الجمهورية الجزائرية الديمقراطية الشعبية

**MINISTERE DE L'ENSEIGNEMENT SUPERIEUR**
**ET DE LA RECHERCHE SCIENTIFIQUE**

**UNIVERSITE YAHIA FARES DE MEDEA**

**FACULTE DETECHNOLOGIE**

**Département du Génie Électrique**

# Mémoire de Fin d'Études

**Présenté pour l'obtention du Diplôme**
**de Master en Électronique**

*Spécialité*: Instrumentation

## THÈME

# Réseau de communication à base d'Ethernet pour La supervision et le contrôle appliqués à la domotique

**Proposé et dirigé par,**

Dr. BOUHEDDA Mounir

**Préparé et présenté par,**

Mr. Abaoub Rachid

Mr.Ben allal Mohamed Amine.

*Thèse préparée au sein du Laboratoire des Systèmes Électroniques Avancés (LSEA) de l'Université de Médéa*

AnnéeUniversitaire2017-2018

# *Dédicace*

***Ma Mère, Mon Père***
***Affable, honorable, aimable : vous représentes pour moi***
***Le symbole de la bonté par excellence, la source de tendresse***
***et l ’exemple du***
***dévouement qui n ’a pas cesséé de m ’encourager et de prier***
***pour moi.***

***Soyez sûrs que je continuerai mon chemin.***
***Je vous dédie ce travail en témoignage de mon profond amour.***
***Puisse Dieu, le tout puissant, vous préserver et vous accorder santé,***
***longue vie et bonheur.***
***A mes sœurs***
***A mes frère***

***En t témoignage de l ’attachement, de l ’amour et de***
***L ’affection que je porte pour vous.***
***A tous les membres de ma famille, petits et grands***
***A tous mes amis de proches :***

***Veuillez trouver dans ce modeste travail l ’expression de mon***
***Affection***

**BenAllalMohamed Amine.**

# *Dédicace*

***Ma Mère, Mon Père***
***Affable, honorable, aimable : vous représentes pour moi***
***Le symbole de la bonté par excellence, la source de tendresse et l ’exemple du***
***dévouement qui n ’a pas cessé é de m ’encourager et de prier pour moi.***

***Soyez sûrs que je continuerai mon chemin.***
***Je vous dédie ce travail en témoignage de mon profond amour.***
***Puisse Dieu, le tout puissant, vous préserver et vous accorder santé, longue vie et bonheur.***
***A mes sœurs***
***Ames frère***

***En t témoignage de l ’attachement, de l ’amour et de***
***L ’affection que je porte pour vous.***
***A tous les membres de ma famille, petits et grands***
***A tous mes amis de proches :***

***Veuillez trouver dans ce modeste travail l ’expression de mon Affection***

**AbaoubRachid .**

# Sommaire

Introduction générale ,1.

Chapitre I : les Bus de communication.

I.1 Introduction,2.
I.1 Définition des bus de terrain,2.
I.2 finalités des bus de terrain ,3.
I.3Les différentes topologies des bus de terrain,3.
I.3.1 Topologie en étoile,3.
I.3.2 Topologie en anneau, 4.
I.3.3 Topologie en ligne bus, 4.
I.3.4 Topologie en arbre, 5.
I.4 Techniques de transmission,5.
I.4.1 Cuivre, 5.
I.4.2 Lumière,6.
I.4.3 Radio ,6.
I.5 Types de transmission, 6.
I.6 Les bus de terrain ,7.
I.6.1 Bus CAN ,7.
I.6.1.1 presentation du bus CAN ,7.
I.6.1.2 Le bus CAN (Controller Area Network) , 7.
I.6.2 MODBUS, 8.
I.6.2.1 Principe général ,8.
I.6.2.2 Trame Modus,8.
I.6.3 BUS KNX, 9.
I.6.3.1 KNX, 9.
I.6.3.2 Fonctionnement, 9.
I.6.4 Bus AS-I, 10.
I.6.4.1 Le bus ASI offre de nombreux avantages,10.
I.6.4.2 Le bus AS-I permet donc d'éviter de nombreux inconvénients comme,10.
I.6.5 Profibus,11
I.6.5.1 Le bus PROFIBUS-DP,11.
I.6.5.2 Les protocoles de transmission ,11.
I.7 Réseau Ethernet ,12.
I.7.1 Introduction, 12.
I.7.2 Les différentes normes, 12.
I.7.3 Caractéristiques principales,12.
I.7.4 Trame 802.3 (Frame 802.3),13.
I.7.5 Trame Ethernet_II,14.
I.7.6 Ethernet et le modèle OSI,14.
I.7.7 Couche Physique (niveau 1 du modèle OSI),15.
I.7.8 Concentrateur (hub) et commutateur (switch),15.
I.7.9 La couche Liaison (sous-couches MAC et LLC),15.
I.7.9.1Méthode d'accès CSMA/CD,16.
I.7.9.2 Round Trip Delay,16.
I.7.10Virtual LAN, 17.
I.7.11Exemples de réseaux,17.

I.8 Conclusion,17

CHAPITRE II :développement de système autour d'Arduino.

II.1 Introduction,18.
II.2 Historique, 18.
II.3 Définition,18.
II.4 But,18.
II.5 Utilisation,18.
II.6 Composition ,18.
II.7 Choix et description de la carte Arduino,18.
II.7.1 Choix, 18.
II.7.2 Description de la carte ,19.
II.8 Le langage 21.
II.9 Types des cartes ,20.
II.9.1 Carte Uno, 20.
II.9.2 La carte arduino Méga,20..
II.9.3 Arduino Nano,21..
II.9.4 ArduinoEsplora,21.
II.9.5 La carte seeduino,21.
II.9.6 La carte Arduino pro (par Sparkfun),22.
II.9.7 Carte arduinoBlackwidow,22.
II.9.8 Carte Rainbowduino,22.
II.10 Quelques modules d'Arduino, 23.
II.10.1 Le module Arduino Bluetooth, 23.
II.10.2 Le module shieldArduino Wifi, 24.
II.10.3 Le module shieldArduino Ethernet, 24.
II.11 Les capteurs,24.
II.12 Les drivers, 25.
II.13 Logiciel Arduino, 25.
II.13.1 Principe général d'utilisation ,27.
II.13.2 Description de la barre des boutons ,27.
II.13.3 Transfert des programmes vers la carte Arduino,28.
II.13.4 Le Moniteur Série,30.
II.14 Conclusion ,31.

ChapitreIII : Réalisation pratique

III.1 Introduction,32.
III.2 Définition du cahier de charge,32.
III.3Description de la salle,33.
III.4 Application de supervision et de contrôle,35.
III.5 ContrôleparArduino,36.
III.6 contrôle par application (java / Arduino),37.

III.6.1 Serveur ,37.
III.6.2 Client ,37.
III.6.3 Les sockets,37.
III.6.4 Shield Ethernet ,38.

III.6.4.1 Constitution du ShieldArduino Ethernet,38.
III.6.4.2 La librairie Ethernet, pour se connecter à internet ou réaliser des réseaux filaires Ethernet avec Arduino,38.
III.6.4.3 INSTRUCTIONS DE LA CLASSE ETHERNETSERVER,39.
III.6.4.4 INSTRUCTIONS DE LA CLASSE ETHERNETCLIENT,39.
III.6.5 Principe d'utilisation,40.
III.7 La communication java /Arduino ,40.
III.8Quelques tests expérimentaux ,41.
III.9 Conclusion ,45.
Conclusion générale ,46.
Bibliographie ,47.

# ملخص

الغرضمنهذاالمشروعهوالتحكموالمراقبة في غرفة عبراتصالالإيثرنت. ولهذهالغاية،تم انجازتطبيقجافا للمراقبةو التحكم فيالعديدمنالأجهزةفيالغرفةباستخداموحدةتحكماردوينو. حيث اثبتت التجارب التي قمنا بها فعاليةالنظامالمدروس

**الكلمات المفتاحية :** ارد ويزو,ايثرنات، اتصال، تحكم، مراقبة

# Résumé

L'objectif de ce projet consiste à faire le contrôle etla supervision à travers une communication Ethernet. À cet effet, une application sous Java est réalisée qui prendra en charge la supervision et contrôle de plusieurs dispositifs d'une salle en utilisant un microcontrôleur de type Arduino. Plusieurs tests sont réalisés ont montré l'efficacité du système réalisé.

**Mots-clés : Arduino ,Ethernet, Communication, contrôle, supervision**

# Abstract

The aim of this project is to control and supervise Ethernet connections. To that end, a Java application that monitors and controls many devices in a room is implemented using acontroller based on Arduino Microcontroller. Many realized tests have demonstrated the effectiveness of the developed system.

**Keywords:**Arduino , Ethernet, Communication, control, supervision

# Liste des figures

Figure I. 1: Topologie en étoile,3.
Figure I. 2: Topologie en anneau,4.
Figure I. 3: Topologie en ligne bus,4.
Figure I. 4: Topologie en arbre,5.
Figure I. 5 : Multiplex age temporal ,7.
Figure I. 6 : Architecture maitre/esclave ,8.
Figure I. 7 : Topologie de bus d'AS-i ,10.
Figure I. 8 : supports de transmission dans un profibus,11.
Figure I. 9 : La nature hybride de la gestion d'accès à PROFIBUS : configuration maître-esclaves et passage de jeton entre trois maîtres formant un anneau logique ,12.
Figure I. 10 :Sigle ,13.
Figure I. 11 : architecture de la trame 802.3 ,13.
Figure I. 12 :la trame Ethernet II ,14.
Figure I. 13 :architecture Ethernet et modèle OSI,14.
Figure I. 14 :Méthode d'accès,16.
Figure I. 15 :exemple de réseau Ethernet,17.

Figure II. 1: Description de la carte ArduinoUno,19.
Figure II. 2: Carte ArduinoUno,20.
Figure II. 3: Carte Arduino Méga,20.
Figure II. 4: Carte Arduino nano,21.
Figure II. 5: Carte ArduinoEsplora,21.
Figure II. 6: Carte seeeduino,21.
Figure II. 7: Carte Arduino pro,22**.**
Figure II. 8: Carte ArduinoBlackwidow , 22.
Figure II. 9: Carte Rainbowduino,22.
Figure II. 10: Carte teensy, 23.
Figure II. 11: Carte netduino,23.
Figure II. 12: Le module Arduino Bluetooth,23.
Figure II. 13: Le module shieldArduino Wifi,24.
Figure II. 14: Le module shieldArduino Ethernet,24.
Figure II. 15:les capteurs utilisée ,24.
Figure II. 16:les drivers, 25.
Figure II. 17 :l'interface de logiciel Arduino, 26.
Figure II. 18 :fenêtreterminal série,26.
Figure II. 19 :description de la barre des boutons ,28.
Figure II. 20 :la fenêtre éditeur28.
Figure II. 21 :fenêtre de transfert vers la carte Arduino,29.
Figure II. 22 :la fenêtre de transfert après le test ,30.
Figure II. 23 :le fenêtre de l'IDE Arduino,30.
Figure II. 24 :la fenêtre terminal,31
Figure II. 25 :la fenêtre terminal,31

# Liste des figures

Figure III. 1schéma de communication ,32.
Figure III. 2 Dispositifs existants dans la salle ,33.
Figure III. 3 la maquette ,34.
Figure III. 4 l'interface de connexion ,35.
Figure III. 5 l'interface supervision et le contrôle ,35.
Figure III. 6 le module shield Ethernet ,38.
Figure III. 7 ;connexion serveur/client ,40.
Figure III. 8 Test 1: avant le test,41.
Figure III. 9Test 1:après le test ,42.
Figure III. 10Test 2: avant le test ,24.
Figure III. 11Test 2: après le test,43 .
Figure III. 12Test 3: après le test ,43.
Figure III. 13Test 4: en cachant le capteur de lumière ,44.
Figure III. 14Test 4: avec projection de la lumière ,44.

# Introduction générale

L'évolution de la technologie et du mode de vie nous permet aujourd'hui de prévoir des espaces de travail et de logement mieux adaptés. De même, La majorité des individus, et plus particulièrement les personnes âgées, passent beaucoup de leur temps à domicile, d'où l'influence considérable de l'habitat sur la qualité de vie. L'amélioration du sentiment de sécurité et de confort dans l'habitat apparaît donc comme une tâche d'une grande importance sociale.

La domotique ou encore la maison intelligente est définie comme une résidence équipée de technologies d'informatique, d'automatisme et d'électronique, ambiante qui vise à assister l'habitant dans les situations diverses de la vie domestique en améliorant le confort et simplifiant un certain nombre de tâches.

Pour faire le contrôle et/ou la supervision au sein des habitations intelligentes, chaqueactionneur, détecteur et capteur doit être lié à l'unité de contrôle. Avec le grand nombre d'informations et données qui véhiculent dans le système de supervision et contrôle, assurer une communication sécurisée est un challenge à réaliser.

Dans ce travail, on propose de faire cette communication en utilisant un réseauEthernet on tous les dispositifs peuvent êtrecontrôlés et supervisés à travers ce réseau.La solution réaliséenécessite le développement d'une application pour la supervision et le contrôle ainsi que l'utilisation de microcontrôleurà base d'Arduino pour la liaison des différents dispositifsutilisés [1].

Le mémoire est organisé en trois chapitres. Après une introduction générale, nous donnons une idée générale sur les bus de communication utilisé en domotique. Dans le deuxième chapitre, nous présentons l'Arduino et ses périphériques ainsi que la méthodologie de développement. Le système réalisé ainsi que l'application développée font partie d'une explication détaillée au chapitre 3. Enfin notre document s'achève par une conclusion générale et perspectives.

## I. Introduction

Partout dans la productique et les procédés, la concurrence pousse à réduire les frais et à mobiliser les potentiels de rationalisation restés inexplorés. Dans l'automatisation des procès industriels, l'utilisation du bus de terrain pour diminuer les coûts d'installation électrique des moyens de production a largement fait ses preuves. Par rapport au câblage parallèle, le câblage série sous la forme du bus de terrain, présente de nombreux avantages.

Dans les systèmes d'automatisation complexes, les commandes réparties dans l'installation doivent être reliées entre elles. On utilise des systèmes de bus normalisés pour relier en réseau les différentes commandes. Une telle mise en réseau offre plusieurs avantages dont les principaux sont énumérés[2].

### I.1 Définition des bus de terrain

Le terme bus de terrain est utilisé par opposition au bus informatique. En effet, le bus de terrain est en général beaucoup plus simple, du fait des faibles ressources numériques embarquées dans les capteurs et actionneurs industriels. Il est également plus robuste face aux perturbations externes.

