CouleurTDA5_Disjoncteur,EtatTDA5_Disjoncteur);
Couleur6A_Disjoncteur:=AND(CouleurTDA6_Disjoncteur,EtatTDA6_Disjoncteur);
Couleur1B_Disjoncteur:=AND(CouleurTDB1_Disjoncteur,EtatTDB1_Disjoncteur);
Couleur2B_Disjoncteur:=AND(CouleurTDB2_Disjoncteur,EtatTDB2_Disjoncteur);
Couleur3B_Disjoncteur:=AND(CouleurTDB3_Disjoncteur,EtatTDB3_Disjoncteur);
Couleur4B_Disjoncteur:=AND(CouleurTDB4_Disjoncteur,EtatTDB4_Disjoncteur);
Couleur5B_Disjoncteur:=AND(CouleurTDB5_Disjoncteur,EtatTDB5_Disjoncteur);

Annexe C

```
Couleur6B_Disjoncteur:=AND(CouleurTDB6_Disjoncteur,EtatTDB6_Disjoncteur);
CouleurSSA_Disjoncteur:=AND(CouleurTD_SSA_Disjoncteur,EtatTD_SSA_Disjoncteur);
CouleurRDCA_Disjoncteur:=AND(CouleurTD_RDCA_Disjoncteur,EtatTD_RDCA_Disjoncteur);
CouleurSSB_Disjoncteur:=AND(CouleurTD_SSB_Disjoncteur,EtatTD_SSB_Disjoncteur);
CouleurExt_Disjoncteur:=AND(CouleurTD_Ext_Disjoncteur,EtatTD_Ext_Disjoncteur);
```

(*TD1_AB*)

```
Couleur1A1A_Disjoncteur:=AND(Couleur1A_Disjoncteur,Etat1A_Disjoncteur);
Couleur1A2A_Disjoncteur:=Couleur1A1A_Disjoncteur;
Couleur1A3A_Disjoncteur:=Couleur1A1A_Disjoncteur;
Couleur1A4A_Disjoncteur:=Couleur1A1A_Disjoncteur;
Couleur1A5A_Disjoncteur:=Couleur1A1A_Disjoncteur;
Couleur1A6A_Disjoncteur:=Couleur1A1A_Disjoncteur;
Couleur1A7A_Disjoncteur:=Couleur1A1A_Disjoncteur;
Couleur1A8A_Disjoncteur:=Couleur1A1A_Disjoncteur;
Couleur1A9A_Disjoncteur:=Couleur1A1A_Disjoncteur;
Couleur1A10A_Disjoncteur:=Couleur1A1A_Disjoncteur;
Couleur1B1B_Disjoncteur:=AND(Couleur1B_Disjoncteur,Etat1B_Disjoncteur);
Couleur1B2B_Disjoncteur:=Couleur1B1B_Disjoncteur;
Couleur1B3B_Disjoncteur:=Couleur1B1B_Disjoncteur;
Couleur1B4B_Disjoncteur:=Couleur1B1B_Disjoncteur;
Couleur1B5B_Disjoncteur:=Couleur1B1B_Disjoncteur;
Couleur1B6B_Disjoncteur:=Couleur1B1B_Disjoncteur;
Couleur1B7B_Disjoncteur:=Couleur1B1B_Disjoncteur;
Courant1A1A:=AND(Etat1A1A_Disjoncteur,Couleur1A1A_Disjoncteur);
Courant1A2A:=AND(Etat1A2A_Disjoncteur,Couleur1A2A_Disjoncteur);
Courant1A3A:=AND(Etat1A3A_Disjoncteur,Couleur1A3A_Disjoncteur);
Courant1A4A:=AND(Etat1A4A_Disjoncteur,Couleur1A4A_Disjoncteur);
Courant1A5A:=AND(Etat1A5A_Disjoncteur,Couleur1A5A_Disjoncteur);
Courant1A6A:=AND(Etat1A6A_Disjoncteur,Couleur1A6A_Disjoncteur);
Courant1A7A:=AND(Etat1A7A_Disjoncteur,Couleur1A7A_Disjoncteur);
Courant1A8A:=AND(Etat1A8A_Disjoncteur,Couleur1A8A_Disjoncteur);
Courant1A9A:=AND(Etat1A9A_Disjoncteur,Couleur1A9A_Disjoncteur);
Courant1A10A:=AND(Etat1A10A_Disjoncteur,Couleur1A10A_Disjoncteur);
Courant1B1B:=AND(Etat1B1B_Disjoncteur,Couleur1B1B_Disjoncteur);
Courant1B2B:=AND(Etat1B2B_Disjoncteur,Couleur1B2B_Disjoncteur);
Courant1B3B:=AND(Etat1B3B_Disjoncteur,Couleur1B3B_Disjoncteur);
Courant1B4B:=AND(Etat1B4B_Disjoncteur,Couleur1B4B_Disjoncteur);
Courant1B5B:=AND(Etat1B5B_Disjoncteur,Couleur1B5B_Disjoncteur);
Courant1B6B:=AND(Etat1B6B_Disjoncteur,Couleur1B6B_Disjoncteur);
Courant1B7B:=AND(Etat1B7B_Disjoncteur,Couleur1B7B_Disjoncteur);
Etat1A_Groupe_Disjoncteur:=AND(Etat1A1A_Disjoncteur,Etat1A2A_Disjoncteur,Etat1A3A_Disjoncteur,Etat1A4A_Disjoncteur,Etat1A5A_Disjoncteur,Etat1A6A_Disjoncteur,Etat1A7A_Disjoncteur,Etat1A8A_Disjoncteur,Etat1A9A_Disjoncteur,Etat1A10A_Disjoncteur);
Etat1B_Groupe_Disjoncteur:=AND(Etat1B1B_Disjoncteur,Etat1B2B_Disjoncteur,Etat1B3B_Disjoncteur,Etat1B4B_Disjoncteur,Etat1B5B_Disjoncteur,Etat1B6B_Disjoncteur,Etat1B7B_Disjoncteur);
```

