

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



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

Mémoire de Fin d'Etudes

En vue de l'obtention du diplôme

Master Académique en Electronique
Option : Instrumentation

Thème

Etude et conception d'une carte de contrôle à base
d'ARDUINO pour des applications domotique

Proposé et dirigé par:

M^r. KHELIFA Mohammed-Amin

Présenté par:

AMMOUR Mouloud

Soutenu le : 26/09/2018

Promotion 2018

Dédicace

Je dédie ce modeste travail à:

- **Mes chers parents**
- **Mon frère et ma sœur**
- **Toute ma famille**
- **Mes amis**

AMMOUR Mouloud

Remerciements

Je tiens à remercier vivement mon promoteur Mr.KHELIFA Mohamme-Amin pour ses précieux conseils et son orientation durant tout le long de mon travail.

Je tiens à remercier également tous ceux qui m'ont aidé et contribué de près ou de loin à la réalisation de ce travail.

Je tiens à remercier également tous les enseignants du département Electronique qui ont assuré ma formation pendant mon cycle d'étude.

Je remercie enfin les membres de jury qui me feront l'honneur de juger mon travail.

Sommaire



Table des matières

Table des matières	i
Liste des figures	iv
Liste des tableaux	vii
Introduction générale	1

Chapitre I

Généralités sur l'Internet des Objets IOT

1. Introduction	3
2. Histoire d'internet des objets	4
3. L'objet connecté	6
4. Visions de l'Internet des Objets	8
4.1 La vision orientée objet	8
4.2 La vision orientée Internet	9
4.3 La vision orientée sémantique	9
5. Composants d'un système IOT	9
6. Application de l'Internet des Objets	11
6.1 Ville intelligente	11
6.1.1 Le transport :	12
6.1.2 L'énergie - les Réseaux Intelligents	12
6.1.3 La domotique	13

6.1.4 Les environnements intelligents	13
6.2 La Médecine Intelligente (e-Santé)	14
6.3 Sport	14
6.4 Agriculture Intelligente.....	14
6.5 Industrie Intelligente.....	15
7. Vulnérabilités et menaces dans l'internet des Objets	15
8. Conclusion.....	16

Chapitre II

La carte de Contrôle à base d'ARDUINO

1. Introduction	17
2. Historique.....	17
3. Description de la Carte ARDUINO	18
4. Les différents types de l'ARDUINO	18
5. Domaines d'applications.....	21
6. Les avantages de l'Arduino	22
7. Arduino Uno, pourquoi ?.....	23
8. Partie matérielle	25
9. Partie logiciel	29
9.1 Structure générale du programme (IDE Arduino)	29
9.2 Injection du programme	30
9.3 Description du programme.....	31
9.4 Les variables	32
9.5 Configuration des entrées et des sorties void setup ()	33
9.6 Programmation des interactions void loop ().....	33
9.7 Les étapes de téléchargement du programme.....	34
10. Les Accessoires de la carte Arduino	34
10.1 Le module Arduino Bluetooth.....	35
10.2 Shield Arduino Wifi	34
10.3 XBee.....	35
10.4 Infrarouge	35
10.5 Radio.....	35
10.6 Les capteurs	35

10.7 Les afficheurs LCD	35
10.8 Les relais :	36
11. Conclusion.....	36

Chapitre III

Structure matérielle et logicielle du projet

1. Introduction	37
2. Structure du systeme.....	37
3. Le bloc principal “l’acquisition des données de la domotique”	37
4. Description du système	38
4.1 Partie matérielle	38
4.1.1 Bluetooth	38
4.1.2 Capteur de mouvement.....	40
4.1.3 Capteur de Gaz MQ-2.....	41
4.1.4 Photorésistance (LDR)	42
4.1.5 Le capteur de température LM335.....	43
4.1.6 Le relais	44
4.1.7 Le Servomoteur	44
4.2 La partie logicielle	46
4.2.1 Fritzing	46
4.2.2 Android	47
4.2.3 App inventor	47
5. Conclusion.....	48

Chapitre IV

Conception de la carte du contrôle

1. Introduction	49
2. Conception du système.....	49
2.1 La carte Arduino Uno	49
2.2 Bluetooth	49
2.3 Le capteur de gaz MQ-2	51
2.4 Le capteur de lumière (LDR)	52

2. 5 Le détecteur de mouvement HC-SR 501	54
2. 6 Le capteur de température LM 335.....	55
2.7 Présentation de l'application APK crée sur APP INVENTOR.....	59
3. Conclusion.....	60

Conclusion générale.....	61
-------------------------------------	-----------

Bibliographie.....	63
---------------------------	-----------



LISTE DES FIGURES

Chapitre I

Figure. I.1 Dimensions de l'Internet des objets [5].	3
Figure. I.2 Historique de la technologie IOT.	5
Figure. I.3 Pr�vision Objets Connect�s.	5
Figure. I.4 la lampe DAL.	7
Figure. I.5 thermostat Qivivo.	7
Figure. I.6 Visions de l'Internet des objets.	8
Figure. I.7 Application de l'Internet des Objets.	11
Figure. I.8 Une ville intelligente.	12
Figure. I.9 Les R�seaux Intelligents.	13

Chapitre II

Figure. II.1 Caract�ristiques de la Carte ARDUINO UNO.	24
Figure. II.2 Composition de la Carte ARDUINO UNO [10].	25
Figure. II.3 Composition du μ C ATmega328 [10].	26
Figure. II.4 Interface de programmation d'Arduino.	30
Figure. II.5 Choix de la carte d'Arduino.	31
Figure. II.6 Edition des programmes d'Arduino.	33

Chapitre III

Figure. III.1 Schéma bloc du système.....	38
Figure. III.2 Module Bluetooth HC-06.....	39
Figure. III.4 Capteur de Gaz MQ-2.....	41
Figure. III.5 Photorésistance (LDR).....	42
Figure. III.6 Graphe de la résistance en fonction de la lumière.....	42
Figure. III.7 Capteur de température LM 335.....	43
Figure. III.8 Le module relais.....	44
Figure. III.9 Servomoteur.....	45
Figure. III.10 L'interface du logiciel Fritzing.....	46
Figure. III.11 Le Logo d'Arduino.....	47
Figure. III.12 L'interface graphique d'App Inventor.....	47

Chapitre IV

Figure. IV.1 Représentation graphique de la communication par Bluetooth HC-06.....	50
Figure. IV.2 Schéma électrique de la communication par Bluetooth HC-06.....	50
Figure. IV.3 Représentation graphique du système de détection de gaz (capteur de gaz MQ-2).....	51
Figure. IV.4 Schéma électrique du système de détection de gaz (capteur de gaz MQ-2).....	51
Figure. IV.5 L'organigramme fonctionnel du système de détection de gaz.....	52
Figure. IV.6 Représentation graphique du système d'éclairage automatique externe.....	53
Figure. IV.7 Schéma électrique du système d'éclairage automatique externe.....	53
Figure. IV.8 L'organigramme fonctionnel du système d'éclairage extérieur.....	54
Figure. IV.9 Représentation graphique du système d'éclairage automatique interne.....	55
Figure. IV.10 L'organigramme fonctionnel du système d'éclairage intérieur.....	56
Figure. IV.11 Représentation graphique du branchement du capteur de température LM 335.....	57
Figure. IV.12 Schéma électrique du branchement du capteur de température LM335.....	57
Figure. IV.13 L'organigramme fonctionnel du système de régulation semi-automatique de température.....	58
Figure. IV.14 Page de commande de l'application mobile.....	59
Figure. IV.15 Suite de page de commande de l'application mobile.....	60



LISTE DES TABLEAUX

Tableau. I.1 Composants d'une solution IOT.....	10
Tableau. II.1 Les différents types de l'ARDUINO.....	21
Tableau. III.1 Caractéristiques du Module Bluetooth HC-06.....	40
Tableau. III.2 Caractéristiques du MQ-2.	41
Tableau. III.3 Caractéristiques du LM 335.....	43

Introduction

générale

Introduction générale

Actuellement, avec l'évolution continue des technologies de l'électronique et de l'informatique, les systèmes électroniques isolés tendent à leur disparition. Ils deviennent alors, de plus en plus obsolètes et remplaçables par une nouvelle génération d'objets intelligents, interconnectés : le phénomène d'internet des objets a vu le jour et qui ne fait que commencé. Ainsi l'interconnexion de ces objets de capacités avancées va conduire le monde à une révolution en termes de disponibilité de services, dans de multiples domaines de la vie quotidienne. L'un des domaines les plus affectés par l'émergence de cette technologie est les maisons connectées et la domotique.

Rendre sa maison intelligente, c'est d'abord apporter confort et sécurité aux occupants. À travers des équipements commandés à distance, il est possible de modifier la température, de contrôler l'éclairage ou de vérifier que personne ne rentre chez vous en votre absence. En quelques mots, une maison intelligente est une habitation dont les différents éléments (chauffage, éclairage, multiprises, alarmes, appareils de vidéosurveillance...) sont pilotables depuis des applications mobiles, disponibles sur Smartphone ou sur tablette. Et c'est grâce à ces objets qualifiés de connectés et la domotique que la maison connectée est devenue réalité.

La domotique est un système intelligent, conçu avant tout pour optimiser et automatiser des fonctions techniques et de communication de la maison. Les appareils électriques sont reliés entre eux et peuvent interagir pour réduire la consommation d'énergie, tout en assurant le confort à domicile.

C'est par cette idée qu'on s'est penché par ce projet de fin d'études vers un des domaines de l'internet des objets qui est la domotique et on a décidé de réaliser une conception d'une carte de contrôle à base d'ARDUINO pour des applications domotique.

Pour une bonne présentation de notre mémoire, notre travail est réparti essentiellement en quatre chapitres :

Le premier chapitre est réservé à des généralités sur l'internet des objets. On va donner un petit aperçu sur la définition de l'objet connecté ainsi que les différentes applications de l'internet des objets.

Introduction générale

Dans le deuxième chapitre, on va parler des solutions pour installer un système domotique utilisant une carte électronique à base de microcontrôleur ARDUINO Uno, en faisant une présentation de ses composants, ses domaines d'utilisation, ainsi qu'une description de sa partie logicielle.

Dans le troisième chapitre, les différentes parties à acquérir et à concevoir pour l'aboutissement de ce projet seront détaillées, aussi un petit aperçu sur l'environnement de Fritzing et App inventor (Une description logicielle) est donné.

Finalement, le dernier chapitre, sera consacré à la conception des applications domotiques, commandées par une application Android mobile.

Chapitre I :
Généralités sur
l'Internet des Objets
IOT

1. Introduction

IOT est l'acronyme de « Internet Of Things », ou internet des objets en français. IOT est l'extension d'internet qui n'était qu'un monde virtuel et qui n'interagissait pas ou peu avec le monde physique à des entités et des emplacements existants sur terre possédant une adresse IP via un réseau de communication. Les données générées par ces entités (objets) sont échangées via internet afin d'être exploitées dans divers domaines ; la santé, la domotique, l'agriculture...etc. Les géants de l'informatique parlent de maisons connectées, de villes intelligentes, et de véhicules autonomes [1].

Cette vision de l'Internet des objets introduira une nouvelle dimension aux technologies de l'information et de la communication : en plus des deux dimensions temporelle et spatiale qui permettent aux personnes de se connecter de n'importe où à n'importe quel moment, nous aurons une nouvelle dimension "objet" qui leur permettra de se connecter à n'importe quel objet [5].

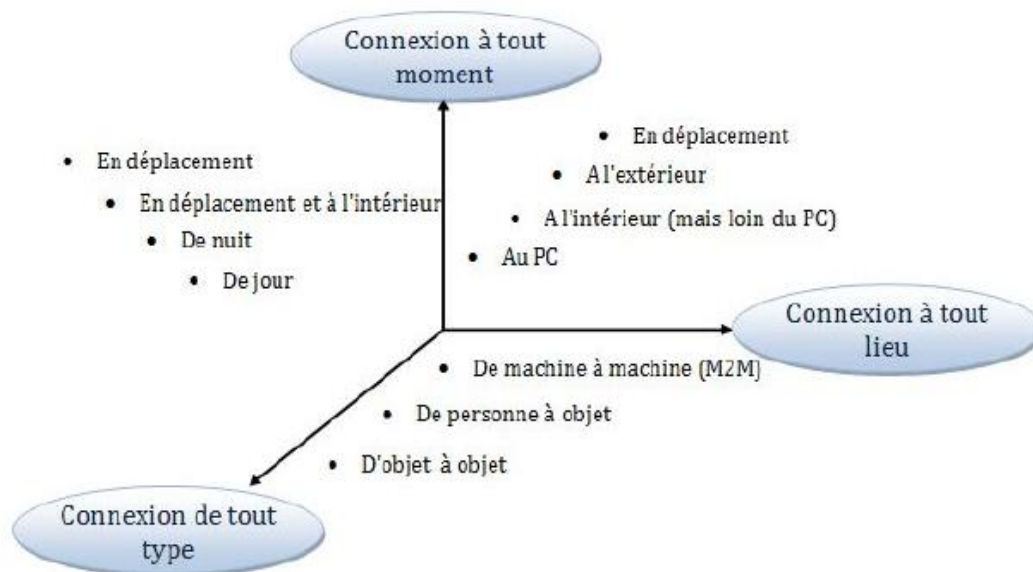


Figure. I.1 Dimensions de l'Internet des objets [5].

Cette évolution est induite par trois moteurs :

- ✓ la prolifération des objets connectés.
- ✓ la croissance exponentielle des applications multiplateformes.
- ✓ la maturité des technologies réseau capables de connecter des milliards de dispositifs, de manière économique et sans efforts [1].

Ce chapitre est consacré à des généralités sur l'électronique embarquée, la domotique, l'internet des objets, les applications IOT, et particulièrement les maisons connectées.

2. Histoire d'internet des objets

L'Internet des Objets est une concrétisation technique de l'informatique où la technologie est intégrée naturellement aux objets du quotidien. Très prometteur, ce concept ouvre la voie vers une multitude de scénarios basés sur l'interconnexion entre le monde physique et le monde virtuel : domotique, e-santé, ville intelligente, logistique, sécurité, etc.

Cependant, comme d'autres concepts prometteurs, celui-ci fait face à un certain nombre de problématiques techniques et non techniques qui nécessitent d'être étudiées pour permettre à l'Internet des Objets d'atteindre son plein potentiel. L'Internet des Objets est un concept concrétisant la vision de l'informatique telle qu'imaginée en 1991 par Mark Weiser, où la technologie deviendra efficace peu à peu dans l'environnement des utilisateurs, intégrée naturellement à l'intérieur des objets du quotidien.

La technologie n'est plus alors représentée par un objet unique, l'ordinateur personnel, mais se présente au contraire sous la forme d'appareils spécialisés et simples d'emploi, capables de communiquer au travers de plusieurs types de réseaux sans fil : liseuses numériques, télévisions et montres connectées, ordinateurs de bord, téléphones intelligents, etc.

À l'origine, le terme Internet des objets a été utilisé pour la première fois en 1999 par Kevin Ashton pour décrire des objets équipés de puces d'identification par radiofréquence (ou puce RFID). Chaque objet, identifié de manière unique et universelle, peut alors être rattaché à un ensemble d'informations le concernant, ces dernières étant lisibles par d'autres machines. Le concept a toutefois évolué avec le temps et s'est généralisé vers une approche consistant à connecter un très grand nombre d'objets du quotidien au réseau Internet, les dotant ainsi d'une identité propre et leur permettant, entre autres, d'offrir des services et de collecter des informations de manière autonome [1].

La figure suivante représente bien l'historique de la technologie de l'internet des objets.

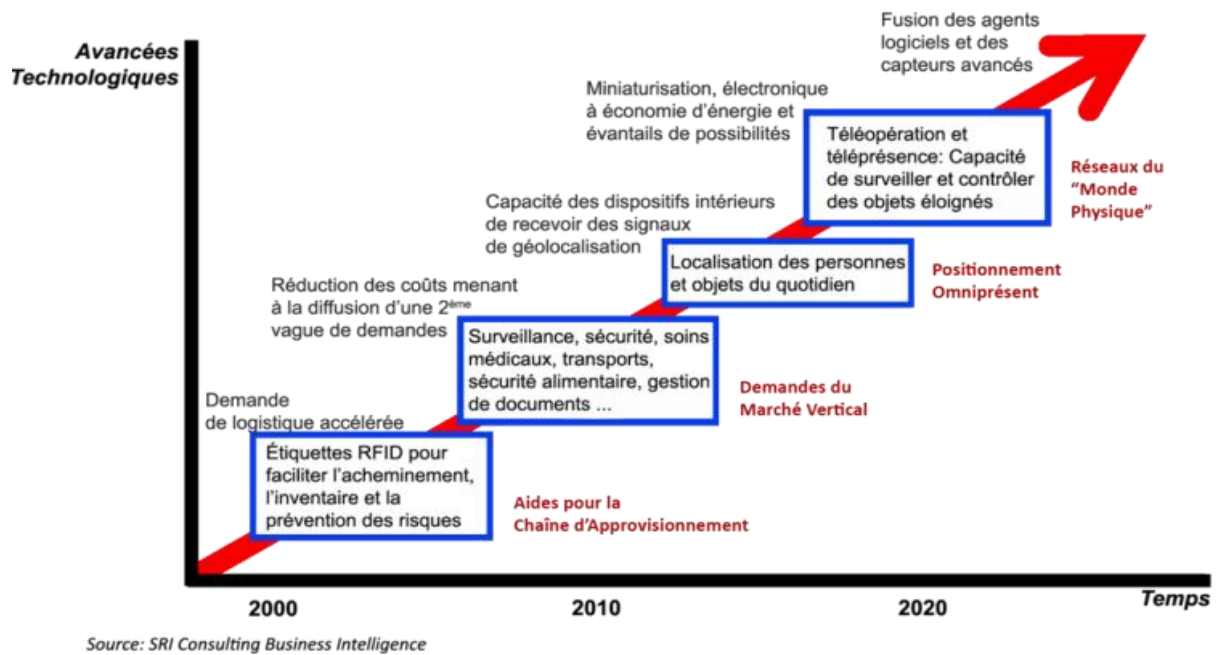


Figure. I.2 Historique de la technologie IOT.

De nos jours, les objets connectés sont partout. D’après les prévisions Cisco, le nombre d’objets connectés est estimé à 50 milliards en 2020, soit plus de 6 fois le nombre d’êtres humains sur terre. Cet important développement est dû à la demande toujours de plus en plus grande ainsi qu’au grand nombre de domaines dans lesquels les objets connectés s’insèrent. Cette importante évolution peut être également mise en relation avec la recrudescence du nombre de smartphones. Leurs usages sont divers et variés, mais avec souvent pour but de rendre notre quotidien plus facile [3].



Figure. I.3 Prévision Objets Connectés.

3. L'objet connecté

Un objet est, avant toute chose, une entité physique ; par exemple, une lampe, une voiture, une cigarette électrique ou un ordinateur .

Il est difficile de définir c'est quoi un objet connecté, tant qu'il englobe d'innombrables choses.

La définition la plus proche de ce terme est la suivante : un objet connecté est un matériel électronique qui peut communiquer des informations avec un Smartphone, une tablette tactile et/ou un ordinateur via une liaison sans fil, Bluetooth ou WiFi, etc., ainsi de pouvoir récupérer, stocker, transférer et traiter, sans discontinuité entre les mondes physiques et virtuels. ses caractéristiques pouvant évoluer au cours du temps (position, niveau de batterie,... etc [1].

Il est raisonnable de considérer que l'Internet des objets est composé d'objets actifs, capables d'accomplir des calculs, d'effectuer des mesures sur l'environnement ou d'influer sur celui-ci, et d'objets passifs qui n'ont pas d'autres aptitudes que celles d'être suivis et détectés par des objets actifs :

Objet passif : Par extension, l'identité d'un objet passif n'est pas directement stockée dans celui-ci, à l'exception de l'identifiant, et nécessite l'utilisation d'une infrastructure tierce capable de stocker ces informations.

Objet actif : Au contraire, un objet actif peut stocker tout ou partie de son identité et échanger directement ces informations avec d'autres objets actifs [1].