Le bus de terrain est un système de réseau industriel pour le contrôle distribué en temps réel. C'est un moyen de connecter des instruments dans une usine de fabrication. Le bus de terrain fonctionne sur une structure de réseau qui permet généralement des topologies en guirlande, étoile, anneau, branche et arborescence. Auparavant, les ordinateurs étaient connectés à l'aide de RS-232(connexions série) par lesquelles seulement deux appareils pouvaient communiquer. Ce serait l'équivalent du schéma de communication 4-20 mA actuellement utilisé qui exige que chaque équipement ait son propre point de communication au niveau du contrôleur, alors que le bus de terrain est l'équivalent des connexions actuelles de type LAN, qui ne nécessitent qu'un seul point de communication au niveau du contrôleur et permettre à plusieurs(centaines) de points analogiques et numériques d'être connectés en même temps. Cela réduit à la fois la longueur du câble nécessaire et le nombre de câbles requis. De plus, étant donné que les dispositifs qui communiquent via un bus de terrain nécessitent un microprocesseur, plusieurs points sont généralement fournis par le même dispositif. Certains périphériques de bus de terrain prennent désormais en charge des schémas de contrôle tels que le contrôle PID du côté de l'appareil au lieu de forcer le contrôleur à effectuer le traitement.[3]

### I.2 finalités des bus de terrain

Un bus de terrain a les finalités suivantes :

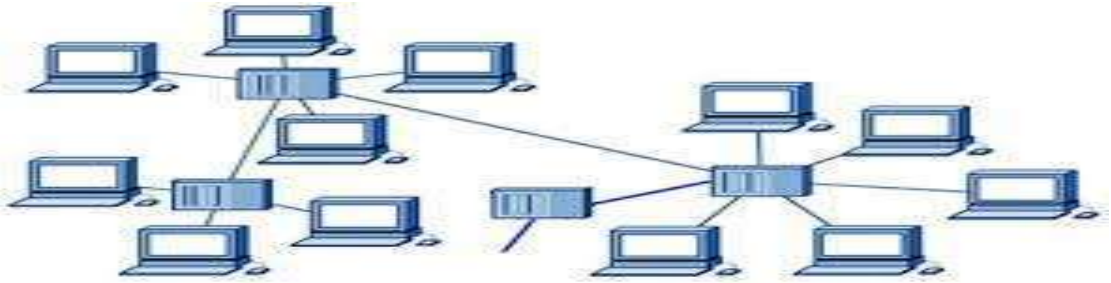
- Simplifier le câblage des capteurs, (pré) actionneurs, équipements de sécurité.
- Mesurer, commander, paramétrer et diagnostiquer de façon Précise.
- Outils de paramétrages et de diagnostic complexes et variés.
- Veiller à la qualité des raccordements, règles de topologies.
- Interopérabilité : de multiples solutions chez de multiples constructeurs.
- Gestion des versions.

### I.3 Les différentes topologies des bus de terrain

Il existe plusieurs topologies pour les bus de terrain.[4]

#### I.3.1 Topologie en étoile

Leréseau de la figure I.1 est un exemple d'un réseau domotique.



**FigureI.1 :** Topologie enétoile.

- **Avantages :**
  - on peut retirer une station sans que le réseau entier ne tombe.
  - extension facile.
  - localisation facile des pannes.
  - débit de transfert extrêmement élevé.
- **Inconvénients:**
  - l'important travail de câblage engendre de gros frais.
  - si le nœud central est défectueux, tout le réseau est indisponible.

### I.3.2 Topologie en anneau

La figure I.2 montre un exemple d'une topologie étoile en anneau.

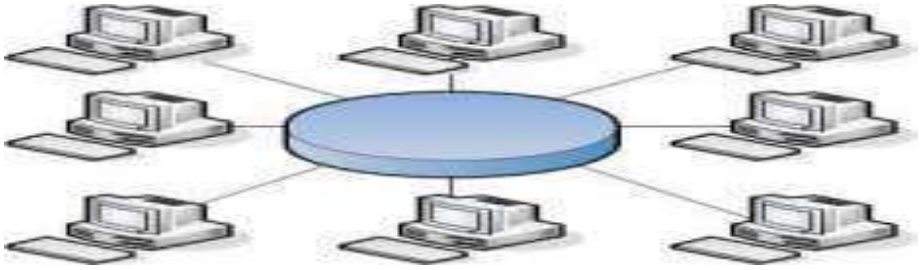


**Figure I.2 :** Topologie en anneau.

- **Avantages :**
  - en raison d'un travail de câblage faible, peu de frais.
  - ajout aisé d'une station par coupure de l'anneau réseau.
- **Inconvénients :**
  - travail de câblage plus onéreux qu'un réseau à topologie en bus.

### I.3.3 Topologie en ligne bus

La figure I.3 donne les différentséléments qui composent une ligne bus.

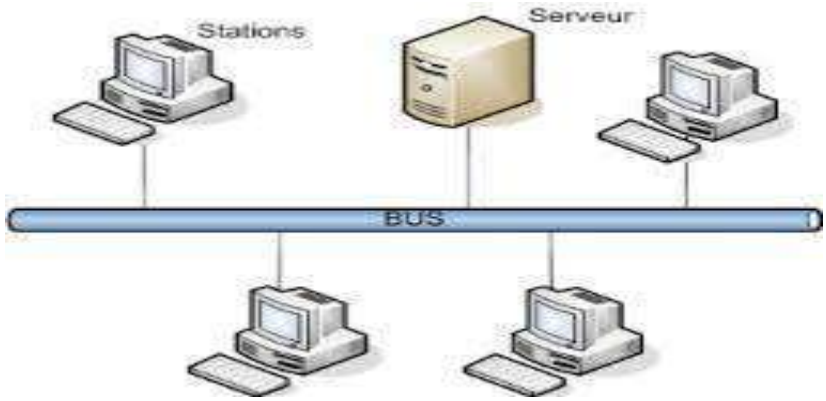



**Figure I.3 :** Topologie en ligne bus.

- **Avantages :**
  - peu de frais, car faible travail de câblage.
  - possibilité de rajouter facilement une station.
- **Inconvénients :**
  - Un problème de câblage sur un des nœuds peu perturber tout le reste du réseau.
  - Performances réduites en cas de charges importantes.
  - Lorsque le support est en panne, c'est l'ensemble du réseau qui ne fonctionne plus.

### I.3.4 Topologie en arbre

La figure I.4 donne un exemple sur un exemple de topologie en arbre.

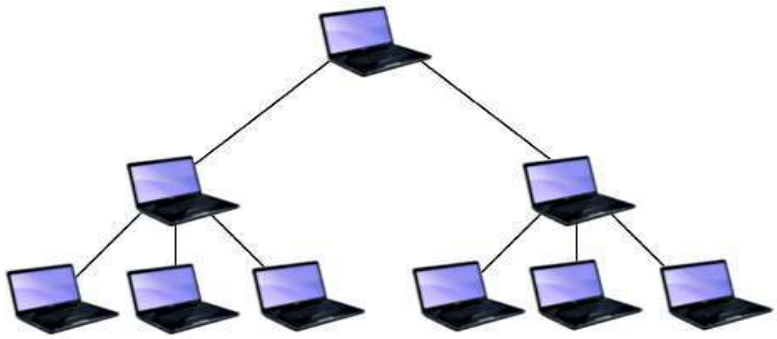


**Figure I.4 :** Topologie en arbre.

- **Avantage :**

- Extension aisée à tout instant.

- Le câblage est d'une grande flexibilité.

- Une défaillance d'un terminal n'affecte pas le reste de l'installation.

- **Inconvénients :**

- La défaillance d'une station périphérique peut entraîner la défaillance d'autres stations.

### I.4 les supports de transmission

### I.4.1 Cuivre

Le support utilisé est une paire de fils de cuivretorsadés.

- **Avantages :**
  - Blindé contre les perturbations extérieures.
  - Torsadé pour prévenir le couplage inductif.
- **Inconvénients :**
  - La résistance électrique du conducteur limite le rayon d'action.

### I.4.2 Lumière

Le support de transmission est des fibres optiques.

- **Avantage :**

- Utilisable sur de grandes distances.

- Pratiquement insensible aux interférences.

- Vitesse de transmission des données très élevée.

- **Inconvénients :**
  - Raccordement compliqué.
  - Plus cher que les paires torsadées.
  - Pose coûteuse en raison de la rigidité du matériel (grand rayon de courbure).

### I.4.3 Radio

Le support de transmission est des ondes radio.

**Avantage :**

- N'a pas besoin d'une ligne de liaison.

- Conception simple.

- **Inconvénients :**

- Le rayonnement électromagnétique incommode les personnes sensibles

- Composant onéreux.

### I.5 Types de transmission :

- **La liaison simplex :** caractérise une liaison dans laquelle les données circulent dans un seul sens, c'est-a-dire de l'émetteur vers le récepteur.
- **La liaison hall-duplex :** caractérise une liaison dans laquelle les données circulent dans un sens ou l'autre, mais pas les deux simultanément.
- **La liaison full-duplex :** caractérise une liaison dans laquelle les données circulent de façon bidirectionnelle et simultanément.
- **La liaison parallèle :** on désigne par liaison parallèle la transmission simultanée N bits.ces bits sont envoyés simultanément sur N voies différentes.
- **La liaison série :** dans une liaison en série, les données sont envoyé bit par bit sur la voie de transmission.
- **La liaison asynchrone :** danslaquelle chaque caractère est émis de façonirrégulière dans le temps (par exemple un utilisateur envoyant en temps réel des caractères saisis au clavier).

- **La liaison synchrone :** danslaquelle émetteur et récepteur sont cadencés la même horloge.

## I.6 Les bus de terrain

### I.6.1 Bus CAN

Le **CAN** est un réseau de terrain(ou réseau embarqué) qui occupe aujourd'hui une position de leader sur le marché automobile. D'autres secteurs utilisent également le bus CAN : [5]

• véhicules industriels ;
• matériel agricole ;
• bateaux ;
• avions ;
• production industrielle
• automatismes ;
•la domotique … .

#### I.6.1.1PRESENTATION DU BUS CAN :

**Multiplexage temporel :** Le multiplexage (figure I.5) temporel consiste à faire circuler sur un même câble (un bus) une multitude d'informations entre les différents équipements (très souvent des calculateurs). Ces équipements communiqueront donc à tour de rôle. Cette technique permet de diminuer une quantité importante de câblage à installer et de coût (en masse, matériaux, main d'œuvre, maintenance).la figure I.5 montre le multiplexage temporal.




**FigureI.5:**Multiplex agetemporal.

#### I.6.1.2Le bus CAN (Controller Area Network)

CAN est un bus série de terrain car il doit fonctionner dans un environnement limité et sévère (milieu industriel, atelier, voiture…) permettant la transmission asynchrone de données numériques. Il s'agit d'un bus multiplexé.

Les particularités de ce bus sont :
• bus multi maîtres où tous les participants ont les mêmes droits ;
• les nœuds (hôtes) de ce bus ne sont pas adressés seuls les messages transmis possèdent un identificateur. Chaque nœud scrute en permanence le bus et décide selon l'identificateur si le message lui est destiné ou pas.
• fiabilité élevée des mécanismes de protection du protocole.

### I.6.2 MODBUS

Le bus Modus répond aux architectures Maître/Esclave. Il a été créé par la société Modibo pour interconnecter les automates programmables. Ce protocole a rencontré beaucoup de succès depuis sa création du fait de sa simplicité et de sa bonne fiabilité.

#### I.6.2.1Principe général

Le bus est composé d'une station Maître et de stations esclaves. Seule la station Maître peut être à l'initiative de l'échange (la communication directe entre stations Esclaves n'est pas réalisable). Le maître peut s'adresser aux esclaves individuellement ou envoyer un message de diffusion générale àtous les esclaves.

Les esclavess renvoient un message (réponse) aux requêtes qui leur sont adressées individuellement. Les requêtes de diffusion générale n'attendent pas de réponses en retour.

Un seul équipement peut émettre sur la ligne à tout moment. Le maître gère l'échange et seul lui peut prendre l'initiative. Il interroge chacun des esclaves successivement.

Aucun esclave ne peut envoyer de message à moins qu'il ne soit invité à le faire.

Le maître répète la question lorsqu'un échange est incorrect et déclare l'esclave interrogé absent si aucune réponse n'est reçue dans un délai donné.

Si un esclave ne comprend pas un message, il envoie une réponse d'exception au maître. Le maître Le maître peut réitérer ou non la requête.[6]

#### I.6.2.2 Trame Modus

Les trames sont de 2 types : [7]

- ✓ Mode RTU: les données sont sur 8 bits.

Le mode de transmission utilisé est le mode RTU3. La trame ne contient ni octet d'en-tête de, ni octet de fin de message.

- ✓ Mode ASCII2: les données sont sur 7 bits (les trames sont donc visibles en hexadécimal et il faut deux caractères pour représenter un octet). Ce dernier mode est quasiment tombé en désuétude.

Le protocole Modus consiste en la définition de trames d'échange.

Le maître peut aussi diffuser un message à tous les esclaves présents sur le réseau. Ceux-ci exécutent l'ordre du message sans émettre une réponse. Le maître envoie une demande et attend une réponse. La figure I.6 donne un exemple sur l'architecture maitre/esclave.

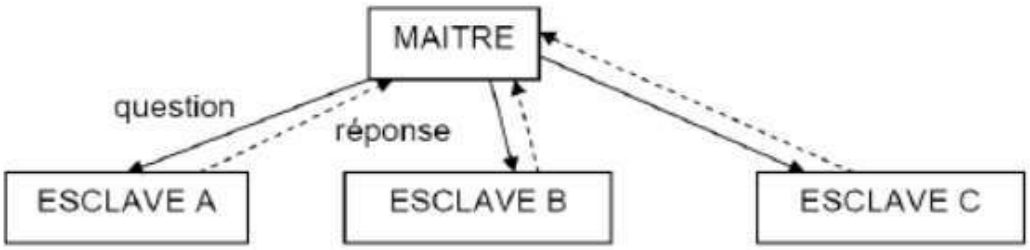



**Figure I.6 :** Architecture maitre/esclave.

### I.6.3 BUS KNX

**I.6.3.1 KNX :** est le standard mondial pour la gestion technique de maison et de batiment.Il reprend le meilleur de Batibus, EIB (européen installation bus) et EHS (européen home Systems qui communique par courant porteur).[8]

Bus KNX peut être utilisé dans toutes sortes de bâtiments comme les espaces de vie et de travail, les bâtiments neufs, les bâtiments classés etc. et dispose de trois mode de configurations:

- ✓ L'A-mode : C'est une mode de configuration automatique à destination de l'utilisateur
- ✓ L'E-mode : Comporte des fonctionnalités standards, à destination de l'installateur
- ✓ Le S-mode : Pour les configuration sur site ou à distance pour les intégrateurs.

Le bus KNX peut être associé à d'autres systèmes comme le Bacnet ou le DALI grâce à des passerelles. Parmi les avantages de KNX, on a :

- La simplicité de la conception.
- La flexibilité de l'installation.
- La maintenance aisée de l'installation.
- La Fiabilité.

### I.6.3.2 Fonctionnement

• L'utilisateur confi gure de manière graphique directement sur les plans du bâtiment.
• Il détermine les fonctions et les charges associées à chaque commande et capteur KNX (boutons poussoirs, thermostats, détecteurs,...)
• Des modèles sont disponibles (par exemple : chambre à coucher type, éclairage constant dans les bureaux). L'utilisateur peut aussi créer ses propres solutions modélisées.
• Le choix des actionneurs KNX est généré automatiquement ou sélectionné manuellement.
• Si un contrôleur IP homeLYnk ou spaceLYnk est intégré dans la confi guration, les interfaces utilisateurs pour les smartphones, tablettes,
écrans tactiles et ordinateurs sont créés automatiquement sous forme de widget.
• La liste du matériel peut être exportée sous forme d'un fi chier .csv.
• Le rapport d'installation peut aussi être édité de manière à constituer un dossier complet à partager avec son équipe ou son client.