(*eclairage exterieur*)

```
IF ModeCrep1=1 THEN
    Allumage_Zone1:=crepuscule;
END_IF;
IF ModeCrep2=1 THEN
```

Annexe C

```
Allumage_Zone2:=crepuscule;
END_IF;
IF ModeCrep3=1 THEN
    Allumage_Zone3:=crepuscule;
END_IF;
IF ModeCrep4=1 THEN
    Allumage_Zone4:=crepuscule;
END_IF;
IF ModeCrep5=1 THEN
    Allumage_Zone5:=crepuscule;
END_IF;
IF ModeCrep6=1 THEN
    Allumage_Zone6:=crepuscule;
END_IF;
IF ModeCrep7=1 THEN
    Allumage_Zone7:=crepuscule;
END_IF;
IF ModeCrep8=1 THEN
    Allumage_Zone8:=crepuscule;
END_IF;
```

(*TD_Ext*)

```
CouleurZone1_Disjoncteur:=AND(CouleurExt_Disjoncteur,EtatExt_Disjoncteur);
CouleurZone2_Disjoncteur:=CouleurZone1_Disjoncteur;
CouleurZone3_Disjoncteur:=CouleurZone1_Disjoncteur;
CouleurZone4_Disjoncteur:=CouleurZone1_Disjoncteur;
CouleurZone5_Disjoncteur:=CouleurZone1_Disjoncteur;
CouleurZone6_Disjoncteur:=CouleurZone1_Disjoncteur;
CouleurZone7_Disjoncteur:=CouleurZone1_Disjoncteur;
CouleurZone8_Disjoncteur:=CouleurZone1_Disjoncteur;
CourantZone1:=AND(EtatZone1_Disjoncteur,CouleurZone1_Disjoncteur);
CourantZone2:=AND(EtatZone2_Disjoncteur,CouleurZone2_Disjoncteur);
CourantZone3:=AND(EtatZone3_Disjoncteur,CouleurZone3_Disjoncteur);
CourantZone4:=AND(EtatZone4_Disjoncteur,CouleurZone4_Disjoncteur);
CourantZone5:=AND(EtatZone5_Disjoncteur,CouleurZone5_Disjoncteur);
CourantZone6:=AND(EtatZone6_Disjoncteur,CouleurZone6_Disjoncteur);
CourantZone7:=AND(EtatZone7_Disjoncteur,CouleurZone7_Disjoncteur);
CourantZone8:=AND(EtatZone8_Disjoncteur,CouleurZone8_Disjoncteur);
EtatExt_Groupe_Disjoncteur:=AND(EtatZone1_Disjoncteur,EtatZone2_Disjoncteur,EtatZone3_Disjoncteur,EtatZone4_Disjoncteur,EtatZone5_Disjoncteur,EtatZone6_Disjoncteur,EtatZone7_Disjoncteur,EtatZone8_Disjoncteur);
```

(*TD2_AB*)

```
Couleur2A1A_Disjoncteur:=AND(Couleur2A_Disjoncteur,Etat2A_Disjoncteur);
Couleur2A2A_Disjoncteur:=Couleur2A1A_Disjoncteur;
Couleur2A3A_Disjoncteur:=Couleur2A1A_Disjoncteur;
Couleur2A4A_Disjoncteur:=Couleur2A1A_Disjoncteur;
Couleur2A5A_Disjoncteur:=Couleur2A1A_Disjoncteur;
Couleur2A6A_Disjoncteur:=Couleur2A1A_Disjoncteur;
Couleur2A7A_Disjoncteur:=Couleur2A1A_Disjoncteur;
Couleur2A8A_Disjoncteur:=Couleur2A1A_Disjoncteur;
Couleur2A9A_Disjoncteur:=Couleur2A1A_Disjoncteur;
```

