

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE

UNIVERSITE M'HAMED BOUGARA-BOUMERDES



Faculté de Technologie

Département Ingénierie des Systèmes Electriques

Mémoire de Master

Présenté par

Mr. GAFSI Djamel Eddine

Filière : Automatique

Spécialité : Automatique et Informatique Industriel

**Etude et réalisation d'un système de commande des
installations électriques pour la Domotique**

Soutenu le 25 /11 /2020 devant le jury composé de:

MASSOUDI	NOURDDINE	MCA	UMBB	Président
NAFAA	FARES	MCB	UMBB	Examineur
AKROUM	HAMZA	MCB	UMBB	Rapporteur

Année Universitaire : 2019/2020

ملخص

في السنوات الاخيرة شهد تصميم الانظمة المدمجة تطورا ملحوظا و قد ادت هذه الثورة الكهربائية الهائلة الى تصميم انظمة تحكم معقدة بشكل متزايد مما يوفر بعض المزايا الواضحة فيما يتعلق بالتحكم عن بعد في الانظمة الكهربائية من الهاتف الذكي لدينا بالصوت.....الخ

الهدف الرئيسي من هذا العمل هو تحقيق نظام التحكم عن بعد على اساس نظام للتحكم في التركيبات الكهربائية للتشغيل الالي للمنزل مثل المحولات الصمامات اجهزة الكمبيوتر و التلفزيون....الخ

الكلمات المفتاحية التحكم عن بعد اردوينو اينو المنزل البلوتوث .

Summary

In recent years the design of embedded systems has undergone a remarkable evolution, this prodigious electric revolution has led to the design of increasingly complex control systems, offering some obvious advantages with regard to the control of remote systems, from our smartphone, by our voice... etc.

The main goal of this work is the realization of a remote control system (using Internet network) based on an Arduino UNO to control electrical installations for home automation, such as: switches, solenoid valves, PCs, TV.... etc.

Keywords: Remote Control, Arduino UNO, WLAN Ethernet Network, Home Automation.

Résumé

Durant ces dernières années la conception des systèmes embarqués a connu une évolution remarquable, cette révolution prodigieuse de l'électrique a conduit à la conception des systèmes de commande plus en plus complexes, offrant certaines avantages évidentes en ce qui concerne la commande des systèmes à distance, à partir de notre Smartphone, par notre voie ...etc.

Le but principal de ce travail est la réalisation d'un système de commande à distance (sous réseau Internet) à base d'un Arduino UNO pour commander des installations électriques pour la domotique, comme par exemple : des interrupteurs, des électrovannes, des PCs, Téléviseuretc.

Mots clés : Commande à distance, Arduino UNO, Réseau Ethernet WLAN, Domotique.

Introduction Général	1
Chapitre I : La domotique	
I.1.Introduction	3
I.2.Définitions de la domotique	3
I.3.Historique	4
I.4.Fonctionnement de la domotique	5
I.5.Différents domaines d’application de la domotique	6
I.5.1. Domotique pour le confort	6
I.5.2.Domotique pour l’énergie	7
I.5.3.Domotique pour la sécurité	7
I.5.4.Domotique pour la santé	9
I.6.Recherche de la domotique	9
I.7.Maison intelligente	10
I.8.L’internet des objets (IOT)	11
I.8.1.Objet connecté	11
I.9.Maison communicante	12
I.9.1. Domotique sans fil	13
I.9.2. Domotique à courant porteur CPL	14
I.9.3. Domotique câblée	15
I.10.Maison intelligente et la domotique	16
I.11.Avantages et inconvénients de la domotique	17
I.11.1.Avantages	17
I.11. 2.Inconvénients	18
II.12.Conclusion	18
Chapitre II : Le Dispositif Programmable Arduino	
II. Introduction	19
II.1.Domaine d'utilisation et ces applications	19
II.2. Avantages d'Arduino	20
II.3.Présentation de la carte	20
II.3.1.Microcontrôleur ATMEL ATMega328	21
II.3.2.Caractéristiques techniques de l’Arduino UNO	22
II.33.Les différents éléments de la carte de commande sont regroupés dans le	24
II.4.Capteurs	24
II.4.1.Présentation des capteurs	25
II.4.2.Communication	28
II.4.2.1.Module Arduino Bluetooth	28
II.4.2.2.Module shield Arduino Wifi	28
II.4.3.Capteur de Température et d'Humidité (DHT22)	29
II.4.3.1.Description du capteur	29
II.4.3.2.Principales caractéristiques de DHT22	30
II.4.4.Capteur de gaz (MQ2) II.5.4.1.Description du capteur	30
II.4.4.1.Principales caractéristiques de gaz/fumée MQ-2	30
II.4.5.Capteur de mouvement (PIR)	31
II.4.5.1.Description du capteur	31
II.4.5.2.Principales caractéristiques de mouvement PIR.	31
II.5.Module de Relais à 4 canaux	31
II.6.Servomoteurs	32
II.6.1.Fonctionnement de servomoteurs	33
II.7.Ventilateur	34
II.8.Description du module de connexion Wifi ESP8266	34