### I.6.4 Bus AS-I

Le bus AS-I (ActuatorsSensors Interface) est un bus de capteurs/actionneurs uniquement de type maître/esclave qui permet de raccorder des capteurs ou pré actionneurs sur un câble spécifique (2 fils) transportant les données et la puissance. Ce bus est totalement standardisé (non propriétaire) et permet d'utiliser des technologies de plusieurs constructeurs (interopérabilité). L'automate est pour cela doté d'un coupleur AS-I.[9]

#### I.6.4.1 Le bus ASI offre de nombreux avantages :

-La réduction du volume de l'armoire de commande.

-La minimisation des erreurs.

-Le maîtres ASI remplace les cartes E/S.

-Suppression des borniers de raccordements.

-Raccorder chaque capteur/actionneur n'importe où, donc en tout point du câble ASI via les modules esclaves sans passer donc par les cartes E/S.

#### I.6.4.2 Le bus AS-I permet donc d'éviter de nombreux inconvénients comme :

-Les nombreux risques d'erreurs.

-Les tests long et laborieux.

-De nombreux borniers de raccordement.

-De raccorder chaque capteur/actionneur directement aux cartes E/S de l'API.

-De réduire les coûts d'étude, de documentation, de fabrication et de maintenance.

La topologie d'AS-i est libre : point à point, en ligne ou en arbre en anneau.la figure I.7 est un exemple d'une topologie de bus AS-I.

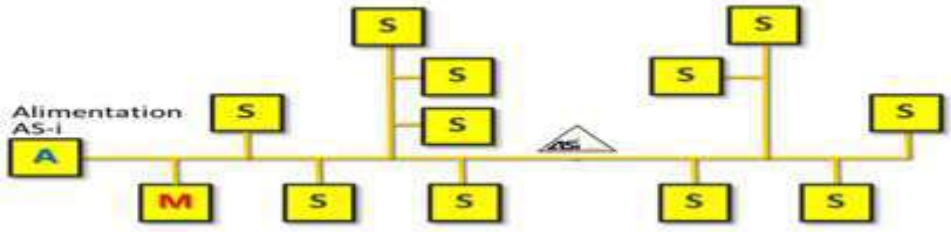



**Figure I.7 :**Topologie de bus d'AS-i

### I.6.5 Profibus

Profibusest le nom d'un bus de terrain inventé par Siemens et devenu peu à peu une norme de communication dans le monde de l'industrie.Profibus autorise le dialogue de matériel multiconstructeurs pour la transmission de données rapides(déterminisme) mais aussi pour les échanges de grandes quantités d'informations (non déterministes).[10]

#### I.6.5.1 Le bus PROFIBUS-DP (DecentralisedPeripheral) (périphérie décentralisée)

Il est utilisé pour la commande de capteurs, d'actionneurs ou d'automates programmables par une commande centrale.
Il est utilisé aussi pour la connexion d'une « intelligence distribuée »,c'est-à-dire la mise en communication de plusieurs automates les uns avec les autres. Les débits peuvent atteindre 12 Mbit/s sur paires torsadées ou fibre optique.
La majorité des automates Siemens disposent d'une interface Profibus-DP pour le dialogue avec le PC de programmation.
Profibus respecte le modèle OSI (niveaux 1,2 et 7).

#### I.6.5.2 Les protocoles de transmission

Profibus possède 2 protocoles de transmission ou profils de communication :

- profil de communication DP (DecentralizedPeriphery) : le plus répandu, simple et performant.
- profil de communication FMS (Fieldbus Message Specification) : plus évolué pour des tâches complexes.

Profibus utilise 3 type de supports de transmission : RS485,IEC 11582et fibre optique.la figure I.8 montre le support de transmission dans un Profibus.




**Figure I.8 :**supports de transmission dans un Profibus.

On reconnaît facilement un réseau Profibus-DP à la couleur de son câble : violet. En l'ouvrant, on distingue 2 fils : un vert et un rouge, nommé "A" et "B". Connecteur habituellment utilisé : DB9.

Profibus est un réseau multimaître :
• maître : pilote la transmission de données sous réserve d'avoir le droit d 'accès au réseau
• esclave : pas le droit d'accès au réseau.Acquitte le message reçu ou transmet des données sur demande du maître Profibus utilise :
• la méthode du jeton pour les communications intermaître : garantit l 'accès au bus au moins une fois à chaque maître dans un temps donné. Le jeton est un droit de parole.
• la méthode maître/esclave : permet au maître possédant le jeton d 'accéder à ses esclaves.

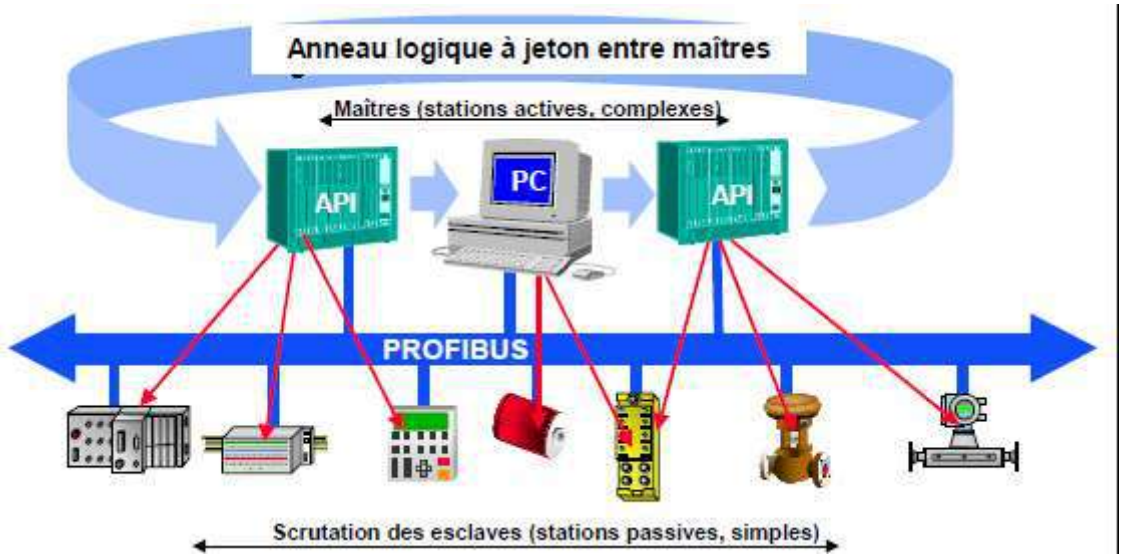



**Figure I.9 :**La nature hybride de la gestion d'accès à PROFIBUS : configuration maître-esclaves et passage de jeton entre trois maîtres formant un anneau logique.

## I.7 RéseauEthernet

### I.7.1Introduction

Mise au point dans les années 80 par XEROX, Intel et DEC, L'architecture Ethernet permet l'interconnexion de matérielsdivers avec de grandes facilités d'extension. [11]

### I.7.2Les différentes normes

- Ethernet à 10 Mbits/s : 10BASET, 10BASE5, 10BASE2 et 10BASEF(802.3)
- FastEthernet à 100 Mbits/s : 100BASET (802.3u)
- Gigabit Ethernet (Gig-E) à 1 Gbits/s : 1000BASE-LX, 1000BASE-SX,1000BASE-CX, 1000BASE-LH (802.3z) et 1000BASE-T (802.3ab)
- Décagigabit Ethernet à 10 Gbits/s (10 Gigabit Ethernet) est déjà àl'étude ...

### I.7.3Caractéristiques principales

- Débit de 10 Mbits/s, 100 Mbits/s, 1 Gbits/s, …
- Transmission bande de base
- Topologie physique (câblage) en étoile (ou en bus pour le coaxial)

- Câbles cuivre (coaxial ou paires torsadées) ou fibre optique (monomode ou multimode).
- Méthode d'accès CSMA/CD (Carrier Sense Multiple Access / Collision Detection)
- Longueur des trames comprises entre 64 et 1518 octets
- Topologie logique en bus
- Gestion des couches 1 et partiellement 2 du modèle OSI

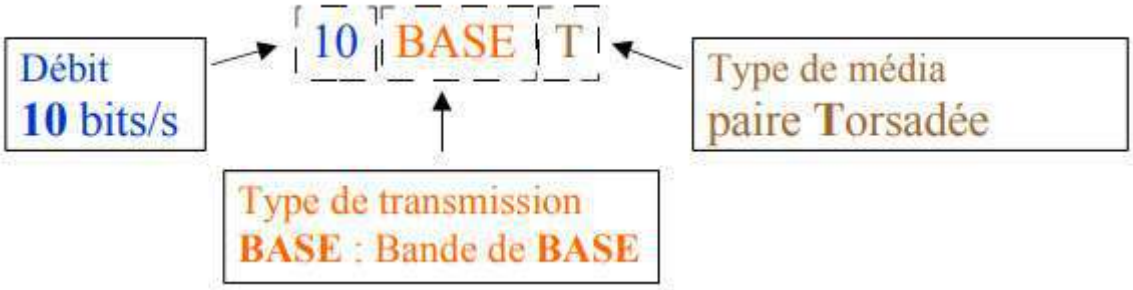



**Figure I.10 :Sigle**

- 10BASE5 : Ethernet 10 Mbits/s standard coaxial (Thick)
- 5 : longueur max. du segment 500 m
- 10BASE2 : Ethernet fin à 10 Mbits/s coaxial (Thin)
  - 2 : longueur max. du segment 185 m

### I.7.4 Trame 802.3 (Frame 802.3)

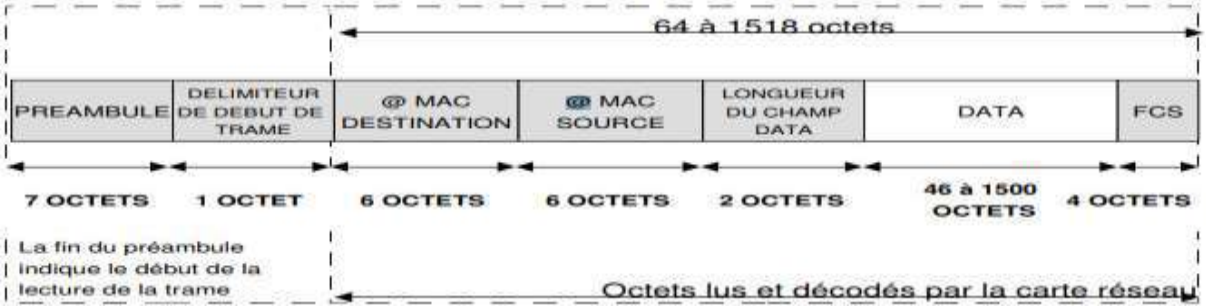



**Figure I.11 :**La trame 802.3 .

- Le préambule, composé d'une succession de 0 et de 1, assure la synchronisation du récepteur sur la trame émise.
- Le délimiteur de trame permet de trouver le début du champ d'adresses.
- Une séquence de contrôle (Frame Check Sequence) calculée suivant un code de
- redondance cyclique (CRC).

### I.7.5 Trame Ethernet_II

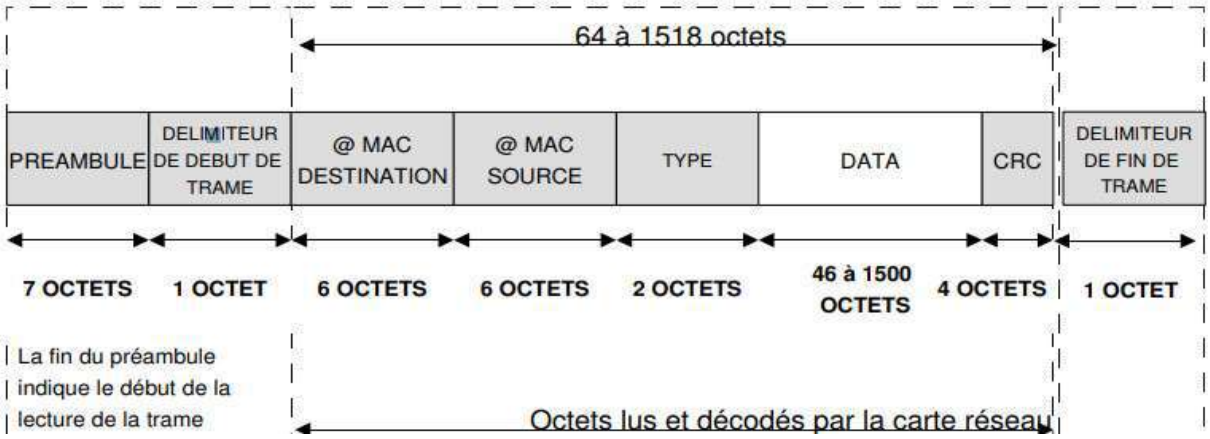



**Figure I.12 :**la trame Ethernet II.

- Le type de protocole de couche supérieure sur 2 octets (Ex: 0800 = IP).
- Le MTU (Maximal Transfer Unit) est un paramètre variable (46 à 1500) qui représente la taille maximale d'un paquet que la trame peut transporter. La couche supérieure réalisera une fragmentation si c'est nécessaire.
- On distingue les deux trames de la manière suivante : si x<= 1 500 alors trame 802.3 avec LLC (et éventuellement SNAP), sinon si x > 1 500 alors trame Ethernet II

### II.7.6Ethernet et le modèle OSI

La Figure I.13 montre une architecture Ethernet et le modèle OSI.

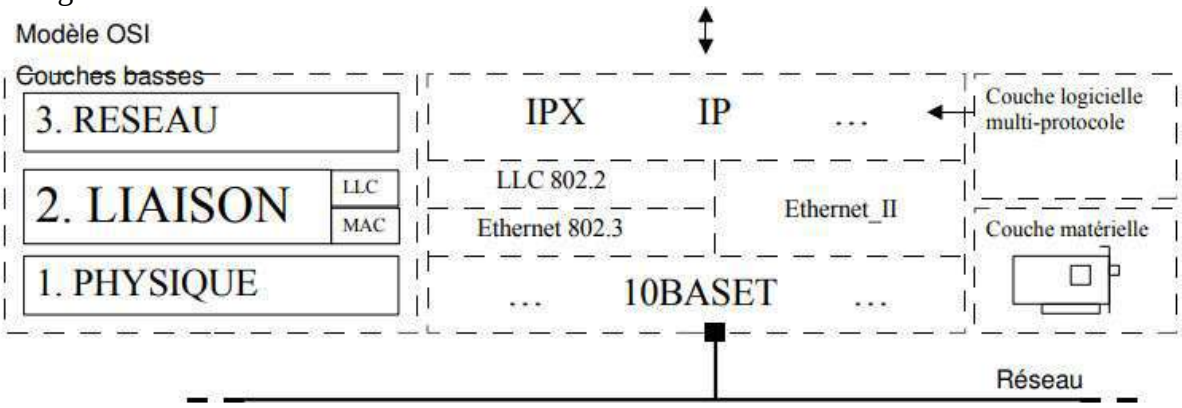



**Figure I.13 :**architecture Ethernet et modèle OSI.

- La norme Ethernet couvre partiellement la couche 2 du modèle OSI qui est, en fait, découpée en 2 sous-couches :
  - La sous-couche MAC (Medium Access Control)
  - La sous-couche LLC (Logical Link Control)
- La sous-couche MAC d'Ethernet gère l'accès au support et offre un ensemble de services à la sous-coucheLLC(Logical Link Control).