Annexe C

```
Couleur2A10A_Disjoncteur:=Couleur2A1A_Disjoncteur;
Couleur2B1B_Disjoncteur:=AND(Couleur2B_Disjoncteur,Etat2B_Disjoncteur);
Couleur2B2B_Disjoncteur:=Couleur2B1B_Disjoncteur;
Couleur2B3B_Disjoncteur:=Couleur2B1B_Disjoncteur;
Couleur2B4B_Disjoncteur:=Couleur2B1B_Disjoncteur;
Couleur2B5B_Disjoncteur:=Couleur2B1B_Disjoncteur;
Couleur2B6B_Disjoncteur:=Couleur2B1B_Disjoncteur;
Couleur2B7B_Disjoncteur:=Couleur2B1B_Disjoncteur;
Courant2A1A:=AND(Etat2A1A_Disjoncteur,Couleur2A1A_Disjoncteur);
Courant2A2A:=AND(Etat2A2A_Disjoncteur,Couleur2A2A_Disjoncteur);
Courant2A3A:=AND(Etat2A3A_Disjoncteur,Couleur2A3A_Disjoncteur);
Courant2A4A:=AND(Etat2A4A_Disjoncteur,Couleur2A4A_Disjoncteur);
Courant2A5A:=AND(Etat2A5A_Disjoncteur,Couleur2A5A_Disjoncteur);
Courant2A6A:=AND(Etat2A6A_Disjoncteur,Couleur2A6A_Disjoncteur);
Courant2A7A:=AND(Etat2A7A_Disjoncteur,Couleur2A7A_Disjoncteur);
Courant2A8A:=AND(Etat2A8A_Disjoncteur,Couleur2A8A_Disjoncteur);
Courant2A9A:=AND(Etat2A9A_Disjoncteur,Couleur2A9A_Disjoncteur);
Courant2A10A:=AND(Etat2A10A_Disjoncteur,Couleur2A10A_Disjoncteur);
Courant2B1B:=AND(Etat2B1B_Disjoncteur,Couleur2B1B_Disjoncteur);
Courant2B2B:=AND(Etat2B2B_Disjoncteur,Couleur2B2B_Disjoncteur);
Courant2B3B:=AND(Etat2B3B_Disjoncteur,Couleur2B3B_Disjoncteur);
Courant2B4B:=AND(Etat2B4B_Disjoncteur,Couleur2B4B_Disjoncteur);
Courant2B5B:=AND(Etat2B5B_Disjoncteur,Couleur2B5B_Disjoncteur);
Courant2B6B:=AND(Etat2B6B_Disjoncteur,Couleur2B6B_Disjoncteur);
Courant2B7B:=AND(Etat2B7B_Disjoncteur,Couleur2B7B_Disjoncteur);
Etat2A_Groupe_Disjoncteur:=AND(Etat2A1A_Disjoncteur,Etat2A2A_Disjoncteur,Etat2A3A_Disjoncteur,Etat2A4A_Disjoncteur,Etat2A5A_Disjoncteur,Etat2A6A_Disjoncteur,Etat2A7A_Disjoncteur,Etat2A8A_Disjoncteur,Etat2A9A_Disjoncteur,Etat2A10A_Disjoncteur);
Etat2B_Groupe_Disjoncteur:=AND(Etat2B1B_Disjoncteur,Etat2B2B_Disjoncteur,Etat2B3B_Disjoncteur,Etat2B4B_Disjoncteur,Etat2B5B_Disjoncteur,Etat2B6B_Disjoncteur,Etat2B7B_Disjoncteur);
(*TD3_AB*)
Couleur3A1A_Disjoncteur:=AND(Couleur3A_Disjoncteur,Etat3A_Disjoncteur);
Couleur3A2A_Disjoncteur:=Couleur3A1A_Disjoncteur;
Couleur3A3A_Disjoncteur:=Couleur3A1A_Disjoncteur;
Couleur3A4A_Disjoncteur:=Couleur3A1A_Disjoncteur;
Couleur3A5A_Disjoncteur:=Couleur3A1A_Disjoncteur;
Couleur3A6A_Disjoncteur:=Couleur3A1A_Disjoncteur;
Couleur3A7A_Disjoncteur:=Couleur3A1A_Disjoncteur;
Couleur3A8A_Disjoncteur:=Couleur3A1A_Disjoncteur;
Couleur3A9A_Disjoncteur:=Couleur3A1A_Disjoncteur;
Couleur3A10A_Disjoncteur:=Couleur3A1A_Disjoncteur;
Couleur3B1B_Disjoncteur:=AND(Couleur3B_Disjoncteur,Etat3B_Disjoncteur);
Couleur3B2B_Disjoncteur:=Couleur3B1B_Disjoncteur;
Couleur3B3B_Disjoncteur:=Couleur3B1B_Disjoncteur;
Couleur3B4B_Disjoncteur:=Couleur3B1B_Disjoncteur;
Couleur3B5B_Disjoncteur:=Couleur3B1B_Disjoncteur;
Couleur3B6B_Disjoncteur:=Couleur3B1B_Disjoncteur;
Couleur3B7B_Disjoncteur:=Couleur3B1B_Disjoncteur;
Courant3A1A:=AND(Etat3A1A_Disjoncteur,Couleur3A1A_Disjoncteur);
Courant3A2A:=AND(Etat3A2A_Disjoncteur,Couleur3A2A_Disjoncteur);
Courant3A3A:=AND(Etat3A3A_Disjoncteur,Couleur3A3A_Disjoncteur);
Courant3A4A:=AND(Etat3A4A_Disjoncteur,Couleur3A4A_Disjoncteur);
```