II.8.1.Mise sous tension	35
II.8.2.Principales caractéristiques d’ESP8266	35
II.9.Les afficheurs LCD	36
II.10.Potentiomètres	36
II.11.Module pour carte SD / Micro SD	37
II.11.1.Description	37
II.11.2.Caractéristiques	37
II.12.woofer	38
II.13. Clavier matriciel	39
II.13.1. Principe du clavier matriciel .	39
II.13.2. Détection des touches	39
II.13.3. Bronchement avec Arduino UNO	39
II.14.Présentation du logiciel	40
II.14.1.IDE Arduino	40
II.14.1.1.Les boutons	41
II.14.2.Langage Arduino	41
II.15.Fonctionnalité de base	44
II.15.1.Les entrées/sorties	44
II.16.Gestion du temps	46
II.17.Les interruptions	46
II.18.Description du programme	47
II.18.1. Les étapes de téléchargement du programme	49
II.19.Conclusion	49

Chapitre III : Développement d’une application

III.1.Introduction	50
III.2.le système Androïde	51
III.3.Application Android dans les Smartphones	51
III.3.1.Structure d’une application App Inventor	55
III.4.Explication et démarche	56
III.5.Présentation des fonctions de système	56
III.5.1.Fonction de point d’accès (Passerelle)	56
III.5.2.Fonction de la sécurité	56
III.5.3.Fonction d’ouverture du garage et la porte principale	56
III.5.4.Fonction d’éclairage	56
III.5.5.Fonction de l’acquisition de la température et d’humidité	56
III.5.6.Fonction de ventilation	56
III.5.7.Fonction de détection de gaz/fumée	57
III.6.Présentation de l’interfaçage Arduino-PC	57
III.7.Conclusion	58

Chapitre VI : Réalisation du système

VI.1.Introduction	59
VI.2.Réalisation pratique	59
VI.2.1. Montage sur plaque d’essai	59
VI..Les problèmes de la réalisation pratique	59
VI.4.Circuit Imprimé Final	60
VI.5.Réalisation du système	60
VI.5.Conclusion	61
	62

Figure I.1 : Illustre du confort	7
Figure I.2 : Détecteur de fumée	8
Figure I.3: Alarme piscine	8
Figure I.4 : Alarmes anti-intrusion	8
Figure I.5 : Schéma représentatif de fonctionnement général des équipements d'une Maison intelligente	10
Figure I.6 : La lampe DAL (premier objet connecté)	11
Figure I.7 : Thermostat Qivivo	12
Figure I.8 : Le lego de protocole radio Zwave	13
Figure I.9 : Le lego de HomeEasy	13
Figure I.10: Le lego de X2D (courant porteur)	13
Figure I.11: Le lego de l'io-Home	14
Figure I.12: Le lego de réseau Zigbee	14
Figure I.13: courant porteur CPL	14
Figure I.14 : Le Bus de terrain KNX	15
Figure I.15 : Kit Alarme sans fil Vidéo Domonial Standard HONEYWELL	16
Figure I.16 : Led ampoule sans fil	17
Figure I.17 : Moniteur d'énergie connecté	17
Figure II.1: Datasheet ATMega328	21
Figure II.2: La carte Arduino UNO	23
Figure II.3: Schéma simplifié de la carte Arduino UNO	23
Figure II.4 : Module shield Bluetooth	28
Figure II.5: Module shield wifi	28
Figure III.4: Image réelle du capteur de Température et d'Humidité DHT22	29
Figure III.5: Capteur de gaz/fumée MQ-2	30
Figure III.6: Capteur de mouvement PIR	31
Figure III.8 : Module de relais à 4 canaux	32
Figure III.9 : Ventilateur 5V	34
Figure III.10: Le module Wifi ESP8266 – ESP-D1	34
Figure II.11: Afficheur LCD.	37
Figure II.12 : Le module carte SD / Micro SD	37
Figure II.13. Woofer	38
Figure II.14 : Clavier matriciel (4*4) et son schéma de principe	39
Figure II.15 : Bronchement du clavier avec l'Arduino UNO	39
Figure II.16 : IDE Arduino.	40
Figure II.17: Un code minimal.	41
Figure II.18: Signal PWM.	45
Figure.III.1 : Image de l'application sur Mobile	50
Figure III.2 : Première interface de la création App Inventor	52
Figure III.3 : Schéma global du Screene 1	52
Figure III.4: En-tête d'éditeur de blocks App Inventor	53
Figure III.5 : Editeurs de blocks App Inventor	54
Figure III.6 : Echantillon d'un composant sous App Inventor	54
Figure III.7 : Schéma global du Scratch	55
Figure III.8 : Photo réelle de notre application	55
Figure III.9: Image réel de l'interface de l'application de commande.	57
Figure III.10: Fenêtre d'affichage des résultats de système sur le moniteur sérié.	58

