

République Algérienne Démocratique et Populaire

Ministère de L'enseignement Supérieur et de la Recherche Scientifique

Université Mouloud Mammeri De Tizi-Ouzou



Faculté De Génie Electrique Et D'informatique
Département D'automatique

Mémoire De Fin D'étude De MASTER ACADÉMIQUE

Domaine : Sciences et Technologies

Filière : Génie électrique

Spécialité : Commande Des Systèmes

Présenté par

Mlle INGUEL MALHA

Thème

Conception et réalisation d'un système

Domotique par GSM

Mémoire soutenu publiquement le 17/09/2017 devant le jury composé de :


TOUAT Mohand Achour Maitre conférence Encadreur

U.M.M.Tizi-ouzou

À cœur vaillant rien d'impossible
À conscience tranquille tout est accessible
Quand il ya la soif d'apprendre
Tout vient à point à qui sait attendre
Quand il ya le souci de réaliser un dessein
Tout devient facile pour arriver à nos fins
Malgré les obstacles qui s'opposent
En dépit des difficultés qui s'interposent
Les études sont avant tout
Notre unique et seul atout
Ils représentent la lumière de notre existence
L'étoile brillante de notre réjouissance
Comme un vol de gerfauts hors de charnier natal
Nous partons ivres d'un rêve héroïque et brutal
Espérant des lendemains épiques
Un avenir glorieux et magique
Souhaitant que le fruit de nos efforts fournis
Jour et nuit, nous mènera vers le bonheur fleuri.



Remerciements



Avant tout, je remercie le bon dieu, qui m'a donnée le courage, la volonté, la patience et la santé nécessaire pour affronter les différentes épreuves de la vie durant mes années d'études.

Au terme de mon projet de fin d'études, j'exprime ma profonde gratitude à Monsieur le Doyen de la Faculté de Génie Électrique Et d'Informatique, mes enseignants et tout le cadre administratif pour leurs efforts considérables, spécialement ceux du département Automatique.

Mon vif remerciement à Mr TOUAT Mohand Achour, mon promoteur, sans qui, ce mémoire n'aura pas eu lieu. Je le remercie aussi pour son soutien scientifique et hors scientifique et la confiance qu'il m'a accordée depuis que je suis au département d'automatique. Je me souviendrai toujours de lui comme un homme patient, judicieux directive avec les étudiants. En fin, rien de ce que j'écrirai ici ne sera suffisant pour lui exprimer ma gratitude.

Ma gratitude s'adresse également à Mr MOULOUDJ Kamel pour son aide très consistant ainsi que pour l'intérêt avec lequel il a suivi la progression de mon travail, pour ses conseils efficaces, ses judicieuses directives et pour les moyens qu'il a mis à ma disposition matérielles et morales pour la réussite de ce travail tout au long de ma période de projet.

Je remercie, également les membres de jury d'avoir accepté d'évaluer mon travail.


Mes sincères remerciements vont aussi à tout le personnel de l'école privée 2INT pour leur soutien et leur encouragement.

Sans oublier Ouchaoua Nadir et lamrache Massi pour leur aide précieux.

Ma gratitude va tout particulièrement aussi à mes sœurs, mon frère, mes chers parents, qui ont été toujours derrière moi et sans eux je ne serai pas ce que je suis devenu.

Tiguercha Rafik bien évidemment, pour son amour sans limites, son calme et sa sérénité très réconfortants tout au long de mes études, pour sa gaieté et sa joie de vivre permanentes, pour ses idées remarquables aussi et notre complicité primordiale. Merci Rafik de m'avoir pleinement "secondé" dans cette grande aventure. Famille, amis et connaissances enfin, pour tous vos petits mots toujours très encourageants.

Vers la fin, il m'est très agréable d'exprimer toutes ma reconnaissance pour ceux qui m'ont entouré de près ou de loin pendant mes années études pour leur soutien, leur aide et, surtout, pour leur sympathie. Qu'ils trouvent ici l'expression de ma profonde reconnaissance et mon profond respect.





Dédicaces

A la mémoire de mon grand-père que je n'ai jamais connu.

A la mémoire de ma grand-mère qui vient de nous quitter.

A mes chers parents qui ont tant donné
Pour leur immense soutien, leur grand amour, leur sacrifices
Et leurs prières.

Qu'ils acceptent ici l'hommage de ma gratitude, qui, si grande qu'elle
puisse être, ne sera jamais à la hauteur de leur tendresse et leur
dévouement.

A mon cher frère Tahar

A mes chères sœurs Ouerdia et son mari Djamel, kheloudja, Dyhia,
Vous aviez toujours cru en moi, et c'est dans votre présence que j'ai
puisé la volonté de continuer.

A Rafik et sa famille

A toute ma famille

A toutes mes chères amies et à tous mes chers amis

A tous ceux qui m'ont aidé de près ou de loin

Pour tous vos petits mots toujours très encourageants.

Sommaire

Introduction générale

Chapitre I : Introduction à la domotique par GSM

I-1 Introduction	1
I-2 Historique de la domotique.....	2
I-3 Définition et principe de fonctionnement.....	2
I-3-1 Les capteurs.....	3
I-3-2 Actionneurs	4
I-3-3 Relation capteurs actionneurs.....	4
I-4 Les technologies utilisées	5
I-4-1 La technologie bus filaire.....	5
I-4-2 Le courant porteur en ligne (CPL)	6
I-4-3 La radiofréquence.....	7
I-5 Les fonctions de la domotique.....	7
I-6 Système automatisé	10
I-6-1 Définition	10
I-6-2 Les composants d'un système automatisé.....	10
I-6-3 Représentation simplifiée d'un système automatisé.....	12
I-7 Le GSM	13
I-7-1 Historique.....	13
I-7-2 Présentation	14
I-7-3 Architecture de GSM	15
I-7-3-1 Le sous-système radio (BSS).....	15
I-7-3-2 Le sous-système réseau (NSS).....	16
I-7-3-3 Le sous-système exploitation OSS (OperationSub-System)	16
I-8 Principe de fonctionnement de la domotique par GSM	17
I-8-1 Commande par GSM.....	17
I-8-2 Surveillance par GSM	17
I-9 Conclusion.....	17

Chapitre II : Description de la partie matérielle et logiciel

II-1 Introduction	19
-------------------------	----

II-2 Structure du système	19
II-3 Description de la partie matérielle	20
II-3-1 la carte Arduino UNO	20
II-3-1-1 Présentation de la carte Arduino UNO	20
II-3-1-2 Caractéristiques de la carte UNO	21
II-3-1-3 Les différents composants de la carte UNO	22
II-3-2 Bluetooth (HC-05)	28
II-3-3 Capteur de température (LM 35)	28
II-3-4 Capteur ultrason (HC-SR04).....	31
II-3-5 Capteur de Gaz (MQ 135).....	33
II-3-6 Servomoteur	35
II-4 Description de la partie logicielle.....	36
II-4-1 Le logiciel Arduino	36
II-4-1-1 Présentation de «IDE »	37
II-4-1-2 Programmer avec Arduino.....	38
II-4-1-3 Syntaxe des commandes Arduino.....	40
II-4-2 Fritzing.....	41
II-4-3 Le système Androïde	41
II-4-4 APP INVENTOR.....	42
II-4-4-1 Historique de logiciel App Inventor	43
II-4-4-2 Un commencement avec APP INVENTOR.....	43
II-5 Conclusion.....	51

Chapitre III : Description de la partie matérielle et logiciel

III-1 Introduction.....	52
III-2 Réalisation du système.....	52
III-2-1 Les organigrammes de notre système.....	53
III-2-2 Présentation des programmes sur IDE.....	58
III-2-3 Quelques simulation virtuelle des composants sous Fritzing.....	62
III-2-4 Réalisation de l'application APP INVENTOR (Androïde).....	63
III-3 Conclusion	66
Conclusion générale	

Liste des figures :

- Figure I-1** : Exemple d'une maison intelligente
- Figure I-2** : Circulation des informations dans la domotique
- Figure I-3** : Relation capteurs actionneurs
- Figure I-4** : Installation domotique bus filaires
- Figure I-5** : Installation CPL
- Figure I-6** : Les fonctions de la domotique
- Figure I-7** : Exemple d'un système automatisé
- Figure I-8** : Architecture d'un réseau GSM
- Figure II-1** : Schéma bloc de notre système
- Figure II-2** : La carte Arduino UNO
- Figure II-3** : Les composants de la carte Arduino UNO
- Figure II-4** : Microcontrôleur ATmega328
- Figure II-5** : Datasheet d'ATmega328
- Figure II-6** : Constitution de la carte Arduino UNO
- Figure II-7** : Vue du Module Bluetooth HC-05
- Figure II-8** : Vue du Capteur LM35
- Figure II-9** : Capteur Ultrason HC-SR04
- Figure II-10** : Principe de fonctionnement d'un capteur Ultrason HC-SR04
- Figure II-11** : Capteur de gaz MQ-5
- Figure II-12** : Vue d'un Servomoteur.
- Figure II-13** : l'angle de l'axe du servomoteur en fonction de la largeur de l'impulsion
- Figure II-14** : L'interface de Fritzing
- Figure II-15** : Différentes fenêtres d'APP INVENTOR
- Figure II-16** : Création de nouveau projet sur APP INVENTOR
- Figure II-17** : Première interface de la création APP INVENTOR
- Figure II-18** : Résumer des quatre étapes
- Figure II-19** : En-tête d'éditeur de blocks APP INVENTOR
- Figure II-20** : Éditeurs de blocks APP INVENTOR
- Figure II-21** : Échantillon d'un composant sous APP INVENTOR
- Figure II-22** : L'organisation générale de l'interface de développement
- Figure III-1** : Schéma synoptique de la communication Bluetooth-Android
- Figure III-2** : Programme principale
- Figure III-3** : Organigramme d'initialisation
- Figure III-4** : Organigramme Reception serie
- Figure III-5** : Organigramme gestion des lampes
- Figure III-6** : Organigramme gestion de porte
- Figure III-7** : Organigramme d'événements
- Figure III-8** : Sous programme Reception serie
- Figure III-9** : Sous programme gestion des lampes
- Figure III-10** : Sous programme de gestion de porte
- Figure III-11** : Sous programme de détection de Voleur
- Figure III-12** : Sous programme d'événements
- Figure III-13** : Sous programme d'initialisation
- Figure III-14** : Simulation de capture de température LM35
- Figure III-15** : Simulation de capture Ultrason HC-04
- Figure III-16** : Simulation d'un Servo Moteur
- Figure III-17** : Simulation de capture de gaz
- Figure III-18** : Choix du mode de commande
- Figure III-19** : Les différentes interfaces de l'application (Mode local)

Bibliographie

- [1] www.blog.univers-domotique.com/2014/06/la-domotique-40-ans-d-evolution-pour-transformer-un-vieux-reve-en-realite. Consulter le : Aout/2017
- [2] www.knx-automation.com/domotique/les-fonctions-domotiques-de-la-maison-intelligente
- [3] Boudellal, Méziane smart home [Texte imprimé] : habital connecte, installations domotiques et multimédia. Consulter le : Aout/2017
- [4] www.maison-et-domotique.com/10902-les-avantages-de-la-domotique-en-tete. Aout/2017
- [5] www.sirlan.com/livreblanc.pdf Consulter le : Aout/2017
- [6] BEN LOUNES Saïd, AHNOUNE Djamel « Conception et réalisation d'un système de vidéo- surveillance à base de la carte ARDUINO UNO>>, Mémoire fin d'étude à L'UMMTO, BENNAMENE.K 2014.
- [8] www.forumatena.org/files/livresblancs/les%20réseaux%20mobiles%20-%20GSM.pdf
- [09] C. Tavernier, « Arduino applications avancées ». Version Dunod.
- [10] <http://www.acm.uiuc.edu/sigbot/tutorials/2009-11-17-arduino-basics>. Consulter le: mars 2015.
- [11] <http://www.generationrobots.com/fr/152-arduino>. consulter le : mars 2015.
- [12] S.V.D.Reyvanth, G.Shirish, « PID controller using Arduino »
- [13] <http://fr.flossmanuals.net/arduino/index>
- [14] T.Karvinen, K.Karvinen, V.Valtokari « Les capteurs pour arduino et Raspberry Pi » DUNOD, 2014
- [15] Lucien Bachelard, User'sManual du HC-SR04/www.manuel-esteban.com, Le 28 novembre 2015
- [16] Hippolyte Weisslinger (olyte), Landrault (Eskimon) Arduino : Premiers pas en informatique embarquée, le Blog d'Eskimon Edition du 19 juin 2014
- [17] Simon Monk, Fritzing for inventors : Take Your Electronics Project from Prototype to Product by Simon Monk, McGraw-Hill Education TAB ; 1 edition, 2015.
- [18] www.fritzing.org Consulter le juin/2017
- [19] www.Appinventor.mit.edu

[20] www.itechnofrance.wordpress.com consulter le : Juillet/2017

[21] <http://www.ti.com/lit/ds/symlink/lm35.pdf>consulter le : Juillet/2017

[22]www.gotronic.fr/ultrason consulter le : Juillet/2017

[23] www.eduscol.education.fr consulter le : Aout/2017

[24]<http://itechnofrance.wordpress.com/2013/05/05/utilisation-du-servomoteur-sg90-avec-l-arduino/>

Introduction générale

Introduction générale :

La technologie a investi plusieurs domaines de la vie. En effet les habitations du futur répondent à une probable insatisfaction innée de l'homme qui croit augmenter son pouvoir sur les choses par la technique. On voit donc que sa maison «s'adapte» à lui et à ses besoins.

Au même titre que nous gérons notre budget, organisons notre temps et nos activités, la domotique nous offre la possibilité d'opter pour une maison qui s'adaptera à notre rythme de vie et à nos habitudes. Avec les nouvelles technologies domotiques, l'homme peut quitter son domicile pour le week-end et avant de fermer la porte, il appuie sur un seul bouton pour éteindre les éclairages oubliés, commander et contrôler à distance des machines, des systèmes d'alarme et de surveillance, de commander des portes et des fenêtres ou d'allumer des lampes...

Il est possible de contrôler et de commander des systèmes à distance en ayant recours au réseau GSM. Pour répondre à cette évolution majeure, nous avons créé une carte qui permet de contrôler n'importe quel appareil ou machine à travers un smart phone ou une tablette.

Le smart phone occupe la première place d'objets qui ne nous quittent pas donc notre travail se concentre sur l'utilisation de ce dernier avec bien sur sa liaison avec un système ou une carte de commande, telle que l'Arduino.

En effet, la «maison intelligente » répond à des besoins, elle facilite notre quotidien mais elle peut aussi créer des besoins. Plus nous faisons appel à la technologie, plus nous en sommes tributaires : elle nous rend de plus en plus dépendant d'elle en nous donnant l'illusion de nous rendre plus libres car il devient difficile de s'en passer.

Le but de cette réalisation est d'avoir le contrôle sur les différents éléments de la maison et d'offrir le confort, la sécurité et l'économie d'énergie pour le propriétaire. La différence est dans les moyens de communication exploités.

Ce mémoire est structuré en trois chapitres dont l'agencement reflète la démarche utilisée et les étapes de conception.

Le premier chapitre est consacré à la «Présentation générale de la domotique par GSM ».

En accord avec les objectifs de mémoire, le deuxième chapitre est consacré à la description des différents logiciels et matériels utilisés dans notre réalisation.

Le troisième chapitre est réservé exclusivement à la description de la réalisation pratique, ainsi que l'application ANDROÏDE.

Enfin, on terminera avec une conclusion générale qui résumera l'intérêt de notre projet.



Chapitre I :

Introduction à la domotique par GSM

I-1 Introduction

La domotique est le domaine technologique qui traite de l'automatisation du domicile, d'où l'étymologie du nom qui correspond à la contraction des termes "domicile" et "automatique".

La gestion d'une maison automatisée consiste-en des techniques qui font appel à l'informatique et l'électronique pour assurer : la sécurité, la surveillance et le confort de la maison, l'économie de l'énergie, commander facilement les différents appareils de la maison et recevoir des informations sur ces dernier quelque soit la distance de votre domicile.

L'objectif de ce chapitre est de faire appel à la domotique par GSM afin de clarifier les concepts clés de notre thématique.

I-2 Historique de la domotique

La domotique, signe extérieur de richesse soutenue par une exigence accrue des ménages en termes de confort, la domotique fait véritablement son apparition dans les années 80.

Les avancées dans les domaines de l'électronique (miniaturisation des composants), de l'informatique (arrivée des premiers ordinateurs dans les foyers) et de la communication (numérisation des réseaux, minitel) sont à l'origine de cette nouvelle technologie.

La maison en 1990 est automatisée : la télécommunication commence à se développer. Les réseaux informatiques deviennent sans fil grâce au wifi. Certaines actions sont planifiées (ouvertures des volets à une heure donnée, allumage des lumières quand on passe à proximité ou quand le soleil se couche...), les télécommandes remplacent les interrupteurs, et les alarmes s'associent à la télésurveillance [1].

Le secteur de la domotique ne cesse de croître depuis 2000. La maison est désormais pilotable à distance ! Avec le développement d'internet, les automatismes sont programmés et commandés via un appareil mobile (Smartphone, tablette, ordinateur). La maison est obéissante et les solutions « sur-mesure » permettent de proposer un écosystème de produits parfaitement adaptés aux besoins du client.

En d'autres termes, la domotique se démocratise, aidé par les professionnels du bâtiment (architectures, artisans, thermiciens, etc.) qui sont de plus en plus nombreux à se former pour offrir conseils et compétences à leur clientèle [1].

I-3 Définition et principe de fonctionnement

C'est l'ensemble des techniques et technologies de l'électronique, de l'automatisme, de l'électricité, de la mécanique, des télécommunications et de l'informatique appliquées à la gestion automatisée des bâtiments individuels et collectifs.