Annexe C

Courant3A5A:=AND(Etat3A5A_Disjoncteur,Couleur3A5A_Disjoncteur);
Courant3A6A:=AND(Etat3A6A_Disjoncteur,Couleur3A6A_Disjoncteur);
Courant3A7A:=AND(Etat3A7A_Disjoncteur,Couleur3A7A_Disjoncteur);
Courant3A8A:=AND(Etat3A8A_Disjoncteur,Couleur3A8A_Disjoncteur);
Courant3A9A:=AND(Etat3A9A_Disjoncteur,Couleur3A9A_Disjoncteur);
Courant3A10A:=AND(Etat3A10A_Disjoncteur,Couleur3A10A_Disjoncteur);
Courant3B1B:=AND(Etat3B1B_Disjoncteur,Couleur3B1B_Disjoncteur);
Courant3B2B:=AND(Etat3B2B_Disjoncteur,Couleur3B2B_Disjoncteur);
Courant3B3B:=AND(Etat3B3B_Disjoncteur,Couleur3B3B_Disjoncteur);
Courant3B4B:=AND(Etat3B4B_Disjoncteur,Couleur3B4B_Disjoncteur);
Courant3B5B:=AND(Etat3B5B_Disjoncteur,Couleur3B5B_Disjoncteur);
Courant3B6B:=AND(Etat3B6B_Disjoncteur,Couleur3B6B_Disjoncteur);
Courant3B7B:=AND(Etat3B7B_Disjoncteur,Couleur3B7B_Disjoncteur);
Etat3A_Groupe_Disjoncteur:=AND(Etat3A1A_Disjoncteur,Etat3A2A_Disjoncteur,Etat3A3A_Disjoncteur,Etat3A4A_Disjoncteur,Etat3A5A_Disjoncteur,Etat3A6A_Disjoncteur,Etat3A7A_Disjoncteur,Etat3A8A_Disjoncteur,Etat3A9A_Disjoncteur,Etat3A10A_Disjoncteur);
Etat3B_Groupe_Disjoncteur:=AND(Etat3B1B_Disjoncteur,Etat3B2B_Disjoncteur,Etat3B3B_Disjoncteur,Etat3B4B_Disjoncteur,Etat3B5B_Disjoncteur,Etat3B6B_Disjoncteur,Etat3B7B_Disjoncteur);

(*TD4_AB*)