Figure VI.1: Montage sur plaque d'essayer	59
Figure VI.2: Circuit sur ARES	60
Figure VI.3: Photos de la maquette réalisée (vue externe)	61
Figure VI.4: Photos d'intérieur de maquette	61
Figure III.5: Photos de la réalisation finale de maquette de maison	62

Tableau II.1: Différents éléments de la carte de commande	24
Tableau II.2: Représente différents shields qui connectée avec arduino	27
Tableau II.3: Servomoteurs utilisés dans noter projet.	33
Tableau II.4: Les types de base	42
Tableau II.4: Les structures de contrôle	43

Introduction générale

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

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

Elle assure différentes fonctions :

- **Fonction de confort**, en optimisant de l'éclairage de telle façon à multiplier les ambiances et d'adapter l'intensité de l'éclairage au besoin du moment, et ainsi en programmant les équipements électroménagers et multimédia.

- **Fonction d'économie d'énergie**, en mettant en veille les dispositifs de chauffage quand les habitants sont absents ou adapter automatiquement l'utilisation des ressources électriques en fonction des besoins des résidents afin de diminuer les gaspillages de ressources énergétiques suivi des consommations et optimisation des tarifs.

- **Fonction de sécurité** ; en outre, un autre but essentiel de l'application des technologies d'information aux maisons est la protection des individus. Cela est rendu possible par des systèmes capables d'anticiper des situations potentiellement dangereuses ou de réagir aux événements mettant en danger l'intégrité des personnes.

Notre mission porte sur la réalisation d'un contrôleur intelligent permettant de piloter la domotique à partir d'une application à distance en créant un Web Server entre les différents équipements du système domotique. Les objectifs de notre projet peuvent être divisés essentiellement selon les axes suivants qu'on va les aborder sous des chapitres. Dans ce travail nous essayons de réaliser un système de commande à distance (sous réseau Internet) à base d'un Arduino UNO pour commander des installations électriques pour la domotique (Une maison intelligente) comprenant plusieurs capteurs et actionneurs tels que: capteur de Température, Humidité, Capteur de Gaz/Fumée, Servomoteurs, relais ...etc.

La carte d'acquisition à base d'Arduino UNO est pour but de transférer les données de ces capteurs vers un smart phone et commander des actionneurs aussi, Les cartes Arduino sont conçues pour réaliser des prototypes et des maquettes de cartes électroniques pour l'informatique embarquée.