Elle pilote de façon intelligente l'ensemble des systèmes automatisés présents dans les habitations individuelles et collectives, on parle de « maison intelligente » (Voir figure-1-).

Le nom vient de la contraction de maison en latin (Domus) et d'informatique, automatique ou technique.

Le principe de la domotique consiste à faire communiquer ensemble et entre eux les équipements électriques d'un bâtiment. Nous parlons alors de bâtiment intelligent ou de bâtiment communiquant.

La domotique permet de superviser, de programmer les fonctions du bâtiment afin de répondre à vos attentes en termes de sécurité, de confort, de gestion d'énergie et de communication. Elle participe également à l'aide au maintien à domicile des personnes âgées ou handicapées en facilitant leur quotidien.

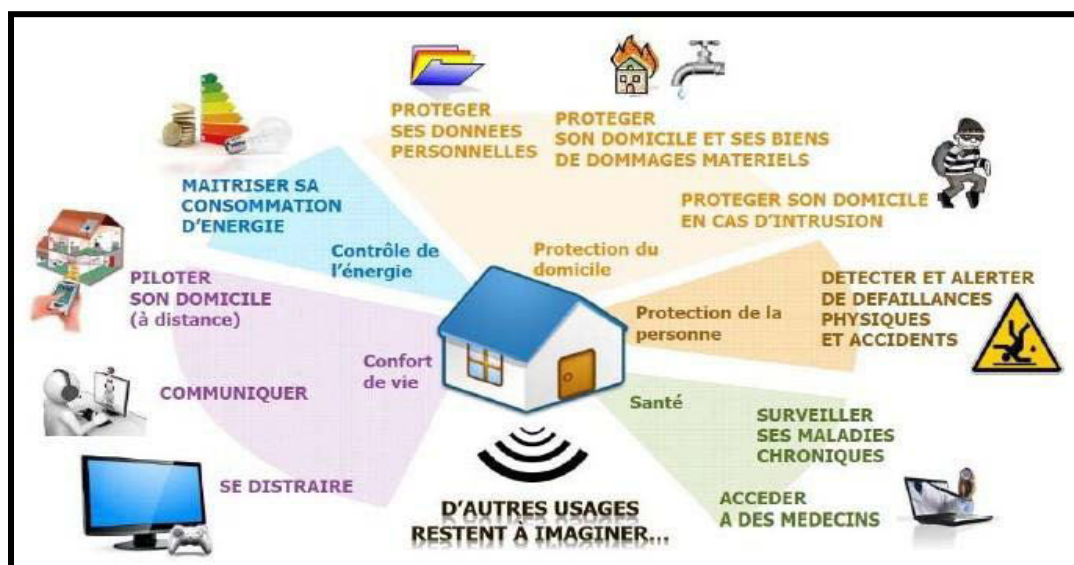


Figure I-1 : Exemple d'une maison intelligente

L'installation domotique peut être pilotée localement ou à distance depuis votre Smartphone, un écran tactile ou encore un ordinateur. (Voir la **Figure I-2**)

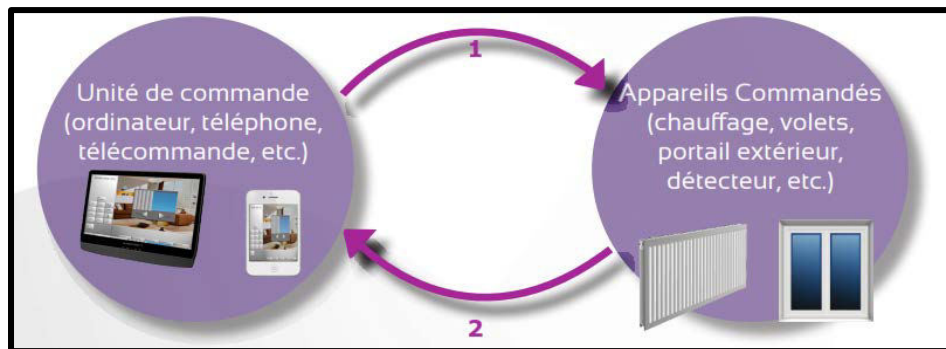


Figure I-2 : Circulation des informations dans la domotique [5]

- La liaison 1 représente le circuit de commande, transmission des informations pour la réalisation d'une tâche à partir du tableau électrique, le wifi, Bluetooth,...etc.
- La liaison 2 représente l'information sur l'état des appareils.

Le principe de fonctionnement d'un système domotique contient certains dispositifs pour récupérer les informations pour commander une maison, et pour cela un système domotique fait appel aux capteurs et aux actionneurs.

I.3.1 Les capteurs :

Un capteur est un dispositif transformant une grandeur physique à une autre grandeur utilisable, tel qu'une tension électrique ou une intensité ; On fait souvent la confusion entre capteur et transducteur : le capteur est au minimum constitué d'un transducteur.

Le capteur est souvent le premier dispositif de la chaîne d'acquisition, il s'agit d'une simple interface entre un processus physique et une information manipulable.

Si l'on s'intéresse aux phénomènes physiques mis en jeu dans les capteurs, on peut classer ces derniers en deux catégories :

- **Les capteurs actifs** : Fonctionnant en générateur, un capteur actif est généralement fondé dans son principe sur un effet physique qui assure la conversion en énergie

électrique de la forme d'énergie à la grandeur physique à prélever : énergie thermique, mécanique ou de rayonnement [7].

- **Les capteurs passifs** : Ils ont besoin dans la plupart des cas d'apport d'énergie extérieure pour fonctionner .Ce sont des capteurs modélisables par une impédance, une variation du phénomène physique étudié engendre une variation de l'impédance. Il faut leur appliquer une tension pour obtenir un signal de sortie [7].

I.3.2 Actionneurs :

Dans une machine ou un système de commande, semi automatique ou automatique, un actionneur est l'organe de la partie opérative qui, dès qu'il reçoit un ordre de la partie commande via un éventuel capteur ou pré-actionneur, convertit l'énergie qui lui est fournie en un travail utile à l'exécution de tâches, éventuellement programmées, d'un système automatisé.

En d'autres termes, un actionneur est l'organe fournissant la force nécessaire à l'exécution d'un travail ordonné par une unité de commande.

Il existe trois types d'actionneur :

- Actionneur électrique.
- Actionneur pneumatique.
- Actionneur hydraulique.

I.3.3 Relation capteurs actionneurs :

Les capteurs sont des informateurs traducteurs, ils détectent une variation de l'environnement de la partie opérative et la traduit en une information interprétable (grandeur électrique) par la partie commande a fin de contrôler les actionneurs qui transforme une forme d'énergie en une autre dans le but d'obtenir l'énergie souhaité (énergie mécanique), comme le montre la figure ci-dessous.

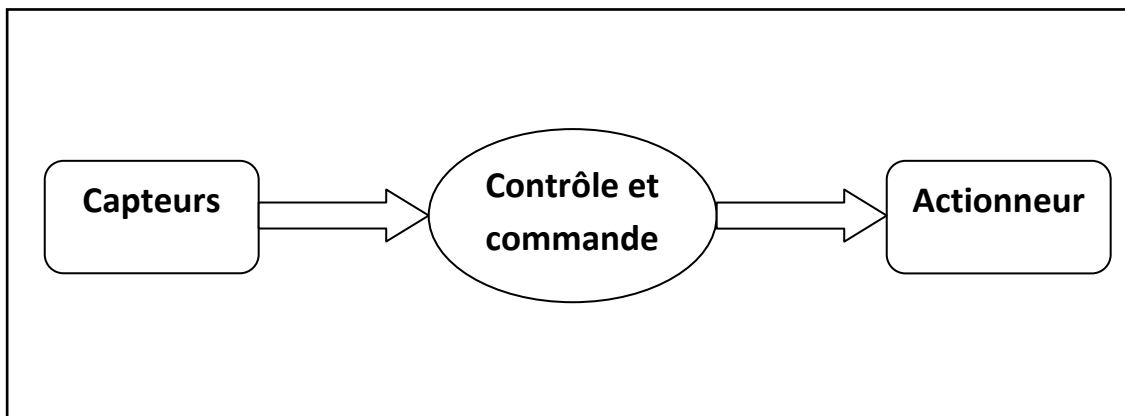


Figure I-3 : Relation capteurs actionneurs

I.4 Les technologies utilisées

I.4.1 La technologie bus filaire

La technologie bus filaire, est souvent utilisée dans la construction ou la rénovation de bâtiments en raison de l'installation d'un bus filaire.

Cette technologie veille à ce que tous les composants communiquent entre eux avec le même langage afin qu'ils puissent échanger des informations, les analyser et les traiter.

L'installation de ce dispositif est composée de deux réseaux :

- un réseau bus filaire reliant les capteurs aux actionneurs
- un réseau d'alimentation reliant les actionneurs au courant.

Comme la montre la figure ci-dessous [5] :

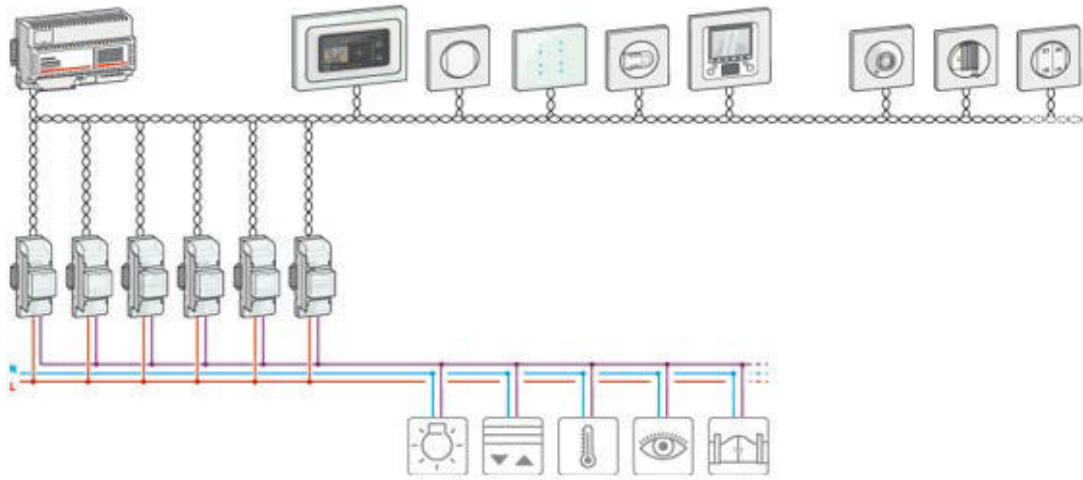


Figure I-4 : Installation domotique bus filaires

I.4.2 Le courant porteur en ligne (CPL) :

La technologie du courant porteur en ligne (CPL) permet le transfert et l'échange d'informations et de données en passant par le réseau électrique existant.

L'installation est composée d'émetteurs et de récepteurs connectés au réseau électrique qui communiquent entre eux.

Et l'utilisation de cette technologie ne nécessite pas de travaux particuliers [5].

(Voir Figure I-5)



Figure I-5 : Installation CPL

I.4.3 La radiofréquence

Avec la technologie radiofréquence, la transmission d'informations s'effectue sans fils. En utilisant les ondes radio, l'émetteur (une commande sans fil) peut ainsi piloter un récepteur (interrupteur, prise...). L'avantage de la radiofréquence est qu'elle permet de faire évoluer une installation électrique sans grands travaux.

I-5 Les fonctions de la domotique

Les fonctions suivantes peuvent être réalisées grâce aux technologies intégrées dans la domotique. (Voir **Figure I-6**)

- Sécurité
- Surveillance
- Gestion de l'énergie
- Scénarisation des actions
- Communication
- Confort



Figure I-6 : Les fonctions de la domotique

a- La fonction de sécurité

Contrairement à un système d'alarme traditionnel, une centrale domotique agit sur toute l'installation électrique de l'habitation [2].

Elle dissuade les intrus en simulant une présence par l'allumage aléatoire des éclairages, de la radio ou l'ouverture des volets durant la journée,...

Les accès à un commerce sont contrôlés et enregistrés et, si une intrusion est détectée, la centrale prend les mesures qui s'imposent :

- Sirène
- Allumage de tous les éclairages de la maison
- Appel d'un centre de surveillance, d'un voisin ou d'un téléphone mobile.

A partir d'un téléphone, il est possible d'écouter et de s'adresser directement aux intrus grâce au haut-parleur de la centrale.

b- La fonction de surveillance

Pour surveiller un domicile, plusieurs capteurs sont utilisés pour détecter les anomalies :

- Inondation
- Incendie
- Fuite de gaz
- Coupure de courant

La centrale intervient instantanément pour couper les alimentations, remonter les stores, couvrir la piscine, appeler les numéros d'urgence ou faire retentir la sirène si l'occupant est présent [2].

c- La fonction de gestion d'énergie

La programmation des seuils de température est l'une des principales sources d'économie, elle peut être journalière ou hebdomadaire. Des sondes de température renseignent la centrale sur les valeurs de température dans chaque pièce [2].

Elle tient compte de la présence d'un occupant et s'adapte automatiquement à son emploi du temps :

- Extinction des éclairages inutiles
- Réglage de l'intensité lumineuse en fonction de l'activité...

Un simple appel téléphonique suffit pour augmenter la température de chauffage en prévision de l'arrivée des occupants.

d- La fonction de scénarisation

Au moment de quitter un habitat ou un commerce, la mise en fonction de l'alarme déclenche une série de contrôles et d'actions, (centralisations des commandes) :

- Extinction ou allumage de toutes les lumières.
- Vérification de la fermeture de toutes les fenêtres.
- Allumage de la lumière extérieure durant quelques minutes s'il fait nuit...

e- La fonction de communication

La domotique permet de communiquer entre l'occupant et son domicile à partir :

- Téléphone (GSM)
- Ordinateur (Internet)

f- La fonction de confort

Ouvrir le portail sans descendre de voiture, allumer automatiquement la lumière en ayant les bras chargés de paquets, arroser automatiquement le jardin, ouvrir ou fermer les volets ou les stores, programmer une ambiance sonore dans la maison, disposer de la télévision dans plusieurs pièces, déclencher la préparation du café avant le réveil, adapter la température aux conditions extérieures...[5]

I-6 Système automatisé**I-6-1 Définition**

Après avoir été mis en fonctionnement par l'homme, un système automatisé est un ensemble de composants qui fonctionne seul (sans interventions supplémentaires) pour effectuer des actions programmées (prévues à l'avance) et répétitives (toujours les mêmes prévues dans le programme).

I-6-2 Les composants d'un système automatisé

Un système automatisé est constitué d'un opérateur (l'utilisateur), d'une partie commande (ordinateur) et d'une partie opérative avec des capteurs et actionneurs.

L'opérateur est l'utilisateur du système qui donne des consignes (ce qu'il veut obtenir) à la partie commande et reçoit des signaux l'informant sur les états du système (température, position ...).

La partie commande joue le rôle du "cerveau" du système qui pilote (ordres) les actionneurs de la partie opérative, reçoit des informations venant des capteurs (comptes rendus) et de l'opérateur (consignes) et informe ce même opérateur (signaux).

Ex : programmeur de la machine à laver, circuit électronique (automate) de l'ascenseur ...

La partie opérative exécute les ordres qu'elle reçoit de la partie commande grâce aux actionneurs et recueille des informations sur les états du système grâce aux capteurs.

- Les actionneurs sont des appareils qui convertissent l'énergie qui leur est fournie en un travail utile, une action (mouvement, chaleur, lumière, son ...)



Moteur
Produire un mouvement



Ampoule
Produire de la lumière



Sirène
Émettre un son



Radiateur
Produire de la chaleur

- Les capteurs : Ce sont des appareils qui traduisent une information physique en un signal compréhensible (compatible) par la partie commande (Ordinateur, automate, circuit électronique).



Détecteur infrarouge
Détecter une présence



Interrupteur, Bouton poussoir
Détecter une instruction



Sonde de température
Mesurer une température



Capteur de luminosité
Mesurer la luminosité



Capteur fin de course
Détecter une position

I-6-3 Représentation simplifiée d'un système automatisé

Exemple : éclairage automatique.

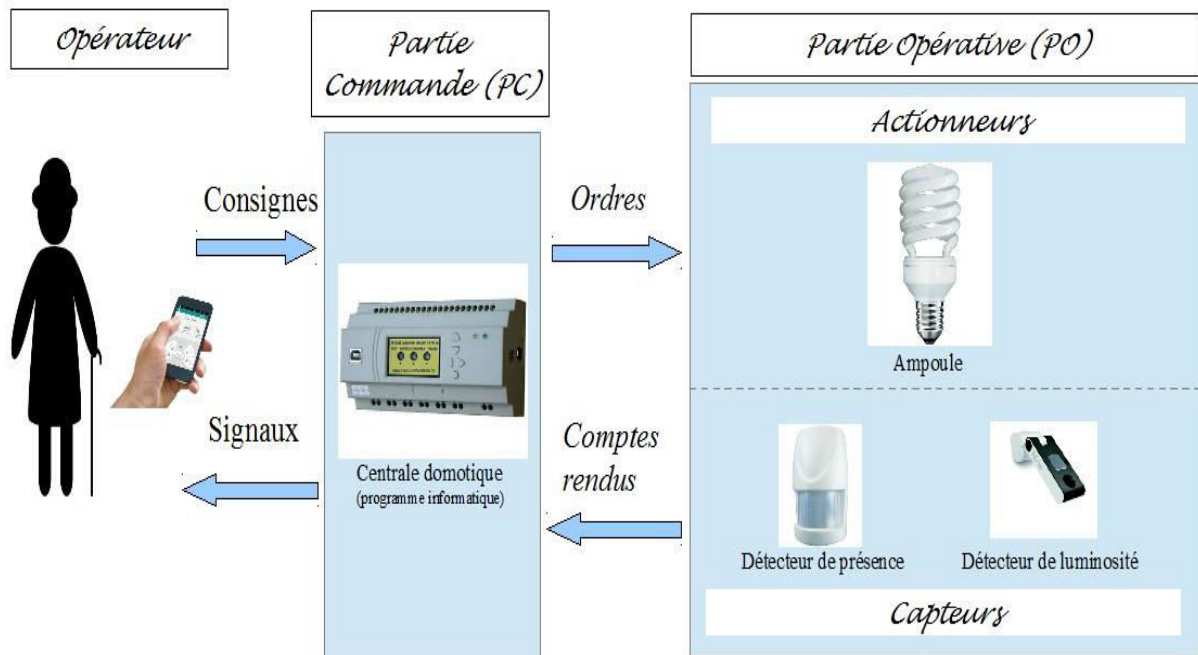


Figure I-7 : Exemple d'un système automatisé

1. L'opérateur donne des consignes à la centrale domotique (mise en marche/arrêt du fonctionnement automatique, allumage forcé de l'ampoule, extinction forcé de l'ampoule ...).
2. La centrale domotique exécute le programme informatique qu'elle contient en fonction des consignes et des comptes rendus qu'elle reçoit et donne en conséquence des ordres aux actionneurs (ordres Allumer l'ampoule ou Éteindre l'ampoule).
3. Les capteurs détectent les grandeurs physiques "présence d'une personne" et "luminosité dans la pièce" et renvois des comptes rendus à la centrale (Présence d'une personne, Absence d'une personne, Forte luminosité, Faible luminosité ...)