Couleur4A1A_Disjoncteur:=AND(Couleur4A_Disjoncteur,Etat4A_Disjoncteur);
Couleur4A2A_Disjoncteur:=Couleur4A1A_Disjoncteur;
Couleur4A3A_Disjoncteur:=Couleur4A1A_Disjoncteur;
Couleur4A4A_Disjoncteur:=Couleur4A1A_Disjoncteur;
Couleur4A5A_Disjoncteur:=Couleur4A1A_Disjoncteur;
Couleur4A6A_Disjoncteur:=Couleur4A1A_Disjoncteur;
Couleur4A7A_Disjoncteur:=Couleur4A1A_Disjoncteur;
Couleur4A8A_Disjoncteur:=Couleur4A1A_Disjoncteur;
Couleur4A9A_Disjoncteur:=Couleur4A1A_Disjoncteur;
Couleur4A10A_Disjoncteur:=Couleur4A1A_Disjoncteur;
Couleur4B1B_Disjoncteur:=AND(Couleur4B_Disjoncteur,Etat4B_Disjoncteur);
Couleur4B2B_Disjoncteur:=Couleur4B1B_Disjoncteur;
Couleur4B3B_Disjoncteur:=Couleur4B1B_Disjoncteur;
Couleur4B4B_Disjoncteur:=Couleur4B1B_Disjoncteur;
Couleur4B5B_Disjoncteur:=Couleur4B1B_Disjoncteur;
Couleur4B6B_Disjoncteur:=Couleur4B1B_Disjoncteur;
Couleur4B7B_Disjoncteur:=Couleur4B1B_Disjoncteur;
Courant4A1A:=AND(Etat4A1A_Disjoncteur,Couleur4A1A_Disjoncteur);
Courant4A2A:=AND(Etat4A2A_Disjoncteur,Couleur4A2A_Disjoncteur);
Courant4A3A:=AND(Etat4A3A_Disjoncteur,Couleur4A3A_Disjoncteur);
Courant4A4A:=AND(Etat4A4A_Disjoncteur,Couleur4A4A_Disjoncteur);
Courant4A5A:=AND(Etat4A5A_Disjoncteur,Couleur4A5A_Disjoncteur);
Courant4A6A:=AND(Etat4A6A_Disjoncteur,Couleur4A6A_Disjoncteur);
Courant4A7A:=AND(Etat4A7A_Disjoncteur,Couleur4A7A_Disjoncteur);
Courant4A8A:=AND(Etat4A8A_Disjoncteur,Couleur4A8A_Disjoncteur);
Courant4A9A:=AND(Etat4A9A_Disjoncteur,Couleur4A9A_Disjoncteur);
Courant4A10A:=AND(Etat4A10A_Disjoncteur,Couleur4A10A_Disjoncteur);
Courant4B1B:=AND(Etat4B1B_Disjoncteur,Couleur4B1B_Disjoncteur);
Courant4B2B:=AND(Etat4B2B_Disjoncteur,Couleur4B2B_Disjoncteur);
Courant4B3B:=AND(Etat4B3B_Disjoncteur,Couleur4B3B_Disjoncteur);
Courant4B4B:=AND(Etat4B4B_Disjoncteur,Couleur4B4B_Disjoncteur);
Courant4B5B:=AND(Etat4B5B_Disjoncteur,Couleur4B5B_Disjoncteur);

Annexe C

Courant4B6B:=AND(Etat4B6B_Disjoncteur,Couleur4B6B_Disjoncteur);
Courant4B7B:=AND(Etat4B7B_Disjoncteur,Couleur4B7B_Disjoncteur);
Etat4A_Groupe_Disjoncteur:=AND(Etat4A1A_Disjoncteur,Etat4A2A_Disjoncteur,Etat4A3A_Disjoncteur,Etat4A4A_Disjoncteur,Etat4A5A_Disjoncteur,Etat4A6A_Disjoncteur,Etat4A7A_Disjoncteur,Etat4A8A_Disjoncteur,Etat4A9A_Disjoncteur,Etat4A10A_Disjoncteur);
Etat4B_Groupe_Disjoncteur:=AND(Etat4B1B_Disjoncteur,Etat4B2B_Disjoncteur,Etat4B3B_Disjoncteur,Etat4B4B_Disjoncteur,Etat4B5B_Disjoncteur,Etat4B6B_Disjoncteur,Etat4B7B_Disjoncteur);

(*TD5_AB*)