À noter que le premier objet connecte était la lampe DAL (Figure I.4), lancé en 2003 par *Rafi Haladjian*. Sensible au toucher et au bruit, cette lampe communiquait des informations sur la météo, la bourse, la pollution, des alertes Google et même des messages grâce à neuf LED de couleur [2].



Figure. I.4 la lampe DAL.

Les fonctions proposées aujourd'hui vont beaucoup plus loin que la simple annonce de la météo. Prenons l'exemple du thermostat Qivivo (Figure I.5), qui permet non seulement de piloter son chauffage à distance, mais également d'obtenir un diagnostic de sa consommation d'énergie, des conseils d'optimisation et même d'être mis en relation avec des professionnels. Les usages ont donc su se développer et, aujourd'hui, les objets connectés sont partout. Ils sont particulièrement appropriés dans certains domaines. afin d'étudier les possibilités d'amélioration de son logement [2].



Figure. I.5 thermostat Qivivo.

4. Visions de l'Internet des Objets

L'Internet des Objets émerge des travaux menés dans plusieurs domaines de recherche dont les différentes « visions » convergent vers un même but. On identifie trois visions majeures : une vision orientée objet, une vision orientée Internet et une vision orientée sémantique. Comme montré sur la Figure I.6, chaque vision possède ses concepts et ses technologies clés.

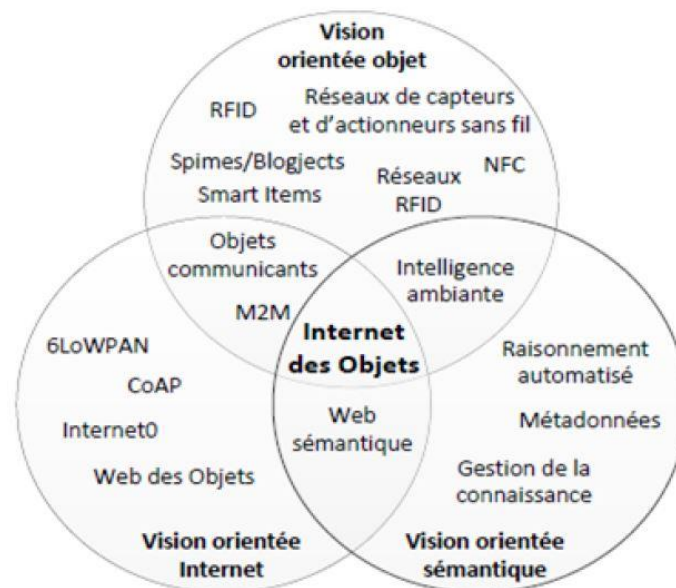


Figure. I.6 Visions de l'Internet des objets.

4.1 La vision orientée objet

Cette vision met l'accent sur les objets physiques et les systèmes qui y sont embarqués, avec notamment les problématiques suivantes :

- identifier les objets de manière unique et, de fait, les doter d'une identité propre ;
- permettre aux objets d'acquérir des informations sur leur environnement en utilisant des capteurs, pour ensuite échanger et traiter ces informations afin d'influer sur cet environnement au travers d'actionneurs.

L'Internet des objets est vu comme le concept émergeant de la mise en réseau des différents objets dans le but de faciliter les interactions entre les êtres humains et les objets, ainsi que les interactions entre objets eux-mêmes.

4. 2 La vision orientée Internet

Elle suggère que cette interconnexion devrait se faire spécifiquement au travers du réseau Internet, considérant que celui-ci est déjà un « réseau de réseaux » qui connecte un très grand nombre de machines au moyen de protocoles standards conçus pour de nombreux usages.

Cette vision s'attache donc à étudier comment le protocole IP (Internet Protocol) peut être adapté pour des systèmes embarqués caractérisés par de faibles ressources matérielles, notamment au travers de nouveaux standards. Par extension, si l'on connecte les objets physiques au réseau Internet, il devient possible de les intégrer aussi au Web pour bénéficier de ses avantages (Web des objets).

4. 3 La vision orientée sémantique

Elle se détache des problématiques techniques propres aux objets physiques et aux réseaux permettant de les faire communiquer, pour se concentrer sur la représentation, l'organisation et le stockage des données relatives à l'Internet des Objets, notamment :

- les identités de chaque objet physique (caractéristiques, états passés et présents, etc.) et les relations qui peuvent unir ces objets entre eux ;
- les flux d'informations acquis par ces objets sur leur environnement, par exemple au moyen de capteurs [1].

5. Composants d'un système IOT

L'internet des objets n'est pas une technologie à part-entière mais plutôt un système intégrant plusieurs autres systèmes. Lier un objet ou un lieu à Internet est un processus plus complexe

que la liaison de deux pages Web. Divers composant sont de mise, L'IOT en exige sept :

- 1) Une étiquette physique ou virtuelle pour identifier les objets et les lieux.
- 2) Un moyen de lire les étiquettes physiques, ou de localiser les étiquettes virtuelles.
- 3) Un dispositif mobile (smartphone, tablette, ordinateur portable).
- 4) Un logiciel additionnel pour le dispositif mobile.
- 5) Un réseau sans fil de type 2G, 3G ou 4G.
- 6) L'information sur chaque objet lié.
- 7) Un affichage pour regarder l'information sur l'objet lié.

Le tableau suivant résume les principaux systèmes technologiques nécessaires à l'implantation d'une solution IOT.

Tableau. I.1 Composants d'une solution IOT.

Type de système	Enjeux	Technologies employées
Identification	Reconnaître chaque objet de façon unique et recueillir les données stockées au niveau de l'objet.	Code-barres, URI, GPS, radioidentification(RFID), ADN...etc.
Capteurs	Recueillir des informations présentes dans l'environnement pour enrichir les fonctionnalités du dispositif.	Luxmètre, capteur de proximité, thermomètre, hydromètre, accéléromètre, gyroscope...etc.
Connexion	Connecter les systèmes entre eux.	Câbles, fréquences radio, Bluetooth, Wi-Fi, ZigBee, ZWave,NFC...etc.
Intégration	Intégrer les systèmes pour que les données soient transmises d'une couche à l'autre.	Middleware (simple et complexe), analyse décisionnelle des systèmes complexe.
Traitement de données	Stocker et analyser les données pour lancer des actions ou pour aider à la prise de décisions.	Tableur, base de données, entrepôt de données, progiciel de gestion intégré(PGI).
Réseaux	Transférer les données dans les mondes physiques et virtuels.	Internet

6. Application de l'Internet des Objets

L'Internet des Objets est influencé par les différents scénarios d'utilisation qui sont considérés aujourd'hui par les mondes scientifique et industriel. Quelques exemples courants sont présentés dans la figure I.7 à titre d'illustration. Chacun d'entre eux possède ses propres objectifs et ses propres contraintes, d'où l'intérêt de considérer l'Internet des objets non pas sous le prisme d'une application spécifique, mais comme la combinaison des besoins d'une multitude de scénarios. Ces scénarios peuvent prendre place dans des espaces privés, au sein desquels les interactions avec l'extérieur sont strictement contrôlées, ou dans des espaces publics ouverts à tous (par exemple, la détection des places de parking vides), avec toute la gamme de nuances qui peut exister entre les deux.

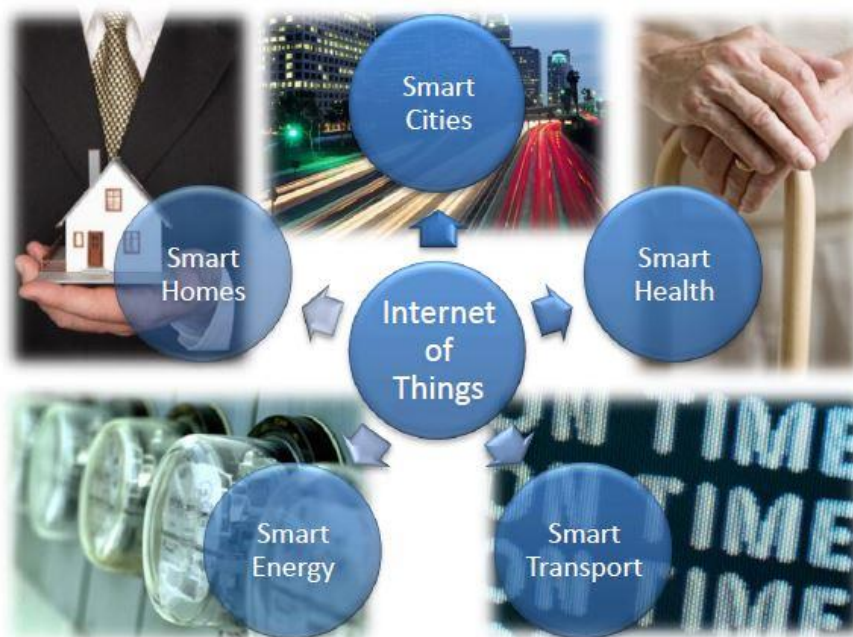


Figure. I.7 Application de l'Internet des Objets.

Parmi les domaines d'application d'Internet des objets, on peut citer :

6.1 Ville intelligente

Une ville intelligente ou encore « smart city » en Anglais, désigne une ville utilisant les technologies de l'information et de la communication (TIC) pour « améliorer » la qualité des services urbains ou encore réduire ses coûts. Ce concept désigne un type de développement urbain apte à répondre à l'évolution ou l'émergence des besoins des institutions, des entreprises et des citoyens, tant sur le plan économique, social, qu'environnemental [4].

Une ville intelligente serait capable de mettre en oeuvre une gestion des infrastructures : Les parcmètres intelligents, les éclairages de rue connectés, les systèmes de distribution d'eau équipés d'instruments et les réseaux électriques intelligents permettront de réduire le trafic, d'économiser l'énergie et l'eau et de plus adapter le réseau électrique aux énergies renouvelables [7].



Figure. I.8 Une ville intelligente.

Parmi les important domaine de la ville intelligente on peut citer :

6.1.1 Le transport :

Dans ce domaine l'IdO appuiera les efforts actuels autour des véhicules intelligents au service de la sécurité routière et l'aide à la conduite. Cela portera sur la communication inter-véhicule et entre véhicules et infrastructure routière. L'IdO constituera ainsi un prolongement naturel des « systèmes de transport intelligents » et leurs apports en termes de sécurité routière, confort, efficacité de la gestion du trafic et économie du temps et de l'énergie [5]

6.1.2 L'énergie - les Réseaux Intelligents

Les Réseaux Intelligents ou « **Smart Grids** » sont des réseaux d'électricité qui, grâce à des technologies informatiques, ajustent les flux d'électricité entre fournisseurs et consommateurs. En collectant des informations sur l'état du réseau, les Smart Grids contribuent à une adéquation entre production, distribution et consommation [4].

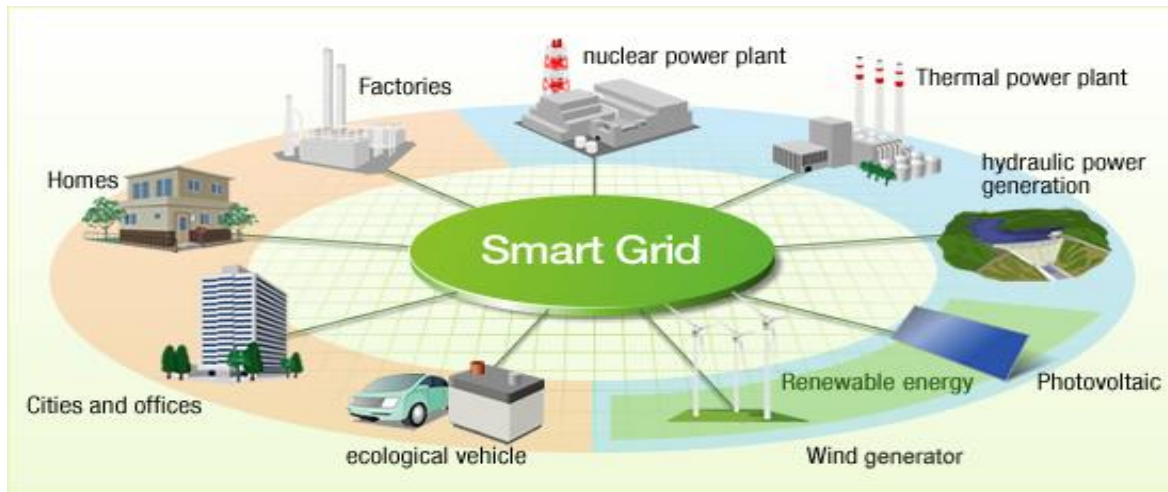


Figure. I.9 Les Réseaux Intelligents.

6.1.3 La domotique

On regroupe sous l'appellation Domotique l'ensemble des technologies de l'électronique, de l'informatique et des télécommunications qui sont utilisées dans les domiciles pour rendre ceux-ci plus «intelligents». La domotique vise donc à intégrer dans un tout cohérent différents systèmes assurant des fonctions de sécurité, de confort, de gestion d'énergie, de communications, de divertissement, d'éducation, etc. qu'on retrouve dans une maison.

La domotique, c'est donc tout ce qui concerne l'automatisation de la maison. Ce domaine est désormais très à la mode depuis que nous avons tous Internet à la maison et sur notre téléphone. Les smartphones et les tablettes permettent d'accéder simplement à la programmation de son chauffage, ses volets, ses caméras, ... [9].

6.1.4 Les environnements intelligents

Les environnements intelligents sont des environnements dans lesquels des détecteurs et actionneurs ont été intégrés, pour réagir à des événements et s'adapter aux personnes présentes. Par exemple, une maison intelligente peut adapter la température et l'éclairage en fonction de la santé, de l'humeur et des préférences des personnes et animaux à l'intérieur de chaque pièce. Un exemple sur ce concept intelligent c'est le *système de détection des incendies de forêts* [4].

6.2 La Médecine Intelligente (e-Santé)

Le domaine médical peut aussi se servir des objets connectés en intégrant des applications pour réduire quelques éléments de dépenses pour les remplacer par d'autres il permet aussi de favoriser l'hospitalisation à domicile, qui assurera le contrôle et le suivi des signes cliniques des patients par la mise en place des réseaux personnels de surveillance, ces réseaux seront constitués de bio-captures posés sur le corps des patients ou dans leurs lieux d'hospitalisation. Cela facilitera la télésurveillance des patients qui permettras de réduire les erreurs médicales, optimiser la consommation de médicaments ou encore leur prise régulière, et même encourager la prévention de certaines maladies, l'internet des objets permettre aussi de suivre sa tension, son rythme cardiaque, la qualité de sa respiration ou encore sa masse grasseuse, et d'autres objets connectés médicaux, brosse a dent connectée ou encore, le scanner qui calcule le nombre de calories dans votre assiette, l'identification des allergies et des médicaments administrés aux patients, la localisation des docteurs et des patients dans l'hôpital,... etc.

6.3 Sport

Les objets connectés et applications mobiles s'invitent dans le quotidien des sportifs de haut niveau. Nombreux sont ceux à en avoir utilisé au cours de leur préparation en se servant d'un bracelet intelligent pour mesurer ses battements cardiaques, atteindre le geste parfait ou encore évaluer la qualité de son sommeil. On note que certains objets connectés [6].

6.4 Agriculture Intelligente

Dans ce domaine, des réseaux de capteurs interconnectés à l'IdO peuvent être utilisés pour la supervision de l'environnement des cultures. Ceci permettra une meilleure aide à la décision en agriculture, notamment pour optimiser l'eau d'irrigation, l'usage des intrants, et la planification de travaux agricoles. Ces réseaux peuvent être aussi utilisés pour récolter les informations utiles sur l'état du sol, taux d'humidité, taux des sels minéraux, etc. et envoyer ces informations au fermier pour prendre les mesures nécessaires garantissant la bonne production ainsi lutter contre la pollution de l'air, du sol et des eaux et améliorer la qualité de l'environnement en général [5].

6.5 Industrie Intelligente

Le concept de l'usine intelligente correspond à une nouvelle façon d'organiser les moyens de production. L'intelligence logicielle ne se trouve plus dans les serveurs mais embarquée dans ces capteurs. Autonomes, ces derniers envoient des informations toutes les secondes et non toutes les dix minutes, ce qui assure une communication continue et instantanée entre les différents outils et postes de travail intégrés dans les chaînes de production et d'approvisionnement. L'utilisation de capteurs communicants apporte à l'outil de production une capacité d'autodiagnostic et permet ainsi son contrôle à distance.

Par ailleurs, ils ne se connectent pas à un réseau GSM mais à un réseau bas débit conçu pour l'IoT. Leur consommation d'énergie est trente fois moindre. Ce qui allonge d'autant leur durée de vie qui se mesure désormais en années et non plus en mois. Leur coût est aussi moindre [8].

7. Vulnérabilités et menaces dans l'internet des Objets

Si l'internet des objets facilite la vie quotidienne des personnes et présente de multiples avantages, il pose aussi plusieurs problèmes quant à la sécurité et à la vie privée en cas de manque de surveillance et de protection physique de ces appareils.

Ces objets du quotidien deviennent, alors des risques potentiels d'attaque sur la sécurité des données et les réseaux, de plus l'apparition de nouvelles menaces qui toucheront directement à l'intégrité des objets eux-mêmes et les vies privées des personnes.

Les enjeux des cyberattaques sont élevés. Il est essentiel de comprendre et de contrer les nouveaux types de conséquences. Par exemple :

- ✓ Une attaque contre un appareil médical peut induire des dommages physiques, voire la perte d'une vie.
- ✓ Le brouillage d'un capteur utilisé pour des compteurs intelligents pourrait provoquer une panne de courant et générer divers inconvénients causés par le manque d'électricité.
- ✓ Le piratage d'une voiture sans chauffeur risquerait de se traduire par une perte de contrôle du véhicule sur une autoroute bondée.

- ✓ L'exploitation des faiblesses d'un appareil IoT pourrait permettre à un individu malveillant d'accéder à des données hautement sensibles telles que des renseignements bancaires, personnels ou de santé.

8. Conclusion

Dans ce chapitre, nous avons introduit l'internet des objets, les objets connectés ainsi que les entités impliquées dans un système IOT. Enfin nous avons conclu ce présent chapitre avec les applications des IOT.

Dans le chapitre suivant nous nous focaliserons sur la carte de contrôle ARDUINO, les différentes cartes existantes, en justifiant notre choix et en présentant ses avantages.

Chapitre II :

La carte de Contrôle à base d'ARDUINO

1. Introduction

L'objectif dans ce mémoire est de concevoir une carte de contrôle à base d'ARDUINO en exploitant et bien sûr en profitant de sa technologie Open-Hardware, qui est sans doute le moyen le plus bas prix existant sur le marché.

Le système Arduino donne la possibilité d'allier les performances de la programmation à celles de l'électronique. Plus précisément, pour programmer des systèmes électroniques. Le gros avantage de l'électronique programmée c'est qu'elle simplifie grandement les schémas électroniques et par conséquent, le cout de la réalisation, mais aussi la charge de travail à la conception d'une carte électronique.

2. Historique

L'Arduino est né en 2005 dans le cadre d'un projet étudiant dans une petite ville du nord de l'Italie. Massimo Banzi enseigne dans une école de Design à Ivrea en Italie, et souvent ses étudiants se plaignent de ne pas avoir accès à des solutions bas prix pour accomplir leurs projets de robotique. Banzi en discute avec David Cuartielles, un ingénieur Espagnol spécialisé sur les microcontrôleurs...Ils décident de créer leur propre carte en embarquant dans leur histoire un des étudiants de Banzi, David Mellis qui sera chargé de créer le langage de programmation allant avec la carte. En deux jours, David écrira le code! Trois jours de plus et la carte était créé...Ils décidèrent de l'appeler Arduino (un bar fréquenté par les élèves à proximité de l'école) : cela devient un hit, tout de suite auprès des étudiants et c'est ainsi qu'un modeste projet d'étudiants bouleversa le milieu de l'électronique, en rendant la technologie plus accessible.

Aujourd'hui, vous pouvez acquérir une carte Arduino pour un prix raisonnable ou la construire vous-même sans avoir de droit de licence à payer. Tous les schémas et les codes sources sont diffusés sous licence libre.