4. La centrale domotique informe l'opérateur par des signaux (voyants, SMS, appel téléphonique avec message enregistré, animation sur l'écran ...) sur l'état du système (Ampoule allumée, éteinte, présence/absence d'une personne dans la pièce, forte/faible luminosité dans la pièce ...).

I.7 Le GSM

I.7.1 Historique

Des étapes majeures ont marqué l'histoire du GSM et de la radiotéléphonie :

- 1887. Le physicien allemand Heinrich Hertz (1857-1894) découvre les «ondes hertziennes», ce sont les ondes radio.
- 1982. La Conférence Européenne des Postes et Télécommunication (C.E.P.T) réserve des fréquences dans la gamme des 900MHz pour les télécommunications mobiles et créé le Groupe Spécial Mobile (G.S.M).
- 1987. 13 opérateurs européens signent le « Memorandum Of Understanding » (MOU) qui prévoit la mise en service de réseaux mobiles conformes au GSM.
- 1989. Le G.S.M préconise l'introduction de la bande de fréquence 1800MHz afin de permettre l'utilisation de la norme DCS 1800.
- 1991. Le ministère français des PTT autorise France Telecom et la Société Française de Radiophonie (SFR) à exploiter un réseau du type GSM.
- 1992. 17 pays et 26 opérateurs européens ont adopté le système G.S.M. De plus, une première ouverture vers un marché mondial s'effectue par la ratification de l'Australie au MOU.
- 1993. L'abréviation GSM signifie désormais « Global System for Mobile communication ». Ce changement marque ainsi le tournant entre la phase d'élaboration de la norme et sa phase de commercialisation.
- Octobre 1993. Le GSM de deuxième génération apparaît. Cette évolution propose de nouveaux services tels que les renvois d'appel ou la limitation d'appel.

- 1994. Les Etats Unis réservent la bande des 1900MHz à la norme GSM. Une adaptation de la norme leur permettra d'utiliser le réseau à cette bande de fréquence.
- 1995. Démarrage de la phase 2+ du GSM.
- 1997. Le DCS1800 est renommé GSM1800 pour refléter l'apparition de services à vitesse plus élevée. Les réseaux bi bandes se multiplient.
- 2005. Le réseau GSM totalise plus de 1,2 Milliard d'abonnés dans le monde.

I.7.2 Présentation

Global System for Mobile Communications (GSM) historiquement « Groupe spécial mobile » est une norme numérique de seconde génération pour la téléphonie mobile. Le groupe de travail chargé de la définir a été établi en 1982 par la Conférence européenne des administrations des postes et télécommunications (CEPT).

Elle a été spécifiée et mise au point par l'ETSI (Européen Télécommunications Standard Institut) pour la gamme de fréquences des 900 MHz. Une variante appelée Digital Communication System (DCS) utilise la gamme des 1 800 MHz. Cette norme est particulièrement utilisée en Europe, en Afrique, au Moyen-Orient et en Asie. Deux autres variantes, en 850 MHz et en 1 900 MHz PCS (personale communications services), sont également utilisées. La protection des données est assurée par les algorithmes de chiffrement A5/1 et A5/2.

Tel qu'il a été conçu, le réseau GSM est idéal pour les communications de type « voix » (téléphonie). Le réseau étant commuté, les ressources ne sont allouées que pour la durée de la conversation, comme lors de l'utilisation de lignes téléphoniques fixes. Les clients peuvent soit acheter une carte prépayée, soit souscrire un abonnement.

Sous l'égide de l'organisation 3GPP la norme GSM a ensuite été étendue pour prendre en charge de plus hauts débits et le transport de données en mode « paquet » par les extensions GPRS (General Packet Radio Services) puis EDGE (Enhanced Data rates for GSM Evolution). Ces deux modes peuvent cohabiter avec le mode « voix commutée » du GSM et utilisent les mêmes antennes et les mêmes bandes de fréquence.

I.7.3 Architecture de GSM

L'architecture d'un système GSM se décompose en trois sous-systèmes qui sont bien schématisés dans la figure suivante :

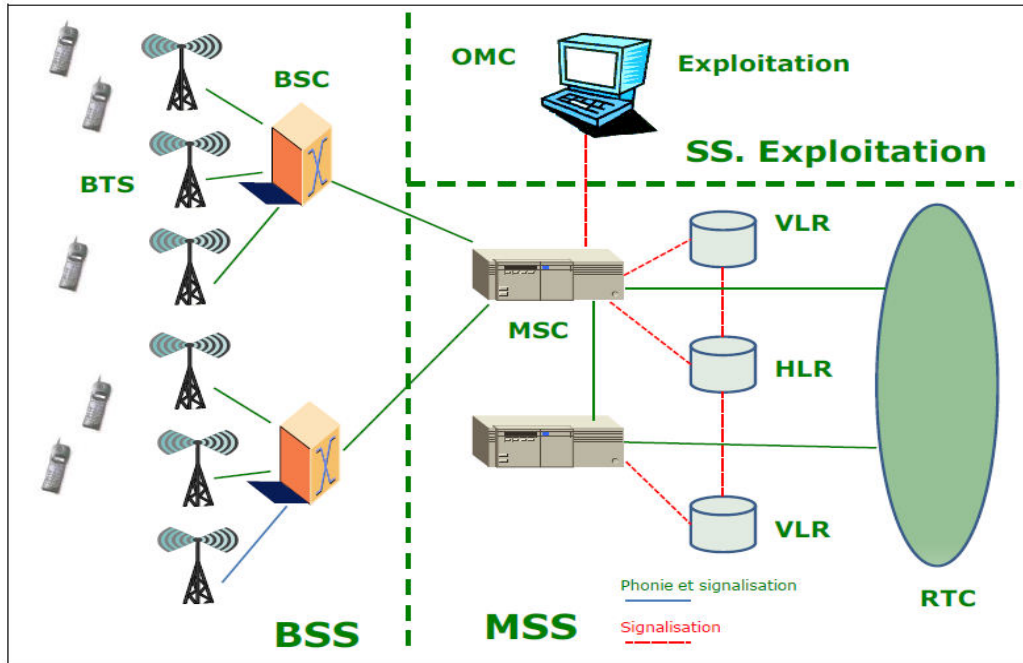


Figure I-8 : Architecture d'un réseau GSM

I.7.3.1 Le sous-système radio (BSS)

Il gère la partie radio des communications et se compose d'émetteurs-récepteurs radio (BTS) contrôlés par une BSC.

La radio BTS (Base Transceiver Station) est un ensemble d'émetteurs-récepteurs radio (TRX), sans grande intelligence. Elle gère la couche physique de l'interface air.

Le BSC il contrôle les stations de base et assure la commutation entre les ressources terrestres et radio [8].

I.7.3.2 Le sous-système réseau (NSS)

Il gère le traitement des appels, la mobilité et l'acheminement de/vers les réseaux filaires. Il se compose de commutateurs radio (MSC) et d'un certain nombre de bases de données HLR et VLR.

Le MSC est un commutateur radio qui gère l'ensemble des communications dans le réseau GSM.

Le HLR (Home Location Register) est une base de données qui centralise les caractéristiques et les informations de localisation des usagers du réseau GSM.

Le VLR (Visitor Location Register) : cette entité contient les données de travail relatives aux abonnés présents dans la zone du MSC. Il permet de minimiser l'accès au HLR.

Le MSC (Mobile-services Switching Centre) est un commutateur qui gère l'établissement des communications entre un mobile et un autre MSC, Il dialogue avec le VLR pour gérer la mobilité des usagers [8].

I.7.3.3 Le sous-système exploitation OSS (Operation Sub-System)

Il contrôle les droits d'accès au réseau, les droits des usagers et assure l'interface homme-machine d'exploitation. Il gère aussi le maintien en conditions opérationnelles du réseau et la remontée des alarmes

La diversité des équipements présents dans un réseau GSM tant sur le plan de leur type (émetteurs-récepteurs, commutateurs, bases de données) que la multiplicité des fournisseurs pousse à adopter une approche structurée et hiérarchique. La norme GSM présente deux niveaux :

Les OMC (Operation and Maintenance Center) et les NMC (Network Management Center).

Le NMC permet l'administration générale de l'ensemble du réseau par un contrôle centralisé, alors que les OMC permettent une supervision locale des équipements [8].

I.8 Principe de fonctionnement de la domotique par GSM

La domotique par GSM est basée sur deux points essentiels, la commande et la surveillance par GSM

I.8.1 Commande par GSM

La commande par GSM peut connecter /déconnecter à distance, tous les matériels électrique comme moteurs, pompes, portes, portails, chauffages, climatisation, etc...

Ainsi tous les systèmes utilisant le GSM se connectent au réseau de téléphonie et comme chaque téléphone portable dispose de sa propre carte SIM et son numéro, et c'est pour ça la commande par GSM est protégée et seuls les numéros que vous autorisez seront habilités à commander votre maison.

I.8.2 Surveillance par GSM

La surveillance par GSM vous permet de contrôler tous les paramètres de votre maison, bureau, entreprise, coupure de courant, variation de température, mouvement ou intrusion, fuite de gaz, etc.... sans vous déplacer, on recevant des messages ou des notifications sur votre téléphone.

I.9 Conclusion :

Les définitions des concepts mises aux claires dans ce chapitre, nous ont permis de comprendre le fonctionnement théorique de la domotique par GSM d'une façon générale. Certains principaux matérielle utilisée pour la réalisation méritent d'être étudiés d'une façon plus particulière afin d'acquérir une meilleur maitrise du fonctionnement de la domotique.

C'est l'objectif du chapitre suivant.

Chapitre II :
Description de la
partie matérielle
et logiciel

II-1 Introduction

L'objectif de ce chapitre est de mettre l'accent sur la description des différents matériels et logiciels qu'on a utilisés pour la réalisation, programmation et création d'une application Androïde. A savoir :

- La carte Arduino
- Le Bluetooth
- Les capteurs
- Les actionneurs et d'autres éléments...
- L'environnement Arduino, Fritzing, APP INVENTOR

II-2 Structure du système

Le système conçu comporte un bloc principal constitué d'une carte à microcontrôleur du type Arduino relié à un système de communication de type Bluetooth et GSM et il est commandé par une application Androïde. Cette carte est connecté à un ensemble de périphériques constitués des capteurs et des actionneurs (figure-9-) : notre système est composé de :

a- Capteurs

- Capteur ultrason : pour réaliser un système d'alarmes.
- Capteur de gaz : pour surveiller les fuites de gaz.
- Capteur de température : donne des informations sur la température ambiante.
- Capture de lumière : allumer automatiquement la lumière.

b- Actionneurs

- Servomoteur : fonctionnement du moteur est asservi à la position de l'axe.
- Carte relais : pour allumer et éteindre des lampes

c- Le module d'acquisition des données de la domotique

C'est la partie *hardware* du système à base de la carte Arduino UNO, qui prend en charge la réception et l'envoi des différentes informations entre les périphériques : c'est lui qui gère et maintien le fonctionnement du circuit et tous les capteurs où les actionneurs qui sont connectés.

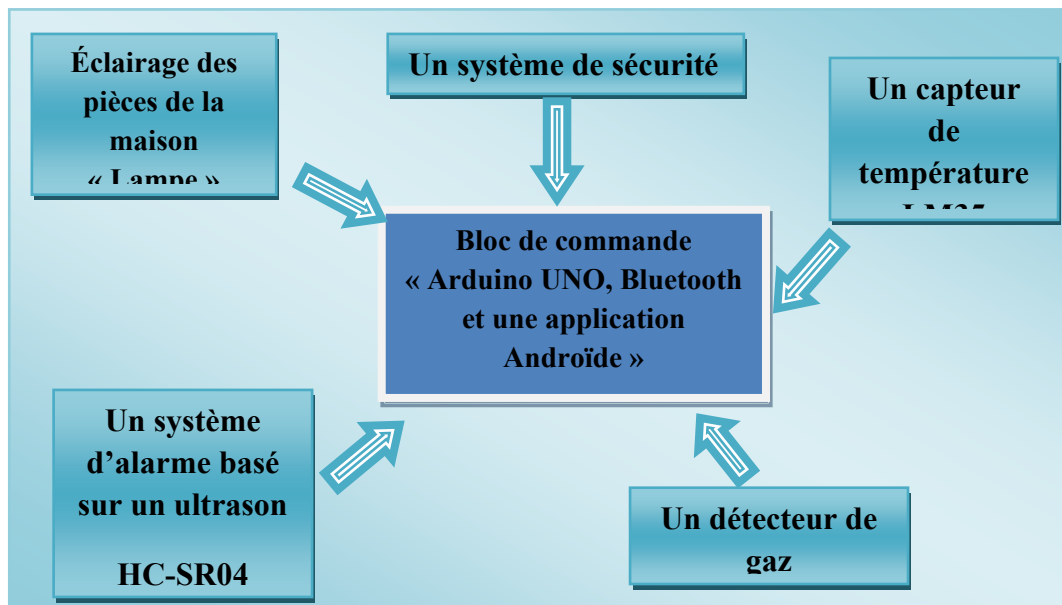


Figure II-1 : Schéma bloc de notre système

II-3 Description de la partie matérielle

II-3-1 La carte Arduino UNO

II-3-1-1 Présentation de la carte

Arduino est une plate-forme de prototypage d'objets interactifs à usage créatif constituée d'une carte électronique et d'un environnement de programmation.

Sans tout connaître ni tout comprendre de l'électronique, cet environnement matériel et logiciel permet à l'utilisateur de formuler ses projets par l'expérimentation directe avec l'aide de nombreuses ressources disponibles en ligne.

Arduino est utilisé dans beaucoup d'applications comme l'électrotechnique industrielle et embarquée ; le modélisme, la domotique mais aussi dans des domaines différents comme l'art contemporain et le pilotage d'un robot, commande des moteurs et faire des jeux de lumières, communiquer avec l'ordinateur, commander des appareils mobiles (modélisme) [11] [13].

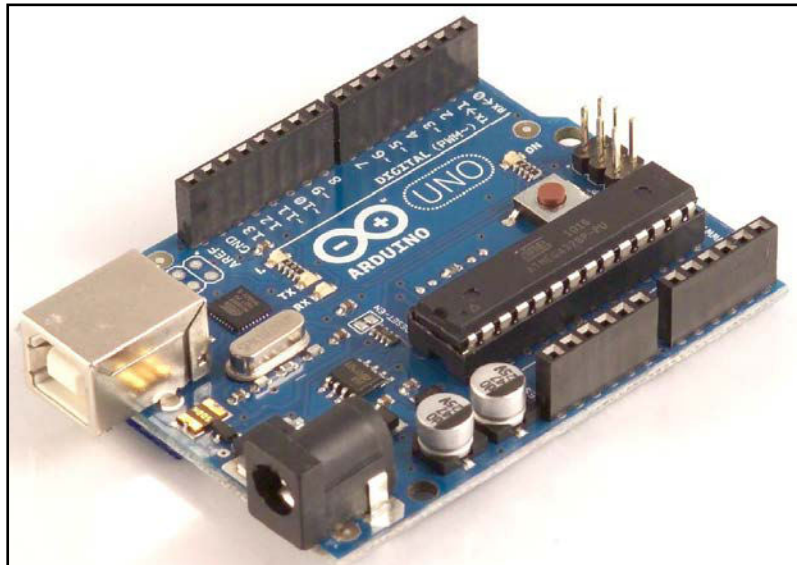


Figure II-2 : La carte Arduino UNO

II-3-1-2 Caractéristiques de la carte Arduino UNO

Micro contrôleur	ATmega328
Tension d'alimentation interne	5V
Tension d'alimentation (recommandée)	7 à 12V, limites = 6 à 20 V
Entrées/sorties numériques	14 dont 6 sorties PWM
Entrées analogiques	6
Courant max par broches E/S	40 mA
Courant max sur sortie 3,3V	50mA
Mémoire Flash	32 KB dont 0.5 KB utilisée par le bootloader
Mémoire SRAM	2 KB
Mémoire EEPROM	1 KB
Fréquence horloge	16 MHz

II-3-1-3 Les différents composants de la carte Arduino UNO

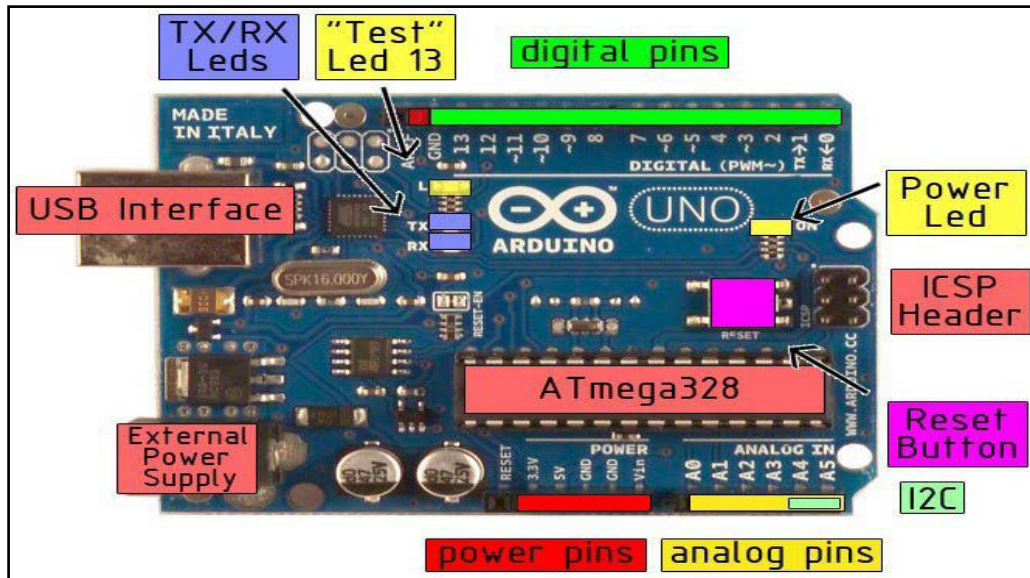


Figure II-3 : Les composants de la carte Arduino UNO

a- Le Microcontrôleur ATmega328

a-1 Description

Un microcontrôleur ATmega328 est un circuit intégré qui rassemble sur une puce plusieurs éléments complexes dans un espace réduit au temps des pionniers de l'électronique. Aujourd'hui, en soudant un grand nombre de composants encombrants ; tels que les transistors ; les résistances et les condensateurs tout peut être logé dans un petit boîtier en plastique noir muni d'un certain nombre de broches dont la programmation peut être réalisée en langage C. la figure-12- montre un microcontrôleur ATmega 328, qu'on trouve sur la carte Arduino [9].