Couleur5A1A_Disjoncteur:=AND(Couleur5A_Disjoncteur,Etat5A_Disjoncteur);
Couleur5A2A_Disjoncteur:=Couleur5A1A_Disjoncteur;
Couleur5A3A_Disjoncteur:=Couleur5A1A_Disjoncteur;
Couleur5A4A_Disjoncteur:=Couleur5A1A_Disjoncteur;
Couleur5A5A_Disjoncteur:=Couleur5A1A_Disjoncteur;
Couleur5A6A_Disjoncteur:=Couleur5A1A_Disjoncteur;
Couleur5A7A_Disjoncteur:=Couleur5A1A_Disjoncteur;
Couleur5A8A_Disjoncteur:=Couleur5A1A_Disjoncteur;
Couleur5A9A_Disjoncteur:=Couleur5A1A_Disjoncteur;
Couleur5A10A_Disjoncteur:=Couleur5A1A_Disjoncteur;
Couleur5B1B_Disjoncteur:=AND(Couleur5B_Disjoncteur,Etat5B_Disjoncteur);
Couleur5B2B_Disjoncteur:=Couleur5B1B_Disjoncteur;
Couleur5B3B_Disjoncteur:=Couleur5B1B_Disjoncteur;
Couleur5B4B_Disjoncteur:=Couleur5B1B_Disjoncteur;
Couleur5B5B_Disjoncteur:=Couleur5B1B_Disjoncteur;
Couleur5B6B_Disjoncteur:=Couleur5B1B_Disjoncteur;
Couleur5B7B_Disjoncteur:=Couleur5B1B_Disjoncteur;
Courant5A1A:=AND(Etat5A1A_Disjoncteur,Couleur5A1A_Disjoncteur);
Courant5A2A:=AND(Etat5A2A_Disjoncteur,Couleur5A2A_Disjoncteur);
Courant5A3A:=AND(Etat5A3A_Disjoncteur,Couleur5A3A_Disjoncteur);
Courant5A4A:=AND(Etat5A4A_Disjoncteur,Couleur5A4A_Disjoncteur);
Courant5A5A:=AND(Etat5A5A_Disjoncteur,Couleur5A5A_Disjoncteur);
Courant5A6A:=AND(Etat5A6A_Disjoncteur,Couleur5A6A_Disjoncteur);
Courant5A7A:=AND(Etat5A7A_Disjoncteur,Couleur5A7A_Disjoncteur);
Courant5A8A:=AND(Etat5A8A_Disjoncteur,Couleur5A8A_Disjoncteur);
Courant5A9A:=AND(Etat5A9A_Disjoncteur,Couleur5A9A_Disjoncteur);
Courant5A10A:=AND(Etat5A10A_Disjoncteur,Couleur5A10A_Disjoncteur);
Courant5B1B:=AND(Etat5B1B_Disjoncteur,Couleur5B1B_Disjoncteur);
Courant5B2B:=AND(Etat5B2B_Disjoncteur,Couleur5B2B_Disjoncteur);
Courant5B3B:=AND(Etat5B3B_Disjoncteur,Couleur5B3B_Disjoncteur);
Courant5B4B:=AND(Etat5B4B_Disjoncteur,Couleur5B4B_Disjoncteur);
Courant5B5B:=AND(Etat5B5B_Disjoncteur,Couleur5B5B_Disjoncteur);
Courant5B6B:=AND(Etat5B6B_Disjoncteur,Couleur5B6B_Disjoncteur);
Courant5B7B:=AND(Etat5B7B_Disjoncteur,Couleur5B7B_Disjoncteur);
Etat5A_Groupe_Disjoncteur:=AND(Etat5A1A_Disjoncteur,Etat5A2A_Disjoncteur,Etat5A3A_Disjoncteur,Etat5A4A_Disjoncteur,Etat5A5A_Disjoncteur,Etat5A6A_Disjoncteur,Etat5A7A_Disjoncteur,Etat5A8A_Disjoncteur,Etat5A9A_Disjoncteur,Etat5A10A_Disjoncteur);
Etat5B_Groupe_Disjoncteur:=AND(Etat5B1B_Disjoncteur,Etat5B2B_Disjoncteur,Etat5B3B_Disjoncteur,Etat5B4B_Disjoncteur,Etat5B5B_Disjoncteur,Etat5B6B_Disjoncteur,Etat5B7B_Disjoncteur);

(*TD6_AB*)

Couleur6A1A_Disjoncteur:=AND(Couleur6A_Disjoncteur,Etat6A_Disjoncteur);