### I.7.7 Couche Physique (niveau 1 du modèle OSI)

- Les fonctions réalisées, sur le support, par le transceiver sont les suivantes :
  - émission et réception de signaux suivant un codage (ex : Manchester).
  - détection de présence de signaux.
  - détection de collisions.
- Les codages en bande de base vont essentiellement avoir pour rôle dediminuer la largeur de bande du signal binaire (moins perturbée), detransposercelle-ci vers des fréquences plus élevées (les moins affaiblies) etd'utiliser les transitions du signal afin d'assurer une transmission synchrone(qui permettront au récepteur de synchroniser sa propre horloge).
- La détection de collision se fait par écoute du support. Lorsque la tension surle câble est plus élevée que la tension maximale pouvant être générée parun seul transceiver, une collision est détectée .

### I.7.8Concentrateur (hub) et commutateur (switch)

- Un concentrateur (hub) est le nœud central d'un réseau informatique. Il s'agit d'un dispositif électronique permettant de créer un réseau informatique local de type Ethernet.
- Cet appareil permet la connexion de plusieurs équipements sur une même ligne de communication, en régénérant le signal, et en répercutant les données émises par l'un vers tous les autres. C'est donc un simple répéteur de données par opposition au commutateur réseau (switch), qui dirige les données uniquement vers la machine destinataire.
- Un commutateur réseau (switch) est un équipement qui relie en étoile plusieurs segments (câbles ou fibres) dans un réseau informatique. Il s'agit le plus souvent d'un boîtier disposant de plusieurs ports Ethernet : il a la même apparence qu'un concentrateur (hub).

  **Principe :** Une opération XOR (ou exclusif) est réalisée entre l'horloge et les données, d'où une transition systématique au milieu de chaque bit du signal binaire.

### I.7.9La couche Liaison (sous-couches MAC et LLC)

- Pour les normes IEEE 802, la sous-couche LLC (IEEE 802.2) est indépendante des trois implantations possibles de la sous-couche MAC qui sont :
  - CSMA/CD (IEEE 802.3),
  - Token Bus (IEEE 802.4) et
  - Token Ring (IEEE 802.5).
- La sous-couche LLC permet notamment l'aiguillage des données vers la couche supérieure.
- Dans le cas d'une trame Ethernet_II, la souscouche LLC est vide.

### II.7.9.1Méthode d'accès CSMA/CD

- La méthode d'accès utilisée sur les réseaux Ethernet est CSMA/CD (Carrier Sense Multiple Access / Collision Détection) . Sur ce type de réseau, il est possible que 2 ou plusieurs stations détectent le support libre, décident retransmettre en même temps et ce qui provoque une collision :cette situation pose problème.
- Autres paramètres :
  - la taille du JAM (signal de renforcement de collision) est fixée à 32 bits-times (contenu quelconque).
  - le délai minimum inter trames (ou IFS) est de 9,6 µs soit 96bits-times.

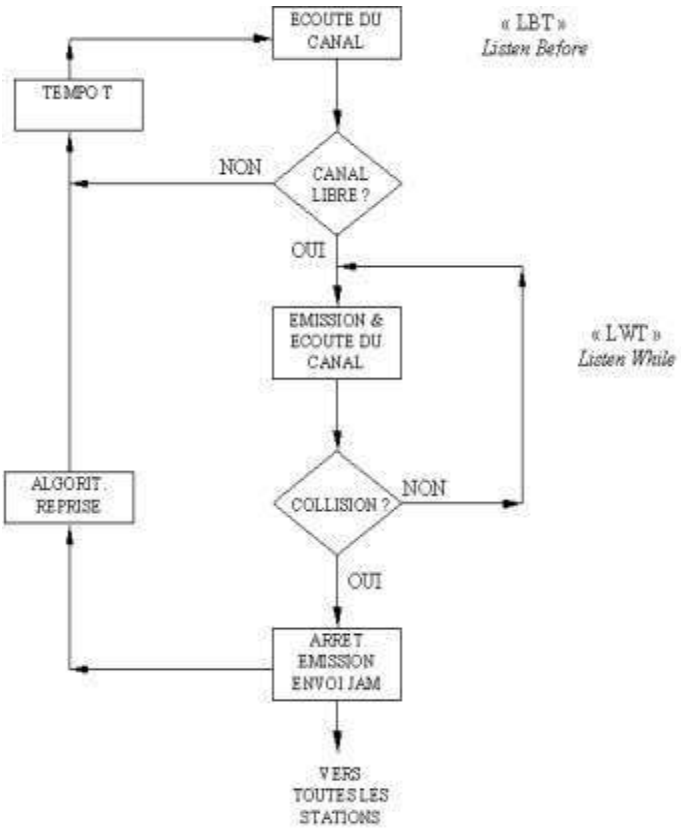



**Figure I.14:**Méthode d'accès.

### I.7.9.2Round Trip Delay

- Un paramètre intervient pour la gestion correcte des collisions : la taille du réseau ou plus exactement le round trip Delay (RTD ou Tb).
- Il correspond au temps nécessaire :
  - à la propagation d'une trame d'un bout à l'autre du réseau (propagation aller) ;
  - à la détection d'une éventuelle collision à l'extrémité du réseau ;
  - à la propagation en retour de l'information de collision (propagation retour) .

- Contrairement à un concentrateur (hub), un commutateur ne se contente pas de reproduire sur tous les ports chaque trame qu'il reçoit. Il sait déterminer sur quel port il doit envoyer une trame, en fonction de l'adresse MAC à laquelle cette trame est destinée. Les commutateurs sont souvent utilisés pour remplacer des concentrateurs :
  - un hub 100 Mbits/s partagera sa bande passante entre chaque port .

• un switch 100 M bits/s permettra de disposer de cette bande passante pour chaque port.

### I.7.10Virtual LAN

- Un réseau virtuel, communément appelé VLAN (pour Virtual LAN), est un réseau informatique logique indépendant. De nombreux VLAN peuvent coexister sur un même commutateur réseau (switch). Les VLAN fonctionnent au niveau de la couche 2 du modèle OSI.
- Création de VLAN :
  - VLAN par port (Port-based VLAN) : on affecte chaque port à un VLAN
  - VLAN par adresse MAC (MAC address-based VLAN) : on affecte des adresse MAC à un VLAN et le switchmaintient une table @ MAC <-> VLAN.
  - VLAN par adresse de niveau 3 : on affecte une adresse de niveau 3 à un VLAN (le commutateur doit donc accéder à ces informations et cela provoque un fonctionnement moins rapide que les VLAN par port ou par MAC).

- Intérêt des VLAN :
  - Segmentation : réduire la taille d'un domaine de diffusion (broadcasta)
  - Flexibilité : Possibilité de travailler au niveau 2 (couche liaison) (Adresse MAC) ou au niveau 3 (IP). Fonctionnant normalement au niveau de la couche 2 du modèle OSI, un VLAN peut être configuré pour se connecter directement un réseau IP, ce qui donne l'impression de travailler plutôt au niveau de la couche 3 (commutateur de niveau 3).
  - Sécurité : permettre de créer un ensemble logique isolé pour améliorer la sécurité. Le seul moyen pour communiquer entre des machines appartenant à des VLAN différents est alors de passer par un routeur.

### I.7.11Exemples de réseaux

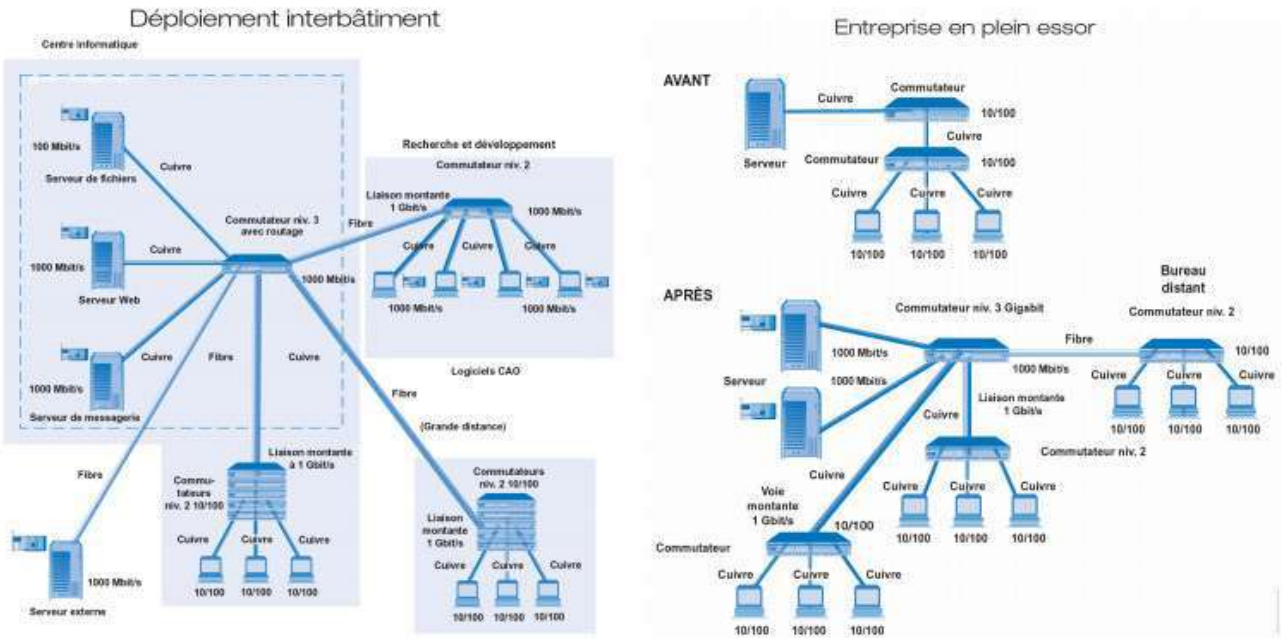



**Figure I.15 :**exemple de réseau Ethernet.

### I.8 Conclusion :

Un aperçu est donné sur les bus de terrain utilisé en industrie et en domotique. En essayant de donner le principe de fonctionnement ainsi que les avantageset inconvénients de chaque type.

## II.1 Introduction

Arduino est un circuit imprimé il englobé des composant le plus important un microcontrôleur qui est le cerveau de la carte programmé pour tester et produire des signaux électriques, de manière à effectuer plusieurs tâches comme les techniques d'électroniques.Arduino peut être un élément primordial dans la chaine de communication où il peut assurer la lecture des données, commander les actionneurs selon un algorithme et surtout faire le lien entre l'application et les dispositif à travers un module de communication.

## II.2Historique

L'Arduino a été inventé 2005 par l'enseignent Massimo Banzi dans une école de dessin à Iveaen Italie avec l'aide de David Cuartielle ingénieur en microcontrôleurs et aussi leur étudiant Mellis qui est spécialiste dans les langages de programmation.

## II.3Définition

La carte Arduino est une plateforme utilisée pour réaliser des projets électroniques plus développé. Elle est composée d'un circuit physique programmables est dit microcontrôleurs et de logiciel utilisé pour créer et télécharger le code de l'ordinateur à la carte.

## II.4 But

L'Arduino est un système qui permet de lier les performances de programmation à celle de l'électronique tout en s'simplifiant les schémas électroniques et les couts de réalisation.

## II.5Utilisation

Des développeurs des systèmes embarqués, de contrôle, de communication ou tout système à base de microcontrôleur.

## II.6Composition

Il nous permet de réaliser plusieurs projets tel que :

- Contrôler les appareils domestiques ;
- Robotique ;
- Jeu de lumière ;
- Communiquer avec PC ;
- Télécommander un appareil mobile ;
- Etc…

## II.7 Choix et description de la carte Arduino

### II.7.1Choix

On a plusieurs paramètres pour choisir l'Arduino :

- Le prix : ils ne sont pas couteux avec un rapport de qualité bonne.
- Logiciel : gratuit et développer par le java. Il est simple à utiliser et nous permet de programmer la carte Arduino.
- Le matériel : consiste une carte électronique et les schémas sont disponibles.
- La compatibilité : le logiciel et la carte sont compatibles avec la programmation.

### II.7.2 Description de la carte

Elle contient les éléments suivants (Figure II.1) :

- Le microcontrôleur : considéré comme le cerveau de la carte.
- L'alimentation : C'est celle du microcontrôleur : 5V régulée (port USB) ou 7 à 12V provenant d'alimentation externe.
- Visualisation : par des LED de taille millimétrique. Elles sont là pour tester le matériel avec le branchement avec microcontrôleur et les autres LED sont pour émission et réception quand le téléchargement de programme dans le microcontrôleur.
- La connectique : Les connecteurs sont :
  - 0 à 13 : Entrée/Sortie numérique ;
  - A0 à A5 : Entrée/Sortie Analogique ;
  - GND : la masse (0V) ;
  - 5V : l'alimentation +5V ;
  - 3.3V : alimentation + 3.3V ;
  - Vin : alimentation non stabilisée.

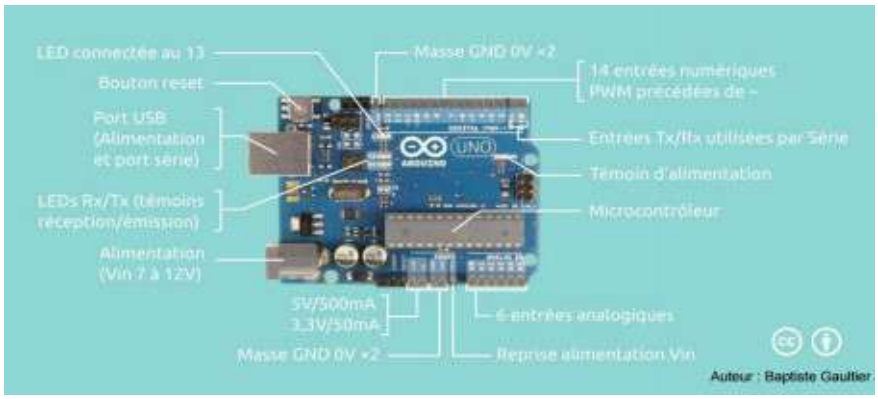



**Figure II.1 :**Description de la carte ArduinoUno.

## II.8Le langage

Le programme est lié à une série d'instruction élémentaires sous forme texte donc la carte lit après exécute les instructions par un ordre un après l'autre.

- Un ordinateur.
- Une carte Arduino.
- Programme lié à l'Arduino.

Syntaxe du langage : C et C++ qui est le suivant :

Code minimal : son rôle de devisé le programme en deux parties .

La fonction :Setup () : est considéré comme fonction d'initialisation on l'appelle une seul fois au début du programme.

Loop() : c'est pour écrire le cœur du programmes. Elle est appelée en permanence : en boucle infinie.

Les instructions : sont des lignes contenant des codes, exemple :« fait ceci »« Fait cela »

Les points-virgules (;) : pour finir les instructions.

Les accolades [ ] : sont utilisées pour les fonctions ; les boucles. Elles sont obligatoires.

Les commentaires : // cette ligne a un commentaire. /* pour plusieurs lignes

Les variables : Les variables booléennes peuvent prendre deux valeurs soit vraie ou faux donc. Si une variable vaut (0) on la considère comme variable booléennes fausse et si une

variable prend n'importe quelle valeur différente de zéro on la considère comme variable booléenne vraie.

- char (variable 'caractère').
- Int (variable 'nombre entier').
- long (variable 'nombre entier de très grande taille').
- string (variable 'chaine de caractères').
- Array (tableau de variables).