A présent l'Arduino est devenu presque un réflexe dès lors qu'on souhaite s'initier le monde de l'électronique numérique. Arduino a permis de démocratiser ce domaine et surtout d'en faciliter l'accès [10].

3. Description de la Carte ARDUINO

Le module Arduino est un circuit imprimé en matériel libre : les plans sont consultables et tout le monde peut fabriquer une copie exacte de l'Arduino ce qui veut dire que vous pouvez en faire ce que voulez du moment que vous partagez ou distribuez votre travail sous les mêmes conditions. Contrairement à la plupart des objets manufacturés de notre environnement comme les machines à laver, les téléphones, les ordinateurs, seuls le nom et le logo sont réservés, dont les plans de la carte elle-même sont publiés en licence libre dont certains composants de la carte : comme le microcontrôleur et les composants complémentaires qui ne sont pas en licence libre.

Un microcontrôleur programmé peut analyser et produire des signaux électriques de manière à effectuer des tâches très diverses. Arduino est utilisé dans beaucoup d'applications comme l'électrotechnique industrielle et embarquée ; le modélisme, la domotique mais aussi dans des domaines différents comme le pilotage d'un robot, commande des moteurs et faire des jeux de lumières, communiquer avec l'ordinateur, commander des appareils mobile.

En gros, on y branche des capteurs, le programme de l'Arduino traite les informations données par le capteur et déclenche des actions (comme allumer ou éteindre, augmenter, diminuer...).

Chaque module d'Arduino possède un régulateur de tension +5 V et un oscillateur à quartz 16 MHz (ou un résonateur céramique dans certains modèles). Pour programmer cette carte, on utilise l'logiciel IDE Arduino [11].

4. Les différents types de l'ARDUINO

Il existe trois types de cartes ARDUINO :

- **Les officielles** qui sont fabriquées en Italie par le fabricant officiel : Smart Projects
- **Les compatibles** qui ne sont pas fabriqués par Smart Projects, mais qui sont totalement compatibles avec les Arduino officielles.
- **Les autres** fabriquées par diverse entreprise et commercialisées sous un nom différent (Freeduino, Seeduino, Femtoduino, ...) [12].

Depuis que le module Arduino a vu le jour, plusieurs versions de cartes ont été réalisées, on cite quelques-unes afin de mieux comprendre l'évaluation de ce produit scientifique et académique :

SERIAL ARDUINO : programmé avec une connexion série par un connecteur DB9 et utilisant un microcontrôleur ATmega8.

ARDUINO EXTREME : programmable via une connexion USB et utilisant un ATmega8.

ARDUINO MINI : une version miniature de l'ARDUINO utilisant un ATmega168 de type CMS.

ARDUINO NANO : une version encore plus petite de l'ARDUINO alimenté par USB et utilisant un ATmega168 (avant la version 3) ou ATmega328 (à partir de la version 3.0) de type CMS

LILYOAP ARDUINO : une conception de type minimaliste pour permettre une application portable utilisant un ATmega168 de type CMS.

ARDUINO NG : programmable via une connexion USB et utilisant un ATmega8.

ARDUINO NG plus : programmable via une connexion USB et utilisant un ATmega168.

ARDUINO BLUETOOTH (BT) : programmable via une connexion Bluetooth et utilisant un ATmega328.

ARDUINO DIECIMILA : possède une interface USB et utilise un ATmega168 dans un boîtier format DIL28. (16 ko de mémoire flash, 1 ko de mémoire SRAM et 0,5 ko de mémoire EEPROM).

ARDUINO DUEMILANOVE(2009) : utilise un Atmega168 et est alimenté par le connecteur USB ou une alimentation externe avec commutation automatique. La nouvelle version est équipée d'un ATmega328 (32 ko de flash, 2 ko de SRAM et 1 ko d'EEPROM).

ARDUINO MEGA: est équipé d'un ATmega1280 de type CMS pour avoir des Entrées/Sorties supplémentaires et de la mémoire (128ko flash, 8ko SRAM, 4ko EEPROM).

ARDUINO UNO : utilise un ATmega328 comme les derniers modèles de DUEMILANOVE, mais alors que le DUEMILANOVE utilisait une puce FTDI pour la programmation via un connecteur USB, l'UNO utilise une puce ATmega8U2 programmé comme un convertisseur série.

ARDUINO MEGA2560: est équipé d'un ATmega2560 de type CMS, augmentant la mémoire totale disponible à 256 ko. Il est équipé aussi de la nouvelle puce USB ATmega8U2.

ARDUINO ETHERNET : est une carte ARDUINO UNO intégrant un chip Wiznet W5100 pour rajouter la connectivité Ethernet intégré.

ARDUINO LEONARDO : est une version bas coût de l'ARDUINO UNO à base d'un ATmega32U4.

ARDUINO DUE : est une évolution de l'ARDUINO Mega2560 avec un microcontrôleur 32 bits AtmelSAM3X (ARM 32 bits Cortex-M3).

ARDUINO ESPLORA : est une carte dérivée de l'ARDUINO LEONARDO. Elle a la forme d'une manette de jeu. Contrairement aux autres ARDUINO, c'est une carte « tout-en-un » qui comporte de nombreux capteurs (température, accélération, lumière, microphone, potentiomètre...) ainsi que 4 boutons poussoirs, un potentiomètre et un joystick analogue [13].

Le tableau suivant (Tableau II.1) regroupe les différentes cartes existantes avec leurs caractéristiques (type du Microcontrôleur, taille de la Mémoire interne, nombre de broches PIN, ...etc).

Tableau. II.1 Les différents types de l'ARDUINO.

Arduino	Microcontrôleur	Flash ko	E/S numériques	PWM	Entrée analogique	Type d'interface USB
Diecimila	ATmega168	16	14	6	6	FTDI
Due	Atmel SAM3X8E	512	54	12	12	ATmega16u2
Duemilanove	ATmega168/328P	16/32	14	6	6	FTDI
Esplora	ATmega32U4	32	N/A	N/A	N/A	ATmega32U4
Fio	ATmega328P	32	14	6	8	Aucune
Leonardo	ATmega32U4	32	20	7	12	ATmega32U4
LilyPad	ATmega168V or ATmega328V	16	14	6	6	Aucune
Mega	ATmega1280	128	54	15	16	FTDI
Mega2560	ATmega2560	256	54	15	16	ATmega8U2
Nano	ATmega168	16	14	6	8	FTDI
Uno	ATmega328P	32	14	6	6	ATmega16U2
Zero	ATSAMD21	256	14	12	6	ATmega32U2

5. Domaines d'applications

Le système Arduino nous permet de réaliser un grand nombre de choses, qui ont une application dans tous les domaines, nous pouvons donner quelques exemples :

- Construire des systèmes physiques interactifs qui utilisent des logiciels et du matériel pouvant s'interfacer avec des capteurs et des actionneurs.
- Électronique industrielle et embarquée.
- Art / Spectacle.
- Domotique.

- Robotique.
- Modélisme.
- Education, Etc.

6. Les avantages de l'Arduino

Il existe dans le commerce, une multitude de plateformes qui permettent de réaliser les mêmes fonctions que l'Arduino .Notamment les microcontrôleurs « PIC» du fabricant Micro chip disponibles pour l'électronique programmée. Néanmoins l'Arduino reste au haut du tableau des cartes électroniques utilisées dans le monde grâce à ses multiples avantages : Le système Arduino, nous donne la possibilité d'allier les performances de la programmation à celles de l'électronique. Plus précisément, nous allons programmer des systèmes électroniques. Les principaux avantages de l'électronique programmée sont :

- **Le prix**

En vue des performances qu'elles offrent, les cartes Arduino sont relativement peu coûteuses, ce qui est un critère majeur pour le débutant.

- **Un environnement de programmation clair et simple**

L'environnement de programmation Arduino (le logiciel Arduino) est facile à utiliser pour les débutants, tout en étant assez flexible pour que les utilisateurs avancés puisse en tirer profit également. Pour les enseignants, il est basé sur l'environnement de programmation Processing : les étudiants qui apprennent à programmer dans cet environnement seront déjà familiarisés avec l'aspect du logiciel Arduino.

La possibilité d'ajouter des Shields spécifiques (cartes filles) pour ajouter de nouvelles fonctionnalités, il suffit de brancher un Shield supplémentaire correspondant à la fonction désirée.

- **La liberté**

C'est un bien grand mot, mais elle définit de façon assez concise l'esprit de l'Arduino. Elle constitue en elle même deux choses :

Le logiciel open source : le logiciel et le langage Arduino sont gratuits et publiés sous licence open source, disponible pour être complété par des programmeurs

expérimentés. Le logiciel de programmation des modules Arduino est une application JAVA multi plateformes (fonctionnant sur tout système d'exploitation), servant d'éditeur de code et de compilateur, et qui peut transférer le programme au travers de la liaison série (RS232, Bluetooth ou USB selon le module).

Le matériel open source : les cartes Arduino sont basées sur les Microcontrôleurs Atmel ATMEGA8, ATMEGA168, ATMEGA 328, les schémas des modules sont publiés sous une licence créative Commons, et les concepteurs des circuits expérimentés peuvent réaliser leur propre version des cartes Arduino, en les complétant et en les améliorant. Même les utilisateurs relativement inexpérimentés peuvent fabriquer la version sur plaque d'essai de la carte Arduino, dont le but est de comprendre comment elle fonctionne pour économiser le coût. Cette liberté a une condition : le nom « Arduino » ne doit être employé que pour les cartes « officielles ». En somme, nous ne pouvons pas fabriquer notre propre carte sur le modèle Arduino et lui assigner le nom « Arduino ». Les cartes non officielles, on peut les trouver et les acheter sur Internet et sont pour la quasi totalité compatibles avec les cartes officielles Arduino [10].

- **La compatibilité**

Le logiciel, tout comme la carte, est compatible sous les plateformes les plus courantes (Windows, Linux et Mac), contrairement aux autres outils de programmation du commerce qui ne sont, en général, compatibles qu'avec Windows.

- **La communauté**

La communauté Arduino est impressionnante et le nombre de ressources à son sujet est en constante évolution sur internet. De plus, on trouve les références du langage Arduino ainsi qu'une page complète de tutoriels sur le site arduino.cc (en anglais) et arduino.cc (en français) et plusieurs forum internet spécialisés sur l'Arduino.

7. Arduino Uno, pourquoi ?

Pour modéliser nos objets connectés, une carte électronique où des capteurs de toutes sortes peuvent être branchés est nécessaire. Les cartes électroniques à bas prix présentes sur le marché sont les cartes Arduino.

Notre choix s'est porté sur la carte Arduino Uno pour les différents avantages déjà cités, ajoutant à ça les nombreux pins d'entrées/sorties, que possède la carte, où différents capteurs peuvent être branchés. Ainsi que la possibilité d'ajouter des Shields spécifiques (cartes filles) pour ajouter de nouvelles fonctionnalités. Comme les communications doivent s'effectuer sans fil, le Shield adéquat sera choisi. De plus si le projet évolue et que de nouvelles fonctionnalités doivent être ajoutées (liaison Ethernet par exemple), il suffit de brancher un Shield supplémentaire correspondant à la fonction désirée.

Catégorie	Valeur
Microcontrôleur	ATmega 328
Fréquence d'horloge	16 MHz
Tension de service	5 V
Tension d'entrée (recommandée)	7-12 V
Tension d'entrée (limites)	6-20 V
Ports numériques	14 entrées et sorties (6 sorties commutables en MLI)
Ports-analogiques	6 entrées analogiques
Courant maxi. par broche d'E/S (c.c.)	40 mA
Courant maxi. par broche 3,3 V	50 mA
Mémoire	32 Ko Flash, 2 Ko SRAM, 1 Ko EEPROM
Chargeur d'amorçage	0,5 Ko (en mémoire Flash)
Interface	USB
Dimensions	6,86 cm × 5,3 cm

Figure. II.1 Caractéristiques de la Carte ARDUINO UNO.

Un module Arduino se compose de deux choses principales : le matériel et le logiciel.

- La partie matérielle s'agit d'une carte électronique généralement construite autour d'un microcontrôleur ATMEL AVR, et de composants complémentaires qui facilitent la programmation et l'interfaçage avec d'autres circuits. Chaque module possède au moins un régulateur linéaire 5V et un oscillateur à quartz 16 MHz. Le microcontrôleur est préprogrammé avec un bootloader de façon à ce qu'un programmeur dédié ne soit pas nécessaire.
- La partie logiciel permet de programmer la carte Arduino. Il offre une multitude de fonctionnalités pour programmer la partie matérielle.

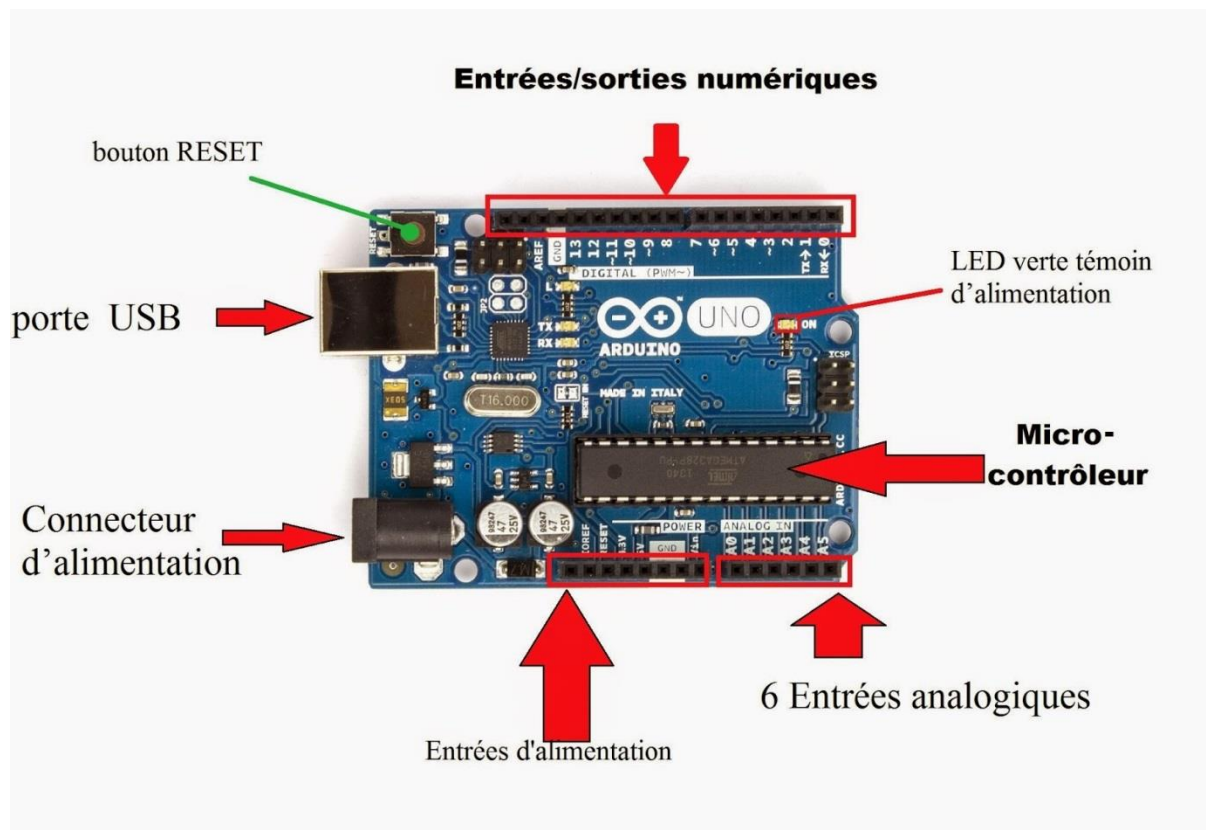


Figure. II.2 Composition de la Carte ARDUINO UNO [10].

8. Partie matérielle

Le Microcontrôleur ATmega328 : Le microcontrôleur utilisé sur la carte est le ATmega328. C'est un microcontrôleur *Atmel* de type *AVR* 8 bits. Il 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 sur :

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).

SRAM : c'est la mémoire dite "vive", elle va 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.

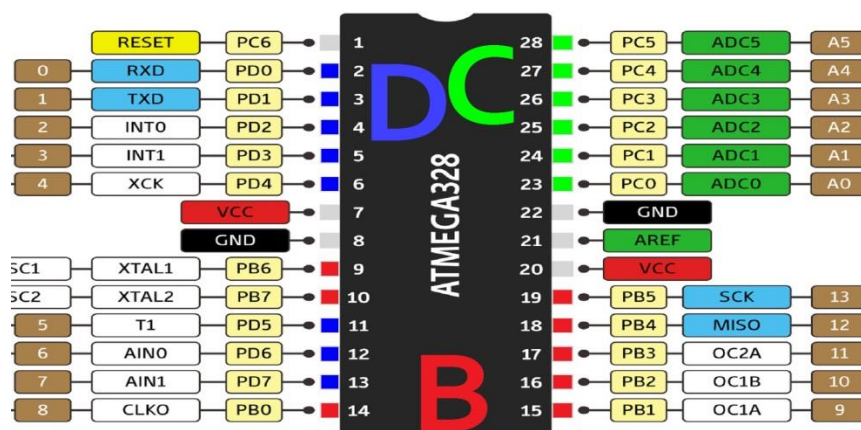


Figure. II.3 Composition du μ C ATmega328 [10].

Alimentation : La carte Arduino Uno peut-être alimentée soit via la connexion USB (qui fournit 5V jusqu'à 500mA) ou à l'aide d'une alimentation externe. La source d'alimentation est sélectionnée automatiquement par la carte.

L'alimentation externe (non-USB) peut être soit un adaptateur secteur (pouvant fournir typiquement de 3V à 12V sous 500mA) ou des piles (ou des accus). L'adaptateur secteur peut être connecté en branchant une prise 2.1mm positif au centre dans le connecteur jack de la carte. Les fils en provenance d'un bloc de piles ou d'accus peuvent être insérés dans les connecteurs des broches de la carte appelées Gnd (masse ou 0V) et Vin (Tension positive en entrée) du connecteur d'alimentation.

La carte peut fonctionner avec une alimentation externe de 6 à 20 volts. Cependant, si la carte est alimentée avec moins de 7V, la broche 5V pourrait fournir moins de 5V et la carte pourrait être instable. Si on utilise plus de 12V, le régulateur de tension de la carte pourrait chauffer et endommager la carte. Aussi, la plage idéale recommandée pour alimenter la carte Uno est entre 7V et 12V.

Les broches d'alimentation sont les suivantes :

- 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). Vous pouvez 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.
- 3V3. 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
- GND. Broche de masse (ou 0V) [15].

Entrées et sorties numériques : La carte UNO est composée de 14 broches numériques (numérotées des 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)`.

De plus, certaines broches ont des fonctions spécialisées :

- **Communication Serie**: Broches 0 (RX) et 1 (TX). Utilisées pour recevoir (RX) et transmettre (TX) les données séries de niveau TTL. Ces broches sont connectées aux broches correspondantes du circuit intégré ATmega8U2 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.
- **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. Voir l'instruction `attachInterrupt()` pour plus de détails.

- **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 de 6 entrées analogiques (numérotées de 0 à 5), chacune pouvant fournir une mesure d'une résolution de 10 bits (càd 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. Néanmoins, les broches analogiques peuvent être utilisées en tant que broches numériques : elles sont numérotées en tant que broches numériques de 14 à 19.

Il existe aussi deux autres broches disponibles sur la carte :

- **AREF** : Tension de référence pour les entrées analogiques (si différent du 5V). Utilisée avec l'instruction **`analogReference()`**.
- **Reset** : Mettre cette broche au niveau BAS entraîne la réinitialisation (= le redémarrage) du microcontrôleur. Typiquement, cette broche est utilisée pour ajouter un bouton de réinitialisation sur le circuit qui bloque celui présent sur la carte. (6)

Communication : La carte Arduino Uno dispose de toute une série de facilités pour communiquer avec un ordinateur, une autre carte Arduino, ou avec d'autres microcontrôleurs. L'ATmega 328 dispose d'une UART (Universal Asynchronous Receiver Transmitter ou émetteur-récepteur asynchrone universel en français) pour communication série de niveau

TTL (5V) et qui est disponible sur les broches 0 (RX) et 1 (TX). Un circuit intégré ATmega8U2 sur la carte assure la connexion entre cette communication série vers le port USB de l'ordinateur et apparaît comme un port COM virtuel pour les logiciels de l'ordinateur.