Figure II-4 : Microcontrôleur ATmega328

Le microcontrôleur ATmega328 est constitué par un ensemble d'éléments qui ont chacun une fonction bien déterminée. Il est en fait constitué des mêmes éléments que sur la carte mère d'un ordinateur. Globalement, l'architecture interne de ce circuit programmable se compose essentiellement de :

- **La mémoire Flash:** C'est Celle qui contiendra le programme à exécuter. Cette mémoire est effaçable et réinscriptible mémoire programme de 32Ko (dont bootloader de 0.5 ko).
- **RAM:** c'est la mémoire dite "vive", elle vas contenir les variables du programme. Elle est dite "volatile" car elle s'efface si on coupe l'alimentation du microcontrôleur. Sa capacité est 2 ko.
- **EEPROM:** C'est le disque dur du microcontrôleur. On y enregistre des infos qui ont besoin de survivre dans le temps, même si la carte doit être arrêtée. Cette mémoire ne s'efface pas lorsque l'on éteint le microcontrôleur ou lorsqu'on le reprogramme [12].

a-2 Caractéristique [20]

Vitesse de processeur	16MHz
Taille de mémoire de programme	32KB
Taille de mémoire, RAM	2KB
Taille d'EEPROM	1KB
Nombre de broches	28
Type de boîtier MCU	DIP
Nombre d'E/S	20
Type d'interface embarquée	I2C, SPI, UART
Tension d'alimentation min	1.8V
Tension d'alimentation max	5.5 V
Type de packaging	Pièce

a-3 Principe de fonctionnement [20]

L'ATMega328 ce compose de 28 pattes qui sont :

1 : Reset : C'est pour le bouton poussoir de réinitialisation.

2 : E/S numérique réservé pour RX (la réception).

3 : E/S numérique réservé pour TX (la transmission).

7-20 : VCC : c'est l'alimentation avec 5 V.

8-22 : GND : c'est la masse.

9-10 : Crystal (quartz)

21 : AREF (Analog référence).

23-24-25-26-27-28 : Des entrées analogiques.

4-5-6-11-14-15-16-17-18-19 : Entrées /sorties numérique .

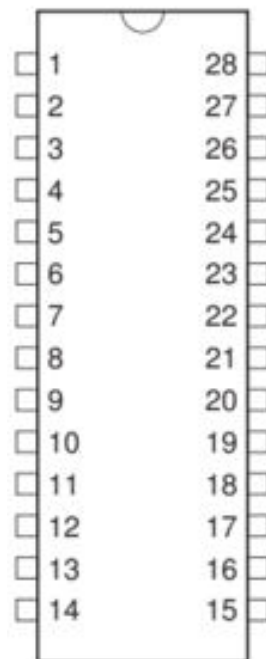


Figure II-5 : Datasheet d'ATMega328

b- Les sources d'alimentation de la carte Arduino UNO

On peut distinguer deux genres de sources d'alimentation (Entrée Sortie) et cela comme suit :

□ **VIN** La tension d'entrée positive lorsque la carte Arduino est utilisée avec une source de tension externe (à distinguer du 5V de la connexion USB ou autre source 5V régulée). On peut alimenter la carte à l'aide de cette broche, ou, si l'alimentation est fournie par le jack d'alimentation, accéder à la tension d'alimentation sur cette broche.

□ **5V** La tension régulée utilisée pour faire fonctionner le microcontrôleur et les autres composants de la carte (pour info : les circuits électroniques numériques nécessitent une tension d'alimentation parfaitement stable dite "tension régulée" obtenue à l'aide d'un composant appelé un régulateur et qui est intégré à la carte Arduino). Le 5V régulé fourni par cette broche peut donc provenir soit de la tension d'alimentation VIN via le régulateur de la carte, ou bien de la connexion USB (qui fournit du 5V régulé) ou de tout autre source d'alimentation régulée.

□ **3.3V** Une alimentation de 3.3V fournie par le circuit intégré FTDI (circuit intégré faisant l'adaptation du signal entre le port USB de votre ordinateur et le port série de l'ATmega) de la carte est disponible: ceci est intéressant pour certains circuits externes nécessitant cette tension au lieu du 5V. L'intensité maximale disponible sur cette broche est de 50mA [12].

c- Les entrées /sorties

Cette carte possède 14 broches numériques (numérotée de 0 à 13) peut être utilisée soit comme une entrée numérique, soit comme une sortie numérique, en utilisant les instructions `pinMode`, `digitalWrite` et `digitalRead` du langage Arduino. Ces broches fonctionnent en 5V. Chaque broche peut fournir ou recevoir un maximum de 40mA d'intensité et dispose d'une résistance interne de "rappel au plus" (pull-up) (déconnectée par défaut) de 20-50 KOhms. Cette résistance interne s'active sur une broche en entrée à l'aide de l'instruction `digitalWrite` (broche, HIGH).

En plus, certaines broches ont des fonctions spécialisées :

• Interruptions Externes :

Broches 2 et 3. Ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, sur un front montant ou descendant, ou sur un changement de valeur.

-Impulsion PWM (largeur d'impulsion modulée) :

Broches 3, 5, 6, 9, 10, et 11. Fournissent une impulsion PWM 8-bits à l'aide de l'instruction `analogWrite`.

• **SPI** (Interface Série Périphérique): Broches 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ces broches supportent la communication SPI (Interface Série Périphérique) disponible avec la librairie pour communication SPI. Les broches SPI sont également connectées sur le connecteur ICSP qui est mécaniquement compatible avec les cartes Mega.

• **I2C** : Broches 4 (SDA) et 5 (SCL). Supportent les communications de protocole I2C (ou interface TWI (Two Wire Interface - Interface "2 fils"), disponible en utilisant la librairie `Wire/I2C` (ou TWI - Two-Wire interface - interface "2 fils").

• **LED** : Broche 13. Il y a une LED incluse dans la carte connectée à la broche 13.

Lorsque la broche est au niveau HAUT, la LED est allumée, lorsque la broche est au niveau BAS, la LED est éteinte.

La carte UNO dispose 6 entrées analogiques (numérotées de 0 à 5), chacune pouvant fournir une mesure d'une résolution de 10 bits (c.à.d. sur 1024 niveaux soit de 0 à 1023) à l'aide de la très utile fonction `analogRead` du langage Arduino. Par défaut, ces broches mesurent entre le 0V (valeur 0) et le 5V (valeur 1023), mais il est possible de modifier la référence supérieure de la plage de mesure en utilisant la broche AREF et l'instruction `analogReference` du langage Arduino.

La carte Arduino UNO intègre un fusible qui protège le port USB de l'ordinateur contre les surcharges en intensité (le port USB est généralement limité à 500mA en intensité).

Bien que la plupart des ordinateurs aient leur propre protection interne, le fusible de la carte fournit une couche supplémentaire de protection. Si plus de 500mA sont appliqués au port USB, le fusible de la carte coupera automatiquement la connexion jusqu'à ce que le court-circuit ou la surcharge soit stoppé [11].

d- Les ports de communications

La carte Arduino UNO a de nombreuses possibilités de communications avec l'extérieur. L'Atmega328 possède une communication série UART TTL (5V), grâce aux broches numériques 0 (RX) et 1 (TX).

On utilise (RX) pour recevoir et (TX) transmettre (les données séries de niveau TTL).

Ces broches sont connectées aux broches correspondantes du circuit intégré ATmega328 programmé en convertisseur USB – vers – série de la carte, composant qui assure l'interface entre les niveaux TTL et le port USB de l'ordinateur.

Comme un port de communication virtuel pour le logiciel sur l'ordinateur, La connexion série de l'Arduino est très pratique pour communiquer avec un PC, mais son inconvénient est le câble USB, pour éviter cela, il existe différentes méthodes pour utiliser ce dernier sans fil :

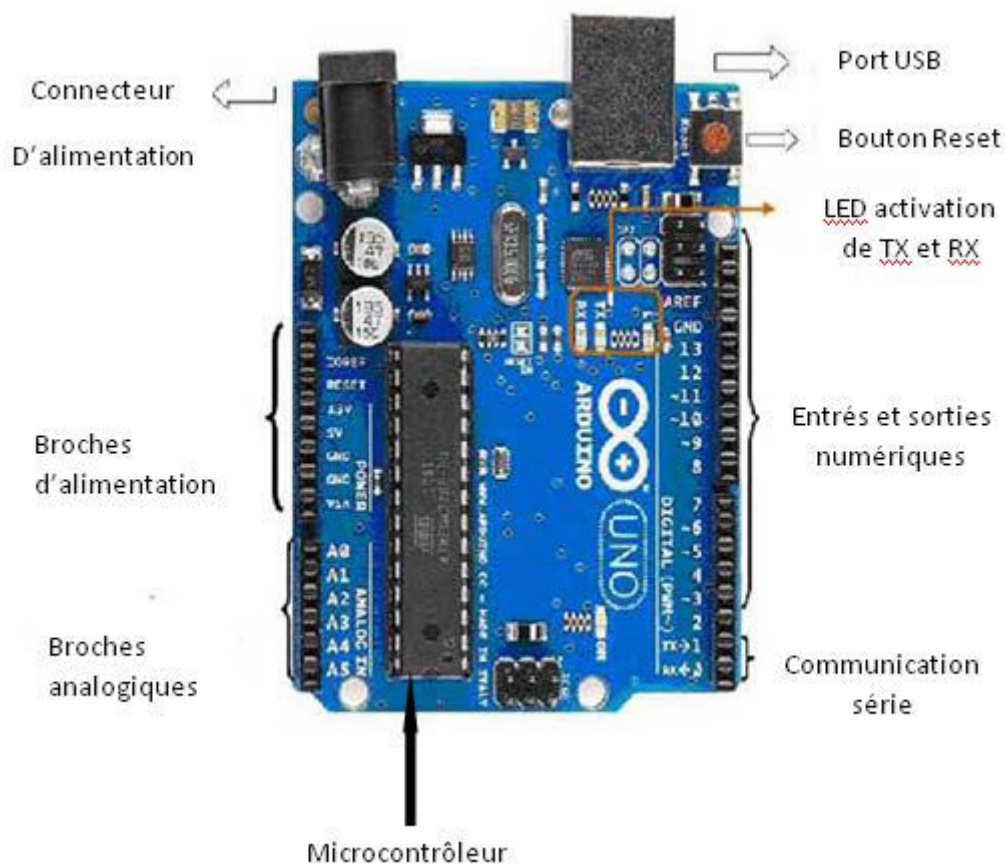


Figure II-6 : Constitution de la carte Arduino UNO

II-3-2 Bluetooth HC-05 :

a- Description

Le Bluetooth est un protocole de communication sans fil, permettant l'échange bidirectionnel de données à une distance bien précise (15 à 20 mètres), en utilisant des ondes radio UHF sur une bande de fréquence de 2.4 GHz. Son objectif est de simplifier les connexions entre les appareils électroniques en supprimant les liaisons filaires.

La norme « Bluetooth » fut mise au point en 1994 par le fabricant Suédois Ericsson à Lund. Le terme signifie littéralement en français : « dent bleue » [21].

Le module HC-05 n'est pas plus gros qu'un pouce. Il est en fait un montage d'un module **Bluetooth** sur un petit **P.C.B**, cela permet de s'affranchir de certaines contraintes comme la soudure du module (qui est très délicate), la conversion 5V et 3.3V, la régulation de l'alimentation (3.3V de nouveau) ou encore l'ajout de **LED** de signal, tout cela est déjà intégré.



Figure II-7 : Vue du Module Bluetooth HC-05

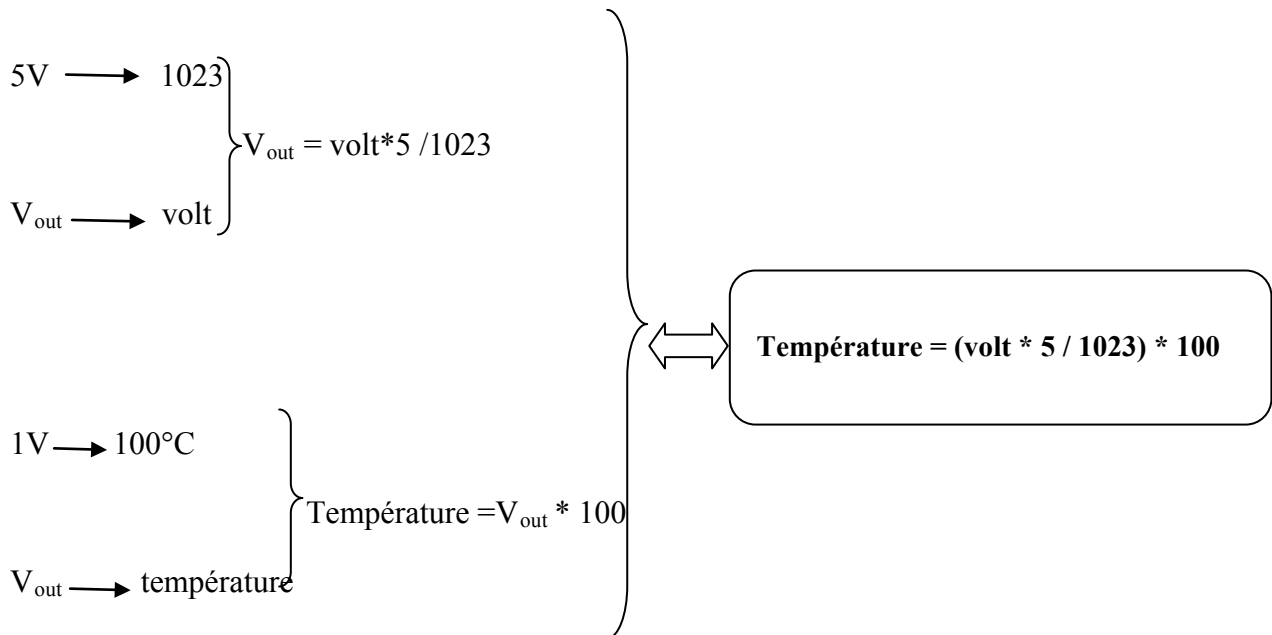
II-3-3 Capteur de température LM 35

a- Description

Le capteur de température LM35 est un capteur de température analogique fabriqué par Texas Instruments, il est extrêmement populaire en électronique, car précis, peu coûteux, très simple d'utilisation et d'une fiabilité à toute épreuve, Il peut fonctionner dans une plage de -55°C à +150°C.

$$\text{Température} = (\text{volt} * 5 / 1023) * 100$$

Volt : est la valeur analogique(V_{out}) converti par le CAN en une valeur numérique [22]



II-3-4 capteur ultrason (HC-SR04)

a- Description

Le HC-SR04 est un capteur qui utilise les ultrasons pour déterminer la distance d'un objet de 2cm à 400cm. La mesure s'effectue "sans contact" et dispose d'une précision de 3mm dans les conditions optimales. Le module inclus un émetteur et un récepteur ultrason ainsi qu'un système de contrôle [23].

b- Caractéristique [24]

Nombre de pattes	4
Dimension	45mm x 20mm x 15mm
Alimentation max	5.5 V DC
Alimentation min	4.5 V DC
Plage de mesure	2cm à 400cm
Courant de fonctionnement max	10 mA
Courant de fonctionnement min	20 mA
Résolution de la mesure	0.3 cm
Angle de mesure	15°
Largeur d'impulsion sur l'entrée de déclenchement	10 μ s
La fréquence des ultrasons	40 KHZ
TRIG	Entrée de déclenchement de la mesure (trigger INPUT)
ECHO	Sortie de mesure donnée en écho (ECHO OUTPUT)
GND	Masse de l'alimentation



Figure II-9 : Capteur Ultrason HC-SR04

c- Principe de fonctionnement

1- Nous envoyons une impulsion sur l'entrée Trig

2- Le module émet une onde sonar composée d'une série de 8 impulsions à 40 kHz.