## II.9Types des cartes

Il y a trois types de cartes :

- « officielles » qui sont fabriquées en Italie par le fabricant officiel : Smart Projects
- « compatibles » qui ne sont pas fabriqués par Smart Projects, mais qui sont totalement compatibles avec les Arduino officielles.
- « autres » fabriquées par diverse entreprise et commercialisées sous un nom différent (Freeduino, Seeduino, Femtoduino, …).

Les différentes cartes : on commence par la plus utilisée

**II.9.1 Carte Uno :** c'est la carte base d'Arduino (Figure II.2), elle est simple à utiliser.




**Figure II.2 :**CarteArduinoUno.

**II.9.2 La carte Arduino Méga**

C'est une carte Arduino officiel mais avec beaucoup plus d'entrées / sorties et 2fois plus de mémoire qu'une carte Arduino UNO (Figure II.3). Elle contient :

- 54 entrées/ sorties numériques
- 16 entrées analogiques
- 256 ko de mémoire flash
- 6 interruptions extérieures




**Figure II.3 :**CarteArduino Méga.

### II.9.3 Arduino Nano

L'Arduino Nano est essentiellement un ArduinoUno de dimension 1.85x4.23cm, disposé pour une utilisation directe sur breadboard. Il embarque son propre contrôleur USB permettant une communication série très facile (Figure II.4) .



**Figure II.4 :**CarteArduino nano.

### II.9.4 ArduinoEsplora

La carte Esplora est dérivée de la carte Arduino Leonardo et est basée sur un ATMega 32U4. Elle contient une implantation d'image elle est utilisée généralement dans les manettes de jeux (dimension : 6.1x16.51cm) (Figure II.5)



**Figure II.5:**CarteArduinoEsplora.

### II.9.5 La carte seeduino

C'est type Arduino constituée de :
AT méga 328p au format CMS avec des entrées analogique el plus A6 et A7. Des portes 12c et série pour faciliter le câblage(Figure II.6)



**Figure II.6 :**Carteseeeduino.

### II.9.6 La carte Arduino pro (par Sparkfun)

Pour les circuits finit elle ne contient pas des ports USB juste le nécessaire pour fonctionner. Elle n'est pas chère, sa taille est réduite (Figure II.7) .




**Figure II.7 :**CarteArduino pro**.**

### II.9.7 Carte ArduinoBlackwidow :

Créé par Asynclabs, c'est une carte Arduino avec module wifi (Figure II.8)



**Figure II.8 :**CarteArduinoBlackwidow.

### II.9.8 Carte Rainbowduino :

c'est une carte créés par Seeedstudion, c'est une matrice 8x8 de LED (Figure II.9)



**Figure II.9 :**CarteRainbowduino.

Les cartes dites compatible avec Arduino sont :

- **Teensy++ 2.0 (par PJRC)**

Carte compatible (logiciel) avec Arduino(Figure II.10) .



**Figure II.10 :**Carteteensy.

- **Carte netduino**

elle ressemble à la forme classique d'Arduino mais elle diffère de langage car elle utilise (C# ou V.net)(Figure II.11)



**Figure II.11 :**Cartenetduino.

## II.10 Quelques modules d'Arduino

Le constructeur a suggéré qu'une telle carte doit être dotée de plusieurs ports de communications ; on peut éclaircir actuellement quelques types

### II.10.1 Le module Arduino Bluetooth

Le Module Microcontrôleur Arduino Bluetooth est la plateforme populaire Arduino avec une connexion sérielle Bluetooth à la place d'une connexion USB, très faible consommation d'énergie, très faible portée (sur un rayon de l'ordre d'une dizaine de mètres), faible débit, très bon marché et peu encombrant(Figure II.12)



**Figure II.12 :**Le module Arduino Bluetooth.

### II.10.2 Le module shieldArduino Wifi

Le module ShieldArduino Wifi permet de connecter une carte Arduino à un réseau internet sans fil Wifi (Figure II.13) .



**Figure II.13 :**Le module shieldArduino Wifi.

### II.10.3 Le module shieldArduino Ethernet

Le module shieldArduinoconnecte votre Arduino à Internet en quelques minutes. Il suffit de brancher ce module sur la carte Arduino, la connectez au réseau avec un câble RJ45 et faire quelques ligne de programmation pour la communication(Figure II.14) .



**Figure II.14 :**Le module shieldArduino Ethernet.

## II.11 Les capteurs

Un capteur est une interface entre un processus physique et une information manipulable. Il ne mesure rien, mais fournit une information en fonction de la sollicitation àlaquelle il est soumis. Il fournit cette information grâce à une électronique à laquelle il estassocié (Figure II.15) .



Un capteur de température Un capteur de mouvementUn capteur de lumière

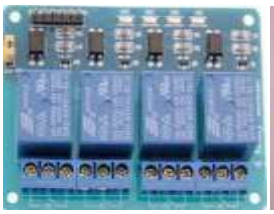
**Figure II.15 :**Les capteursutilisés .

## II.12 Les drivers

Il existe plusieurs drivers comme des cartes auxiliaires qui peuvent être attachées avec l'Arduino afin de faciliter la commande (Figure II.16) ; on peut citer quelques types.


Les moteur électriques


Les afficheurs LCD


Les relais


lesled

**Figure I.16 : L**es drivers.

## II.13 Logiciel Arduino

Description : Ce logiciel il a les fonctions suivantes :

- Ecrire et compiler le programme pour la carte.
- Connecter et transférer les programmes.
- Communiquer avec la carte.

Cet espace de développement intégré (EDI) dédié au langage Arduino et à la programmation des cartes Arduino comporte (Figure II.17) :

- Une**BARRE DE MENUS** comme pour tout logiciel une interface graphique (GUI),
- Une**BARRE DE BOUTONS** qui donne un accès direct aux fonctions essentielles du logiciel et fait toute sa simplicité d'utilisation,
- Un**EDITEUR** (à coloration syntaxique) pour écrire le code de vos programme, avec onglets de navigation,
- Une**ZONE DE MESSAGES** qui affiche indique l'état des actions en cours,
- Une**CONSOLE TEXTE** qui affiche les messages concernant le résultat de la compilation du programme

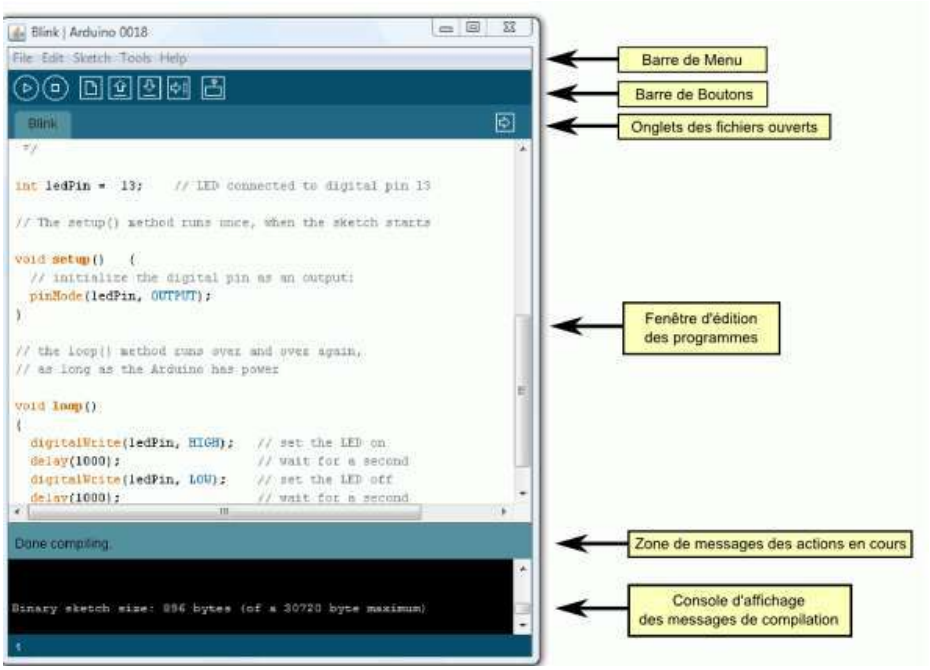



**Figure I.17 :**l'interface de logiciel Arduino.

Le logiciel Arduino intègre également :

- un**TERMINAL SERIE** (fenêtre séparée) qui permet d'afficher des messages textes reçus de la carte Arduino et d'envoyer des caractères vers la carte Arduino. Cette fonctionnalité permet une mise au point facilitée des programmes, permettant d'afficher sur l'ordinateur l'état de variables, de résultats de calculs ou de conversions analogique-numérique : un élément essentiel pour améliorer, tester et corriger ses programmes(Figure II.18) .

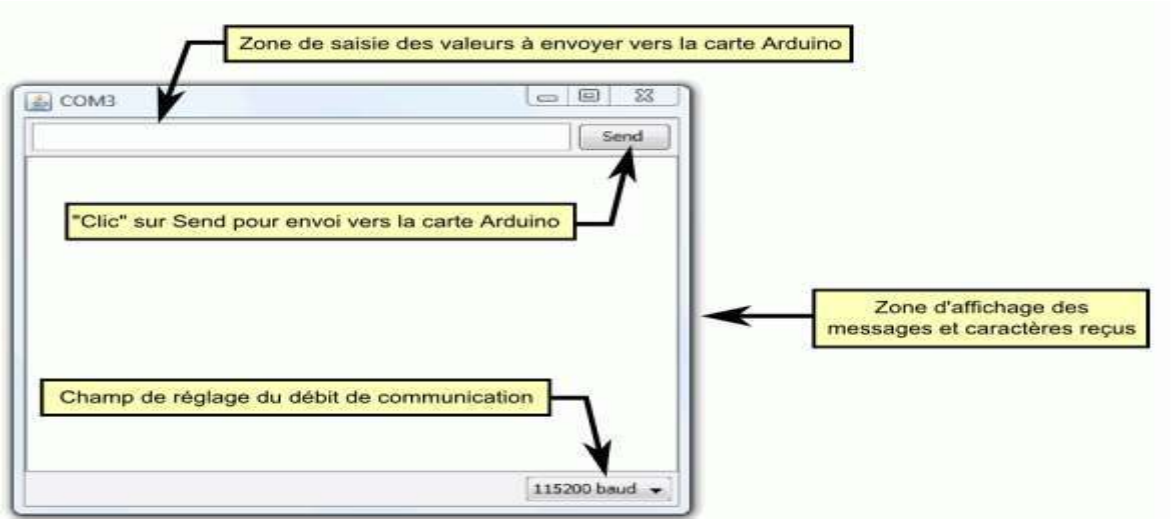



**Figure II.18 :**fenêtreterminal série.

### II.13.1Principe général d'utilisation

Le code écrit avec le logiciel Arduino est appelé un programme (ou une séquence - sketch en anglais) :

- Ces programmes sont écrits dans **l'éditeur de texte**. Celui-ci a les fonctionnalités usuelles de copier/coller et de rechercher/remplacer le texte.
- la**zone de messages** donne l'état de l'opération en cours lors des sauvegardes, des exportation et affiche également les erreurs.
- La **console texte** affiche les messages produit par le logiciel Arduino incluant des messages d'erreur détaillés et autres informations utiles.
- la**barre de boutons** vous permet de vérifier la syntaxe et de transférer les programmes, créer, ouvrir et sauver votre code, et ouvrir le moniteur série.
- la barre des menus vous permet d'accéder à toutes les fonctionnalités du logiciel Arduino.

### II.13.2 Description de la barre des boutons

Le tableau II.1 décrit la barre des boutons



### II.13.3 Transfert des programmes vers la carte Arduino

La (Figure II.19) donne une description détaillée sur la barre des boutons

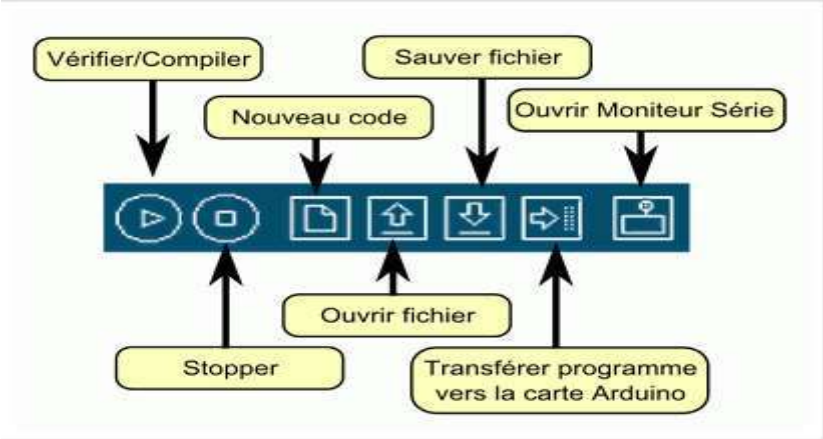



**Figure II.19 :**description de la barre des boutons.

- **Saisir votre programme et vérifier le code**

On suppose ici qu'un programme correctement écrit se trouve dans la fenêtre éditeur. Pour votre première programmation de la carte, aller dans le menu **File>Examples>Digital>Blink**: un programme s'ouvre avec du code dans la fenêtre éditeur.Appuyez alors sur le boutonVerify de la barre d'outilspour lancer la vérification ducode (Figure II.20) :

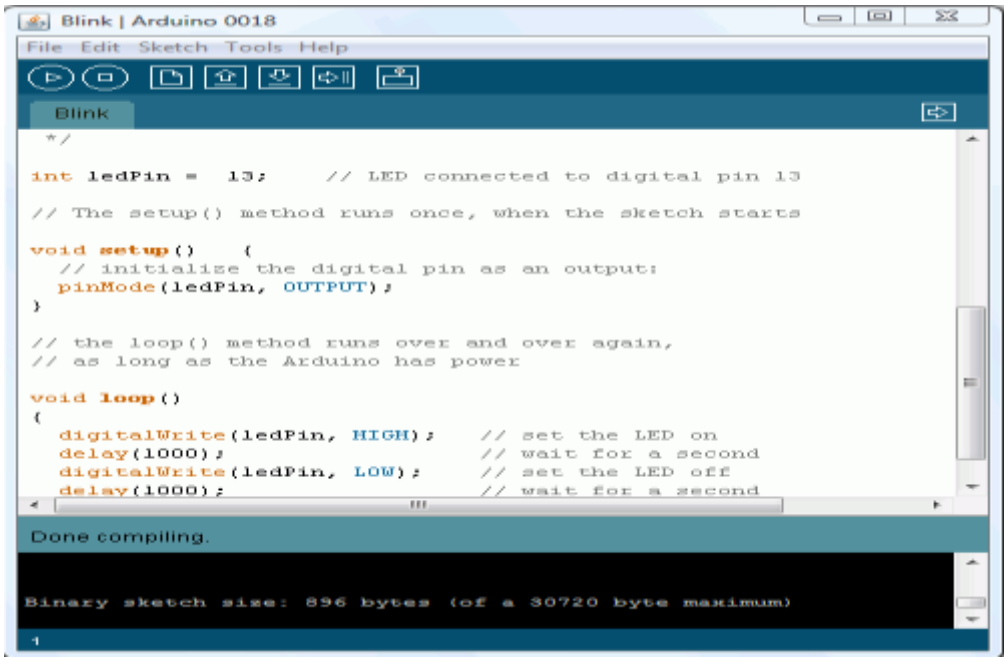



**Figure II.20 :**lafenêtre éditeur.

Si tout va bien, aucun message d'erreur ne doit apparaître dans la console et la zone de message doit afficher **DoneCompiling** attestant que la vérification s'est bien déroulée.