N.B. Le logiciel Arduino inclut une fenêtre terminal série (ou moniteur série) sur l'ordinateur et qui permet d'envoyer des textes simples depuis et vers la carte Arduino. Les LEDs RX et TX sur la carte clignote lorsque les données sont transmises via le circuit intégré USB-vers-série et la connexion USB vers l'ordinateur (mais pas pour les communications série sur les broches 0 et 1). Mais son inconvénient est le câble USB, pour éviter cela, il existe différentes méthodes pour utiliser ce dernier sans fil [15].

9. Partie logiciel

Comme toutes les carte électronique qui, leurs constructions se basent sur un microcontrôleur doivent être dotées d'une interface de programmation comme est le cas de notre carte Arduino Uno.

Sur un ordinateur, le logiciel de programmation de la carte Arduino sert d'éditeur de code (langage proche du C) , avec des fonctions et des bibliothèques spécifiques à ARDUINO (gestions des entrées /sorties). Une fois le programme tapé ou modifié au clavier, il sera transféré et mémorisé dans la carte au travers de la liaison USB. Le câble USB alimente à la fois en énergie la carte et transporte aussi l'information.

L'environnement de programmation open-source pour Arduino peut être téléchargé gratuitement (pour Mac OS X, Windows, et Linux).

9.1 Structure générale du programme (IDE Arduino)

Le logiciel de programmation d'Arduino est connu par son interface souple et simple qu'on peut exécuter sur n'importe quel système d'exploitation déjà cités.

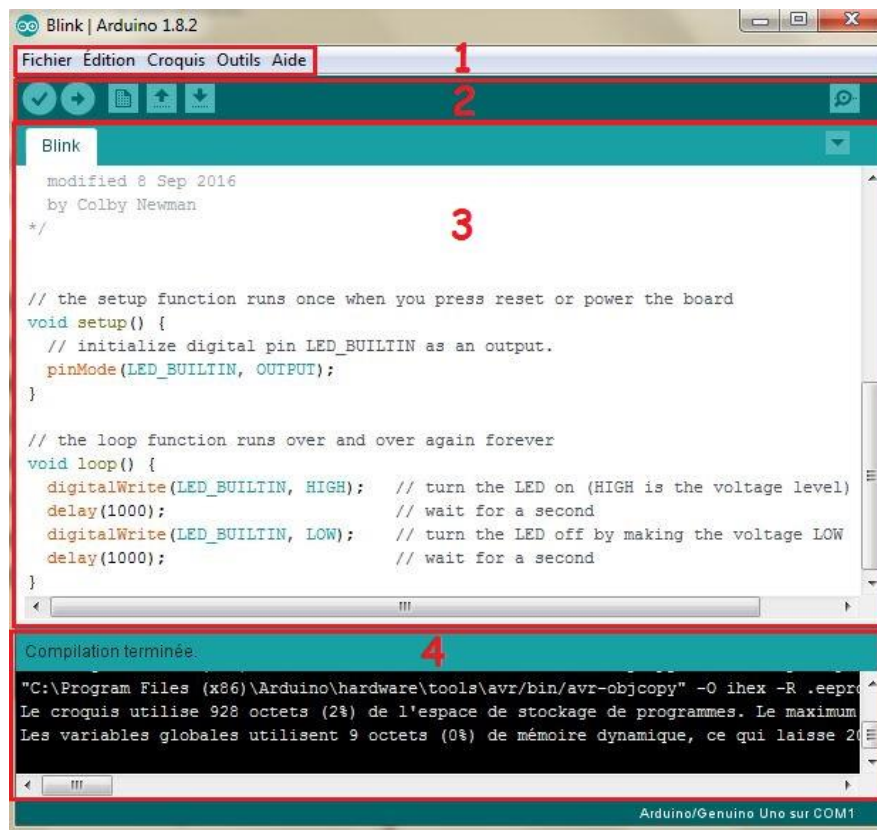


Figure. II.4 Interface de programmation d'Arduino.

La zone 1 : ce sont les options de configuration du logiciel.

La zone 2 : elle contient les boutons qui vont nous servir lorsque on va programmer.

La zone 3 : ce bloc va contenir le programme que nous allons créer.

La zone 4 : celui-ci est important, car il va nous aider à corriger les erreurs dans notre programme: C'est le débogueur ; il nous affiche les messages de compilation.

9.2 Injection du programme

En reliant la carte Arduino au PC, la toute première étape avant d'envoyer un programme dans la carte, il est nécessaire de sélectionner le type de la carte Arduino à utilisée (dans notre cas on sélectionne l'Arduino Uno) (Figure II.5), ainsi que le numéro de port USB (COM3). En suite, on édite un programme (on dit sketch) concernant l'application demandée, on contrôle s'il n'y a pas d'erreurs de syntaxe ou autres puis on transfère le programme sur la carte.

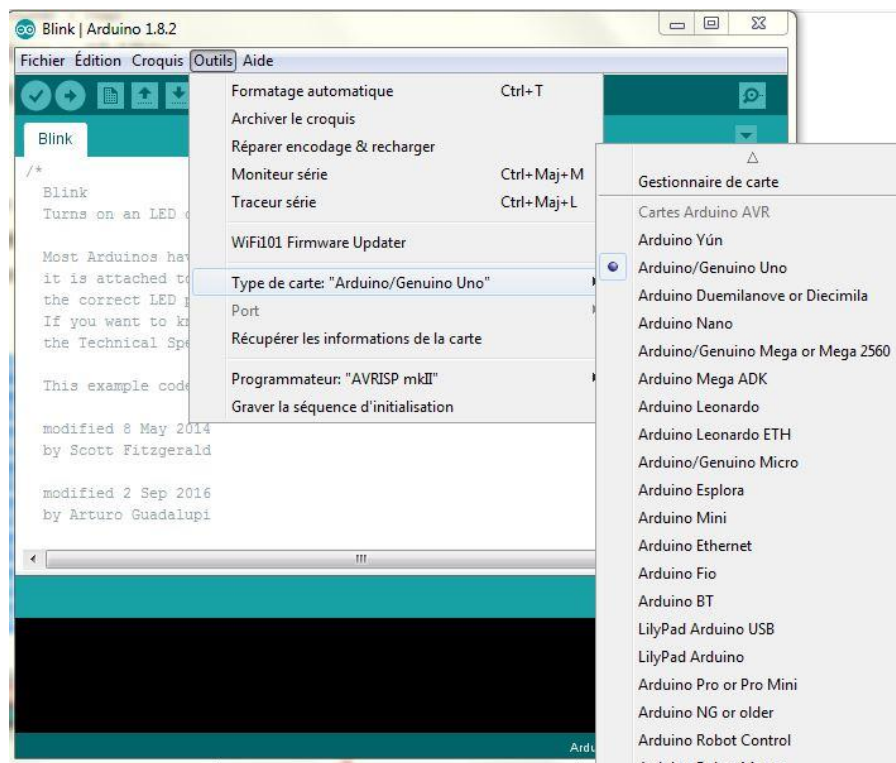


Figure. II.5 Choix de la carte d'Arduino.

9.3 Description du programme

Après avoir téléchargé et installé l'IDE de l'arduino et avant de commencer son premier projet, il est primordial d'apprendre la syntaxe du langage de programmation sur l'IDE de l'arduino. Le langage Arduino est basé sur les langages C/C++.

Un programme Arduino est une suite d'instructions élémentaires sous forme textuelle, ligne par ligne. Le microcontrôleur lit puis effectue les instructions les unes après les autres, dans l'ordre défini par les lignes de code.

Voilà donc, la Structure générale du programme :

- **void setup()** (configuration-préparation) : C'est une partie initialisation et configuration des entrées/sorties. Elle est appelée une seule fois lorsque le programme commence.
- **void loop()** (exécution) : C'est la partie principale contenant le programme. Elle est répétée indéfiniment en boucle infinie.

Les **commentaires** sont, en programmation informatique, des portions du code source ignorées par le compilateur ou l'interpréteur, car ils ne sont pas censés influencer l'exécution du programme : c'est un moyen qu'on utilise pour ne pas s'égarer entre les lignes de codes écrits, et ça permet de retrouver la ligne qu'on veut modifier en cas d'erreurs de syntaxe.

```
// (commentaire simple ligne)
```

```
/* */ (commentaire multi-lignes)
```

Exemples de commentaires : 1 // programme de commande DC moteur

2 // afficher la température en degrésetc.

9.4 Les variables

Une variable est un espace réservé dans la mémoire de l'ordinateur. Elle est caractérisée par un nom qui permet d'y accéder facilement.

Il existe différents types de variables identifiés par un mot-clé dont les principaux sont :

- nombres entiers (**int**)
- nombres à virgule flottante (**float**)
- texte (**String**)
- valeurs vrai/faux (**boolean**).

Un nombre à décimales, par exemple *3.14159*, peut se stocker dans une variable de type float. On Note que l'on utilise un point et non une virgule pour les nombres à décimales. Dans Arduino, il est nécessaire de déclarer les variables pour leurs réserver un espace mémoire adéquat. On déclare une variable en spécifiant son type, son nom puis en lui assignant une valeur initiale (optionnel) [11].

Exemple de déclaration de variable :

```
int ma_variable = 45;
```

```
// int est le type, ma_variable le nom et = 45 assigne une valeur.
```

9.5 Configuration des entrées et des sorties void setup ()

Comme on vient de le cité en haut , le programme Arduino se compose généralement de deux partie ; Et c'est dans la partie void setup () qu'on configure les entrées et les sorties.

Les broches numériques de l'arduino peuvent aussi bien être configurées en entrées numériques ou en sorties numériques, en utilisant l'instruction :

pinMode (broche, état) (configuration des broches)

Exemple de configuration d'une broche en sortie :

```
void setup()
{
  // configure la broche numérique en SORTIE
  pinMode(ledPin, OUTPUT);
}
```

9.6 Programmation des interactions void loop ()

Dans cette partie du programme, on définit les opérations à effectuer, en utilisant des fonctions. Il est important à savoir aussi qu'il est nécessaire de terminer chaque ligne d'instruction par un point virgule sinon elle ne sera pas prise en considération par le compilateur et elle sera ignorée.

Les fonctions qu'on utilise souvent sont :

- **digitalWrite**(broche, état) (écrire un état sur une broche num.)
- **digitalRead**(broche) (lire un état sur une broche num.)
- **int analogRead**(broche) (lire la valeur d'une broche analogique.)
- **analogWrite**(broche, valeur) (PWM: écrire une valeur analogique sur les PIN 9, 10 ou 11)
- **delay(ms)** (attente, en millisecondes)
- **delayMicroseconds(µs)** (attente, en microsecondes)

Niveaux logiques des connecteurs numériques:

- HIGH (état 1)
- LOW (état 0)
- INPUT (configuré en entrée)
- OUTPUT (configuré en sortie)

Exemple: **digitalRead**(ledPin, HIGH) (lire Haut sur une broche ledPin.)

Chaque fois on fait appel à des boucles ; Contrôle et/ou de Condition :

if (si...)

if...else (si...sinon...)

for (pour...)

switch case (dans le cas où...)

while (pendant que ...)

do... while




break

continue

return

9.7 Les étapes de téléchargement du programme

Une simple manipulation enchaînée doit être suivie afin d'injecter un code vers la carte Arduino via le port USB comme suit :

1. On écrit ou on ouvre un programme existant avec le logiciel IDE Arduino. 
2. On vérifie ce programme avec le logiciel Arduino (compilation). 
3. Si des erreurs sont signalées, on modifie le programme.
4. On connecte la carte à l'ordinateur avec le cordon USB.
5. On transfère le programme sur la carte. 
6. On câble le montage électronique.
7. L'exécution du programme est automatique après quelques secondes.
8. On alimente la carte soit par le port USB, soit par une source d'alimentation autonome (pile 9 volts par exemple).
9. On vérifie que notre montage fonctionne.

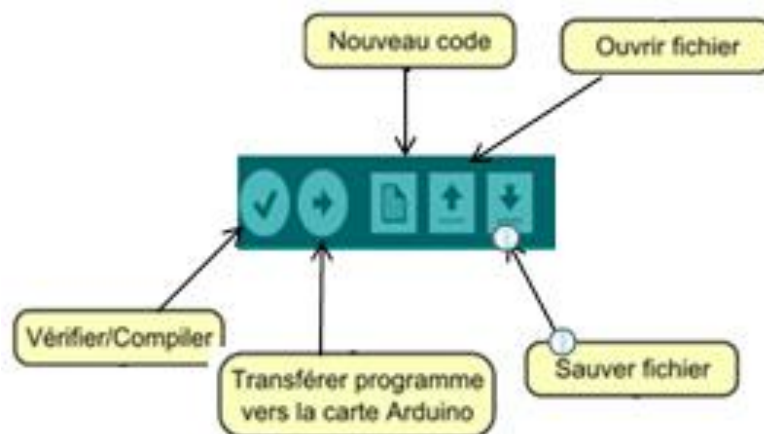


Figure. II.6 Edition des programmes d'Arduino.

10. Les Accessoires de la carte Arduino

Dans un premier temps, une communication point à point entre un utilisateur et un objet connecté est établie. Ensuite une des améliorations possibles peut être la construction d'un réseau d'objets connectés : Communication sans fil. Il existe de nombreuses méthodes pour communiquer sans fil entre différents objets : un Shield doit être ajouté à la carte de base pour accéder aux nouvelles fonctionnalités.

Parmi les protocoles de communications qu'on peut utiliser on cite les suivants :

10.1 Le module Arduino Bluetooth

Le Module Microcontrôleur Arduino Bluetooth est la plateforme populaire Arduino avec une connexion série 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 [16].

10.2 Shield Arduino Wifi

Shield Arduino Wifi permet de connecter une carte Arduino à un réseau internet sans fil Wifi. Il permet d'avoir une grande portée et également un débit bien plus élevé que les autres protocoles. Le principal défaut du protocole Wifi est son importante consommation d'énergie [16].

10.3 XBee

la communication XBee Arduino permet la connexion sans fil par le biais d'un adaptateur XBee.

10.4 Infrarouge

Pour établir une communication infrarouge entre deux modules, il ne faut pas d'obstacle sur le chemin du signal (les ondes ne traversent pas les murs). Or Cette contrainte réduit l'usage des objets connectés.

10.5 Radio

La radio est un moyen de communication par les ondes Radio sans tout en consommant très peu d'énergie.

10.6 Les capteurs

Un capteur est un dispositif transformant l'état d'une grandeur physique observée en une grandeur utilisable, exploitable, telle qu'une tension électrique. On peut dire aussi que c'est une interface entre un processus physique et une information manipulable. Il ne mesure rien, mais fournit une information en fonction de la sollicitation à laquelle il est soumis. Il fournit cette information grâce à une électronique à laquelle il est associé.

10.7 Les afficheurs LCD

Les afficheurs à cristaux liquides, LCD (Liquid Crystal Display), sont des modules qui nécessitent peu de composants externes pour un bon fonctionnement. Ils permettent d'afficher des lettres, des chiffres et quelques caractères spéciaux. Les caractères sont prédéfinis.

10.8 Les relais :

C'est un organe électrique permettant de dissocier la partie puissance de la partie commande : il permet l'ouverture et la fermeture d'un circuit électrique par un second circuit complètement isolé (isolation galvanique) et pouvant avoir des propriétés différentes [17].

11. Conclusion

Dans ce chapitre nous avons donné tout un aperçu sur les cartes ARDUINO, en suite on s'est consacré sur la carte Arduino UNO, nous avons justifié ce choix en présentant ses avantages. Aussi, on a présenté les périphériques utilisés avec la carte pour pouvoir implanter toute une application. Et à la fin nous avons détaillé l'interface de développement (IDE Arduino) qui est une pièce maitresse (partie programmation).

Chapitre III :

Structure matérielle et logicielle du projet

1. Introduction

L'objectif de ce chapitre est de présenter les différentes parties utilisées dans ce mémoire.

A savoir :

- ✓ Le Bluetooth.
- ✓ Le détecteur de mouvement RIP.
- ✓ Le capteur de gaz MQ-2.
- ✓ Le capteur de température LM 335.
- ✓ Le détecteur de lumière LDR et d'autres éléments.
- ✓ L'environnement Fritzing et App inventor.

2. Structure du systeme

Le système réalisé comporte un bloc principal constitué d'une carte à microcontrôleur du type Arduino UNO reliée à un canal de communication de type Bluetooth qui sera connecté et commandé par une application Android. Cette carte est reliée à un ensemble de périphériques constitué de capteurs.

Comme le montre la Figure III.1 , notre système est composé de :

Des Capteurs :

- Détecteur de mouvement : Pour réaliser un système d'éclairage interne automatique.
- Détecteur de lumière LDR : Pour allumer automatiquement la lumière externe.
- Capteur de gaz MQ-2 : Pour surveiller les fuites de gaz.
- Capteur de température LM 335 : Pour surveiller sur la température ambiante afin de la régulariser

Des Actionneurs :

- Servo moteur : Pour ouvrir et fermer la fenêtre.
- Module relais : Pour allumer et éteindre les lampes.

3. Le bloc principal "l'acquisition des données de la domotique"

C'est la partie *hardware* du système, à base de la carte Arduino Uno commandée par l'application Android. L'Arduino Uno traite la réception et l'envoi des différentes informations entre les périphériques et l'application Android : Elle gère et maintient le fonctionnement du circuit où tous les capteurs et les actionneurs y sont connectés.

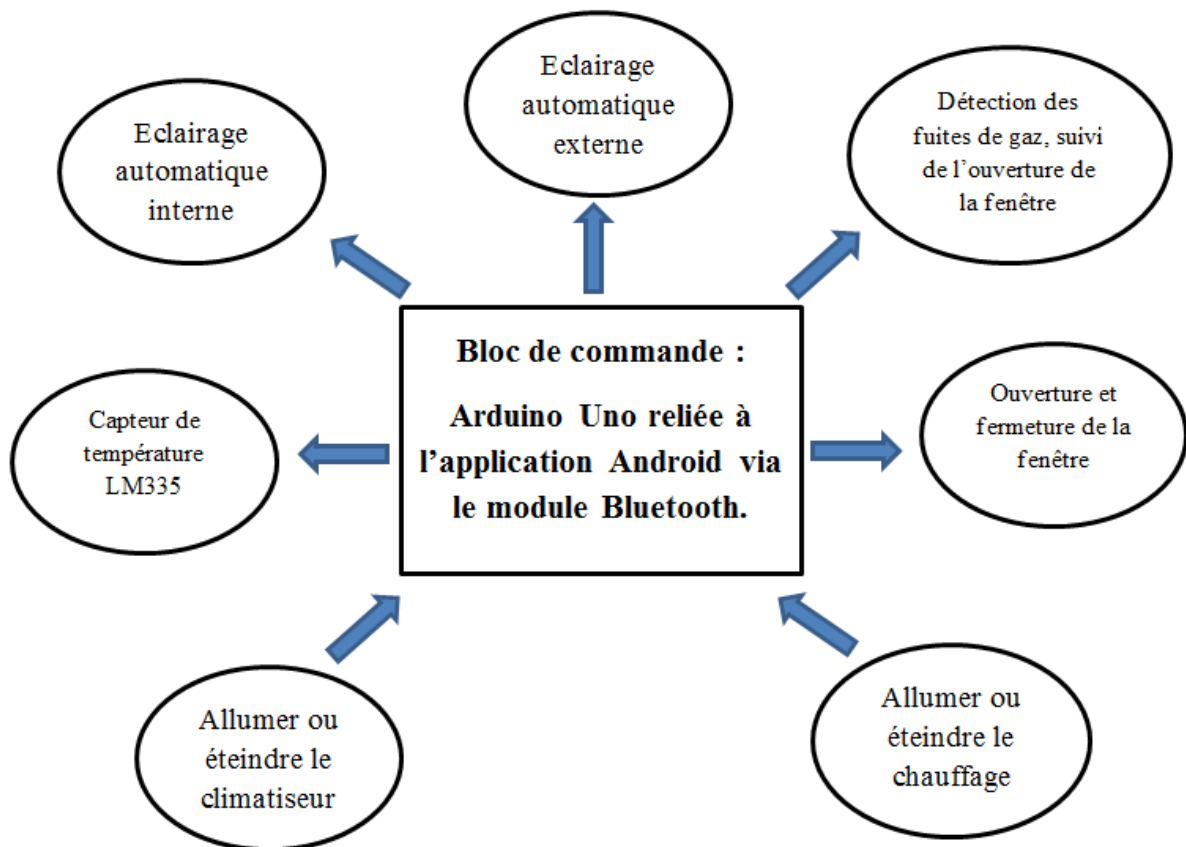


Figure. III.1 Schéma bloc du système.