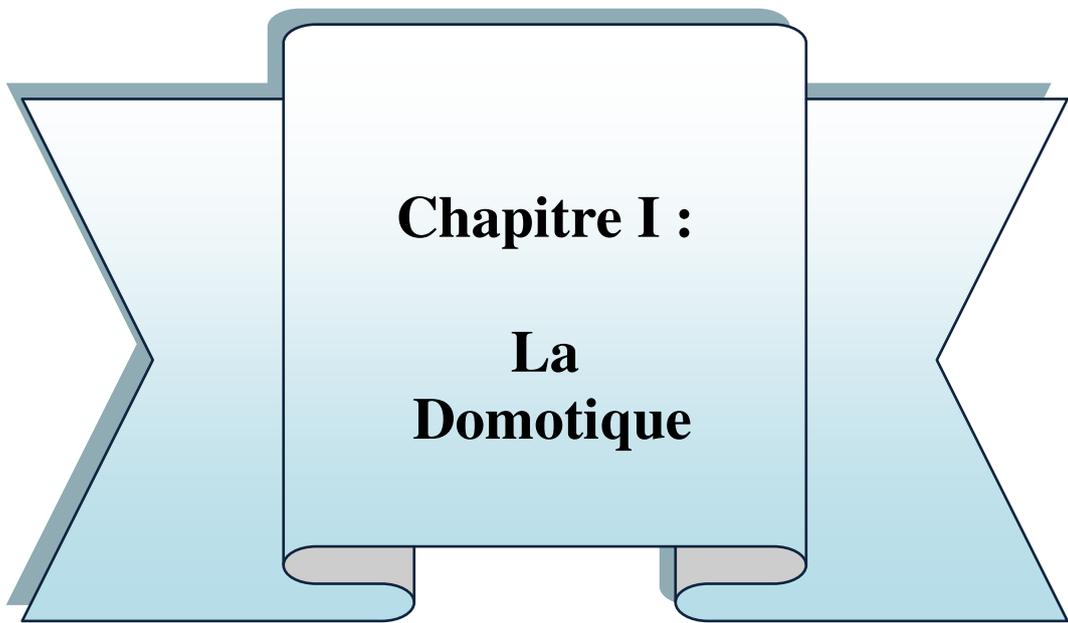
Deux principaux objectifs sont visés: Le premier objectif est de regrouper suffisamment d'informations sur une grande catégorie de cartes d'interfaçage à base de l'Arduino: son langage de programmation, sa construction, son principe de fonctionnement.

Le deuxième objectif consiste à réaliser une connexion sans fil entre la carte Arduino et smart phone à travers la carte WIFI série sans fil.

Notre projet expérimental est donc consiste alors à concevoir et à réaliser un système de maison connectée contrôlable via une application mobile multiplateforme en local ou par internet. Les données sont traitées dans l'unité de traitement et de commande à base de l'Arduino UNO afin de les transférer par un protocole de transfère des données sans fils WLAN (Wireless Local Area Network) ou LAN (Local Area Network) à travers de la carte WIFI série sans fil de type ESP8266 (ESP-D1) au réseau internet.

Ce mémoire est organisé en trois chapitres, organisés comme suit:

- Dans un premiers temps on commence par une introduction générale.
- Dans le premier chapitre, nous allons voir une présentation générale de la domotique ainsi ses secteurs d'application et les différents types de technologies utilisées.
- Le deuxième chapitre est dédiée à la description du projet ; son planification, la présentation du cahier des charges et les enjeux envisagés de notre système domotique.
- Le troisième chapitre est consacré à la description de la partie matérielle du projet, en identifiant le choix du cœur du système domotique. Par la suite on va réaliser une petite application App Inventor, qui va nous permettre de connecter notre système par wifi.
- Le quatrième chapitre est réservé pour l'étude et la réalisation de notre système; d'où on va présenter à la fin de ce chapitre les résultats obtenu après la réalisation de ce système de commande à distance des installations électriques.
- Enfin, on termine par une conclusion générale.



Chapitre I :

La Domotique

I.1.Introduction

Durant ces dernières années la technologie de fabrication des cartes de commande connue une évolution remarquable, cette révolution prodigieuse de la microélectronique a conduit à la fabrication des systèmes de commande de plus en plus complexes, offrant des avantages meilleurs tels que : la simplicité de la programmation, la vitesse d'exécution, les ports d'entrée/sortie... etc. [3]

Avec le développement des équipements électriques du logement, un nombre de plus en plus grand des systèmes électriques permettent de piloter de façon simple et confortable l'ensemble de ces équipements notamment l'éclairage, le climatiseur, les ouvrants, l'arrosage et le système alarme. On appelle ces systèmes « la domotique ».

Les progrès technologiques, notamment de l'informatique, de la télécommunication et de l'électronique ont permis le développement de systèmes de transmission, des commandes à distance et favorisé l'éclosion d'une offre abondante de nouveaux services pour les occupants des logements.

Dans ce chapitre, nous allons voir une présentation générale de la domotique ainsi ses secteurs d'application et les différents types des technologies utilisées.