- **Sélectionner le bon port série (et la bonne carte Arduino...).**

Vous devez également sélectionner le bon port série depuis le menu **Tools > Serial Port** (Outils > Port Série) :

- Sur un Mac, le port série ressemble probablement à quelque chose comme /dev/tty.usbserial-1B1 (pour une carte USB), ou /dev/tty.USA19QW1b1P1.1 (pour une carte série connectée avec un adaptateur USB-vers-Série).
- Sous Windows, c'est probablement COM1 ou COM2 (pour une carte série) ou COM4, COM5, COM7 ou supérieur (pour une carte USB) - pour trouver le bon, vous pouvez chercher dans la rubrique des ports série USB dans la section des ports du panneau de configuration ou du gestionnaire de périphériques.
- Sous Linux, çà devrait être /dev/ttyUSB0, /dev/ttyUSB1 ou équivalent.

- **Clic sur le bouton UPLOAD (Transfert du programme)**



Une fois que vous avez sélectionné le bon port série et la bonne carte Arduino, cliquez sur le bouton **UPLOAD** (Transfert vers la carte) dans la barre d'outils, ou bien sélectionner le menu **File>Upload to I/O board** (Fichier > Transférer vers la carte). Avec les versions récentes (Duemilanove notamment), la carte Arduino va alors automatiquement se réinitialiser et démarrer le transfert. Avec les versions précédentes qui ne sont pas équipées de l'auto-réinitialisation, vous devez appuyer sur le bouton "reset" de la carte juste avant de démarrer le transfert.

Sur la plupart des cartes, vous devez voir les LEDs des lignes RX et TX clignoter rapidement, témoignant que le programme est bien transféré. Durant le transfert, le bouton devient jaune et le logiciel Arduino affiche un message indiquant que le transfert est en cours(Figure II.21) :

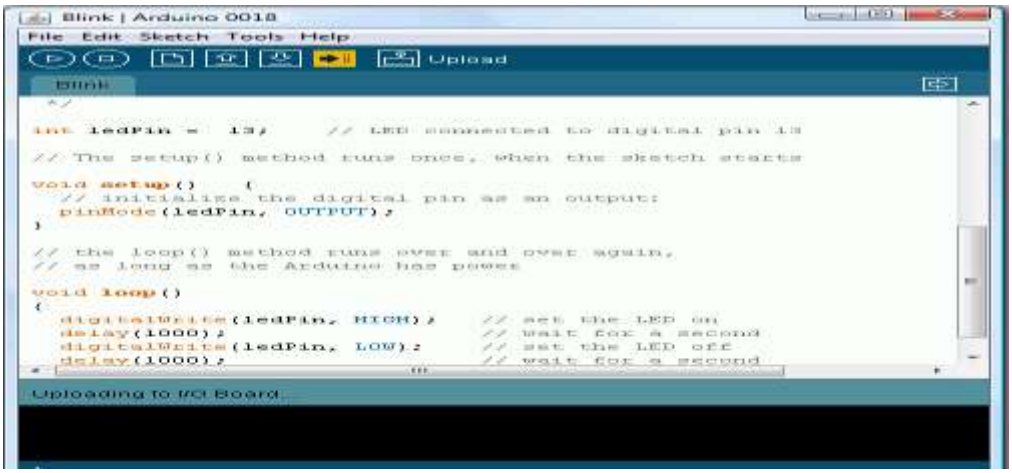



**Figure II.21 :** fenêtre de transfert vers la carte Arduino.

Une fois le transfert terminé, le logiciel Arduino doit afficher un message indiquant que le transfert est bien réalisé, ou montrer des messages d'erreurs (Figure II.22).

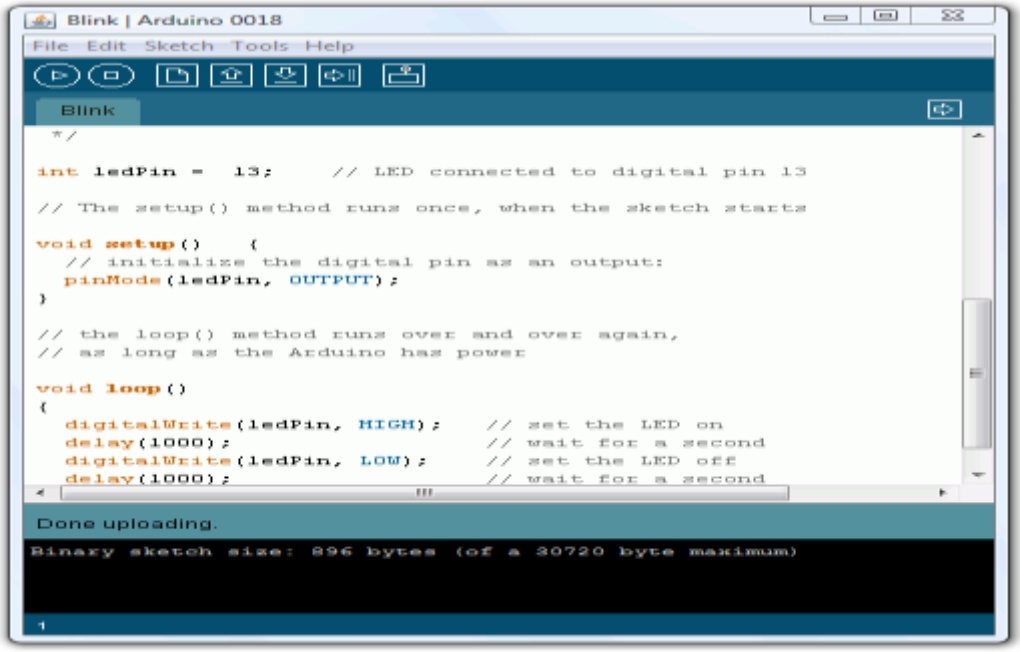



**Figure II.22 :** la fenêtre de transfert après le test.

- **Le programme transféré avec succès se lance**

Quand vous transférez un programme, en utilisant le bootloaderArduino, un petit programme (code binaire) a été chargé dans le microcontrôleur sur votre carte Arduino. Cette technique vous permet comme vous avez pu le voir de transférer votre programme sans aucun matériel externe. Une fois le transfert terminé, le bootloader est actif une petite seconde ("écoute" pour voir si un nouveau programme arrive...) une fois que la carte est réinitialisée à la fin du transfert; puis le dernier programme programmé dans la carte s'exécute.

### II.13.4Le Moniteur Série

Le logiciel Arduino intègre également un **TERMINAL SERIE** (fenêtre séparée) qui permet d'afficher des messages textes reçus de la carte Arduino et d'envoyer des caractères vers la carte Arduino.sur votre ordinateur, il faut ouvrir la fenêtre terminal de l'IDE Arduino : pour ce faire, un simple clic sur le bouton « Sérial Monitor »(Figure II.23) .




**Figure II.23 :**le fenêtre de l'IDE Arduino.

La fenêtre « Terminal » s'ouvre alors (Figure II.24) :

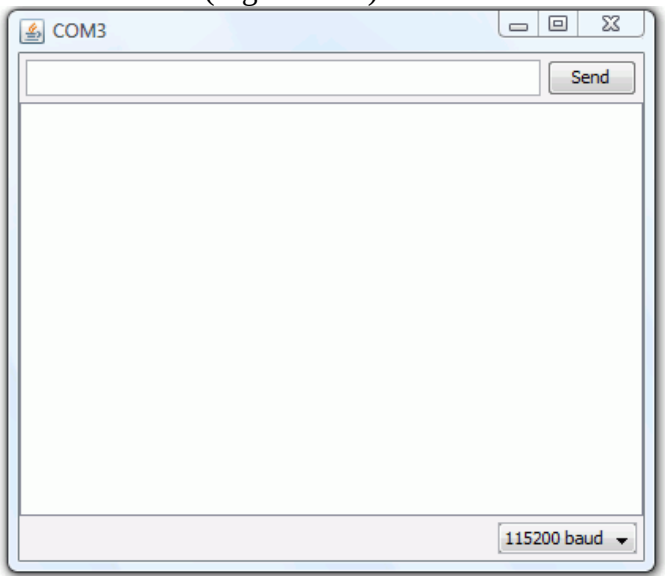



**Figure II.24 :**lafenêtre terminal.

Pour envoyer des données vers la carte, saisir le texte dans le champ de saisie et cliquer le bouton "send" (envoi) ou appuyer sur enter(Figure II.25) .

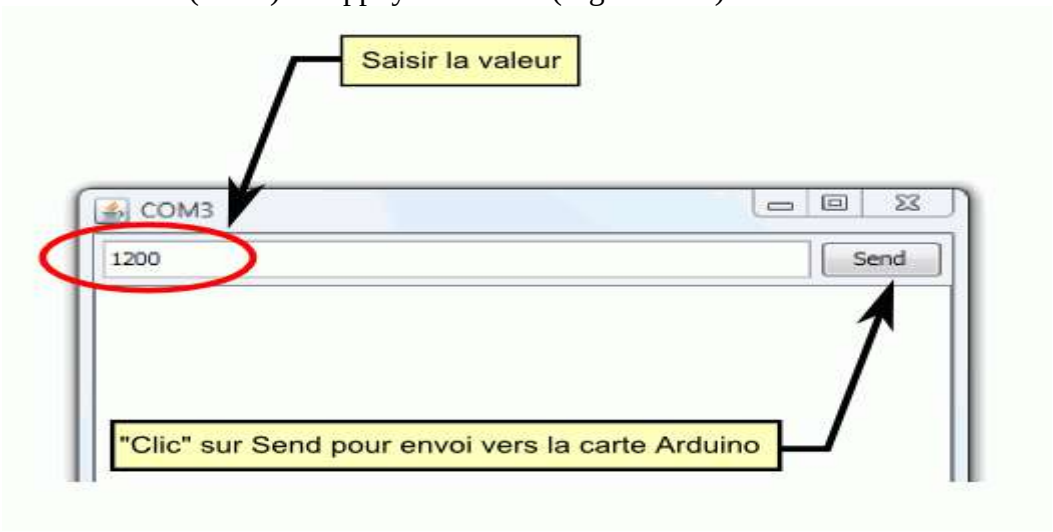



**Figure II.25 :**la fenêtre terminal.

Vous pouvez également communiquer avec la carte Arduino depuis l'interface graphique programmable Processing, Flash, MaxMSP, etc…

## II.14 Conclusion

Dans ce chapitre, une explication détaillée est donnée sur l'utilisation d'Arduino et ses Shields. Arduino est une bonne solution pour faire partie de la chaine de communication; d'une part il permet de relever les données issues des capteurs et détecteurs et d'autre part faire des actions tout en communicant à travers Ethernet en utilisant une application.

# Chapitre III

# *Réalisation pratique*

## III.1 Introduction

Après avoir décrit la partie matérielle de notre dispositif ainsi que la partie électronique et la conception de la carte de commande et contrôle du système, nous sommes arrivés à la partie la plus importante dans notre mémoire qui est la programmation de la carte Arduino ainsi que la réalisation d'une application sous Java. Dans ce chapitre, nous allons aborder cette programmation en passant par définition de cahier de charge.

Le bon fonctionnement de notre système se base essentiellement sur une bonne démarche et une bonne réflexion de notre programme.

## III.2 Définition du cahier de charge

Notre projet a pour but la conception et réalisation d'une application de contrôle et de commande à distance des salles dans un bloc d'administration.Chaque salle contient un dispositif de capteurs, détecteurs et actionneurs autour Arduinoqui contient un Shield Ethernet, ce dernier sert à relier la salle au réseau.

La gestion des salles se fait par un seul réseau dans le but de :

1. Diminuer le câblage
2. Faciliter l'utilisation
3. Économiser le cout
4. Détecter rapidement la panne

Dans notre application,on se limiter àune seule salle.Le système de la salle est gérée par un dispositif Arduino et ce dernier est gérée par l'applicationà travers le réseau l'Ethernet Comme indiqué dans la(figureIII.1)



SalleCablageArduino Ethernet Application sous Java

**Figure III.1** : schéma de communication

## III.3Description de la salle

La salle (Figure III.2) contient les installations comme les climatiseurs de refroidissement, les rideaux des fenêtres, les systèmes d'éclairage, de chauffage et d'aérations. Elle peut également avoir un capteur de température, le niveau de la lumière, le détecteur d'incendie et un détecteur de présence.

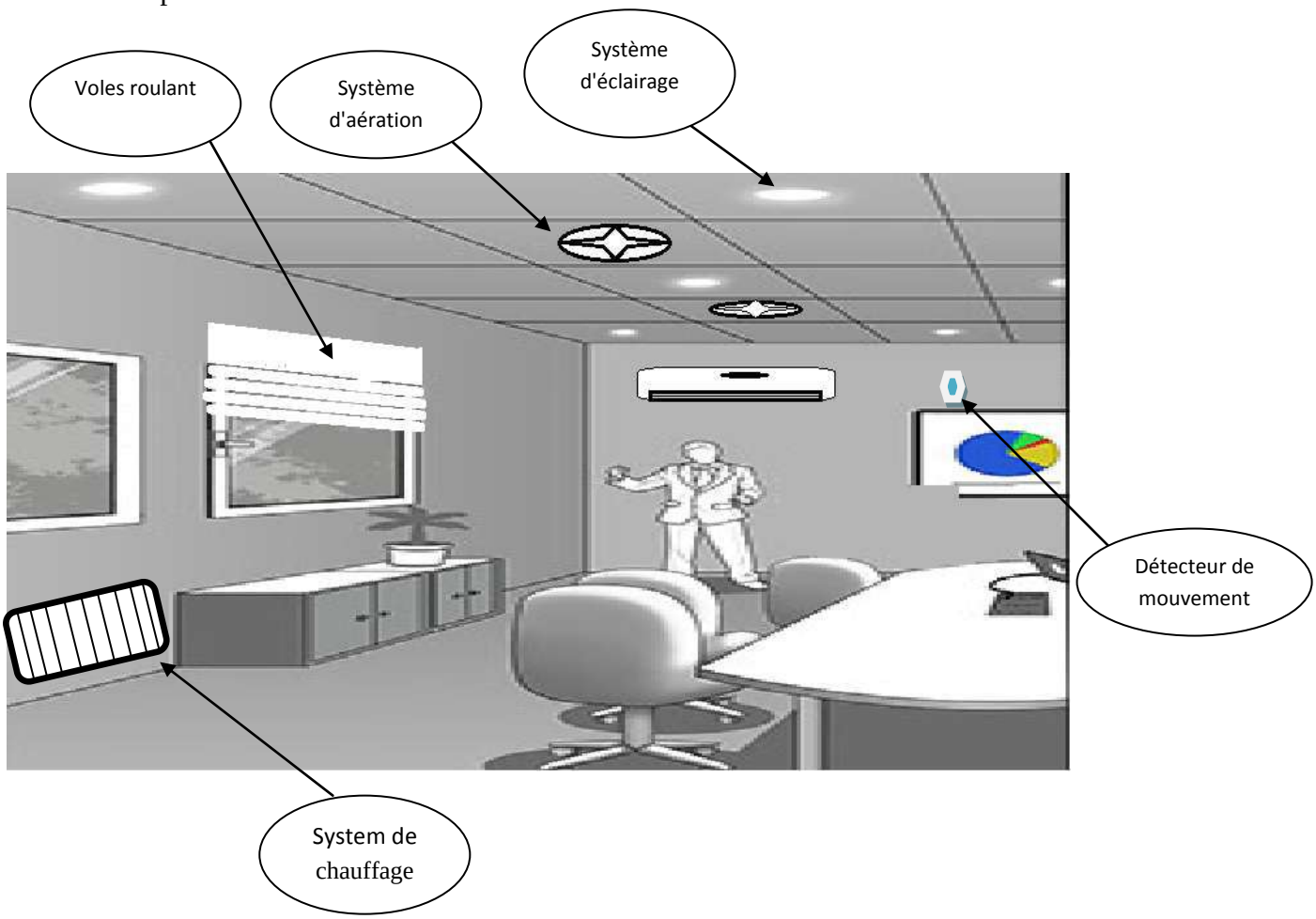



**Figure III.2** : Dispositifsexistants dans la salle

Dans notre projet, la salle a été représentée par petit prototype (Figure III.3)

Le prototype contient les dispositifs suivants

1. Deux capteurs
   A- capteur de température.
   B- capteur de lumière.
2. Deux détecteurs
   C-détecteur de mouvement
   D-détecteur d'incendie
3. Un Arduinouno avec un ShieldEthernet .
4. Une plaque de quatre relai pour la commande 220 v .