4. Description du système

4.1 Partie matérielle

4.1.1 Bluetooth

Le bluetooth est un protocole de communication sans fil qui a vu le jour à la fin des années 1990 et n'a vraiment percé que dans les années 2000. À l'origine, la société Ericsson cherchait à développer un moyen d'utiliser une voie série sans passer par un fil. Petit à petit, ses utilisations se sont étendues pour atteindre ce que l'on connaît aujourd'hui, un moyen de connecter sans fil de nombreux appareils, allant d'une Arduino et sa voie série à un ordinateur, ou pour utiliser un casque audio ou encore une manette de jeu sur une console de salon.

Comme le Wi-Fi, le Bluetooth utilise aussi la même gamme de fréquences : 2.4 GHz . C'est une communication bidirectionnelle, deux modules peuvent communiquer ensemble en même temps. Le comportement utilisé est "maître/esclave". Un esclave pourra parler avec un seul maître, mais un maître pourra dialoguer avec plusieurs esclaves. Pour son utilisation, elle se passe en plusieurs étapes [20]:

- ✓ Le maître se met en mode "reconnaisable"
- ✓ L'esclave trouve le maître et demande à s'y connecter
- ✓ Le maître accepte la connexion
- ✓ Les périphériques sont alors appairés (ou associés)
- ✓ La communication peut commencer

Le module Bluetooth HC-06 (Figure III.2) est l'un des différents modèles qui existent dans le marché. Il permet d'établir une liaison Bluetooth (liaison série) entre une carte Arduino et un autre équipement possédant une connexion Bluetooth (Smartphone, tablette, seconde carte Arduino, etc...). Le module HC-06 est un module "esclave" contrairement au module HC-05 par exemple qui est "maître". Un module "maître" peut demander à un autre élément Bluetooth de s'appairer avec lui alors qu'un module "esclave" ne peut recevoir que des demandes d'appairage. Ce qui convient parfaitement à notre volonté de communiquer entre le Bluetooth implémenté dans la carte Arduino et le smartphone [21].



Figure. III.2 Module Bluetooth HC-06.

Le tableau suivant résume les caractéristiques du Module Bluetooth HC-06 :

Tableau. III.1 Caractéristiques du Module Bluetooth HC-06.

Matériel de Shell	métal dur durable
Dimensions	3,57 x 1,52 cm
Tension d'entrée	3.6V à 6V (jamais plus de 7V)
Courant absorbé	30mA
Distance effective	20M (dans la zone ouverte)
Interface	VCC, GND, RXD TXD
Longueur d'onde	2.4 GHz à 2.8 GHz, ISM Band
Sensibilité de réception	-85dBm
Température	-40 à +105 °C
Les vitesses de transmission possibles (en bauds)	9600, 19200, 38400, 57600, 115200, 230400, 460800, 921600, 1382400.

4.1.2 Capteur de mouvement

Un capteur PIR (capteur passif d'infrarouge) permet de détecter le mouvement d'un corps humain : la chaleur du corps produit suffisamment de lumière infrarouge pour être mesurée). Le capteur peut détecter un mouvement jusqu'à une distance maximale de 6m.

Ce capteur (Figure III.3) nécessite au moins 20 secondes de calibration automatique au démarrage. Durant ces 20 secondes, il est complètement inactif et ne doit pas être obstrué par des sources de chaleurs (main, corps, chat,...etc) [18].

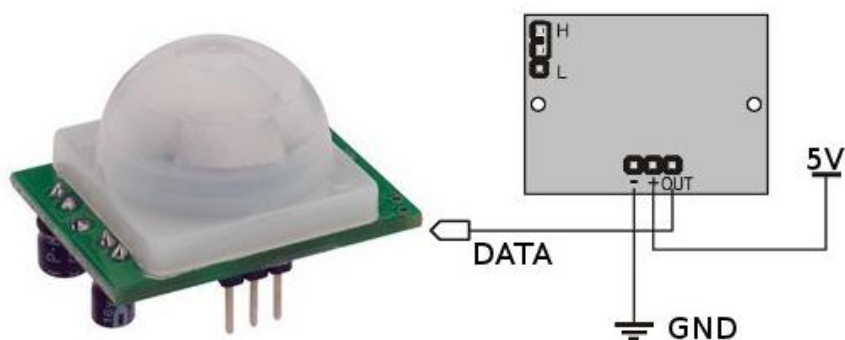


Figure. III.3 Capteur de mouvement.

4.1.3 Capteur de Gaz MQ-2

Le MQ-2 Capteur de Gaz/Fumée est un semi-conducteur qui détecte la présence du gaz/fumée à des concentrations de 300 ppm à 10000 ppm. Il a une grande sensibilité et un temps de réponse rapide. Sa sensibilité peut d'ailleurs être ajustée par potentiomètre.

Ce capteur (Figure III.4) peut fonctionner à des températures allant de -10 à 50 °C et consomme moins de 150 mA à 5 V [18].



Figure. III.4 Capteur de Gaz MQ-2.

Le tableau suivant résume les Caractéristiques du MQ-2 :

Tableau. III.2 Caractéristiques du MQ-2

Alimentation	5v
Puissance	0.8W
Courant	0.16A
Tension de sortie	5V DC
Nombre de broche	4 broches (4 pins)
Dimensions	40x20mm
Type d'Interface	Analogique
Stabilité	Système stable à longue durée de vie.
Circuit de contrôle	Simple
Temps de réponse	rapide et haute sensibilité

4.1.4 Photorésistance (LDR)

Une photorésistance (également appelée LDR : résistance photo-dépendante ou cellule photoconductrice) est un composant électronique composée d'un semi-conducteur à haute résistivité dont la résistivité varie en fonction de la quantité de lumière incidente (Figure III.5) : plus elle est éclairée, plus sa résistivité baisse [22].

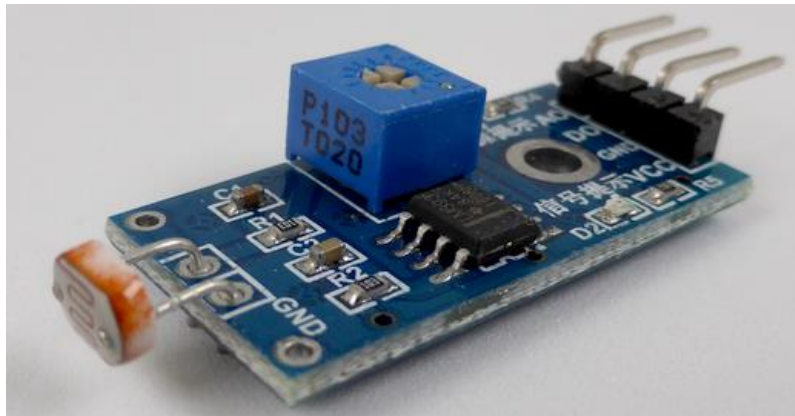


Figure. III.5 Photorésistance (LDR).

Une photorésistance est un composant dont la résistivité dépend de la luminosité ambiante. Dans l'obscurité, la résistance d'une LDR est proche de $1\text{ M}\Omega$. Avec un éclairage intense, la résistance chute fortement (quelques $\text{K}\Omega$). Un pont diviseur permet de récupérer une tension qui sera directement le reflet de la lumière arrivant sur la LDR [22].

On va donc s'en servir pour détecter les fortes variations de la luminosité. On note l'illumination en « lux », voici la figure (xx) qui montre l'évolution en lux en fonction de la résistance :

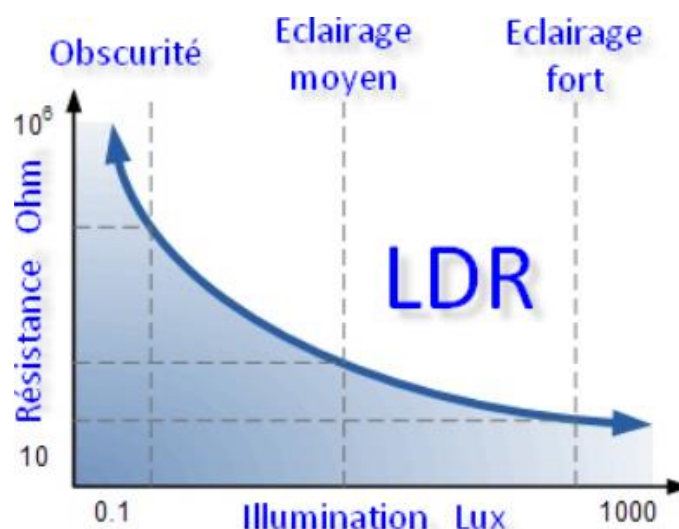


Figure. III.6 Graphe de la résistance en fonction de la lumière.

4.1.6 Le relais

Le relais est un composant électromécanique qui permet de dissocier la partie puissance de la partie commande : il permet l'ouverture/fermeture d'un circuit électrique par un second circuit complètement isolé (isolation galvanique) et pouvant avoir des propriétés différentes.

Le relais (Figure III.8) comporte deux parties :

- ✓ Une bobine qui induit un champ magnétique lorsqu'elle est parcourue par un courant électrique. C'est la partie commande.
- ✓ un contact ou interrupteur mis en mouvement lorsque le champ magnétique est présent. C'est la partie puissance. Le courant circulant à travers ce contact peut atteindre plusieurs Ampère [21].



Figure. III.8 Le module relais.

4.1.7 Le Servomoteur

Le servomoteur (souvent abrégé en « servo », provenant du latin servus qui signifie « esclave ») est un système qui a pour but de produire un mouvement précis en réponse à une commande externe, C'est un actionneur qui mélange l'électronique, la mécanique et l'automatique. Il est composé :

- D'un moteur à courant continu
- D'un axe de rotation

- D'un capteur de position de l'angle d'orientation de l'axe (très souvent un potentiomètre)
- D'une carte électronique pour le contrôle de la position de l'axe et le pilotage du moteur à courant continu

Un servomoteur est capable d'attendre des positions prédéterminées dans les instructions qui lui ont été données, puis de les maintenir. Il a l'avantage d'être asservi en position angulaire, cela signifie que l'axe de sortie du servomoteur respectera la consigne d'instruction que vous lui avez envoyée en son entrée. Même si un obstacle se présente sur la route, qui viendrait à lui faire changer l'orientation de sa trajectoire, le servomoteur essaiera de conserver la position. Pour un ajustement précis de la position, le moteur et son réglage sont équipés d'un système de mesures qui détermine la position courante [23].



Figure. III.9 Servomoteur.

On note quelques différences entre un servomoteur et un simple moteur :

- ✓ Le servomoteur coûte plus cher qu'un moteur.
- ✓ Le servomoteur est plus précis, si on met les bons encodeurs et qu'on règle bien les drivers.
- ✓ Le servomoteur est plus puissant pour la même puissance consommée (rendement de plus de 70%).
- ✓ Le moteur fonctionne en boucle ouverte alors qu'un servomoteur fonctionne en boucle fermée.

Ainsi que quelques avantages :

- ✓ Le fil signal à faible courant peut-être raccordé directement à une sortie du PIC, pas besoin de circuit d'interface.
- ✓ On peut commander l'arrêt, la marche, le sens de rotation et la vitesse du servomoteur à l'aide d'un seul fil, Economie d'E/S.
- ✓ Le servomoteur offre un couple important sous un volume réduit.

4.2 La partie logicielle

4.2.1 Fritzing

Fritzing est un logiciel Open Source développé par l'université de Postdam aux Pays-Bas. Ce logiciel permet de :

- Réaliser des schémas de câblage sur platine d'essai. (figure III.10)
- Saisir des schémas structurels.
- Dessiner des typons.
- Editer des programmes Arduino et les télécharger sur la carte [21].

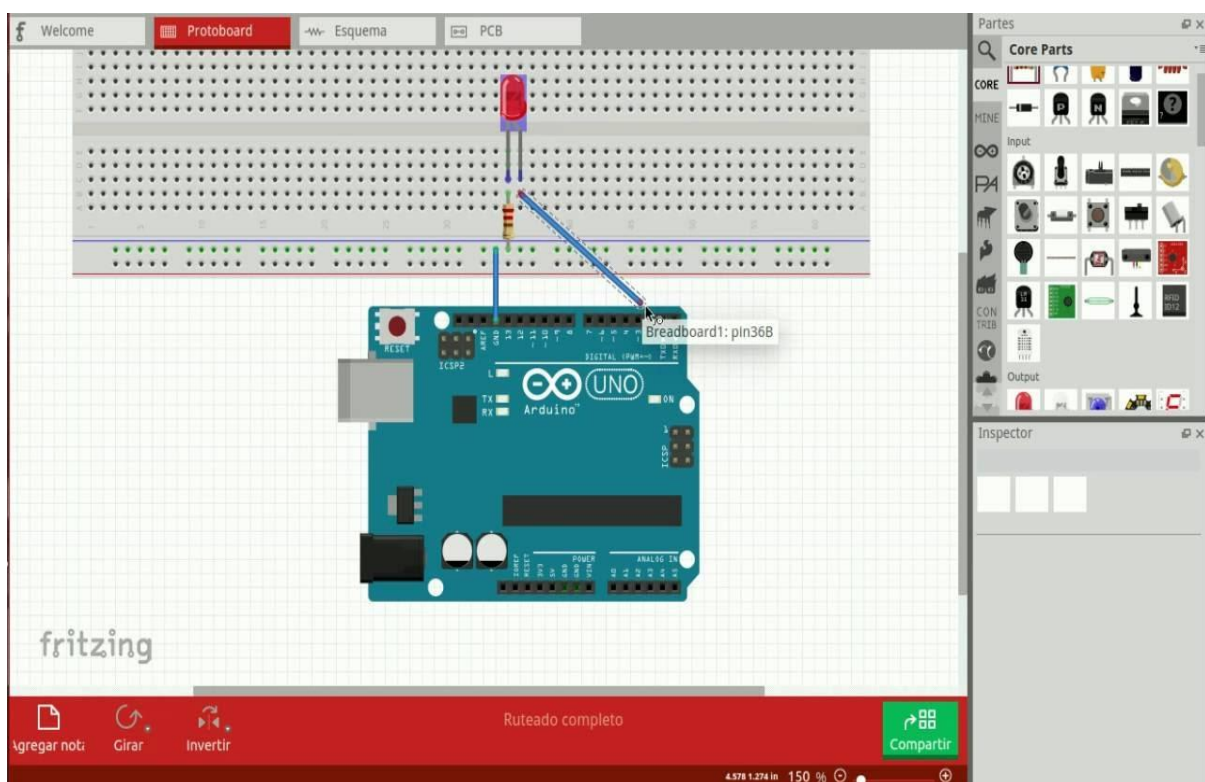


Figure. III.10 L'interface du logiciel Fritzing.

4.2.2 Android

Android, est un système d'exploitation mobile basé sur le noyau Linux et développé actuellement par Google. Lancé en juin 2007 à la suite du rachat par Google en 2005 de la startup du même nom, le système avait d'abord été conçu pour les smartphones et tablettes tactiles, puis s'est diversifié dans les objets connectés et ordinateurs comme les télévisions (Android TV), les voitures (Android Auto), les ordinateurs (Android-x86) et les smartwatch (Android Wear). En 2015, Android est le système d'exploitation mobile le plus utilisé dans le monde avec plus de 80 % de parts de marché dans les smartphones, devant iOS d'Apple [26].



Figure. III.11 Le Logo d'Android.

4.2.3 App inventor

App inventor est un IDE (environnement de développement intégré) qui permet la création d'applications destinées à des systèmes équipés de plates-formes Android. Parmi ces systèmes, on trouve des téléphones ainsi que des tablettes.

Cet environnement de programmation permet une programmation graphique aisée, basée sur l'assemblage de blocs. Des connaissances en programmation orientée objet sont toutefois nécessaires. Cependant, on pourra s'affranchir complètement de la connaissance des noms des propriétés et méthodes liées aux objets. En effet des blocs de propriétés, méthodes et événements seront directement proposés dès la création d'un objet.

L'IDE (Figure III.12) est formée de deux composantes logicielles :

- Un site web permettant de créer un projet : création de la page graphique de la future application (dépôt des objets nécessaires à l'application sur une page de). Ce site Web a besoin d'un compte google (gmail) pour fonctionner ;
- un programme résident sur l'ordinateur, qui permet la saisie du programme (association des blocs liés aux objets précédemment définis dans l'espace web). Ce programme permet également le téléchargement de l'application vers le média désiré (tablette, téléphone...).

A l'origine le projet APP INVENTOR a été créé par le Google Labs... puis abandonné. Aujourd'hui ce concept est repris par le MIT (Massachusetts Institute of Technology).

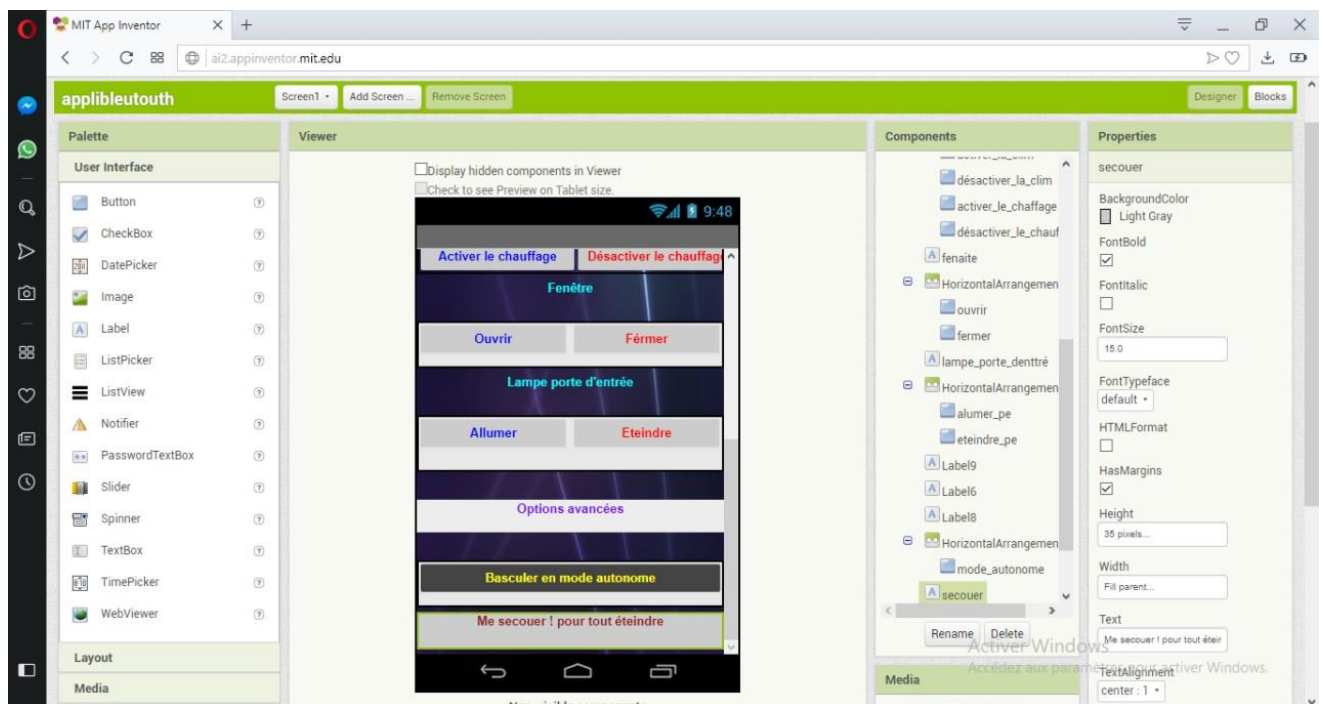


Figure. III.12 L'interface graphique d'App Inventor.