I.2.Définitions de la domotique

Le mot domotique vient de domus qui signifie «domicile » et du suffixe –tique qui fait référence à la technique. la domotique est l'ensemble des techniques de l'électronique, de physique du bâtiment, d'automatisme, de l'informatique et des télécommunications utilisées dans le bâtiment, plus ou moins « interopérables » et permettant de centraliser le contrôle des différents systèmes et sous-systèmes de la maison et de l'entreprise (chauffage, volets roulants, porte de garage, portail d'entrée, prise électrique, etc.). la domotique vise à apporter des solutions techniques pour répondre aux besoins de confort (gestion d'énergie, optimisation de l'éclairage et du chauffage), de sécurité (alarme) et de communication (commandes à distance, signaux visuels ou sonores, etc.) que l'on peut trouver dans les maisons, les hôtels, les lieux publics...etc.[4]

A l'origine, la domotique avait donc pour but d'automatiser sa maison : ouverture et fermeture automatiques des volets, ouverture du portail électrique, gestion du

chauffage, gestion de l'éclairage, etc. Ainsi avant l'ère des Smartphones, il était par exemple possible d'activer son chauffage à distance en passant un coup de téléphone à sa maison, ou encore en lui envoyant un SMS. C'était tout à fait réalisable. Seulement une telle installation était relativement compliquée à mettre en place et, il faut bien l'avouer, coûteuse. Cette époque a malheureusement laissé des traces, puisque pour beaucoup encore aujourd'hui, domotique rime avec cher et compliqué. Pourtant, ce domaine a énormément évolué et il existe de nombreuses solutions simples à mettre en place et tout à fait abordables pour le grand public. [4]

La domotique a surtout elle-même évolué, si bien que le terme est quelque peu dépassé. La domotique servait à automatiser sa maison ; aujourd'hui on parle de domotique 2.0, ou de « maison intelligente », pour bien marquer l'évolution de ce monde. Les différents domaines de la maison ne se contentent plus d'être automatisés et pilotables, ils communiquent ensemble, permettant à la maison de réagir selon différents évènements. [4]

I.3.Historique

Les premiers travaux de domotique sont apparus dans les années 70 avec les problématiques énergétiques dues aux crises pétrolières. Ces crises marquent le début du développement de l'électronique pour les bâtiments. Au départ, la domotique contrôle seulement les prises, l'éclairage et les volets roulants grâce à une télécommande. Au fur et mesure, de nouveaux objets se mettent en réseau comme les thermostats et les alarmes.

Mais c'est véritablement à partir de la fin du 20e siècle, que la domotique va se démocratiser.

Deux raisons expliquent ce développement :

- l'arrivée de l'ordinateur et des technologies de communication dans la maison au début des années 1990; notamment, le déploiement d'Internet qui permet aux ordinateurs de communiquer entre eux.
- Le coût de l'énergie qui augmente suite aux deux crises pétrolières survenues dans les années 70. Désormais, de nouvelles normes forcent les constructeurs privilégier des bâtiments bien mieux isolés pour limiter leur utilisation chauffage.

La domotique intervient donc avec des appareils capables de communiquer entre eux pour surveiller et gérer cette énergie.

Depuis les années 2000, avec le développement des technologies sans fil comme le wifi ou le Bluetooth, la miniaturisation des composants électroniques, l'avènement des appareils mobiles, l'invasion des écrans tactiles et des télévisions connectées, les ingénieurs peuvent désormais proposer au public des produits - objets connectés ou systèmes domotiques – bien plus puissants et simples d'utilisation. [2]

I.4.Fonctionnement de la domotique

Aujourd'hui, les différents objets connectés de la maison ne se contentent plus d'être automatisés et pilotables ; ils interagissent ensemble pour notamment offrir aux habitants des maisons intelligentes un véritable confort d'usage, gagner en sécurité et optimiser la consommation énergétique des bâtiments. En communiquant avec l'habitat, il est possible de régler le chauffage par zones, de simuler à distance une présence, etc. En couplant l'installation avec une télécommande universelle ou avec un simple appui sur une touche sur son Smart phone, le pilotage s'effectue de n'importe où dans la maison, en fonction des besoins.

Concrètement, la domotique consiste à mettre en réseau différents appareils connectés dans une maison et à centraliser les commandes. Ces appareils sont déjà souvent existants : radiateurs, ventilation, éclairage, ... auxquels on ajoute des moyens de communiquer au sein de la maison. Chaque appareil est connecté avec d'autres via un appairage, qui consiste à associer deux ou plusieurs appareils entre eux. Cet appairage permet par exemple de dire à un interrupteur quel groupe de lampes il va devoir allumer. L'appairage peut se faire directement entre deux objets, ou via un boîtier domotique qui sert d'intermédiaire.