5. Deux lampesLED qui représente l'éclairage de la salle :
   E- La LED bleu elle s'allume par notre interface de commande.
   F-LaLED jaune elle s'allumelorsque le détecteur de mouvement détecté un mouvement.
6. Quatrelampes 220 V qui représentent des actions qu'ont faire dans la salle comme:
   G-Les extracteurs
   H-Les climatiseurs
   I-Le système de chauffage
   J-Les volets roulants
7. L'entrée de réseaux 220v
8. Alimentation 5 v pour Arduino

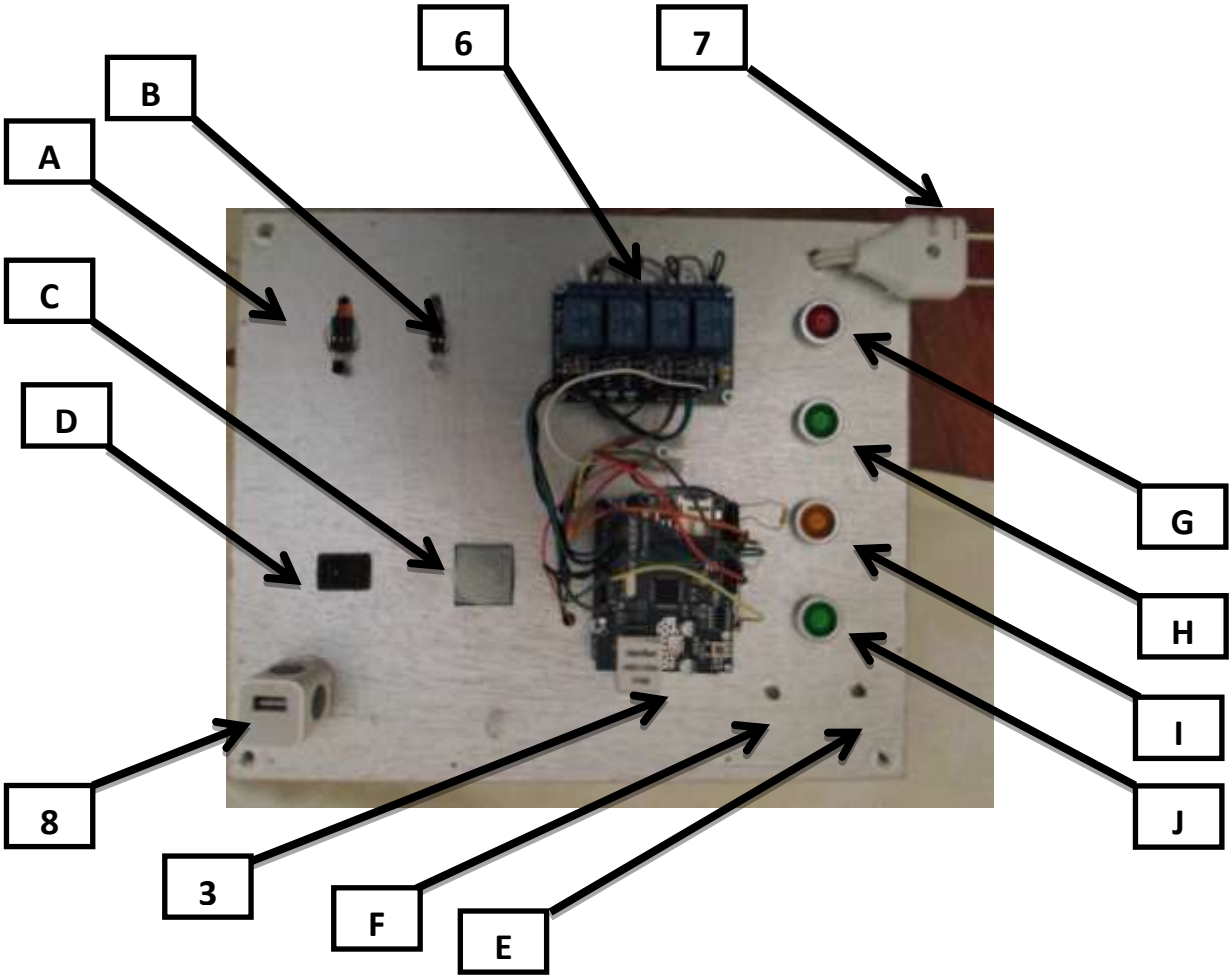



**Figure III.3** : la maquette

## III.4 Application de supervision et de contrôle

Une application a été développée sous Java pour le contrôle et la supervision de la salle. Pour se connecterau système de contrôle et supervisionon doit entrer un mot de passe qui est composé de trois chiffres « 1 2 3 » puiscliquer sur le bouton connexion comme l'indique la(figure III.4)




**Figure III.4** : l'interface de connexion

Lorsque La connexion est établie, une autre interface apparaitra (Figure III.5). On aura le message "Client Connected" et les états des détecteurs, capteurs et des bouton pour contrôler les actions, l'état de chacune des état peut être vu sur l'application

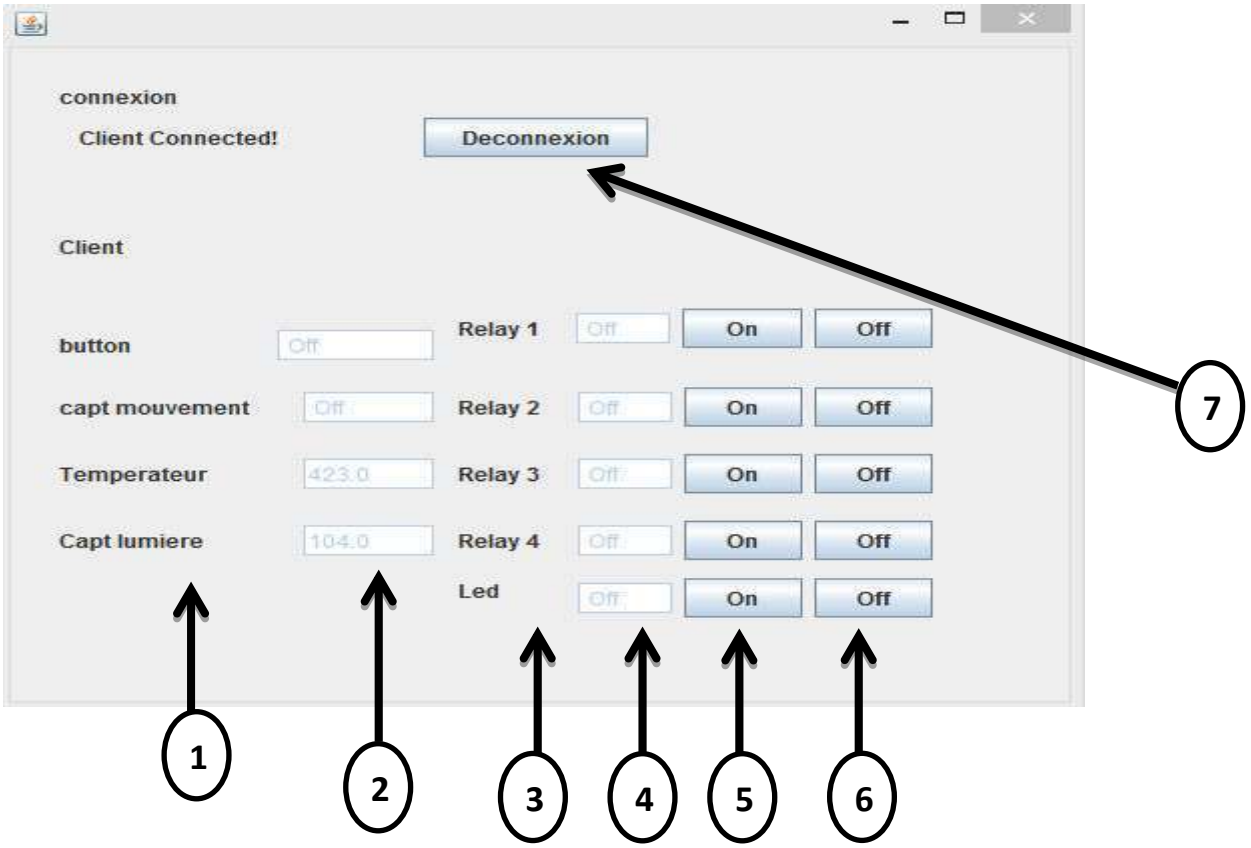



**Figure III.5** : l'interface supervision et le contrôle

**Description de l'application**

1. Type de capteurs et détecteurs (détecteur de mouvement, capteur de température, capteur de lumière , détecteur d'incendie)
2. Valeur qui affichée parles capteur et détecteurs(en °C pour la température, en Lux pour la lumière, et etat "ON" ou "OFF" pour les détecteurs )
3. Actionneurs(relais 1, relais 2, relais 3, relais 4 et une LED)
4. État de chaque actionneur "ON" ou "OFF".
5. Bouton "ON" pour activer l'actionneur.
6. Bouton "OFF"pour désactiver l'actionneur.
7. Bouton de déconnexion.

Le fonctionnement de chaque bouton fera l'objet d'une explication dans l'application.

## III.5 ContrôleparArduino

Le contrôle par Arduino se fait par câblage directe à travers des câbles de cuivre .chaque borne de dispositif sera reliéeàArduino comme suit :

- Capteur de lumièrenon polaire et contient deux pôles
  - Le premier relié au **VCC (5v)**
  - Le deuxième relié au pin analogique **ANALOGUE IN 1**
- Capteur de température polaire et contiens trois pôles
  - Le premier relié au **VCC (5v)**
  - Le deuxième relié au pin analogique **ANALOGUE IN 2**
  - Le dernier relié au **GND**
- Détecteur de mouvement polaire et contiens trois pôles
  - Le premier relié au **VCC (5v)**
  - Le deuxième relié au pin**IN8**d'Arduino
  - Le dernier relié au **GND**
- Détecteur d'incendie (remplacé par un simple interrupteur de deux pôles )
  - Le premier relié au **VCC (+5V)**
  - Le deuxième relié au pin**IN7**d'Arduino
- La carte des relais contient deux pôles d'alimentation et quatre pôles pour chaque relai
  - Le premier relié au **VCC (5v)**
  - L'IN1est relié au pin**IN2**d'Arduino
  - L'IN2est relié au pin**IN3**d'Arduino
  - L'IN3est relié au pin**IN4**d'Arduino
  - L'IN4est relié au pin **IN5**d'Arduino
  - Le dernier est relié au **GND**
- La LED bleu polaire et contient deux pôles
  - **L'anode**est reliée au pin**6**de l'Arduino

- **La cathode est** reliéeau **GND**

- La LED jaune polaire et contient deux pôles
  - **L'anode**reliée au pin**9**de l'Arduino
  - **La cathode** reliée au **GND**

Exemple de fonctionnement de la LED jaune

Lorsque le détecteur de mouvement détecte un mouvement, il envoie un signal via le l'entrée numéro 8 de l'Arduino qui est détecté par l'Arduino.Qui à son tour envoie un signal sous la forme d'une tension de 5 volts au pin numéro 6de l'Arduino pour allumer la LED

## III.6 contrôle par application (java / Arduino)

### III.6.1 Serveur :

Le serveur java initialise la connexion il lance l'écoute sur un port et se met en attente des connexions entrantes pour qu'il l'accepte. Java fournit un package java.net qui traite tout ce qui est réseau, on a besoin seulement de deux classes :

**Java.net.serversocket :** cette classe accepte les connexions venues des clients.

**Java.net.socket :** cette classe permet de se connecter à la machine distante.

On a besoin aussi d'un outil pour saisir, envoyer et recevoir le flux :

**Scanner :** lire les entrées de claviers.

**Bufferedreader :**lire le texte reçuà partir de l'émetteur.

**Printwriter :**envoyer le texte saisi.

### III.6.2 Client :

Le client n'a besoin qu'à la classe socket pour établir la connexion serveur, le constructeur prend en entrée l'adresse IP du serveur et le numéro de port.

### III.6.3 Les sockets :

Pour programmer une application client/serveur, il est recommandé d'utiliser les sockets. Les sockets fournissent une interface qui permet d'utiliser facilement leprotocoles de transport TCP/IP

## III.6.4 Shield Ethernet

LeShieldArduinoEthernetest un module Arduinoadditionnel aux cartesUno**,** qui permet de les rendrecommunicantessur unréseau filaireEthernet**.**

On pourra ainsi créer uneInterface Homme Machine **(**IHM**)** pourpiloter à distanceouvisualiserl'état de notrecarte Arduinoen utilisant uncâble réseaurelié à un réseau local ou Internet**..**

### III.6.4.1 Constitution du ShieldArduino Ethernet

Le module Ethernet (Figure III.6) est composé de :

• Un Port Ethernet (10BaseT/100BaseTX).

• Une circuit intégré Wiznet W5100 supportant le protocole TCP/IP et 4 connexions en même temps.

• Un emplacement pour carte Micro SD pour y stocker ou envoyer des données.

• Des borniers embrochables déportés de votre carte Arduino.

• 7 LEDs permettant d'afficher les informations :

- TX - s'allume lors d'une transmission de données.
- RX - s'allume lors d'une réception de données.
- COLL - s'allume lors d'une collision sur le réseau.
- FULLD - s'allume lorsque la connexion réseau est en Full duplex.
- LINK - s'allume lorsqu'un réseau est connecté et clignote lors de l'échange de données.
- 100M - s'allume lorsque la connexion est en 100Mbits/s.
- PWR - s'allume lorsque l'alimentation de la carte est présente.

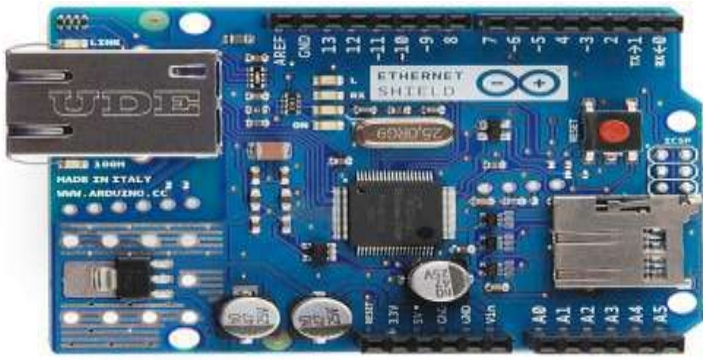



**Figure III.6 :** le module shield Ethernet.

### III.6.4.2 La librairie Ethernet, pour se connecter à internet ou réaliser des réseaux filaires Ethernet avec Arduino

Avec le module Arduino Ethernet, cette librairie permet de connecter une carte Arduino à un réseau Ehternet ! Ceci peut aussi bien être utilisé pour créer un serveur acceptant des connexions entrantes ou un client faisant des requêtes sortantes. Cette librairie supporte jusqu'à 4 connexions simultanées (sortantes, entrantes ou une combinaison des deux).