5. Conclusion

Dans ce chapitre nous avons présenté les détails de la structure du système à concevoir, que ce soit la partie matérielle ou la partie logicielle. Nous avons déterminé les différents composants nécessaires pour la conception de notre projet. Le dernier chapitre sera consacré à la conception des applications domotique.

Chapitre IV :

Conception de la carte du contrôle

1. Introduction

L'objectif de ce chapitre c'est bien l'implantation des différentes applications discutées précédemment tout au long des deux derniers chapitres.

2. Conception du système

A premier lieu, la conception matérielle est faite par parties : chaque partie du système domotique conçu est réalisée et testée séparément sur des Lab d'essai (Breadboard).

Après les avoir conçues et adoptés séparément, on les a regroupées ensemble avec la carte Arduino Uno.

Ce système domotique englobe 09 parties :

- La carte Arduino Uno.
- Le module Bluetooth HC-06.
- Le capteur de gaz MQ-2.
- Le capteur de lumière (LDR).
- Deux détecteur de mouvement (PIR) HC-SR 501.
- Le capteur de température LM 335.
- Commande des lampes.
- Commande de la fenêtre.
- Commande du climatiseur et du chauffage.

L'ensemble de ce système sera commandé par l'application Androïde définie dans la suite.

2.1 La carte Arduino Uno

- Les broches 12 et 13 reliées au Bluetooth.
- Les broches 9, 10, 11 et A1 sont réservées au capteur de gaz MQ-2.
- Les broches 8 et A2 sont réservées au capteur de lumière (LDR).
- Les broches 2, 3, 4, 5, 6, 7 sont réservées pour le détecteur de mouvement HC-SR501.
- Les broches 1, 2 et A3 sont réservées au capteur de température LM3 335.

2.2 Bluetooth

Une fois la communication sans fil est établie, le module Bluetooth permet l'échange bidirectionnel des données à des distances courtes, en utilisant des ondes radio UHF.

Ce type de liaison a pour objectif de simplifier les connexions entre les appareils électroniques en supprimant les liaisons filaires.

Dans ce cas, on va connecter le Bluetooth à une carte Arduino Uno pour permettre l'échange d'information entre cette carte, ses périphériques ainsi que l'application installée sur un Smartphone pour contrôler notre système.

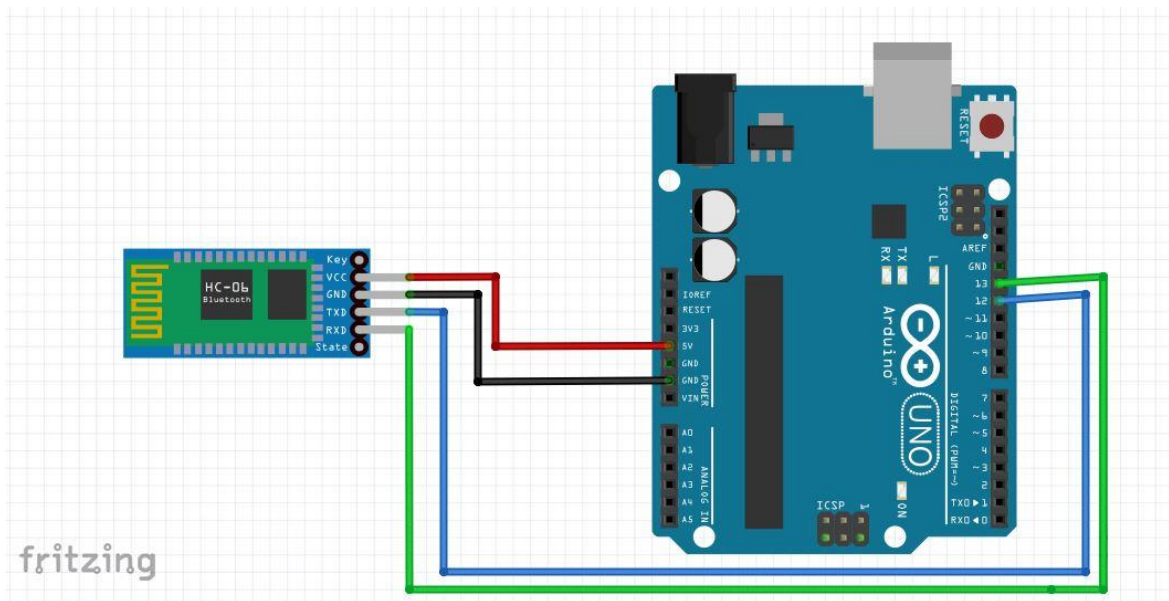


Figure. IV.1 Représentation graphique de la communication par Bluetooth HC-06

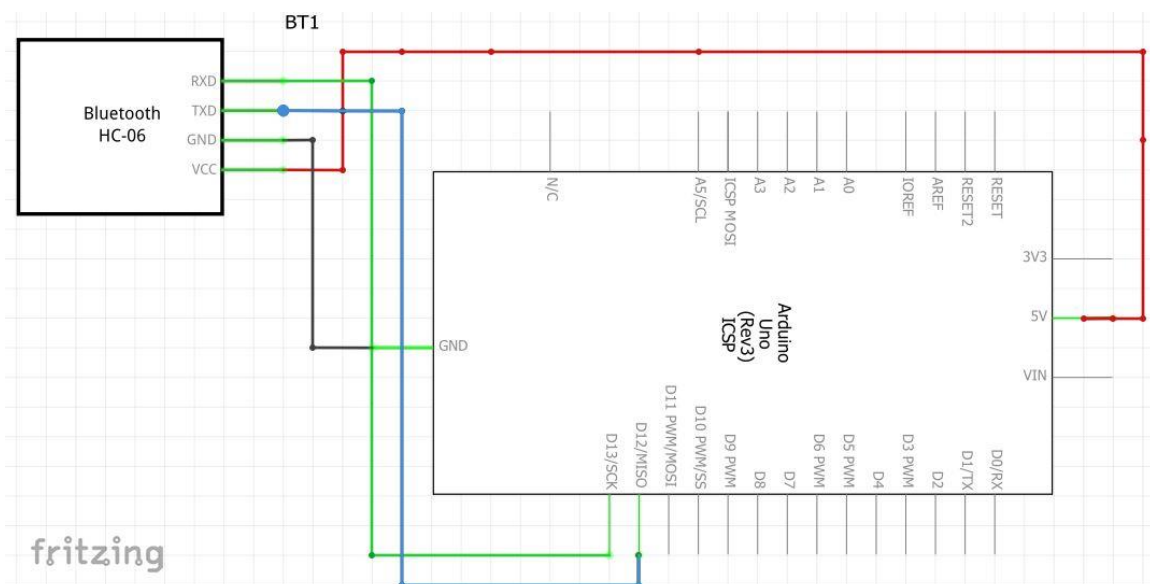


Figure. IV.2 Schéma électrique de la communication par Bluetooth HC-06

2. 3 Le capteur de gaz MQ-2

Pour des raisons de sécurité, on va équiper ce système avec un capteur de gaz et fumée MQ2, pour détecter la fuite de gaz et prendre, si tel est le cas, des mesures de sécurité pour éliminer les dangers qui peuvent survenir dans la maison.

On place ce capteur à l'intérieur de la maison.

En cas où le capteur détecte une fuite de gaz, le buzzer s'active la fenêtre s'ouvrira automatiquement, afin d'aérer la maison et diminuer les risques, en attendant l'intervention du propriétaire.

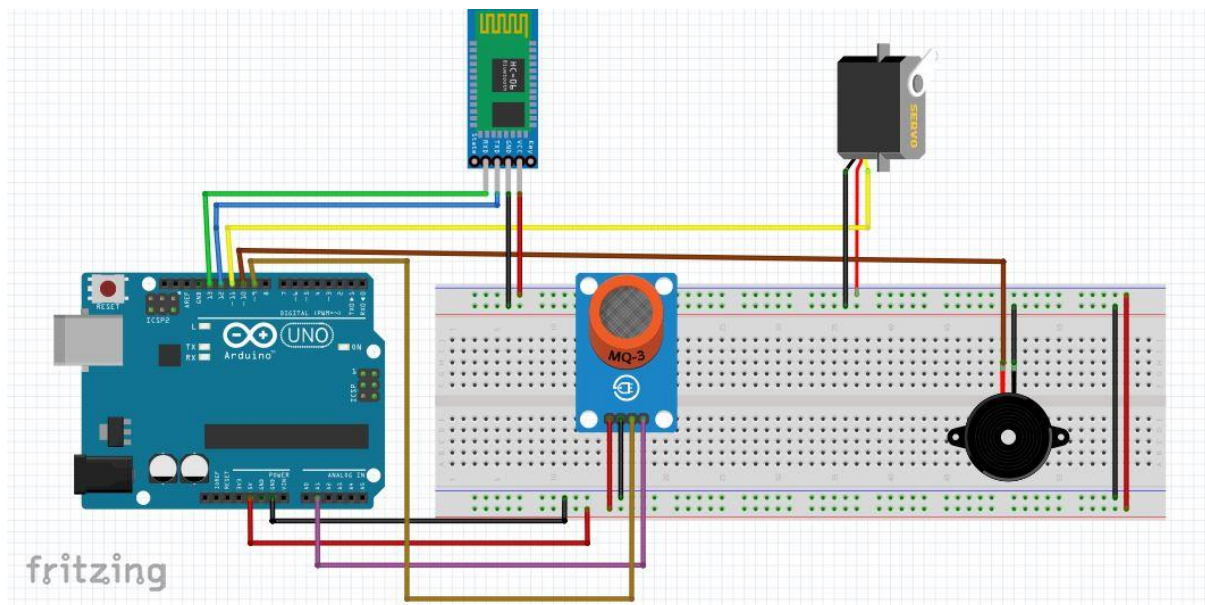


Figure. IV.3 Représentation graphique du système de détection de gaz (capteur de gaz MQ-2).

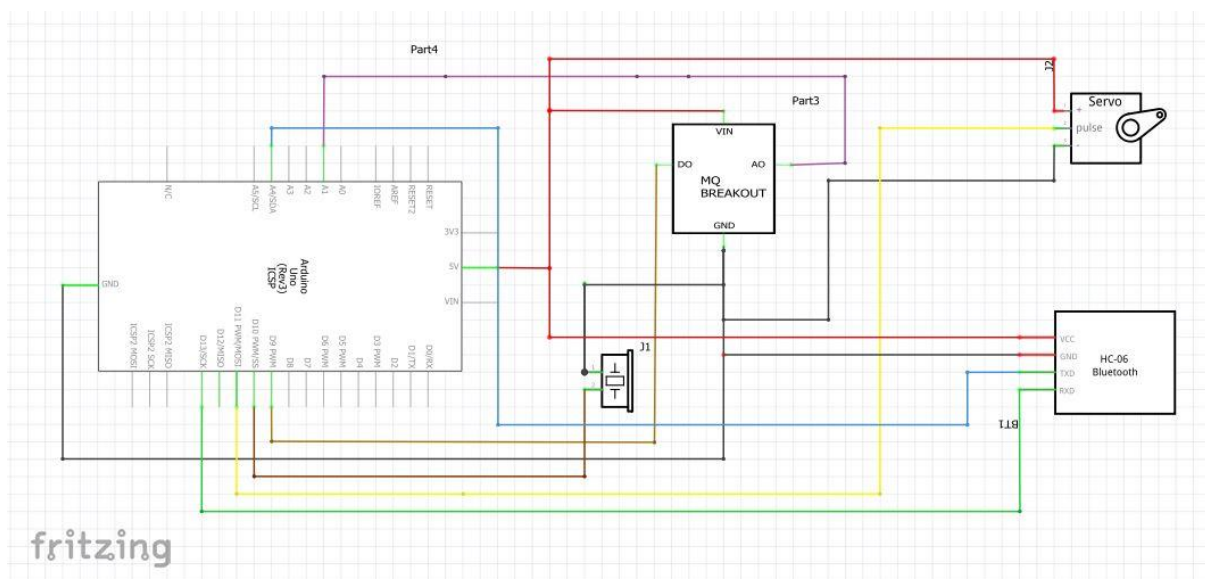


Figure. IV.4 Schéma électrique du système de détection de gaz (capteur de gaz MQ-2).

Le diagramme suivant résume le fonctionnement du système de détection de fuites de gaz avec le capteur de gaz MQ-2 :

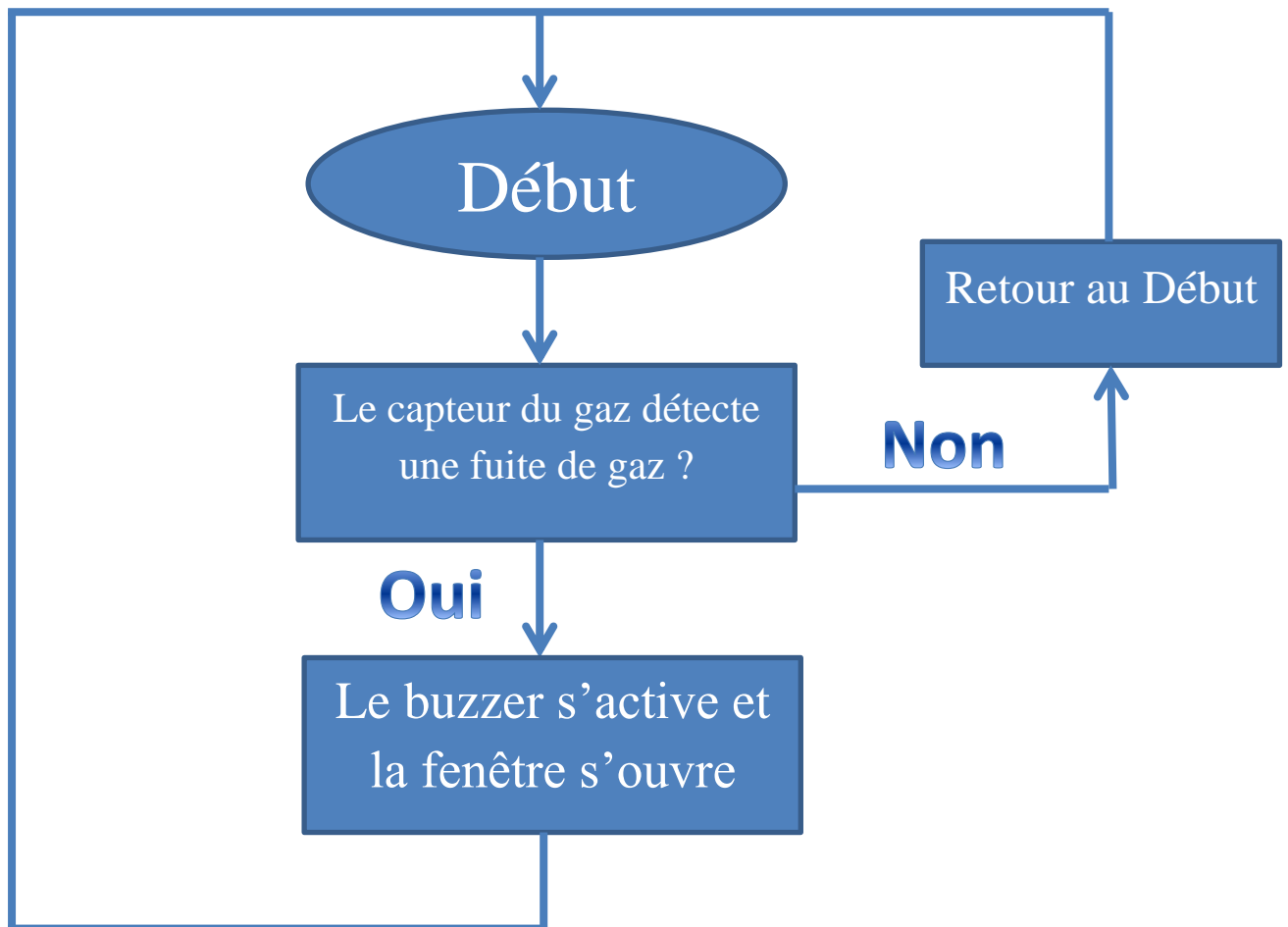


Figure. IV.5 Diagramme fonctionnel du système de détection de gaz.

2. 4 Le capteur de lumière (LDR)

Grâce au capteur de lumière (LDR) reliée à un relais contrôlant l'arrivée de courant à la lampe, on va réaliser un éclairage automatique externe. La LDR va allumer et éteindre automatiquement la lumière de la lampe installée à l'entrée de la porte de notre maison. Lorsque la lumière est forte, l'Arduino reçoit ce changement de la part de la LDR et elle envoie alors un signal de niveau haut au relais pour qu'il s'ouvre ou reste ouvert, et inversement, quand il fait sombre, un signal de niveau bas lui sera envoyé pour le basculer ; ce qui déclenchera l'allumage de la lampe.

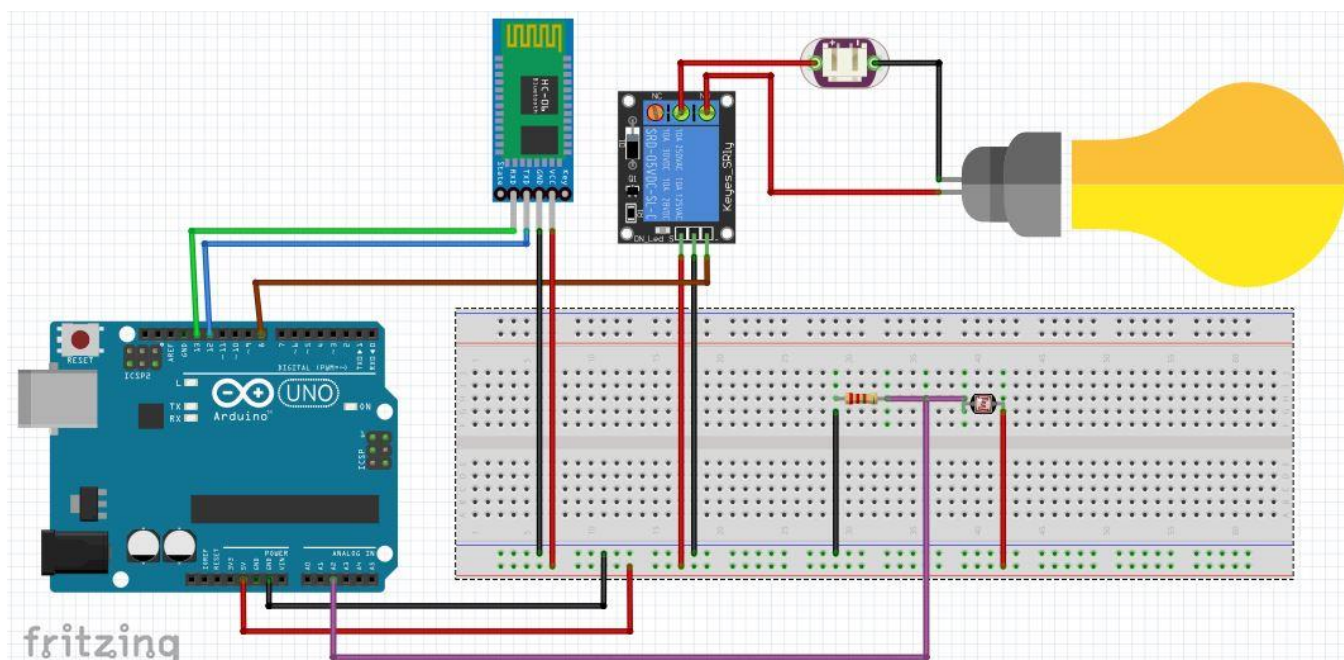


Figure. IV.6 Représentation graphique du système d'éclairage automatique externe.