Annexe C

```
Couleur6A2A_Disjoncteur:=Couleur6A1A_Disjoncteur;
Couleur6A3A_Disjoncteur:=Couleur6A1A_Disjoncteur;
Couleur6A4A_Disjoncteur:=Couleur6A1A_Disjoncteur;
Couleur6A5A_Disjoncteur:=Couleur6A1A_Disjoncteur;
Couleur6A6A_Disjoncteur:=Couleur6A1A_Disjoncteur;
Couleur6A7A_Disjoncteur:=Couleur6A1A_Disjoncteur;
Couleur6A8A_Disjoncteur:=Couleur6A1A_Disjoncteur;
Couleur6A9A_Disjoncteur:=Couleur6A1A_Disjoncteur;
Couleur6A10A_Disjoncteur:=Couleur6A1A_Disjoncteur;
Couleur6B1B_Disjoncteur:=AND(Couleur6B_Disjoncteur,Etat6B_Disjoncteur);
Couleur6B2B_Disjoncteur:=Couleur6B1B_Disjoncteur;
Couleur6B3B_Disjoncteur:=Couleur6B1B_Disjoncteur;
Couleur6B4B_Disjoncteur:=Couleur6B1B_Disjoncteur;
Couleur6B5B_Disjoncteur:=Couleur6B1B_Disjoncteur;
Couleur6B6B_Disjoncteur:=Couleur6B1B_Disjoncteur;
Couleur6B7B_Disjoncteur:=Couleur6B1B_Disjoncteur;
Courant6A1A:=AND(Etat6A1A_Disjoncteur,Couleur6A1A_Disjoncteur);
Courant6A2A:=AND(Etat6A2A_Disjoncteur,Couleur6A2A_Disjoncteur);
Courant6A3A:=AND(Etat6A3A_Disjoncteur,Couleur6A3A_Disjoncteur);
Courant6A4A:=AND(Etat6A4A_Disjoncteur,Couleur6A4A_Disjoncteur);
Courant6A5A:=AND(Etat6A5A_Disjoncteur,Couleur6A5A_Disjoncteur);
Courant6A6A:=AND(Etat6A6A_Disjoncteur,Couleur6A6A_Disjoncteur);
Courant6A7A:=AND(Etat6A7A_Disjoncteur,Couleur6A7A_Disjoncteur);
Courant6A8A:=AND(Etat6A8A_Disjoncteur,Couleur6A8A_Disjoncteur);
Courant6A9A:=AND(Etat6A9A_Disjoncteur,Couleur6A9A_Disjoncteur);
Courant6A10A:=AND(Etat6A10A_Disjoncteur,Couleur6A10A_Disjoncteur);
Courant6B1B:=AND(Etat6B1B_Disjoncteur,Couleur6B1B_Disjoncteur);
Courant6B2B:=AND(Etat6B2B_Disjoncteur,Couleur6B2B_Disjoncteur);
Courant6B3B:=AND(Etat6B3B_Disjoncteur,Couleur6B3B_Disjoncteur);
Courant6B4B:=AND(Etat6B4B_Disjoncteur,Couleur6B4B_Disjoncteur);
Courant6B5B:=AND(Etat6B5B_Disjoncteur,Couleur6B5B_Disjoncteur);
Courant6B6B:=AND(Etat6B6B_Disjoncteur,Couleur6B6B_Disjoncteur);
Courant6B7B:=AND(Etat6B7B_Disjoncteur,Couleur6B7B_Disjoncteur);
Etat6A_Groupe_Disjoncteur:=AND(Etat6A1A_Disjoncteur,Etat6A2A_Disjoncteur,Etat6A3A_Disjoncteur,Etat6A4A_Disjoncteur,Etat6A5A_Disjoncteur,Etat6A6A_Disjoncteur,Etat6A7A_Disjoncteur,Etat6A8A_Disjoncteur,Etat6A9A_Disjoncteur,Etat6A10A_Disjoncteur);
Etat6B_Groupe_Disjoncteur:=AND(Etat6B1B_Disjoncteur,Etat6B2B_Disjoncteur,Etat6B3B_Disjoncteur,Etat6B4B_Disjoncteur,Etat6B5B_Disjoncteur,Etat6B6B_Disjoncteur,Etat6B7B_Disjoncteur);
```

(*TDSS_AB*)

```
CouleurSSA1A_Disjoncteur:=AND(CouleurSSA_Disjoncteur,EtatSSA_Disjoncteur);
CouleurSSA2A_Disjoncteur:=CouleurSSA1A_Disjoncteur;
CouleurSSA3A_Disjoncteur:=CouleurSSA1A_Disjoncteur;
CouleurSSA4A_Disjoncteur:=CouleurSSA1A_Disjoncteur;
CouleurSSA5A_Disjoncteur:=CouleurSSA1A_Disjoncteur;
CouleurSSA6A_Disjoncteur:=CouleurSSA1A_Disjoncteur;
CouleurSSB1B_Disjoncteur:=AND(CouleurSSB_Disjoncteur,EtatSSB_Disjoncteur);
CouleurSSB2B_Disjoncteur:=CouleurSSB1B_Disjoncteur;
CouleurSSB3B_Disjoncteur:=CouleurSSB1B_Disjoncteur;
CouleurSSB4B_Disjoncteur:=CouleurSSB1B_Disjoncteur;
CourantSSA1A:=AND(EtatSSA1A_Disjoncteur,CouleurSSA1A_Disjoncteur);
CourantSSA2A:=AND(EtatSSA2A_Disjoncteur,CouleurSSA2A_Disjoncteur);
```

Annexe C

```
CourantSSA3A:=AND(EtatSSA3A_Disjoncteur,CouleurSSA3A_Disjoncteur);
CourantSSA4A:=AND(EtatSSA4A_Disjoncteur,CouleurSSA4A_Disjoncteur);
CourantSSA5A:=AND(EtatSSA5A_Disjoncteur,CouleurSSA5A_Disjoncteur);
CourantSSA6A:=AND(EtatSSA6A_Disjoncteur,CouleurSSA6A_Disjoncteur);
CourantSSB1B:=AND(EtatSSB1B_Disjoncteur,CouleurSSB1B_Disjoncteur);
CourantSSB2B:=AND(EtatSSB2B_Disjoncteur,CouleurSSB2B_Disjoncteur);
CourantSSB3B:=AND(EtatSSB3B_Disjoncteur,CouleurSSB3B_Disjoncteur);
CourantSSB4B:=AND(EtatSSB4B_Disjoncteur,CouleurSSB4B_Disjoncteur);
EtatSSA_Groupe_Disjoncteur:=AND(EtatSSA1A_Disjoncteur,EtatSSA2A_Disjoncteur,EtatSSA3A_
Disjoncteu
r,EtatSSA4A_Disjoncteur,EtatSSA5A_Disjoncteur,EtatSSA6A_Disjoncteur);
EtatSSB_Groupe_Disjoncteur:=AND(EtatSSB1B_Disjoncteur,EtatSSB2B_Disjoncteur,EtatSSB3B_D
isjoncteur,
EtatSSB4B_Disjoncteur);
```