**Les classes de la librairie**

La librairie Ethernet regroupe 5 classes :

- la classe Ethernet : classet racine permettant l'initialisation du shield Ethernet
- la classe IPAddress : classe représentant une adresse IP au format xxx.xxx.xxx.xxx
- la classe EthernetServer : classe regroupant toutes les fonctions utiles pour l'utilisation du shieldethernet en tant que serveur réseau
- la classe EthernetClient: classe regroupant toutes les fonctions utiles pour l'utilisation du shieldethernet en tant que client réseau

**Les instructions de la librairie Ethernet**

La classe Ethernet initialise la librairie ethernet et les paramètres du réseau. Réalise l'initialisation matérielle du shield Ethernet.

- Le constructeur de la classe Ethernet est implicite
- begin()
- localIP()
- maintain()

- **LA CLASSE IPADDRESS**
  - IPAddress

### III.6.4.3 INSTRUCTIONS DE LA CLASSE ETHERNETSERVER

La classe Server crée des serveurs qui envoient des données vers et reçoivent des données depuis des clients connectés (programmes s'exécutant sur d'autres ordinateurs ou composants).

- Constructeur Server() / EthernetServer()
- begin()
- available()
- write()
- print()
- println()

### III.6.4.4 INSTRUCTIONS DE LA CLASSE ETHERNETCLIENT

La classe Client crée des clients qui peuvent se connecter à des serveurs et envoyer ou recevoir des données.

- Constructeur Client() / EthernetClient()
- if (EthernetClient)
- connected()
- connect()
- write()
- print()
- println()
- available()
- read()

- flush()
- stop()

### III.6.5 Principe d'utilisation

Pour inclure la librairie Ethernet dans un programme, on ajoutera au début du programme les lignes suivantes (la librairie Ethernet nécessite la librairie SPI) :

```
#include <SPI.h>
#include <Ethernet.h>
```

Pour ce faire, le plus simple est d'utiliser le menu du logiciel Arduino Programme > Importer Librairie > Ethernet

Voir également l'instruction #include.

## III.7 La communication java /Arduino :

L'Arduino(le client)se connecte à l'interface (serveur) sur le port 8088à travers l'adresse IP en deux types de communication (figureIII.7 ):



**Figure III.7:** connexion serveur/client

- Demande l'état.

Dans ce cas le serveur envoie les sockets vers le client pour faire le dialogue selon un protocole de communication qui contient descaractères (1 jusqu'àa 9) pour connaitre l'état de chaque capteur ou détecteur; le client (Arduino) vérifie chaque capteur et détecteurpour réennoyer son état au serveur pour l'afficher.

Exemple :

if(c=='4') {client.println("4"+a(led));

Le serveur demande au client l'état de la LED, le client répondessoit ON ou OFF.

- Commande

  Après avoir vérifié l'état de chaque dispositif, leserveur peutdemander au client un changementd'états lorsquel'utilisateurappuie sur l'un des deux boutons (On ou OFF) pour chaque actionneur, s'il appuie sur le bouton"ON", le client reçoituncaractère majuscule sinon le client reçoituncaractèreminuscule, dans ce cas le client ne répond pas au serveur, ilne fait que l'exécution

La mise ajour de chaque état se fait par le serveur sur une boucle infinie de 50ms.

## III.8Quelques tests expérimentaux

Pour vérifier et valider notre application, il faut d'abord alimenter la maquette et la relierau terminal qui contient l'application à travers un câble réseau RJ45

- Brancher la prise d'alimentation (220v).
- Brancher l'ardin a l'alimentation (5v)
- Lancer le programme Arduino (client) et téléverser vers la carte Arduino (se fait une seule fois)
- Placer le Schield Ethernet sur la carte Arduino
- Relier le Schield au PC en utilisent le câble RJ45
- Lancer le serveur et saisir le mot de passe pour connecter au client

- **Test1 : Allumez la LED bleu par l'application**

Après la connexion de notre client Arduino avec le serveur (application)nous verrons cette interface où la lampe n'est pas allumée et l'interface (figure III.8)

Coté serveur Coté client




**Figure III.8:** Test 1: avant le test.

Lorsqu'on clique sur le bouton ON de la LED,la LED s’allume et son"ON" correspondant s'activepour montrer son état (Figure III.9)

Coté serveur Coté client

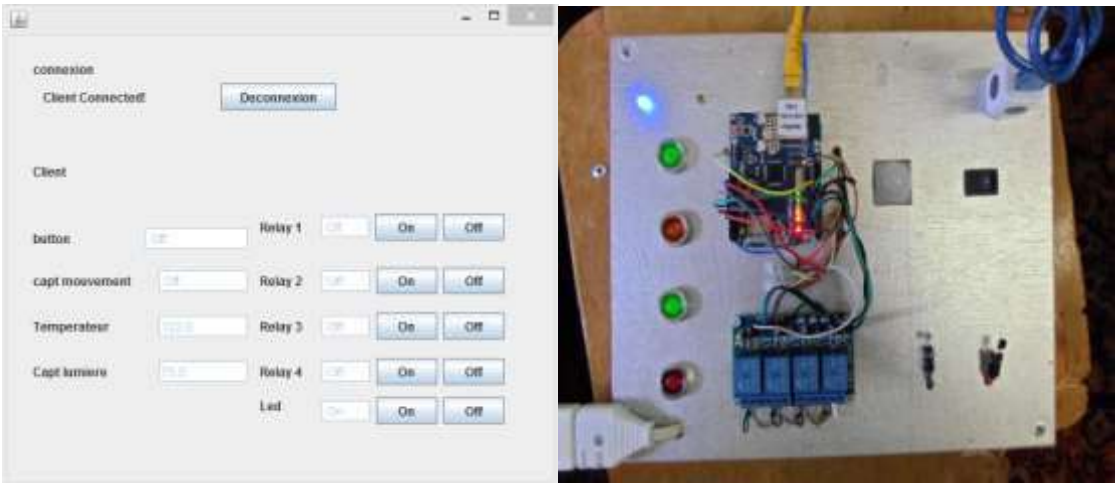



**Figure III.9:** Test 1:après le test .

- **Test 2: Détection de mouvement**

Avant de détecterunmouvement, l'état du détecteur correspondant est indiqué par "OFF" dans l'interfaceet la LED jaune n'est pas allumée (Figure III.10)

Coté serveur Coté client




**Figure III.10:** Test 2: avant le test.

Lorsque le détecteur détecte un mouvement son, état affiche "ON" et la LED jaune s'allume (Figure III.11)

Coté serveur Coté client

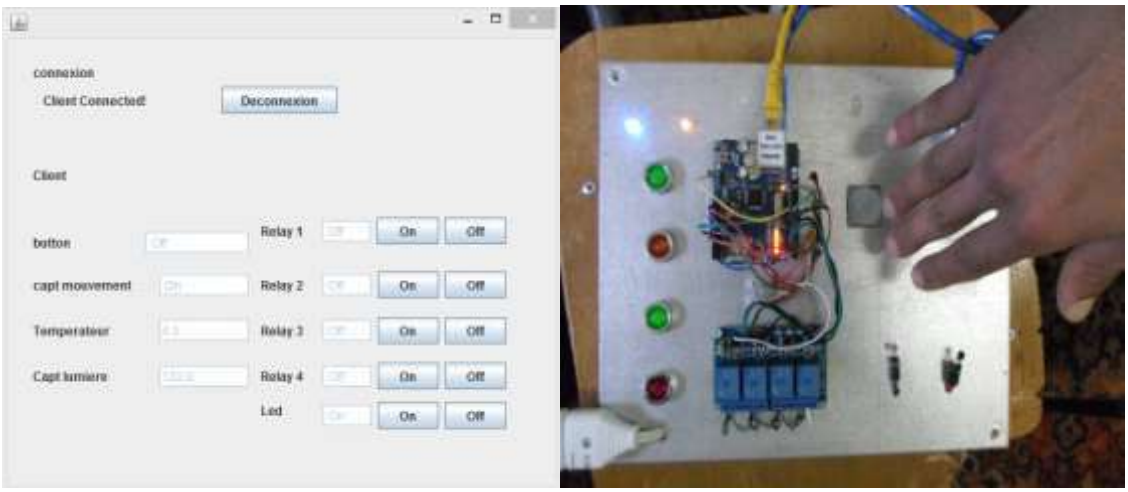



**Figure III.11:** Test 2: après le test .

- **Test 3: Commande d'un actionneur**

L'État de relai est OFF avant le test. Lorsqu'on clique sur le bouton "ON" le relai est activé et ainsi la lampe correspondante s'allume(Figure III.12)

Coté serveur Coté client




**Figure III.12:** Test 3: après le test.

- **Test 4: Capteur de l'intensitélumineuse**

Lorsqu'on cache le capteur, il affiche la valeur (5 Lux) comme indique (figure III.13).

Coté serveur Coté client




**Figure III.13:** Test 4: en cachant le capteur de lumière

Si on projette la lumière sur le capteur, la valeur 479 Lux apparait dans l'interface(figure III.14).

Coté serveur Coté client

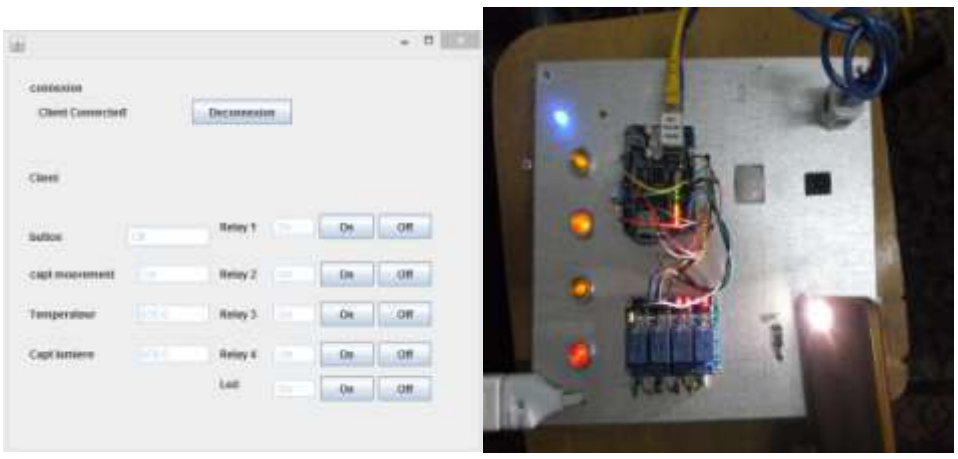



**Figure III.14:** Test 4: avec projection de la lumière.

## III.9 Conclusion

Dans ce travail, nous avons présenté un système de contrôle et de commande à distance très pratique simple et souple à réaliser par Arduino avec un Shield Ethernet. Une application Java est réalisée et installée dans un terminal qui est lié au réseau de communication. À travers cette application, on peut superviser et contrôler tous les dispositifs.

On a pu obtenir des résultats satisfaisants compte tenu des limitations du matériel et des moyens dont nous disposons

# Conclusion Générale

Dans notre projet, on a réussi à réaliser un système de communication Ethernet à base d'Arduino et une application développée sous Java. Ce système de communication est appliqué pour la supervision et contrôle d'une salle (pris comme exemple). À travers l'application, on peut visualiser tous les paramètres de la salle détectés ou captés et on aussi faire activer des actionneurs tel que le : chauffage, climatisation, lumière, …

Pour cela on du tout d'abord maitriser le contrôle local en utilisant Arduino puis créer uneapplication Java serveur en utilisant un module Ethernet quipermetdecréerunréseau local entre les différents organes du système de contrôle.

Ce système est très bénéfique vu qu'il permet de diminuer le cout en consommant considérablement beaucoup moins du fil de cuivre qu'en utilisant la communication classique, facile à utiliser et permet aussi de détecter la panne rapidement

Pourcequiestdesperspectivesdecontinuationplusieursvoiespeuventêtre envisagées.Àtitreindicatif, nous conseillons d'examiner et de développer les points suivants :

- DévelopperuneapplicationAndroidaulieudecelleréaliséedansceprojetdetellefaçonà enrichir le système de contrôle avec d'autres fonctions domotiques.
- Piloterladomotiqueàpartirdesdonnéesreçuesd'unréseaudomotiqueetd'unsystèmed'anal ysesonore ayant la capacité de reconnaître les ordres vocaux. Les actions exécutées parle contrôleur se produisent en réponse aux commandes vocales prononcées par l'habitant ou de manière proactive pour assurer sa sécurité et son confort.
- Danslecasdelareconnaissancedelasituationdel'habitant,connaîtrelapositiondela personne et sa localisation précise dans une salle.
- RéaliserlecontrôledomotiqueenutilisantunmoduleGSMquipermetd'augmenterla possibilité de surveiller l'habitat tout en communiquant via des commandes envoyées par SMS.
- Ajouterunefonctiondedétectiond'intrusiontoutenutilisantunecaméradesurveillanceetave c un traitement d'images pour la détection facial des personnes.

# Bibliographie

**Thèse**

[1] Hamid_Hamouchi,' Conception et réalisation d'une centrale embarquée de la domotique Smart Home',univmohamed V de rabat,2015.

[2]L'innovation invisible dans les transports publics,Le cas du bus électrique à la RATPL'innovation invisible dans les transports publics - Master PIC, Août 2017

**Les sites web**

[3] Bus de terrain, http://dictionnaire.sensagent.leparisien.fr/Bus%20de%20terrain/fr-fr/, consulté le 10 mars 2018.

[4][PDF]Les bus et les réseaux de terrain en automatisme ... - robertponge

,https://www.robertponge.com/telechargements/ebooks/bus.pdf,consulté *le 10 mars 2018*

[5]Document Ressource : Le Bus CAN, http://lewebpedagogique.com/isneiffel/files/2017/05/Bus-CAN.pdf, consulté le 20mars 2018.

[6]Modbus,http://projet.eu.org/pedago/sin/term/8-Modbus.pdf,consultéle 15mars 2018

[7]Modbus,http://projet.eu.org/pedago/sin/term/8-Modbus.pdf,consulté le 15mars 2018.

[8]Cours KNX/EIB - Automation Sense,https://www.automation-sense.com/blog/domtique-objets-connectes/cours-knx-eib.html, consulté le 12mars 2018

[9]19.2 Bus AS-I -IA Portal ,http://tia-portal.weebly.com/192-bus-as-i.html, consulté le 10mars 2018

[10]Profibus, Dictionnaire, http://dictionnaire.sensagent.leparisien.fr/Profibus/fr-fr/, consulté le 10 mars 2018

[11]Ethernet, http://tvaira.free.fr/bts-sn/reseaux/cours/cours-reseaux-ethernet.pdf,consulté le 11 avril 2018.

[12]Shield Arduino Ethernet raccordement et création d'un serveur HTTP,http://electrotoile.eu/arduino_shield_ethernet.php.consulté le 02 avril-2018.