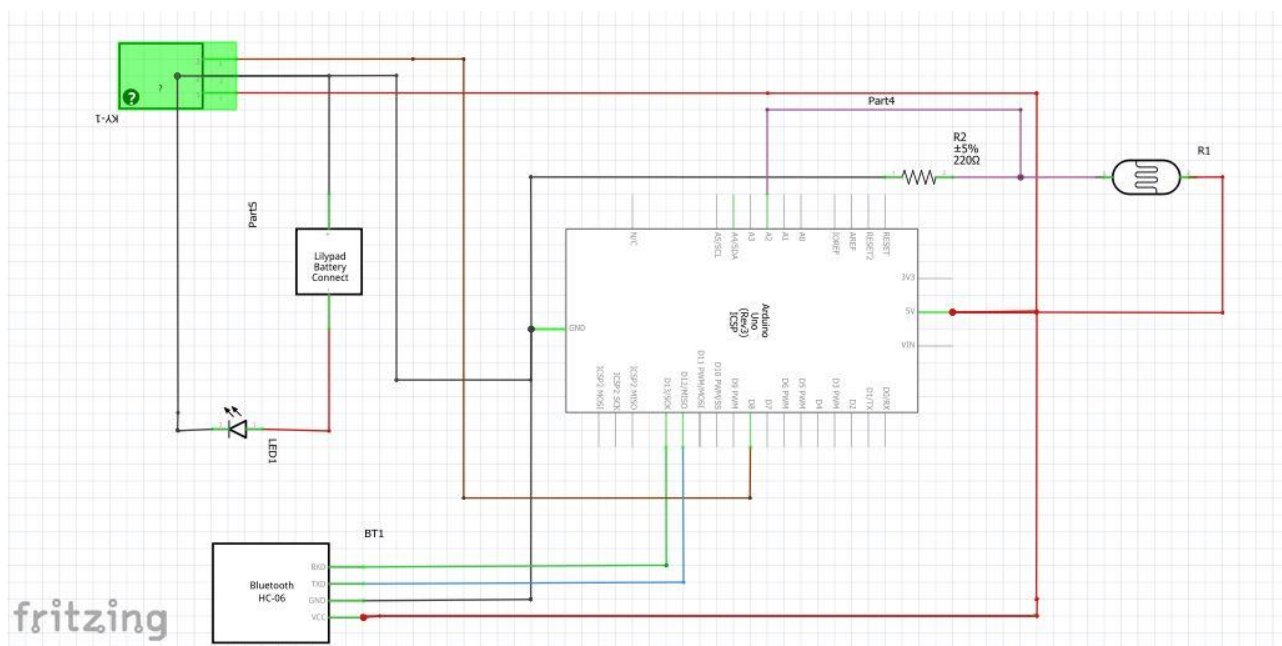


Figure. IV.7 Schéma électrique du système d'éclairage automatique externe.

Le diagramme suivant résume le fonctionnement du système d'éclairage extérieur avec le détecteur de lumière LDR.

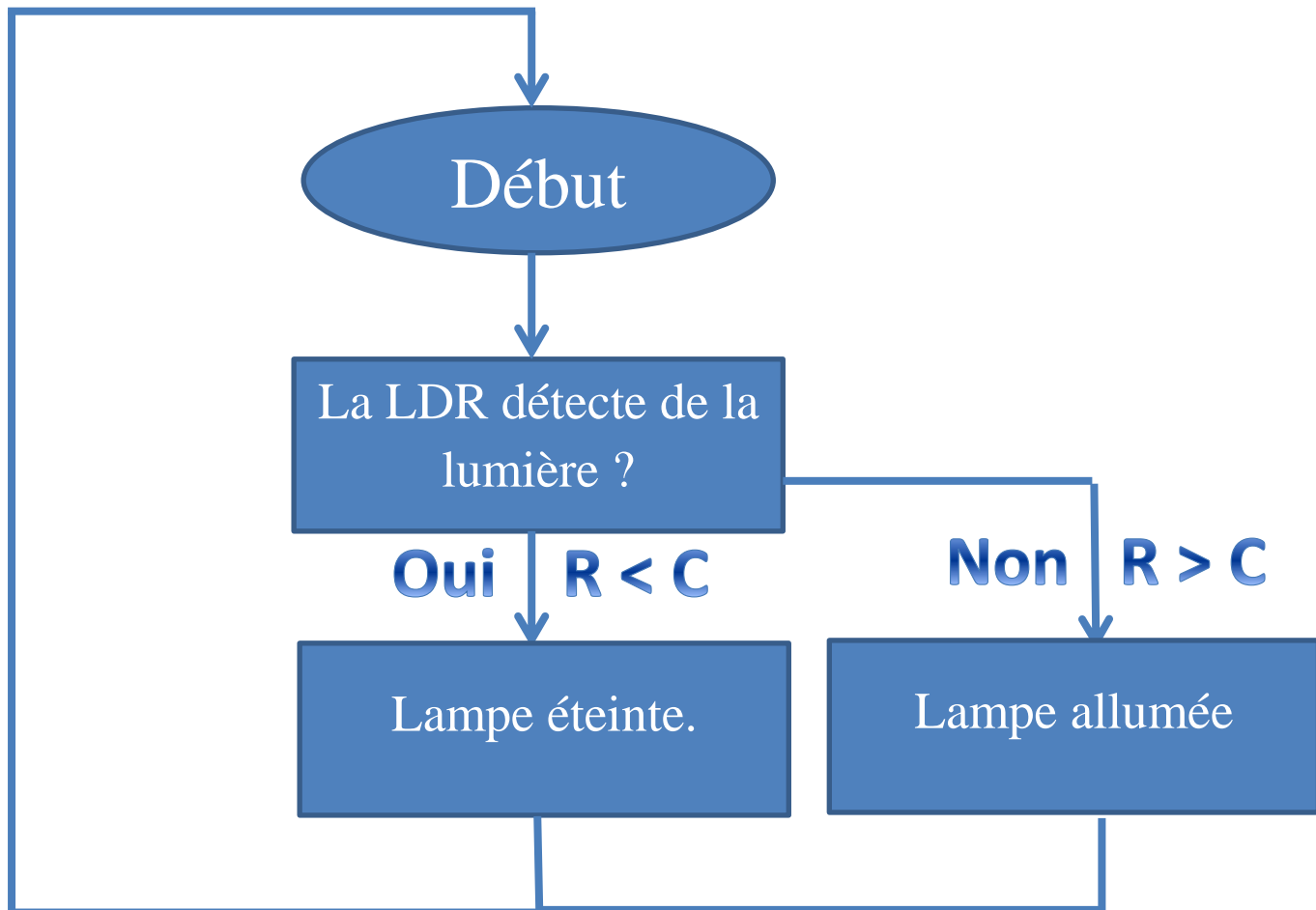


Figure. IV.8 Diagramme fonctionnel du système d'éclairage extérieur.

2. 5 Le détecteur de mouvement HC-SR 501

Le capteur de mouvement HC-SR501 (PIR) qu'on va utiliser est un capteur capable de détecter les mouvements. Il détecte les rayonnements infrarouge dans son champ de vision et en déduit une présence ou un mouvement.

On va utiliser deux de ces capteurs reliés à un relais, afin de réaliser un système d'éclairage automatique intérieur, dans une maison composée, d'une grande chambre non séparée de la cuisine et un bureau.

On pose un des détecteur à l'entrée de la porte. Lorsque qu'il n'y a pas de présence détectée, le HC-SR 501 reste à son état bas, ainsi, le relais reste ouvert et les lampes éteintes. Mais une fois qu'on rentre à la maison et qu'un mouvement est pris en compte par le capteur, le signal passe à l'état haut basculant l'état du relais ce qui déclenchera l'allumage de la première lampe (celle du salon), suivie par celle de la cuisine, après un délai de 5 secondes. Le même cas pour les deux autre lampes placées au deux coin du bureau composé d'un bureau et un atelier de travail.

Lorsque le capteur placé à l'entrée de cette pièce détecte une présence, la première lampe s'allume suivie de la deuxième, après un délai de 5 secondes.

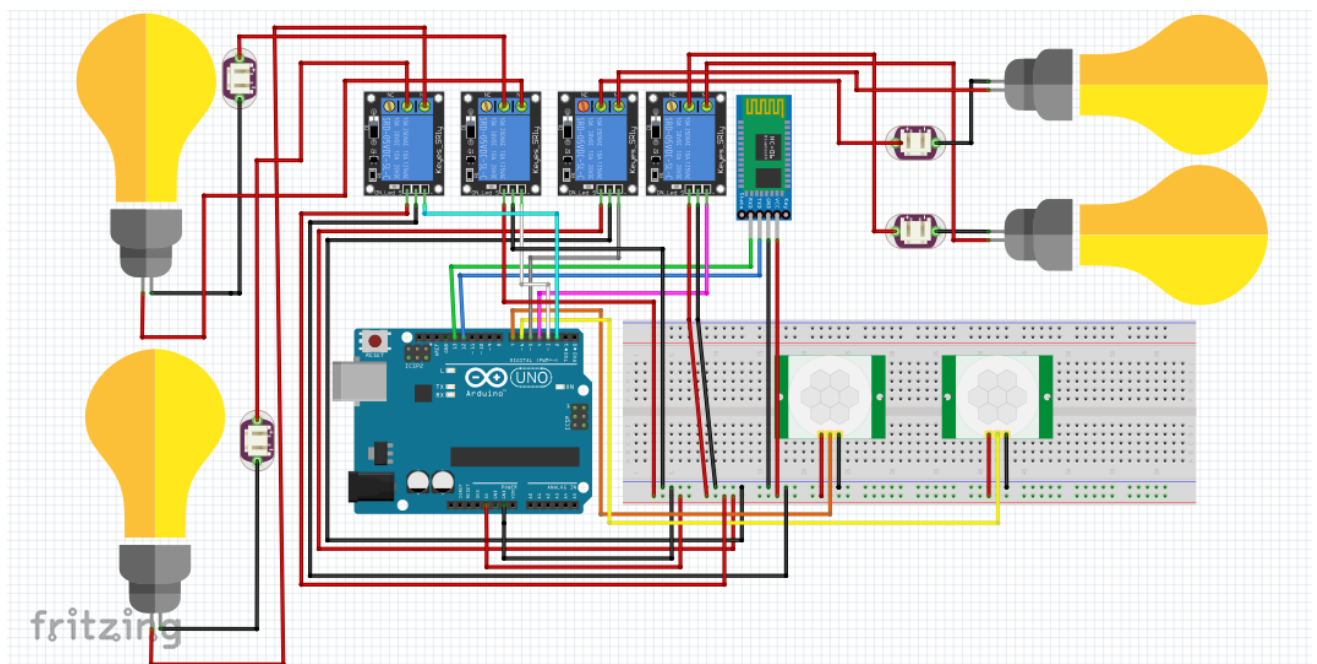


Figure. IV.9 Représentation graphique du système d'éclairage automatique interne .

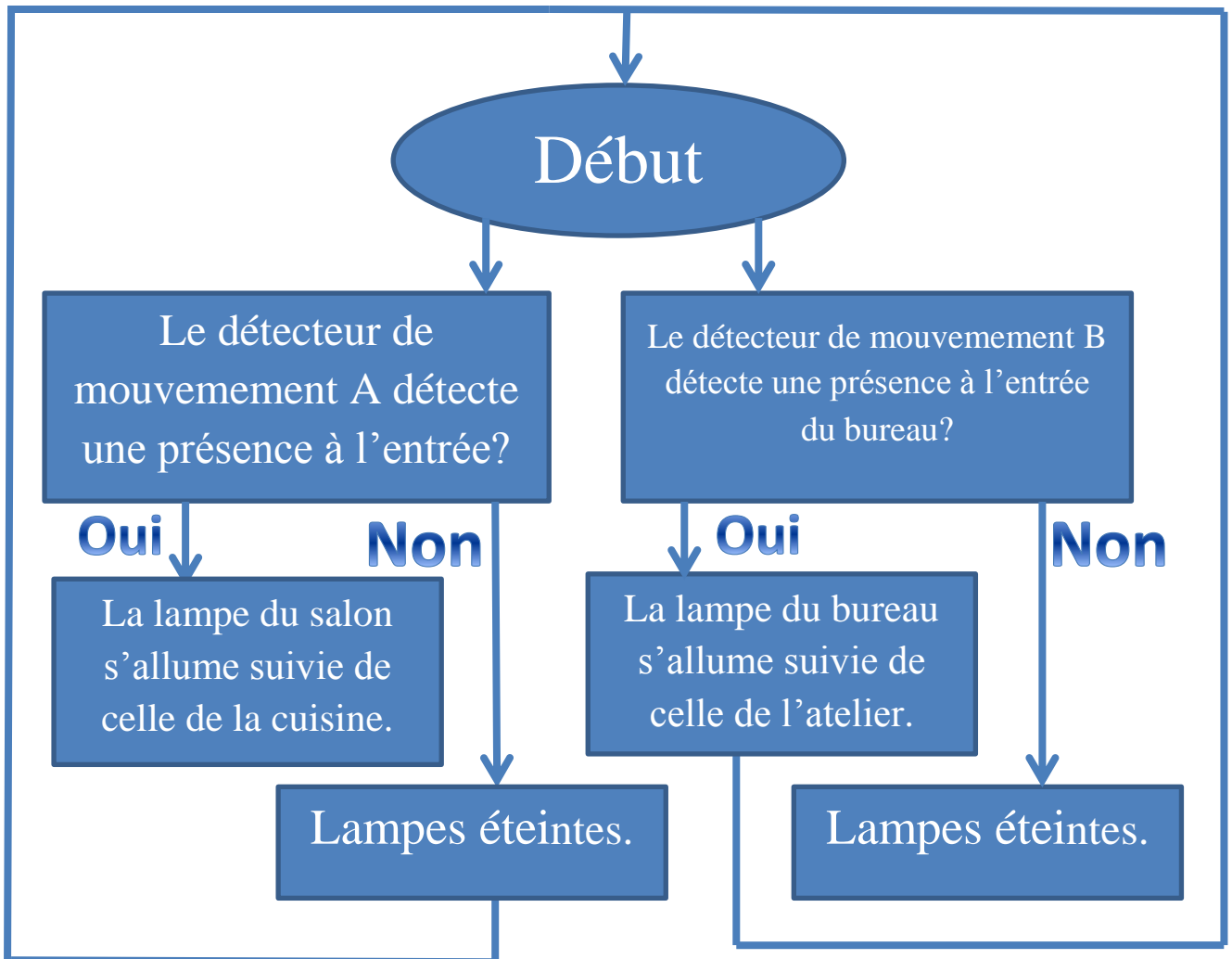


Figure. IV.10 Le diagramme fonctionnel du système d'éclairage intérieur.

2. 6 Le capteur de température LM 335

L'objectif de l'utilisation du capteur de température LM335, est de régulariser la température ambiante de notre maison. En fixant un intervalle de température désirée au sein de notre domicile, allant de 22 degrés Celsius jusqu'à 27 degrés Celsius en été , La carte arduino va comparer la température relevée par le capteur. Et cas où, la température ambiante dépassera celles de l'intervalle, elle enverra un signal sur à notre application mobile, afin d'allumer le climatiseur.

Même principe pendant l'hiver : Lorsque la température ambiante est relevée à 10 degrés Celsius, l'information sera reçue sur l'application et on allume le chauffage.

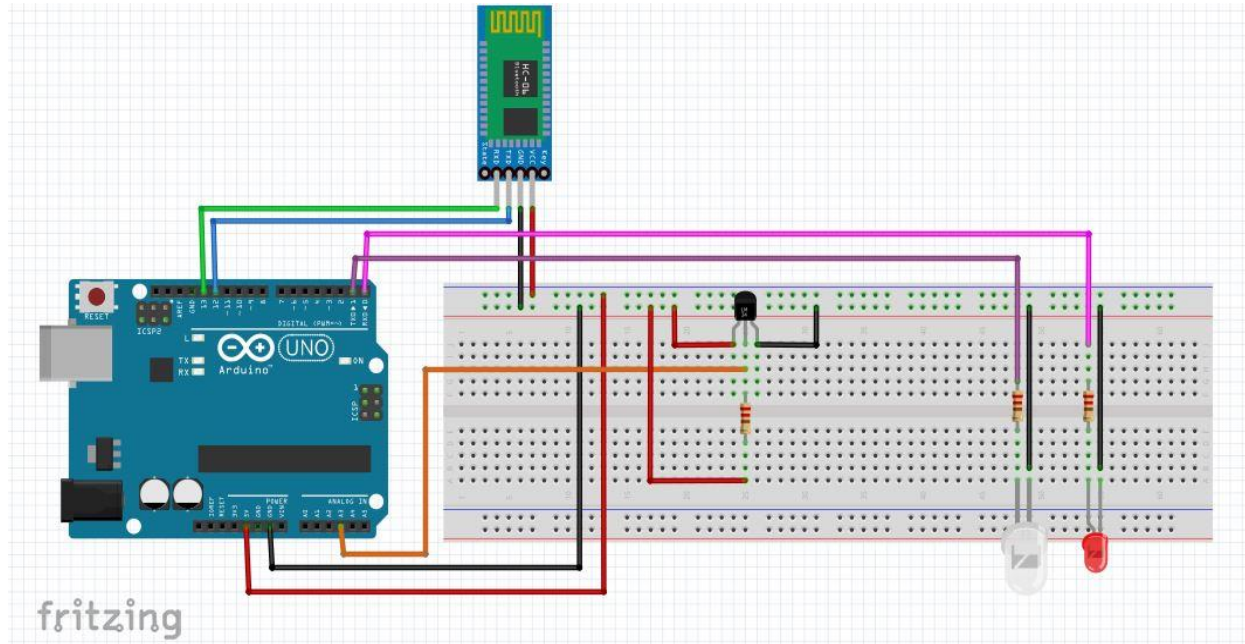


Figure. IV.11 Représentation graphique du branchement du capteur de température LM 335.

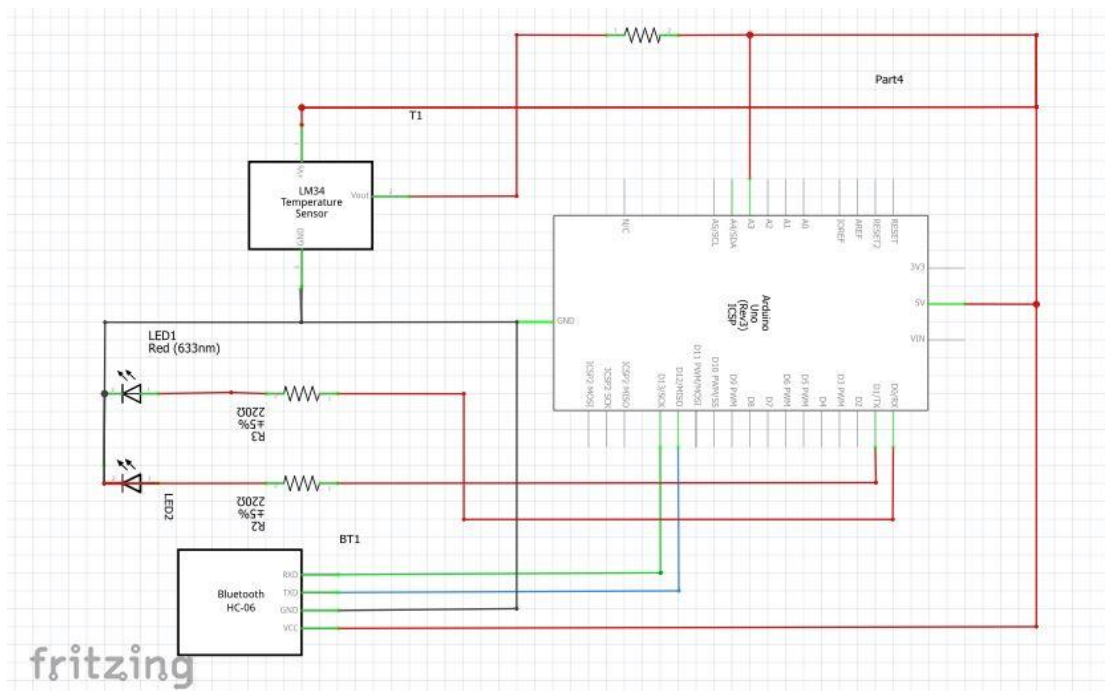


Figure. IV.12 Schéma électrique du branchement du capteur de température LM335.