3- Quand le signal revient, la sortie ECHO passe au niveau haut durant toute la période où l'onde voyage vers l'objet et revient après avoir été réfléchi par se dernier.

Le calcul de la distance parcourue par un son s'effectue par l'équation suivante :

Distance = vitesse x temps (D = V x T)

V : c'est la vitesse de la lumière (34'000 cm/1'000'000μs)

T : c'est le temps écoulé par les ondes ultrason aller /retour, le HC-SR04 donne une durée d'impulsion en dizaines de μs. donc Il faut multiplier la valeur obtenue par 10 μs pour obtenir le temps T **(T= valeur x 10 μs)**

Puisque le son fait un aller-retour. La distance vaut donc la moitié

$$D = 34'000 \text{ cm/1'000'000 } \mu \text{ s} * \text{valeur} \times 10 \mu \text{ s} / 2$$

En simplifiant : **D = 17/100 cm * valeur [23].**

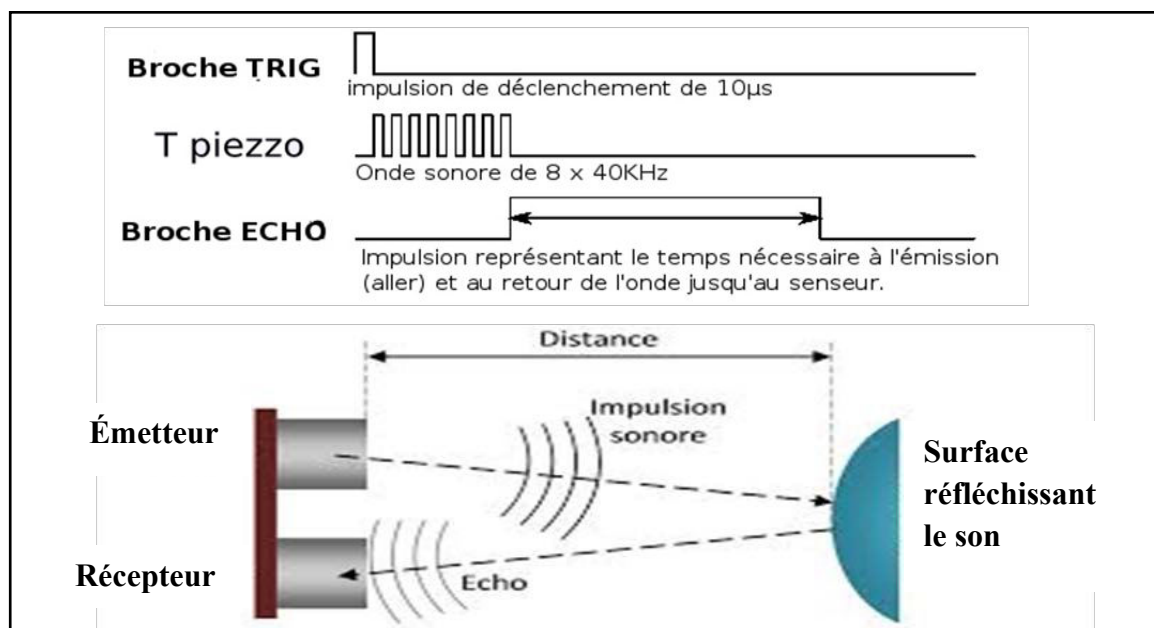


Figure II-10 : Principe de fonctionnement d'un capteur Ultrason HC-SH04

II-3-5 capteur de Gaz (MQ-5)

a- Description

Le MQ-5 est un capteur qui permet de détecter du gaz ou de fumée à des concentrations de 300ppm à 1000 ppm. Après calibration, le MQ-5 peut détecter différents gaz comme le GPL, le butane, le propane, le méthane, l'alcool, l'hydrogène ainsi que les fumées. Il est conçu pour un usage intérieur à température ambiante.

Le MQ-5 doit être alimenté en 5V pour le capteur physico-chimique puisse atteindre sa température de fonctionnement. Il dispose de deux type de sorties :

- Une sortie analogique, dont la tension de sortie varie en fonction de la concentration globale détecte.
- Une sortie numérique, réglable à l'aide d'un potentiomètre à l'arrière du composant. L'utilisateur règle le seuil de détection du module, et ce dernier se déclenche si la concentration dépasse le seuil [16].



Figure II-11 : Capteur de gaz MQ-5

b- Caractéristiques

Nombres de broches	4 pins
Dimensions	35 x 22 mm
Tension d'entrée	5V
Courant absorbé	0.16 A
Tension de sortie	5V
Puissance	0.8 W
Temps de réponse	Rapide

II-3-6 Servomoteur

a- Description

Un servomoteur est un actionneur (moteur) capable de maintenir une opposition à un effort statique et dont la position est vérifiée en continu et corrigée en fonction de la mesure. Le servomoteur est composé de plusieurs éléments visibles :

- Les fils.
- L'axe de rotation sur lequel est monté un accessoire en plastique ou en métal.
- Le boîtier qui le protège.

Aussi de plusieurs éléments que l'on ne voit pas qui se trouvent à l'intérieur du boîtier.

- Un moteur à courant continu.
- Des engrenages pour former un réducteur.
- Un capteur de position de l'angle d'orientation de l'axe (un potentiomètre bien souvent).
- Une carte électronique pour le contrôle de la position de l'axe et le pilotage du moteur à courant continu.

La position est définie avec une limite de débattement d'angle de 180 degrés. Souvent abrégé en « servo » provenant du latin servus qui signifie « esclave ».

Il existe plusieurs servomoteurs, dans notre application on a utilisé un servomoteur de faible puissance (9 Gramme) SG90.

Il contient trois fils, un pour l'alimentation, l'autre pour la masse (GND) et la dernière pour le signal d'entrée (la commande) [16].

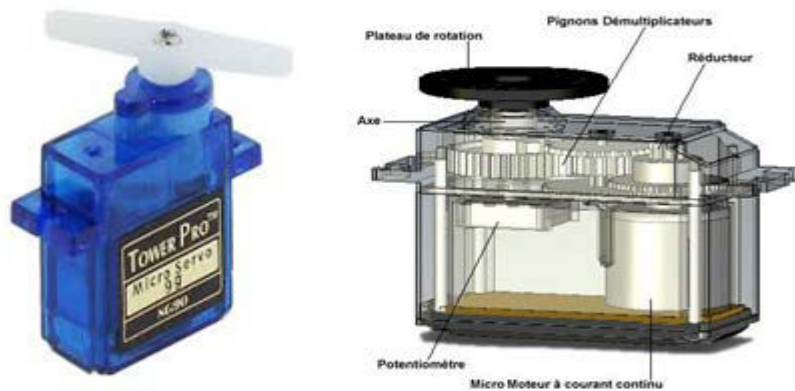


Figure II-12 : Vue d'un servomoteur.

b- Caractéristiques [24]

Nombres de broches	3 fils	
	Marron	Masse
	Rouge	Vcc
	Orange	Commande
Dimension	22mm x 11.5mm x 27mm	
Poids	9 grammes	
Tension d'alimentation Min	4.8V	
Tension d'alimentation Max	6V	
Vitesse	0.12 s/60° sous 4.8V	
Couple	1.2 kg/cm sous 4.8V	
Consommation	125	

c- Principe de fonctionnement d'un servomoteur

Les servomoteurs sont commandés par l'intermédiaire d'un câble électrique qui lui transmette des consignes de position sous forme d'un signal codé en largeur d'impulsion communément appelé PWM. Cela signifie que c'est la durée des impulsions qui détermine l'angle absolu de l'axe de sortie et donc la position de bras de commande du servomoteur. Comme la montre la figure ci-dessous :

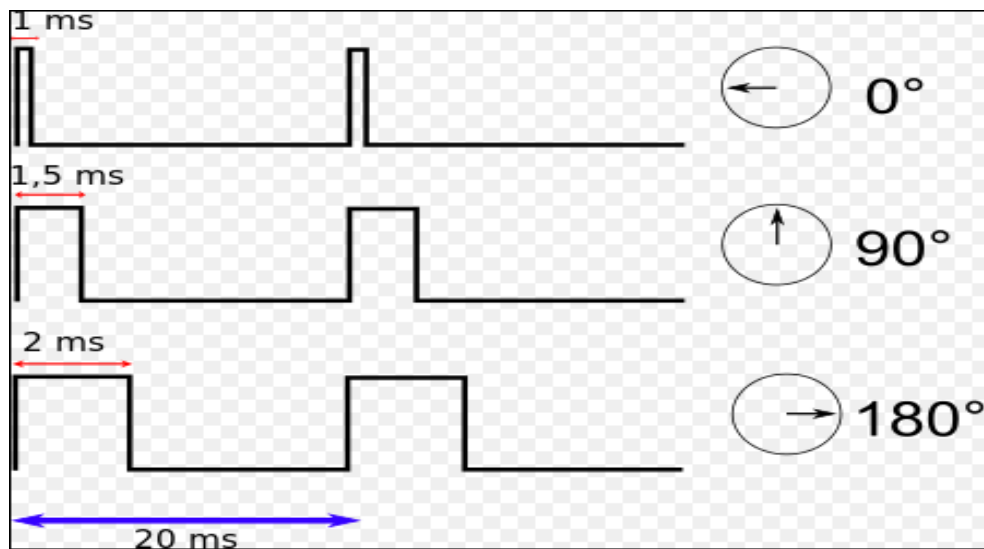


Figure II-13 : l'angle de l'axe du servomoteur en fonction de la largeur de l'impulsion.

Le signal est répété périodiquement, en général toutes les 20 millisecondes, ce qui permet à l'électronique de contrôler et de corriger continuellement la position angulaire de l'axe de sortie, cette dernière étant mesurée par le potentiomètre.

Lorsque le moteur tourne, l'axe du servomoteur change de position, ce qui modifie la résistance du potentiomètre. Le rôle de l'électronique est de commander le moteur pour que la position de l'axe de sortie soit conforme à la consigne reçue.

II-4 Description de la partie logicielle

Notre réalisation software a besoin de deux étapes : la première consiste à un programme qui va s'injecter aux microcontrôleurs de la carte Arduino après avoir été convertie par le IDE en code HEX et la deuxième à un programme qui va se manipuler sous App Inventor et s'installer sous smart phone.

II-4-1 Le logiciel Arduino

Le logiciel de programmation de la carte Arduino de code (langage proche de C). Une fois, le programme tapé ou modifié au clavier, il sera transféré et mémorisé dans la carte à

travers de la liaison USB. Le câble USB alimenté à la fois en énergie la carte et transporte aussi l'information, ce programme appelé IDE Arduino.

Comme n'importe quel langage de programmation, une interface souple et simple est exécutable sur n'importe quel système d'exploitation, Arduino basé sur la programmation en C.

II-4-1-1 Présentation de «IDE »

Le logiciel Arduino est un environnement de développement (IDE) open source et gratuit, téléchargeable sur le site officiel d'Arduino.

L'IDE Arduino permet :

- D'éditer un programme : des croquis (sketch en Anglais),
- De compiler ce programme dans le langage « machine » de l'Arduino,
- De téléverser le programme dans la mémoire de l'Arduino,
- De communiquer avec la carte Arduino grâce au terminal,

Des boutons de commande en haut, une page blanche vierge, une bande noire en bas.

A l'ouverture, l'interface visuelle du logiciel ressemble à ceci :



Mise en œuvre de l'environnement Arduino :

- On conçoit d'abord un programme avec le logiciel Arduino
- On vérifie ce programme avec le logiciel (compilation)
- Des messages d'erreur apparaissent Éventuellement...on corrige puis vérifie à nouveau...
- On enlève le précédent programme sur la carte Arduino (Bouton réinitialisation)
- On envoie ce programme sur la carte Arduino dans les 5 secondes qui suivent l'initialisation.
- L'exécution du programme sur la carte est automatique quelques secondes plus tard ou à ses prochains branchements sur une alimentation électrique (Alim 9/12V ou port USB).

II-4-1-2 Programmer avec Arduino

Un programme utilisateur Arduino est une suite d'instructions élémentaires sous forme textuelle, ligne par ligne.

La carte lit puis effectue les instructions les unes après les autres, dans l'ordre défini par les lignes de code.

Structure d'un programme

Il y a trois phases consécutives :

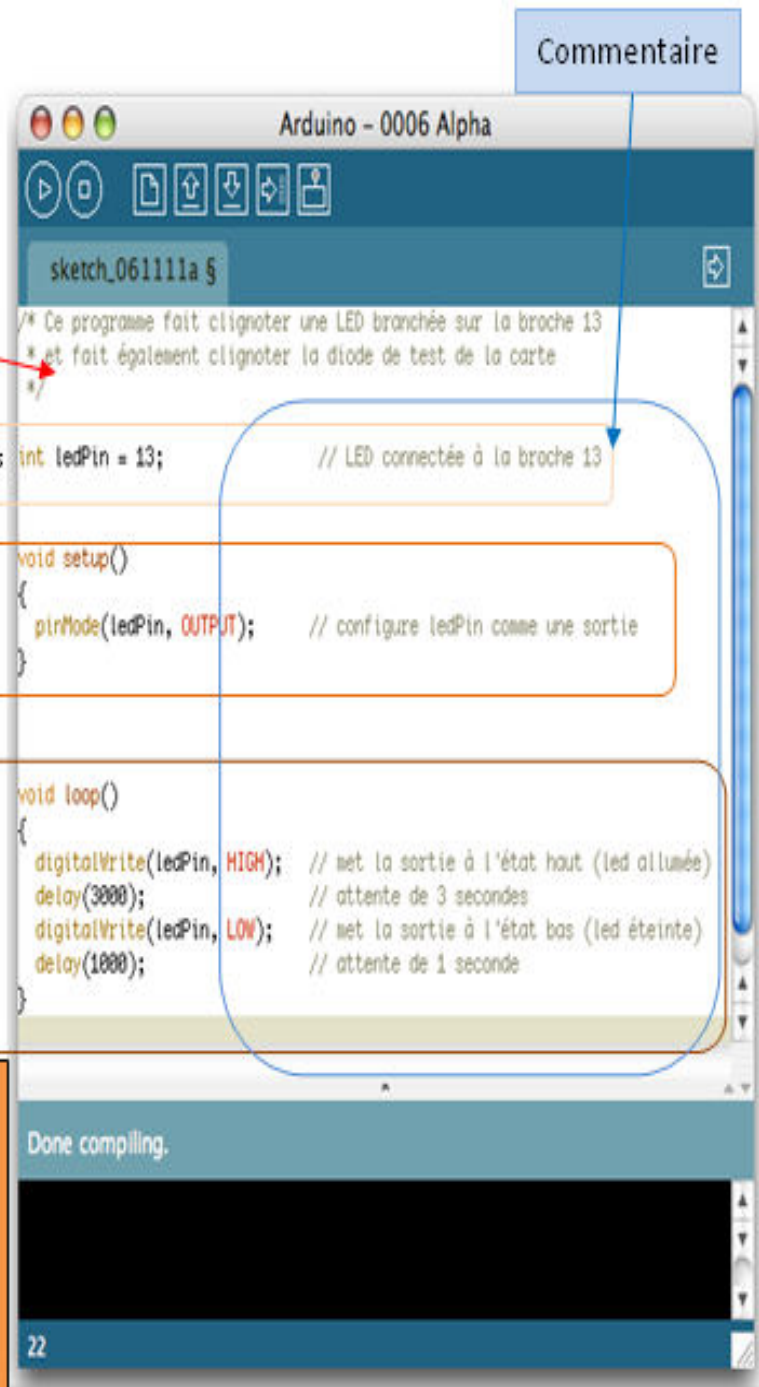
Commentaires multilignes pour se souvenir du patch.

1/La définition des constantes et des variables

2/La configuration des entrées et sorties
Void setup ()

3/La programmation des interactions et Comportements
Void loop ()

Une fois la dernière ligne exécutée, la carte revient au début de la troisième phase et recommence sa lecture et son exécution des instructions successives. Et ainsi de suite. Cette **boucle** se déroule des milliers de fois par seconde et anime la carte.



II-4-1-3 Syntaxe des commandes Arduino

Commentaires

Toujours écrire des commentaires sur le programme : soit en multiligne, en écrivant entre des `/***/`, soit sur une ligne de code en se

Définition des variables :

Pour notre montage, on va utiliser une sortie numérique de la carte, qui est par exemple la 13^{ème} sortie numérique. Cette variable doit être définie et nommée ici : on lui donne un nom arbitraire `BrocheLED`. Le mot de la syntaxe est pour désigner un nombre

Configuration des entrées-sorties`void setup () :`

Les broches numériques de l'Arduino peuvent aussi bien être configurées en entrées numériques ou en sorties numériques. Ici on va configurer `BrocheLED` en sortie.

`PinMode (nom, état)` est une des quatre fonctions relatives aux entrées-sorties

Programmation des interactions`void loop () :`

Dans cette boucle, on définit les opérations à effectuer, dans l'ordre :

- `digitalWrite (nom, état)` est une autre des quatre fonctions relatives aux entrées-sorties numériques.
- `delay (temps en millisecondes)` est la commande d'attente entre deux autres instructions
- Chaque ligne d'instruction est terminée par un point virgule

(Syntaxe en "Rouge" paramètres utilisateur en "Noir")

```
/* Ce programme fait clignoter une LED
branchée sur la broche 13
* et fait également clignoter la diode de test de
la carte
*/
```

```
int BrocheLED = 13 ; // Définition de la valeur 13
et du nom de la broche à utiliser
```

```
void setup ()
{
  pinMode (BrocheLED, OUTPUT) ; // configure
  BrocheLED comme une sortie
}
```

```
void loop ()
{
  digitalWrite (BrocheLED, HIGH) ; // met la
  sortie num. à l'état haut (led allumée)
  delay(3000) ; // attente de 3 secondes
  digitalWrite (BrocheLED, LOW) ; // met la
  sortie num. à l'état bas (led éteinte)
  delay(1000) ; // attente de 1 seconde
}
```

II-4-2 Fritzing

Fritzing est un logiciel open-source multiplateforme permettant de construire des schémas et des circuits électroniques que nous utilisons avec Arduino. Plusieurs vues sont disponibles : platine d'essai, schémas électriques et circuit imprimé.