(*TD_RDC_A*)

```
CouleurRDCA1A_Disjoncteur:=AND(CouleurRDCA_Disjoncteur,EtatRDCA_Disjoncteur);
CouleurRDCA2A_Disjoncteur:=CouleurRDCA1A_Disjoncteur;
CouleurRDCA3A_Disjoncteur:=CouleurRDCA1A_Disjoncteur;
CouleurRDCA4A_Disjoncteur:=CouleurRDCA1A_Disjoncteur;
CouleurRDCA5A_Disjoncteur:=CouleurRDCA1A_Disjoncteur;
CouleurRDCA6A_Disjoncteur:=CouleurRDCA1A_Disjoncteur;
CouleurRDCA7A_Disjoncteur:=CouleurRDCA1A_Disjoncteur;
CouleurRDCA8A_Disjoncteur:=CouleurRDCA1A_Disjoncteur;
CouleurRDCA9A_Disjoncteur:=CouleurRDCA1A_Disjoncteur;
CouleurRDCA10A_Disjoncteur:=CouleurRDCA1A_Disjoncteur;
CouleurRDCA11A_Disjoncteur:=CouleurRDCA1A_Disjoncteur;
CouleurRDCA12A_Disjoncteur:=CouleurRDCA1A_Disjoncteur;
CouleurRDCA13A_Disjoncteur:=CouleurRDCA1A_Disjoncteur;
CouleurRDCA14A_Disjoncteur:=CouleurRDCA1A_Disjoncteur;
CourantRDCA1A:=AND(EtatRDCA1A_Disjoncteur,CouleurRDCA1A_Disjoncteur);
CourantRDCA2A:=AND(EtatRDCA2A_Disjoncteur,CouleurRDCA2A_Disjoncteur);
CourantRDCA3A:=AND(EtatRDCA3A_Disjoncteur,CouleurRDCA3A_Disjoncteur);
CourantRDCA4A:=AND(EtatRDCA4A_Disjoncteur,CouleurRDCA4A_Disjoncteur);
CourantRDCA5A:=AND(EtatRDCA5A_Disjoncteur,CouleurRDCA5A_Disjoncteur);
CourantRDCA6A:=AND(EtatRDCA6A_Disjoncteur,CouleurRDCA6A_Disjoncteur);
CourantRDCA7A:=AND(EtatRDCA7A_Disjoncteur,CouleurRDCA7A_Disjoncteur);
CourantRDCA8A:=AND(EtatRDCA8A_Disjoncteur,CouleurRDCA8A_Disjoncteur);
CourantRDCA9A:=AND(EtatRDCA9A_Disjoncteur,CouleurRDCA9A_Disjoncteur);
CourantRDCA10A:=AND(EtatRDCA10A_Disjoncteur,CouleurRDCA10A_Disjoncteur);
CourantRDCA11A:=AND(EtatRDCA11A_Disjoncteur,CouleurRDCA11A_Disjoncteur);
CourantRDCA12A:=AND(EtatRDCA12A_Disjoncteur,CouleurRDCA12A_Disjoncteur);
CourantRDCA13A:=AND(EtatRDCA13A_Disjoncteur,CouleurRDCA13A_Disjoncteur);
CourantRDCA14A:=AND(EtatRDCA14A_Disjoncteur,CouleurRDCA14A_Disjoncteur);
EtatRDCA_Groupe_Disjoncteur:=AND(EtatRDCA1A_Disjoncteur,EtatRDCA2A_Disjoncteur,EtatR
DCA3A_D
isjoncteur,EtatRDCA4A_Disjoncteur,EtatRDCA5A_Disjoncteur,EtatRDCA6A_Disjoncteur,EtatRDC
A7A_Disj
oncteur,EtatRDCA8A_Disjoncteur,EtatRDCA9A_Disjoncteur,EtatRDCA10A_Disjoncteur,EtatRDCA
10A_Disj
oncteur,
EtatRDCA11A_Disjoncteur,EtatRDCA12A_Disjoncteur,EtatRDCA13A_Disjoncteur,EtatRDCA14A_
Disjoncte
ur);
```

1-2 Partie programmée en FBD (partie simulation)