Le diagramme fonctionnel suivant résume la régulation semi-automatique de la température dans la maison à base du capteur de température LM 335 :

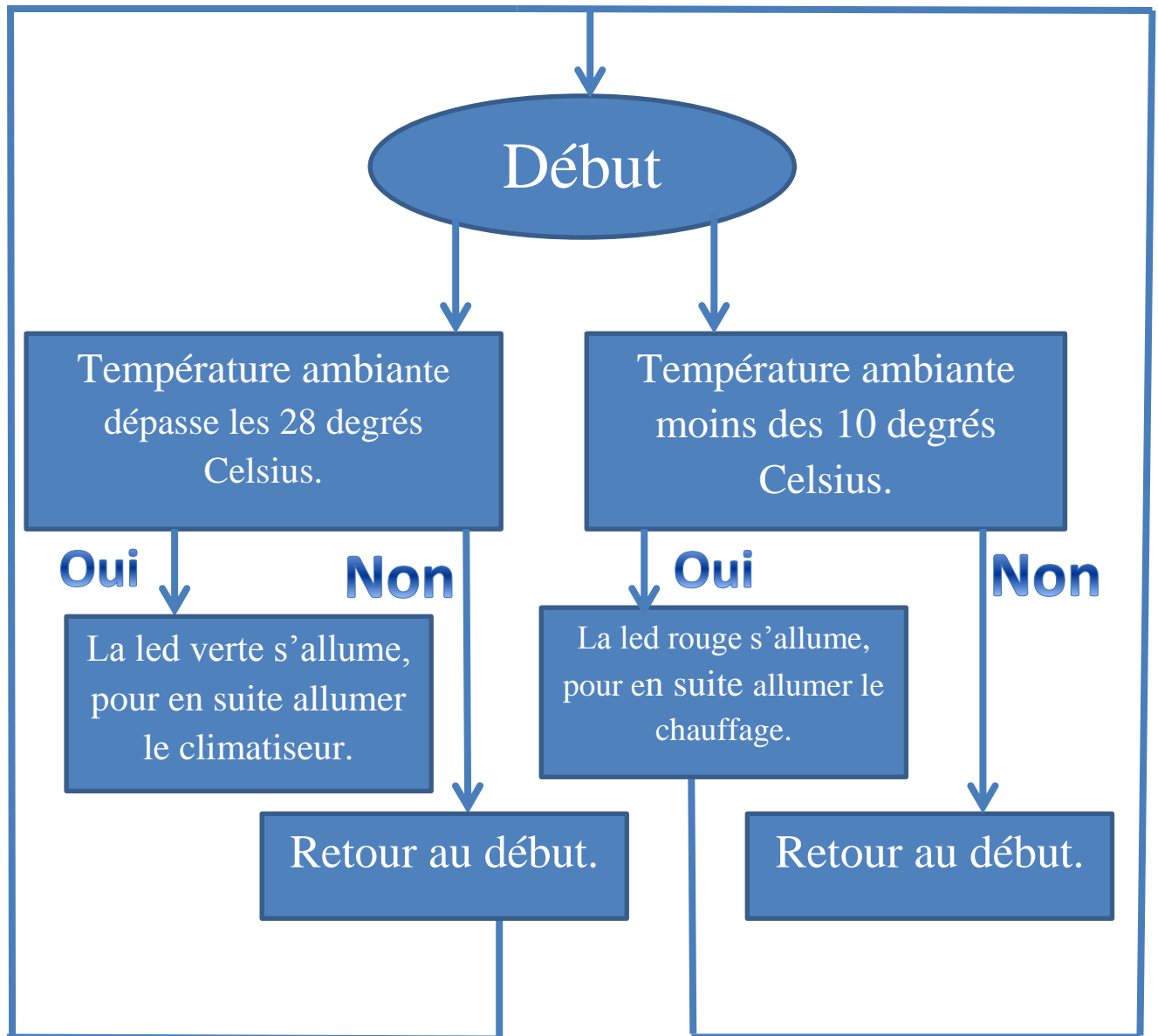


Figure. IV.13 Diagramme fonctionnel du système de régulation semi-automatique de la température.

2.7 Présentation de l'application APK crée sur APP INVENTOR

L'application créée sur APP INVENTOR va nous permettre de commander notre système domotique. Une fois l'application est ouverte, on doit appuyer sur la touche « listes des périphériques » en suite choisir le bluetooth HC-06 utilisé dans notre système et le connecté.

Une fois le bluetooth connecté, une liste de commandes qui sont représentées par la figure. IV.14 et la figure. IV.15, s'affiche.

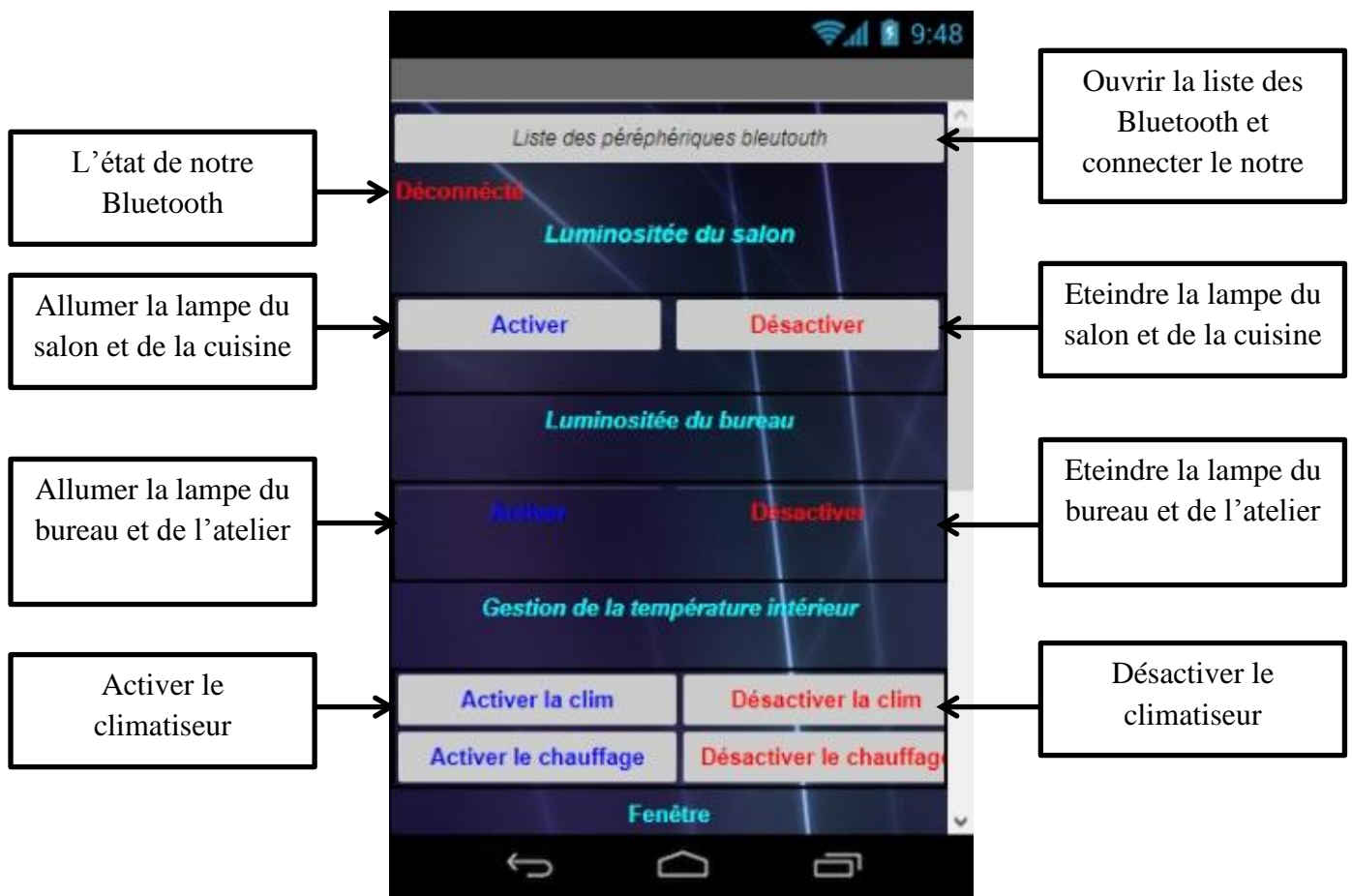


Figure. IV.14 Page de commande de l'application mobile.

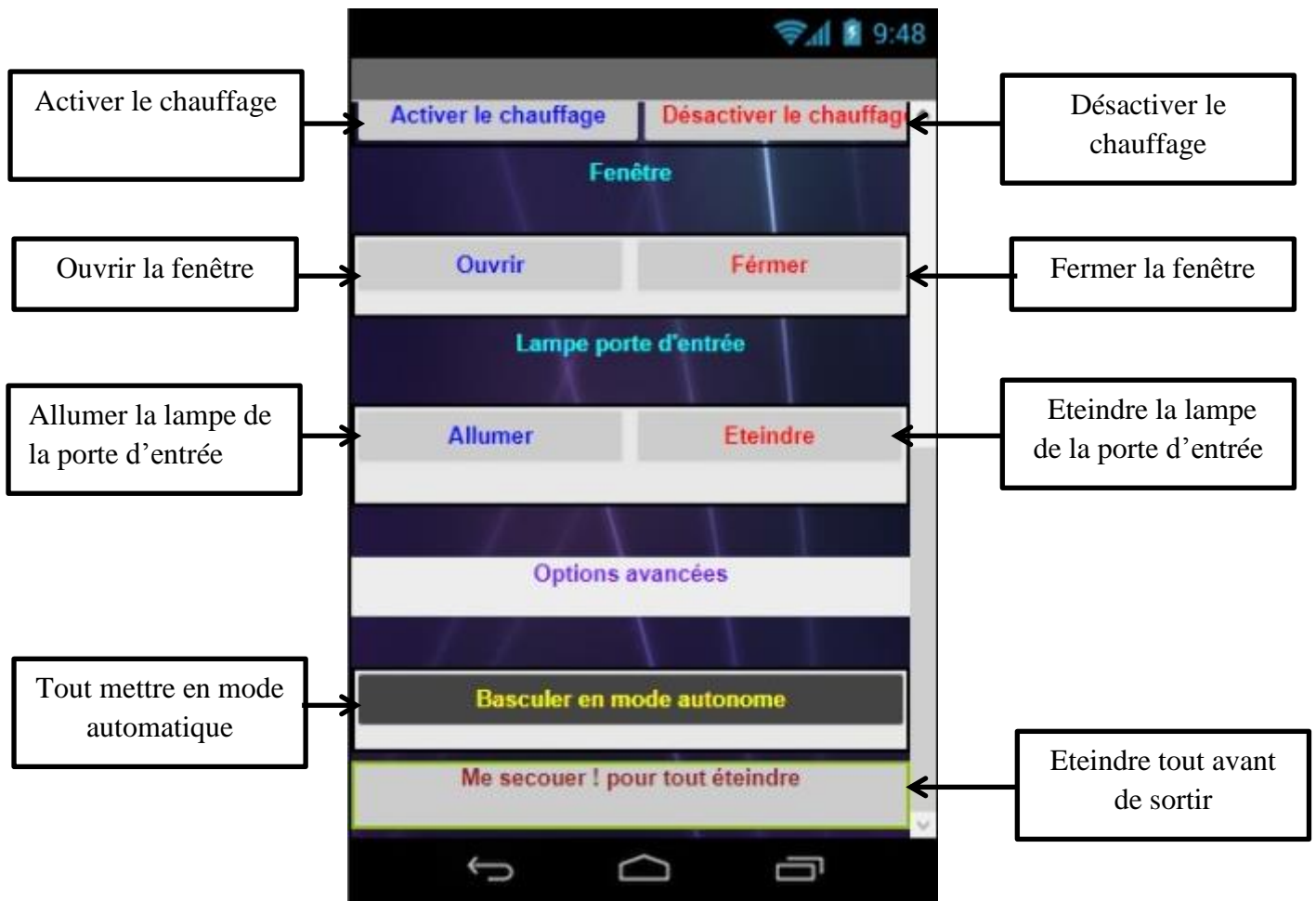


Figure. IV.15 Suite de la page de commande de l'application mobile.

3. Conclusion

Dans ce chapitre on s'est consacré à la conception des différentes applications discutées au deuxième et au troisième chapitre. Aussi, nous avons présenté tout ce qui concerne la structure et les schémas fonctionnels du système pour l'aboutissement de ce projet. Ensuite une vue détaillée du système (cartes et composants) et de l'application (architecture, hiérarchie, entités...etc.) a été mise en évidence. Enfin nous avons conclu par les interfaces permettant de contrôler différentes entités du système.

Conclusions générales

Conclusions générales & Perspectives

Au bout de mon cursus Master d'instrumentation en électronique, j'ai été chargé de réaliser un projet de fin de d'études. Mon travail s'est basé sur l'étude et la conception d'une carte de contrôle à base d'ARDUINO pour des applications.

Ce projet m'a amené à découvrir une nouvelle plateforme de développement et à enrichir mes connaissances théoriques ainsi que pratiques et mon expérience dans le domaine d'internet des objets, et de mieux comprendre les composants et technologies concernés par ce domaine. Ce dernier qui est devenu un grand domaine de recherche et un grand marché de travail au monde.

Que ce soit pour une maison connectée ou pour tout autre projet d'informatique embarquée, la connaissance de l'environnement Arduino est nécessaire pour implanter ce genre de projet. Par conséquent, j'ai présenté, dans le deuxième chapitre les différentes cartes électroniques ainsi que leur environnement logiciel permettant de les programmer.

La conception a été initiée par une mise en évidence de la problématique, l'objectif, et les fonctions du futur système. En suite j'ai établi une analyse détaillée de mon application en analysant chaque partie de mon système conçu séparément, en utilisant le logiciel Fritzing, qui est un standard incontournable dans la discipline de la conception logicielle. J'ai appris aussi, à m'initier à la programmation mobile, en particulier, la conception et la réalisation d'une application mobile sous le système Android, pour commander des objets connectés.

Cela n'empêche pas d'avoir rencontré plein de contraintes et d'obstacles notamment le manque de composants électroniques, ce qui m'a empêché d'ailleurs de réaliser ma maison connectée à temps et me contenter de sa conception.

L'autre problème que j'ai croisé, est le manque de documentations officielles, c'est à dire que j'ai pas trouvé des références académiques, vu que le domaines des systèmes embarquée, à savoir dans mon cas, l'Arduino, Fritzing, App inventor sont en Open-Source, ce qui rend la majorité de la documentions disponible sur internet, rédigée par des gens passionnés par l'électronique embarquée et les nouvelles technologies.

Conclusions générales & Perspectives

Ce travail m'a fait aussi découvrir ma passion pour le domaine de l'électronique embarquée, que je compte nourrir prochainement avec de nouvelles aventures, et des nouveaux mini-projets à réaliser, que ce soit en groupe ou pour l'amour personnel de ce domaine, et bien évidemment des améliorations futures peuvent être apportées à ce projet pour qu'il soit réalisé déjà, en suite enrichir avec plus d'applications domotiques qui répondront aux besoins quotidien. Ainsi que l'utilisation d'un matériel plus puissant comme la carte Raspberry pour remplacer la carte Arduino et le WIFI pour remplacer le Bluetooth qui permettra le pilotage de la maison, via un site web relié à l'application mobile, de n'importe quel endroit au monde où il y aura une connexion internet.

Pour conclure, j'espère par le biais de ce travail, apporter une validation de ces techniques de conception et donner une bonne cause pour mieux explorer ces domaines d'internet des objets et de l'électronique embarquée.

RÉFÉRENCES BIBLIOGRAPHIQUES

1. http://docplayer.fr/amp/1610815-Systeme-de-gestion-de-flux-pour-l-internet-des-objets-intelligents.html#download_tab_content (Mai 2018)
2. Le guide de la maison et des objets connectés Domotique, smart home et maison connectée-notag
3. www.cisco.com/c/dam/global/en_ca/solutions/executive/assets/pdf/internet-of-things-fr.pdf
4. www.Wikipédia.org
5. Y. Challal. Sécurité de l'Internet des Objets : vers une approche cognitive et systémique. PhD thesis, Université de Technologie de Compiègne, 2012
6. <http://www.lefigaro.fr/secteur/high-tech/2016/08/13/32001-20160813ARTFIG00034-ces-objets-connectes-utilises-par-les-sportifs-olympiques.php>
7. <http://www.zdnet.fr/actualites/les-villes-intelligentes-s-appuient-sur-l-iot-mais-c-est-encore-la-confusion-en-matiere-de-securite-39855378.htm>
8. http://technologies.lesechos.fr/decryptage/que-peut-apporter-l-internet-des-objets-a-l-industrie-_f-71.html
9. <https://id2domotique.com/2016/06/15/la-domotique/#more-2>
10. <http://blog.ac-versailles.fr/technopeguy/public/Programmation/Arduino.pdf>
11. <http://etudeetprogrammationcartearduinouno.blogspot.com/>
12. https://zestedesavoir.com/tutoriels/686/arduino-premiers-pas-en-informatique-embarquee/742_decouverte-de-larduino/3414_presentation-darduino/#3-10776_les-cartes-arduino
13. <http://etudeetprogrammationcartearduinouno.blogspot.com/>
14. <https://technologieeducationculture.fr/article.php?sid=128>
15. http://www.monclubelec.fr/pmwiki_reference_arduino/pmwiki.php?n=Main.MaterielUo
16. https://www.flossmanualsfr.net/manuals/view/arduino/ch013_programmer-arduino/
17. <http://geii.iut-troyes.univ-reims.fr/wikigeii/index.php?title=Batak>
18. <http://wiki.t-o-f.info/Arduino/CapteurPIRDeMouvement>
19. <http://fr.hobbytronics.co.uk/mq2-gas-smoke-sensor>
20. <https://eskimon.fr/tuto-arduino-907-utiliser-un-module-bluetooth-hc-05-avec-arduino>
21. <http://pecquery.wixsite.com/arduino>
22. www.framboise314.fr/scratch-raspberry-pi-composants/module-detecteur-de-lumiere-a-ldr
23. <https://www.supinfo.com/articles/single/296-qu-est-ce-qu-servomoteur>
24. <https://fr.scribd.com/document/238089643/Arduino-Capteur-de-Temperature>
25. <http://ww2.ac-poitiers.fr/sciences-ingenieur-sti/spip.php?article88>
26. https://fr.wikipedia.org/wiki/Android#cite_note-lee-7

Titre : Etude et conception d'une carte de contrôle à base d'ARDUINO pour des applications domotique.

Résumé : L'Internet des Objets est une nouvelle technologie qui vise à intégrer le monde virtuel de l'Internet avec le monde physique des objets intelligents communicants, tels que les capteurs et les actionneurs, afin d'améliorer et d'assurer une meilleure accessibilité et exploitation de ces objets intelligents.

À travers une telle évolution, l'expansion de l'Internet d'aujourd'hui touche à plusieurs domaines d'applications comme la sécurité, la santé, les maisons et villes intelligentes, l'agriculture, et les systèmes de logistique et de transportation intelligents.

Dans ce contexte, l'objectif de ce projet de fin d'étude consiste à la conception d'une maison intelligente à base d'une carte microcontrôleur, Arduino Uno, équipée d'une fenêtre, des lampes, d'un servomoteur, d'un buzzer et de quelque capteurs de température, de fumée, de lumière et de présence, et contrôlés à distance par une application Android.

Mots-clés : Internet des Objets, internet of things, Arduino Uno, domotique, capteurs et d'actionneurs, Android..