Le projet dispose d'un site web, et se veut un outil qui permet aux utilisateurs de documenter leurs prototypes et les partager avec d'autres [18]. (Voir Figure II-14).

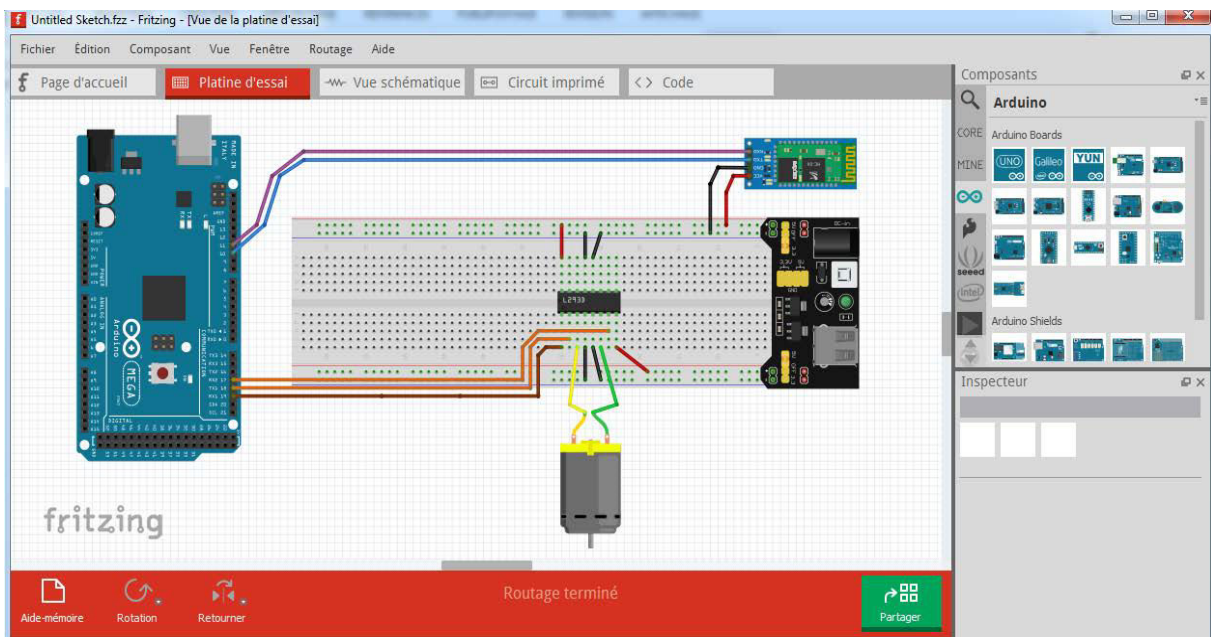


Figure II-14 : L'interface de Fritzing

II-4-3 Le système Androïde

Androïde est un système d'exploitation développé initialement pour les Smart phones. Il utilise un noyau Linux qui est un système d'exploitation libre pour PC et intègre tous les utilitaires et les périphériques nécessaires à un smart phone. Il est optimisé pour les outils Gmail.

Aussi, l'androïde est libre et gratuit et a été ainsi rapidement adopté par des fabricants.

La société Androïde a été rachetée en 2007 par Google. Mais aujourd'hui, l'Androïde est utilisé dans de nombreux appareils mobiles (smart phones). Les applications sont exécutées par un processeur de type ARM à travers un interpréteur JAVA. En plus de cela,

l'androïde concurrence l'opérateur système d'Apple qu'il tend à dépasser en nombre d'utilisateurs.

Androïde évolue pour mieux gérer l'hétérogénéité des appareils qu'il utilise [29].

II-4-4 APP INVENTOR

APP INVENTOR est un outil de développement des applications en ligne pour les smart phones sous androïde et permet à chacun de créer son application personnelle pour le système d'exploitation Androïde qui est développée par Google.

La plateforme de développement est offerte à tous les utilisateurs possédant un compte Gmail. Elle rappelle certains langages de programmation simplifiés basés sur une interface graphique similaire à Scratch. Les informations des applications sont stockées sur des serveurs distants [19]. Elles sont actuellement entretenues par le Massachusetts Institute of Technologie (MIT).

L'environnement de APP INVENTOR contient trois fenêtres sont proposées pendant le développement :

- Une pour la création de l'interface homme machine : elle sera l'allure de l'application (App Inventor Designer) ;
- Une pour la programmation par elle-même : elle permet, par l'assemblage des blocs de créer le comportement de l'application (**App Inventor Block Editor**) ;
- Une pour l'émulateur : qui permettra de tester l'application. L'émulateur permet de remplacer un terminal réel pour vérifier le bon fonctionnement du programme.

La connexion d'un terminal réel sous Androïde permettra ensuite d'y télécharger le programme pour un test réel. Ce terminal pourra aussi bien être un téléphone qu'une tablette ; le comportement du programme sera identique [23].

N.B. : Scratch est un environnement graphique permettant aux débutants de s'initier à la programmation. On y programme le comportement de lutins animés à l'aide de briques visuelles qui s'emboîtent comme des Lego par Drag and Drop [19].

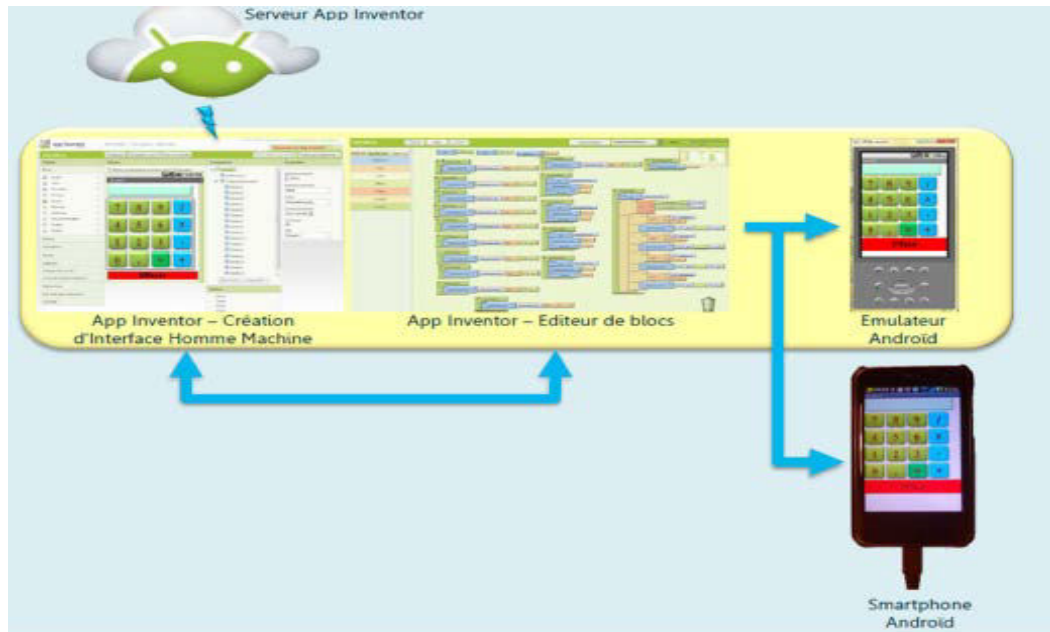


Figure II-15 : Différentes fenêtres d'APP INVENTOR.

II-4-4-1 Historique de logiciel App Inventor

2009 : Début du développement du logiciel App Inventor par Google à partir de recherches dans l'enseignement de l'informatique faites par le MIT, Boston près de New-York. L'objectif de l'enseignement permet à des étudiants qui débutent en informatique d'apprendre à programmer sans se noyer sous le code Java.

2011: Google rend APP INVENTOR open source. Le MIT poursuit le développement.

2012 : Version beta APP INVENTOR diffusé par le MIT encore en version beta aujourd'hui.

II-4-4-2 Un commencement avec APP INVENTOR

Google fournit gratuitement un kit de développement (SDK) prévu pour s'intégrer (sous la forme d'un Plugin) à l'environnement de développement Eclipse (libre). Il permet de développer des applications codées en langage Java pour les différentes versions d'Androïde.

1. Se connecter à Internet.
2. Ouvrir notre navigateur et se connecter au compte Google.
3. Se connecter au site Internet d'APP INVENTOR du MIT : [19]. Cela peut-être assez long parfois car APP INVENTOR est lancée depuis le Web en mode Cloud.
4. Créer un nouveau projet :



Figure II-16 : Création de nouveau projet sur APP INVENTOR

Cliquer sur My Projects (en haut à gauche) / New / Project Name (sans espace) / OK. On aura cette interface (Figure-16-).

A. L'interface graphique

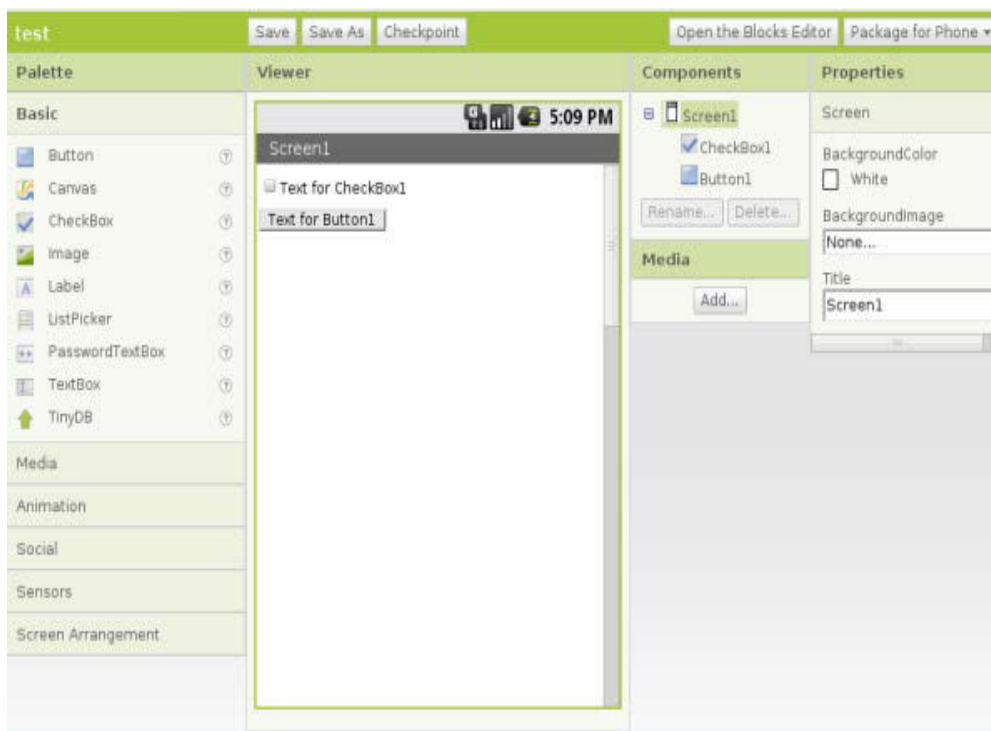
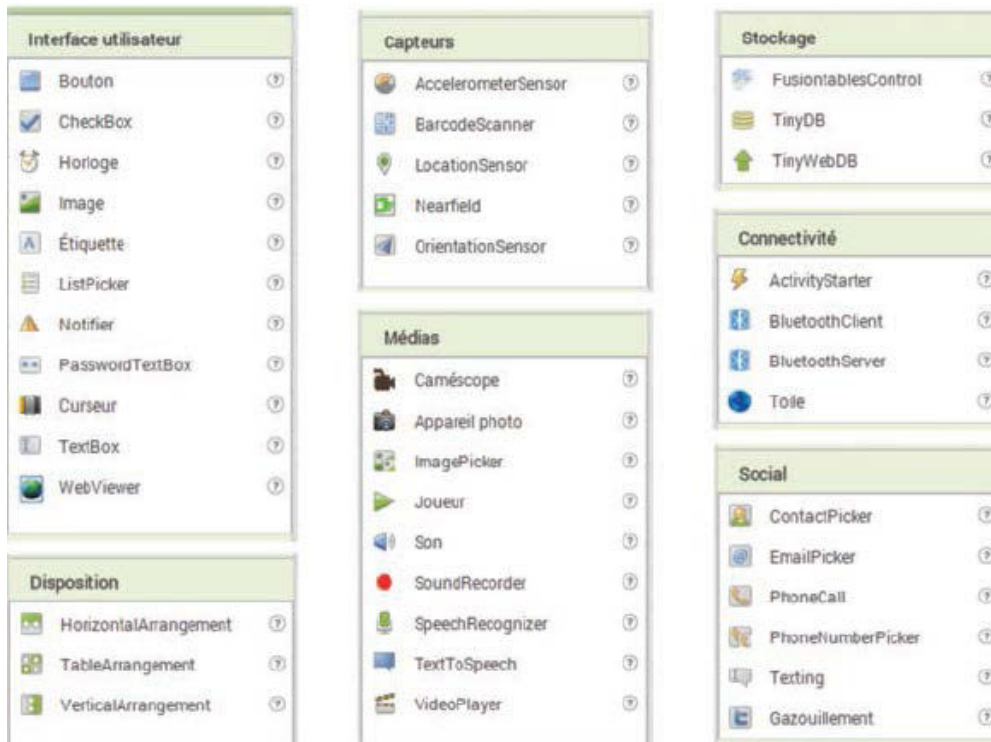


Figure II-17 : Première interface de la création APP INVENTOR

Cette interface graphique contient quatre parties :

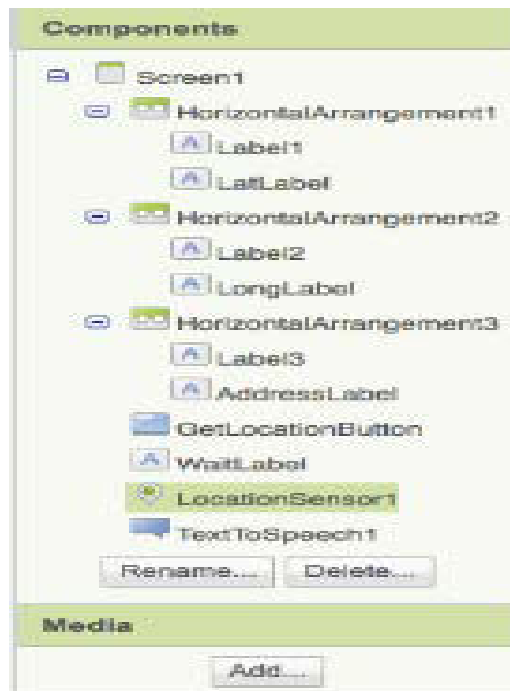
Partie1 : les composants graphiques



Partie2 : C'est la surface du téléphone ajusté automatiquement par APP INVENTOR ou manuellement par nous même en utilisant le composant « Screen arrangement ».



Partie 3 : La liste des éléments et des medias utilisés sur l'écran.



Partie 4 : Les propriétés des différents éléments utilisés par exemple la couleur et la taille du bouton ou texture.



Pour ajouter un bouton sur l'écran d'émulateur virtuel, on clique sans lâcher sur le mot « Bouton » dans la **palette** en déplaçant la souris sur **Viewer** et relâcher le bouton et là un nouveau bouton va apparaître sur le **Viewer**.

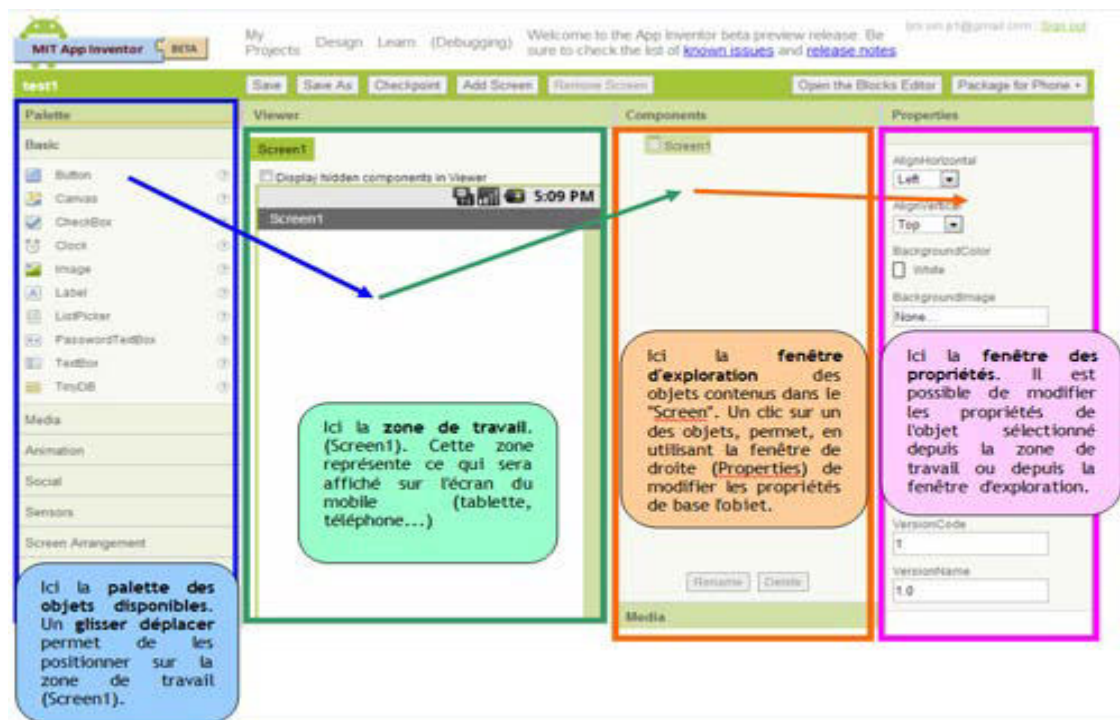


Figure II-18 : Résumer des quatre étapes

B. Éditeur de blocs (fenêtre Scratch)

Une fois les composants de l'écran de téléphone mis en place et désigné, nous passons à la deuxième phase de développement d'une application via App Inventor : l'interface Scratch, pour cela, il faut cliquer sur «Open the Blocks Editor» en haut à droite de la page.

L'interface Scratch permet d'imbriquer des éléments graphiques entre eux pour effectuer la partie programmation de l'application à développer. De cette partie, on peut assembler les différents blocs de l'application et indiquer comment les composants doivent se comporter et qui s'affichent dans l'émulateur virtuel (par exemple ; ce qui se produit quand un utilisateur clique un bouton) déterminant le fonctionnement même de l'application, en réaction à des **événements** (internes ou externes) ou à des **réponses** [29].



Figure II-19 : En-tête d'éditeur de blocs APP INVENTOR

Comme indiqué sur la **Figure-19- APP INVENTOR**, éditeur de blocks, cette interface est très simple et épurée. En effet, en haut ; on retrouve des éléments classiques «Save», «Undo» et «Redo» ainsi qu'un émulateur. Cette interface contient des parties **Built-in** et **blocks**.

La fenêtre à gauche contient les blocs qui assemblent la partie droite de la fenêtre pour décrire le comportement de notre application. Les blocs peuvent être standards (dans l'onglet « Built-in ») ou définis spécifiquement pour l'application (dans l'onglet « My Blocks »). Aussi, la corbeille est utilisée pour jeter des morceaux de pseudo-code et la loupe sert à changer la taille l'écran de l'éditeur.

Chaque élément contient des blocks : (ces pages décrivent les blocs que nous pouvons utiliser lorsque nous construisons nos applications App Inventor. Dans l'onglet « Built-in », nous retrouvons toujours les mêmes éléments :

- **Définition** : morceaux permettant de définir des procédures (avec/sans résultats/attributs).
- **Texte** : morceaux permettant de traiter le texte. Assimilables au type *char* et à la classe *String* en JAVA.
- **Lists** : morceaux permettant de traiter des listes. Assimilables aux sous-classes de *List* en JAVA.
- **Math** : morceaux permettant de traiter des nombres. Assimilables au type *int* et à la classe *Integer* en JAVA.
- **Logic** : morceaux permettant de traiter des booléens. Assimilables au type *boolean* et à la classe *Boolean* en JAVA.
- **control** : outils permettant d'effectuer de la programmation conditionnelle par exemple : dans l'élément logic.

Lorsqu' on clique sur «My Blocks» en haut et à droite de la page : nous obtenons la figure suivante :

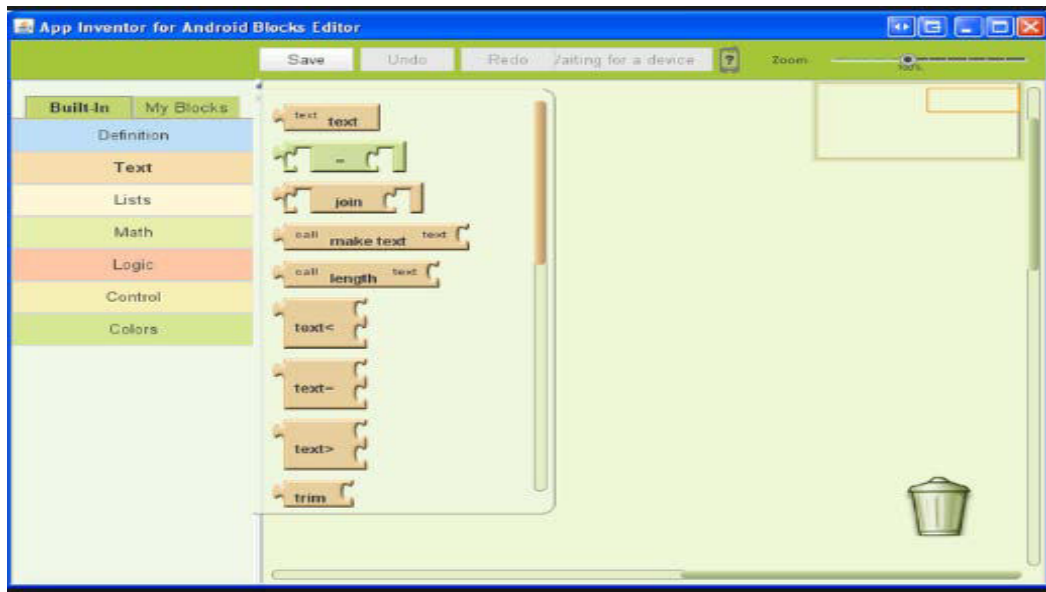


Figure II-20 : Éditeurs de blocks APP INVENTOR

La palette des variables et fonctions est à gauche, l'onglet My blocks propose les fonctions associées aux éléments déposés sur notre écran au préalable. Dans l'onglet « My Blocks », on retrouvera les éléments et leurs accesseurs et fonctions :

- **My Définitions** : variables et procédures globales.
- **Button1** : variables et procédures spécifiques au bouton.
- **Label1** : variables et procédures spécifiques au label.
- **Orientation Sensor1** : variables et procédures spécifiques au capteur.
- **Screen1** : variables et procédures spécifiques à l'écran (assez restreint).

Lorsque on Clique sur l'élément « Button1 » on obtient la figure suivante est cela comme échantillon pour n'importe quel composant associé dans notre application.

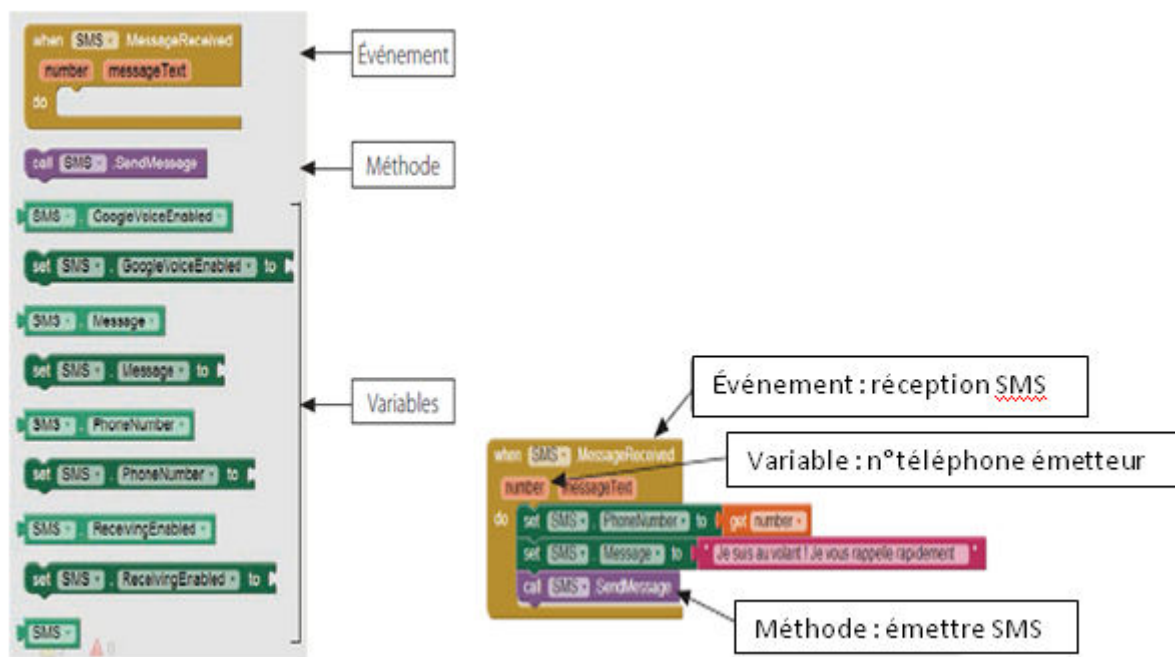


Figure II-21 : Échantillon d'un composant sous APP INVENTOR

A partir de l'onglet «**Package for phone**» ce test assure trois solutions accessibles :

1. L'émulateur : un écran s'affichera sur l'ordinateur.
2. la connexion se fera directement sur le smart phone en wifi.
3. USB : la connexion se fera sur le smart phone via un câble USB.

Pour installer notre application sur un appareil Androïde, on exécute une des trois solutions à partir du bouton "**Package for Phone**" (en haut à droite de l'interface web).



Figure II-22 : L'organisation générale de l'interface de développement

II-5 Conclusion

Dance ce chapitre nous avons présenté les différents matériels et logiciels utilisés dans notre projet pour réaliser : l'application Androïde, la simulation, ainsi les différents programmes.

Le chapitre suivant sera consacré pour la réalisation pratique de notre projet et la simulation de système finale.

Chapitre III

Réalisation

pratique



Realisé par :
M^{elle} INGUEL Malha



III-1 Introduction

Dans ce chapitre nous allons montrer les différentes étapes de la réalisation de notre système, ainsi que les branchements des différents capteurs et actionneurs et leurs différents programmes de configuration. L'organigramme, le déroulement des différentes étapes du programme et la réalisation de l'application androïde seront présentés aussi.

III-2 Réalisation du système

Le Bluetooth est utilisé comme un standard de communication permettant l'échange bidirectionnel de donnée entre la carte Arduino UNO et le Smartphone. La figure ci-dessous montre le schéma synoptique de cette communication.



Figure III-1 : Schéma synoptique de la communication Bluetooth-Android

III-2-1 Les organigrammes de notre système

Avant de passer à la programmation, nous devons réaliser un organigramme qui explique le déroulement des différentes séquences, tant intérieures qu'extérieures : il comportera plusieurs boucles dont la fin d'exécution succède toujours à son commencement.

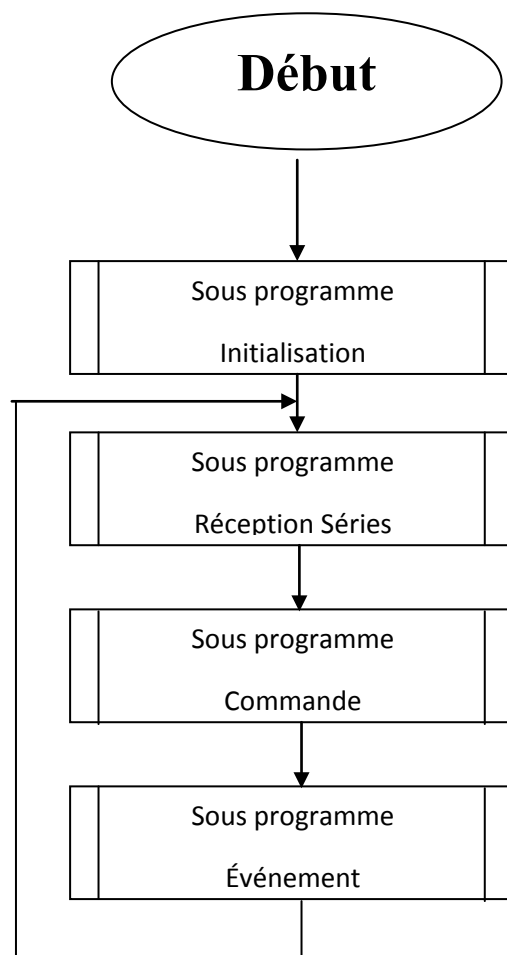


Figure III-2 : Programme principale

N.B Chaque action est symbolisée par un rectangle et chaque test est symbolisé par un losange.

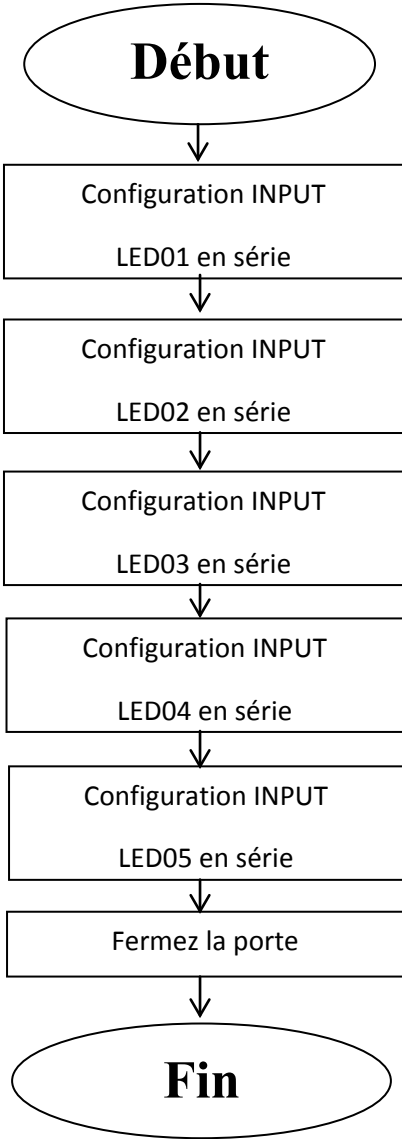


Figure III-3 : Organigramme d’initialisation

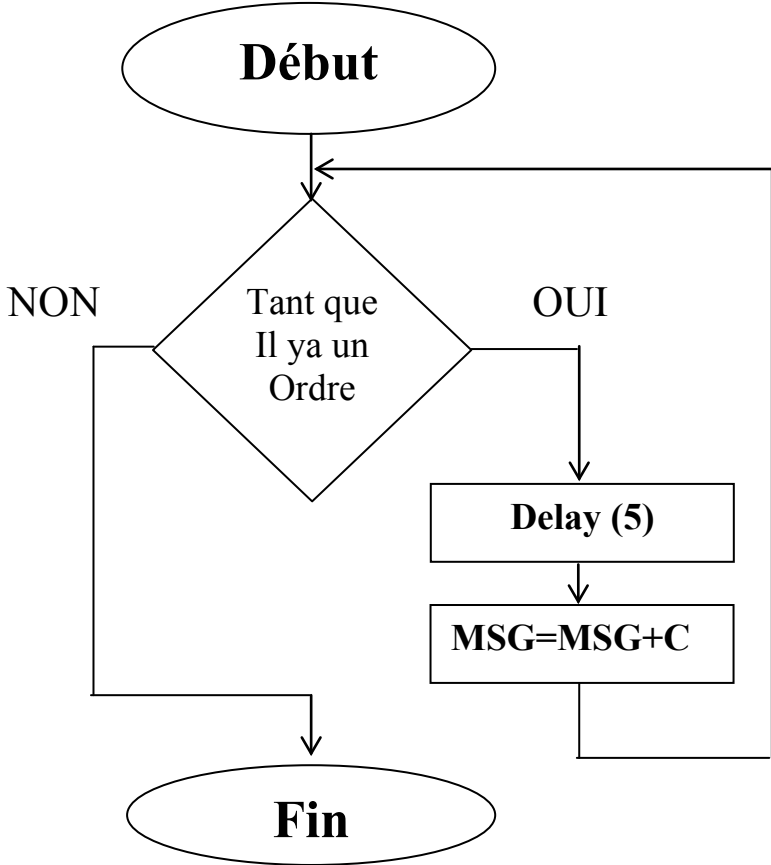


Figure III-4 : Organigramme reception serie

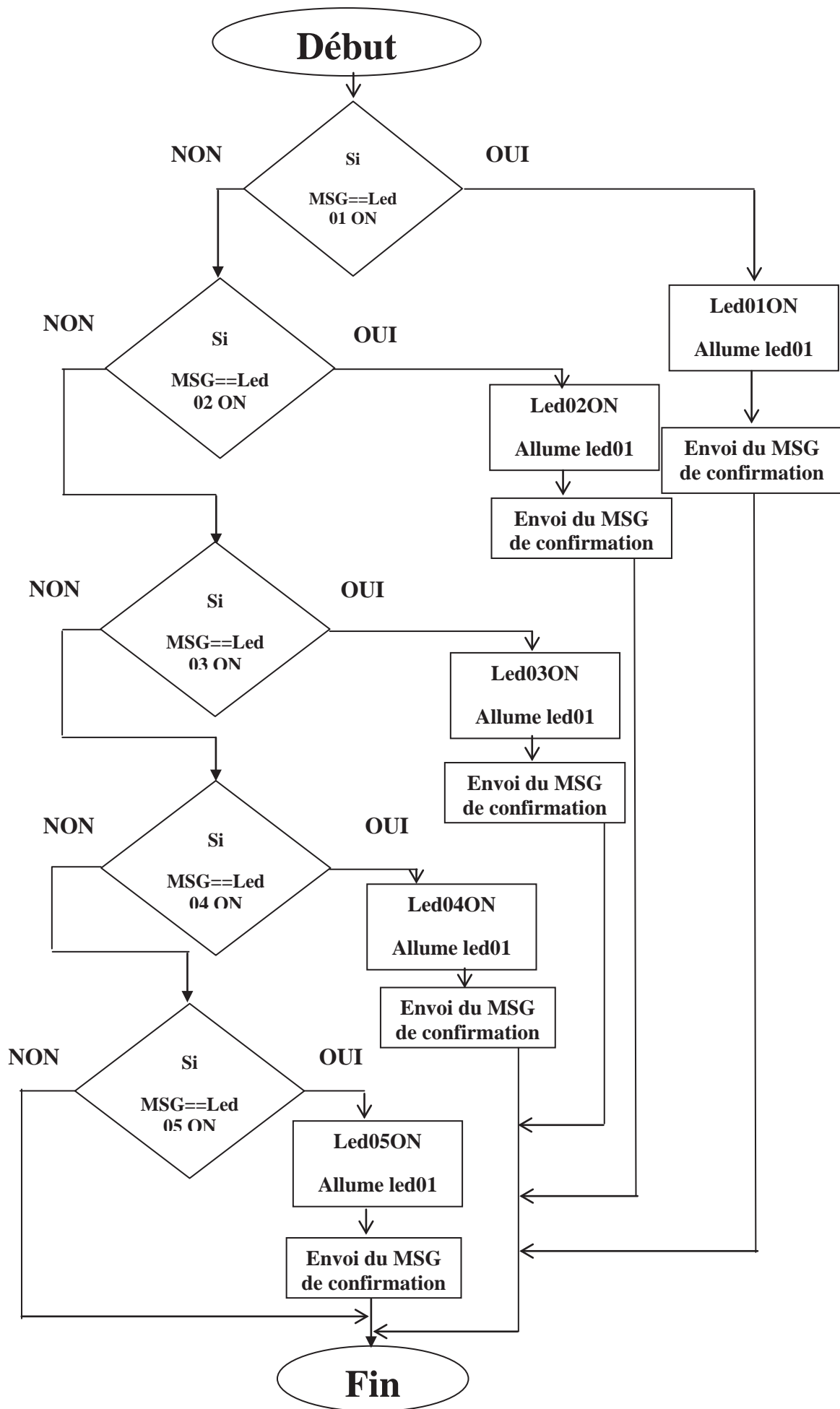


Figure III-5 : Organigramme gestion des lampes

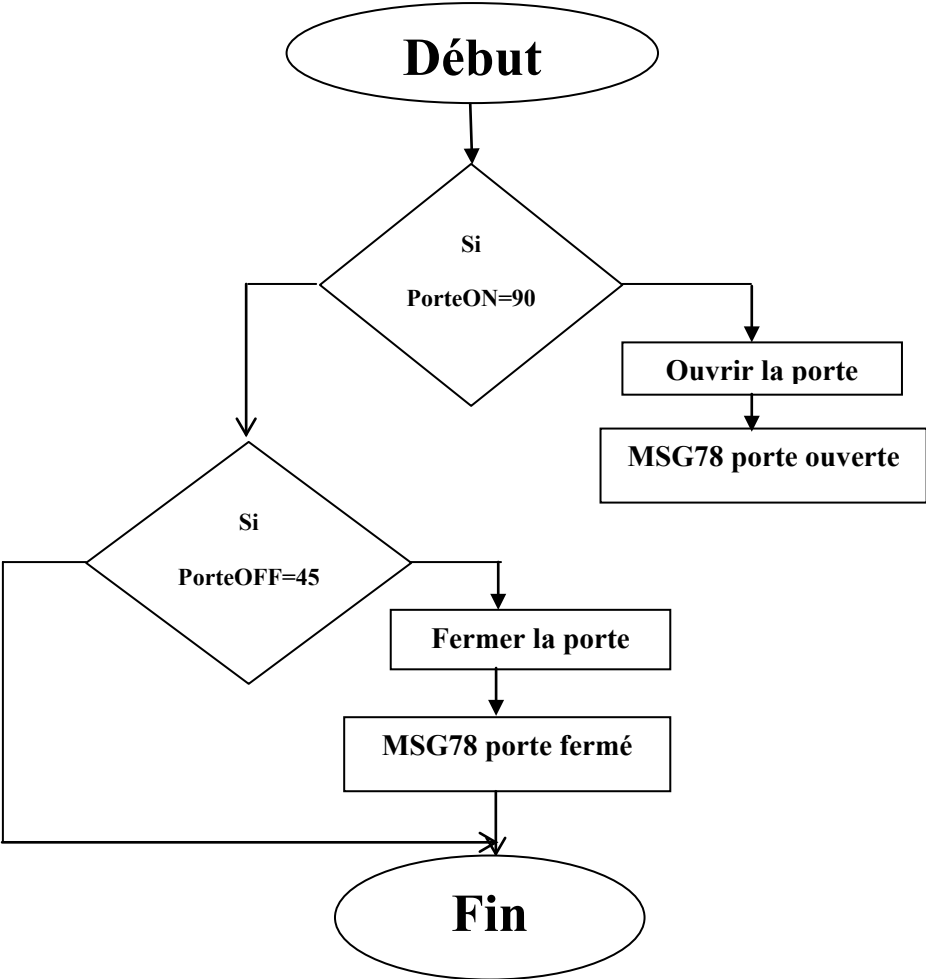


Figure III-6 : Organigramme gestion de porte

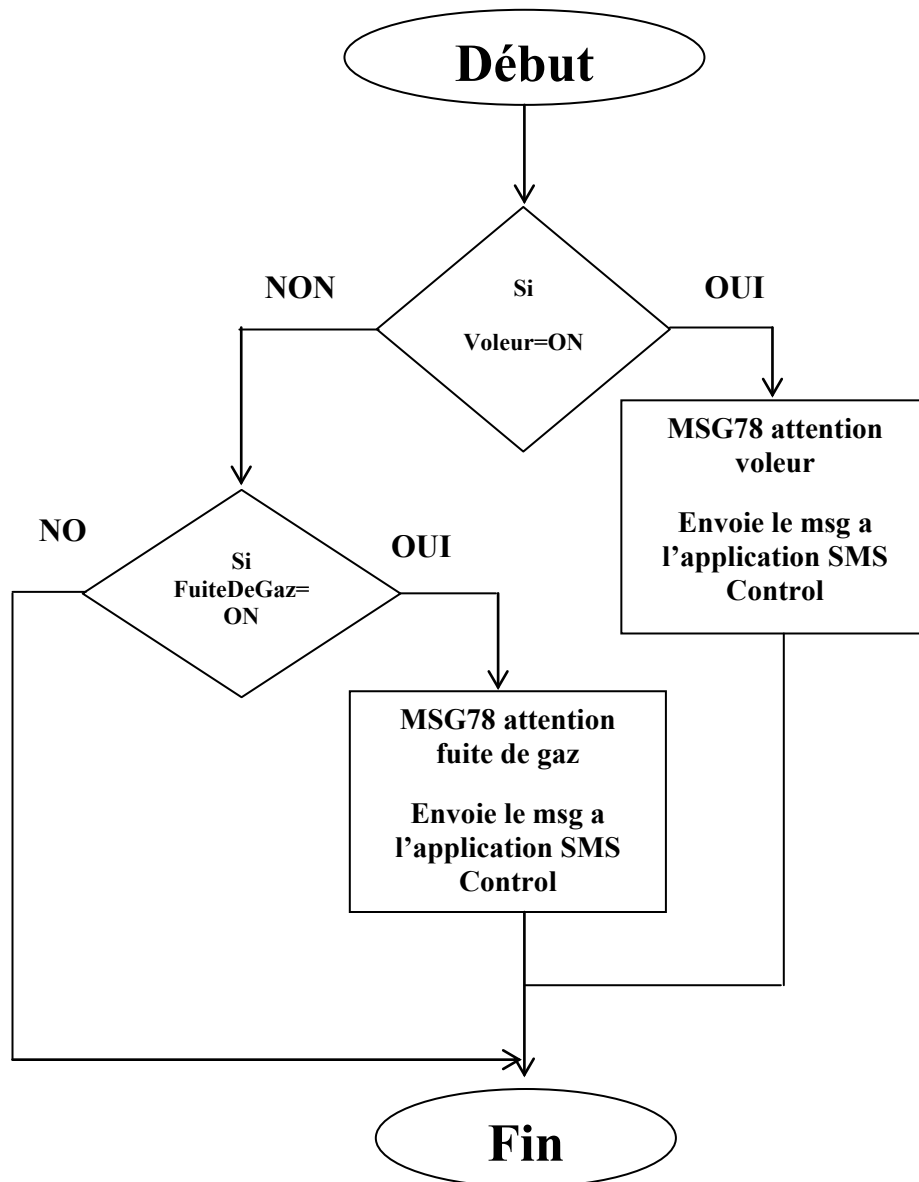


Figure III-7 : Organigramme d'événements

III-2-2 Présentation des programmes sur IDE

Nous avons préféré de photographier l'interface du IDE de l'environnement Arduino dans lequel nous avons simulé notre programme.

```

//#####
void ReceptionSerie() {
    while (Serial.available())
    {
        delay(5);
        char c = Serial.read();
        MSGC = MSGC+c;
    }
}

```

Figure III-8 : Sous programme Reception serie

```

#####
void GestionDeslampes() {
    if(compar("LED01ON")){digitalWrite(LED01,HIGH); Serial.print("MSG78lampe allumer");}
    if(compar("LED02ON")){digitalWrite(LED02,HIGH); Serial.print("MSG78lampe allumer");}
    if(compar("LED03ON")){digitalWrite(LED03,HIGH); Serial.print("MSG78lampe allumer");}
    if(compar("LED04ON")){digitalWrite(LED04,HIGH); Serial.print("MSG78lampe allumer");}
    if(compar("LED05ON")){digitalWrite(LED05,HIGH); Serial.print("MSG78lampe allumer");}
    if(compar("LED01OFF")){digitalWrite(LED01,LOW); Serial.print("MSG78lampe eteinte");}
    if(compar("LED02OFF")){digitalWrite(LED02,LOW); Serial.print("MSG78lampe eteinte");}
    if(compar("LED03OFF")){digitalWrite(LED03,LOW); Serial.print("MSG78lampe eteinte");}
    if(compar("LED04OFF")){digitalWrite(LED04,LOW); Serial.print("MSG78lampe eteinte");}
    if(compar("LED05OFF")){digitalWrite(LED05,LOW); Serial.print("MSG78lampe eteinte");}
}

```

Figure III-9 : Sous programme gestion des lampes

```

//#####
void GestionDePorte() {
    if(compar("PORTEON")) {PORTE_POSITION(90);
    Serial.print("MSG78porte ouverte");}
    if(compar("PORTEOFF")) {PORTE_POSITION(45);
    Serial.print("MSG78porte fermee");}
}

```

Figure III-10 : Sous programme de gestion de porte

```

//#####
bool VOLEUR() {
    bool etat=false;
    if(millis()-CapteurUSMillis>TempsVoleur) {
        TempsVoleur=1000;
        CapteurUSMillis=millis();
        if((CapteurUS()<10)&&(CapteurUS()>1)) {
            VoleurCMP++;    }
    }
    if(VoleurCMP>=5) {
        VoleurCMP=0;
        CapteurUSMillis=millis();
        TempsVoleur=300000;
        etat=true;    }
    return etat;
}

```

Figure III-11 : Sous programme de ditection de Voleur

```

#####
void Evenements() {
    if (VOLEUR()==true) {
        //Serial.print("MSG78VOLEUR");
        Serial.print("MSG78attention voleur");
    }
    if (FuiteDeGAZ()==true) {
        //Serial.print("MSG78GAZ");
        Serial.print("MSG78attention fuite de gaz");
    }
}
}

```

Figure III-12: Sous programme d'évenements

```

#####
void setup() {
    Serial.begin(9600);

    pinMode(LED01,OUTPUT);   pinMode(LED02,OUTPUT);
    pinMode(LED03,OUTPUT);   pinMode(LED04,OUTPUT);
    pinMode(LED05,OUTPUT);

    Serial.println("#####");
    Serial.println("##### SMS CONTROLE ACTIVE... #####");
    Serial.println("#####");
}

```

Figure III-13: Sous programme d'initialisation

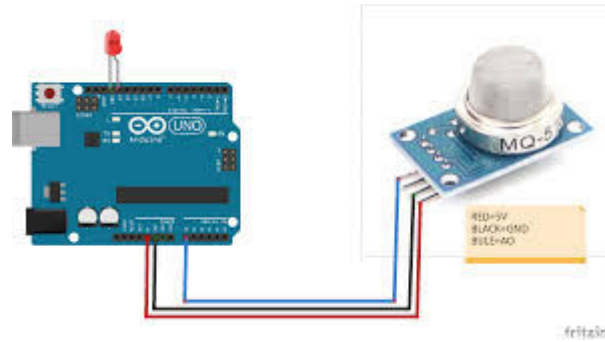


Figure III-17 : Simulation de capture de gaz

III-2-4 Réalisation de l'application APP INVENTOR (Androïde) :

Notre réalisation software a besoin de deux étapes : la première consiste a un programme qui va s'injecter aux microcontrôleurs de la carte Arduino après avoir été convertie par le IDE en code HEX et la deuxième a un programme qui va se manipule sous APP INVENTOR et s'installe sous smart phone.

L'application **SMS contrôle** : c'est une application qui fonctionne sur les tablettes ou Smartphone, ils sont dotée d'un système d'exploitation Androïde avec cette application on peut travailler en deux mode :



Choix du Mode :



Figure III-18 : Choix du mode de commande

- Le mode local qui nous permet d'allumer et d'ouvrir la porte.
- Le mode GSM : qui nous permet d'activer la sécurité et exploiter les capteurs notamment, le capteur ULTRASON HC-SR04 pour vérifier la présence d'intrus est le capteur xxx de gaz qui nous permet de vérifier la présence de gaz.



Figure III-19 : Les différentes interfaces de l'application (Mode local)

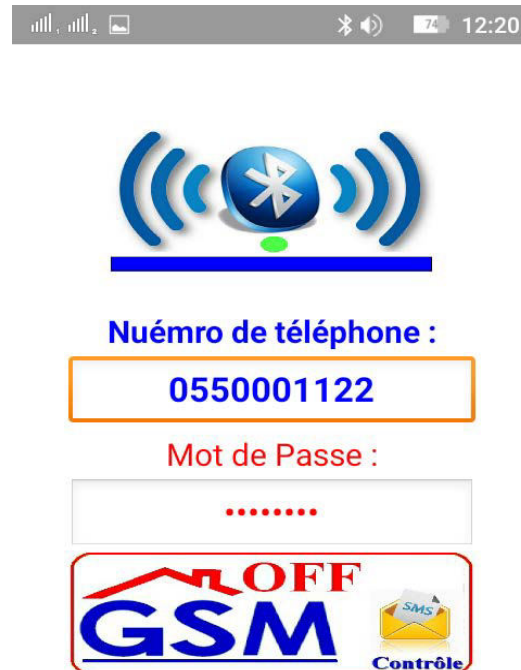
Si on choisit le mode **local** l'application nous donne une interface dotée des boutons pour assurer la gestion des lampes et de la porte. (**Figure III-19**)

Ex : si on clique sur le bouton de la lampe1 l'application traite cet événement et envoie un message « LED01ON » ce message sera envoyé à la carte Arduino, via la liaison Bluetooth lorsque le message est reçu par la carte UNO, cette dernière allume la LED01.

Et si on clique sur le même bouton de l'application SMS contrôle, cette application traite l'événement et bascule le msg « LED01OFF », est ce dernier sera envoyé à la carte arduino via la communication Bluetooth. La carte arduino qui a déjà allumé la « LED01 »Lorsque elle reçoit ce message « LED01OFF » elle va éteindre la LED01.

Et sera le même fonctionnement pour les différentes lampes et la porte.

Si on choisit le mode GSM : une interface apparaît, cette dernière nous permet d'activer ou de désactiver le mode GSM.



Si on active le mode GSM, l'application SMS contrôle, donne la main à la carte arduino pour assurer la protection de notre maison, (des voleurs et en cas ou de fuite de gaz).

Lorsque le capteur ultrason capte un intrus, donne l'information à la carte arduino, cette information, sera traitée par cette dernière, si elle juge qu'il y a un voleur chez soi, automatiquement en temps réel, la carte arduino envoie un message à l'application SMS contrôle via le Bluetooth.

Lorsque l'application SMS contrôle reçoit un message de la carte arduino ; qui signifie la présence d'un voleur, cette application envoie automatiquement un message « Attention monsieur, vous avez un voleur chez vous, Appelez la police », au numéro qui a été déterminé à l'avance dans cette application.

Et enfin, le propriétaire de la maison, quand il est loin de sa maison, notre système va assurer la sécurité de sa maison contre les cambrioleurs, il sera la même chose en cas ou de fuite de gaz, notre système donne aux propriétaires de la maison, la bonne gestion via GSM, et cela :

Quelque soit la distance qu'il l'éloigne, peut donner des commandes à l'application SMS contrôle via SMS accompagnée d'un mot de passe.

III-3 Conclusion

Dans ce chapitre nous avons illustré les différentes étapes de la réalisation pratique de notre système : les organigrammes et les programmes des différents capteurs et actionneurs ainsi que la réalisation de l'application qui automatise notre maison.

Sur le plan pratique, une manipulation adéquate du logiciel « Arduino » nous permet alors d'utiliser un compilateur ; il s'agit de « IDE » ce dernier possède une capacité de créer un code HEX, qui peut être injecté sur son microcontrôleur d'une part ; d'une autre part une simple élaboration de l'environnement App Inventor qui nous pousse à réaliser une application « *.apk » sous smart phone capable de lier une carte Arduino et un smart phone afin d'exécuter des ordres bien définis. On peut conclure que les résultats obtenus sont satisfaisants compte tenu des limitations du matériel et des moyens dont nous disposons.

Conclusion

générale

Conclusion générale :

La domotique est un système révolutionnaire appelé à évoluer encore plus dans le futur. Ce système permet de contrôler une maison entièrement depuis de simples boîtiers disposés un peu partout dans la maison. Le fait de pouvoir relier ces boîtiers à un Smartphone est aussi incroyable car cela permet de contrôler sa maison à distance par exemple depuis son travail ! Ce système pourra donc s'avérer très utile dans le futur, et ce pour une question de confort et sécurité des personnes.

Avant tout, la domotique permet d'améliorer le mode de vie en augmentant le confort et la sécurité de notre domicile, et le but de notre projet est d'assurer ces avantages.

Et pour cela, dans notre travail nous avons pu réaliser un système domotique dont la surveillance et la commande se fait via la technologie GSM, ce système est composé d'une carte électronique qui se communique avec une application qu'on a réalisée sur Smartphone via le Bluetooth, notre application nous permet de recevoir des SMS pour commander notre système via un autre téléphone et envoyer des SMS de notification sur l'état de la maison.

Enfin, ce travail peut être optimisé en incorporant une caméra de surveillance et des électrovannes permettant une sécurité adéquate et de crypter les msg qui s'écoulent entre la carte Arduino et l'application SMS Control.