Chaque groupe d'appareils (éclairage, chauffage, volets roulants, ...) est pilotable via une ou plusieurs applications sur des appareils tels que les Smartphones, tablette, ordinateur ou télécommande. C'est cette application qui, à distance, permet de transmettre une demande (augmentation de la température, éclairage d'une pièce, démarrage de la télévision).

Les objets de l'habitat sont ainsi considérés comme intelligents. Ils sont équipés de capteurs tels que des capteurs de température et de présence pour un thermostat, qui

vont mesurer et détecter les habitudes des personnes vivant dans la maison. Les informations telles que les arrivées, sorties, temps passé dans une pièce des habitants sont toutes enregistrées et envoyées aux radiateurs pour faire adapter la température en fonction des scénarii programmés. Ainsi, on ne chauffe que quand c'est nécessaire.

Les objets peuvent communiquer entre eux par plusieurs moyens, dont les trois plus fréquents sont :

- L'envoi d'information par **un réseau filaire**, tel qu'un réseau informatique, un réseau téléphonique ou un câble dédié (un bus de données)
- Les informations peuvent aussi passer par **des câbles électriques**, ce qu'on appelle le courant porteur
- Ou alors le boîtier peut émettre **des ondes** comme le wifi, le Bluetooth ou les ondes radio.

Le câblage reste, à ce jour, la solution la plus fiable. Mais la domotique sans fil est plus simple à installer. On choisira l'un ou l'autre de ces moyens de communication en fonction des caractéristiques de l'habitat (ancien ou neuf).

Enfin, il est possible d'utiliser plusieurs types de communication en utilisant un boîtier domotique pour transmettre les informations d'un réseau à un autre. [5]

I.5. Différents domaines d'application de la domotique

I.5.1. Domotique pour le confort

Gestion de l'éclairage, gestion du chauffage, gestion des volets roulants, par simple action d'une commande, toutes ces tâches sont simplifiées grâce à la domotique. La domotique permet d'améliorer le confort d'usage. Grâce à une application installée sur son Smartphone, par exemple, les habitants d'une maison connectée peuvent décider de l'heure d'ouverture des volets, de la température des pièces selon l'heure de la journée. Des capteurs installés un peu partout dans la maison détectent la présence des individus et peuvent ainsi donner le signal pour allumer ou éteindre les lumières dans une pièce, activer la température optimale et même aller jusqu'à démarrer une musique d'ambiance dans le salon si les habitants l'ont choisie. [5]



Figure I.1 : illustre du confort

I.5.2.Domotique pour l'énergie

L'un des enjeux de la domotique est d'améliorer significativement l'efficacité énergétique des bâtiments. Les maisons dites « intelligentes » ou connectées sont équipées d'un ensemble de technologies innovantes permettant d'améliorer de manière globale leurs performances énergétiques sans perte de confort.

Parmi ces technologies, de nombreux automatismes : gestion des volets, de la ventilation, gestion des équipements de chauffage rendent les maisons réactives aux conditions extérieures (climat) et intérieures (usage), l'objectif final étant de réduire les dépenses quotidiennes d'énergie tout en préservant le confort des habitants. [5]

I.5.3.Domotique pour la sécurité

Un des domaines d'application de la domotique est la sécurité des biens et des personnes par des systèmes d'alarme qui préviennent d'une part des risques techniques (pannes ou dysfonctionnements des appareils) et d'autre part des éventuelles intrusions dans la maison (cambriolage).En général on trouve :

Alarmes techniques : Les alarmes techniques sont basées sur des capteurs capables de détecter différents incidents tels que des dégagements toxiques, incendie, fuite d'eau, fuite de gaz, etc. Ces différents capteurs sont raccordés à une centrale d'alarme. Les sécurités anti-noyade des piscines font également partie de ces systèmes d'alarme.

Ainsi que certains détecteurs de pannes sur les équipements domestiques (chaudière par exemple).



Figure I.2 : détecteur de fumée



Figure I.3: alarme piscine

Alarmes anti-intrusion : Ce sont en général des capteurs sur les portes (détection d'ouverture) ou dans les pièces détection de présence) qui sont reliés eux aussi à une centrale d'alarme. Ces capteurs peuvent être couplés avec un réseau de caméras numériques de surveillance. Lors d'une intrusion, un message d'alerte peut être envoyé par e-mail ou sur un téléphone portable. [6]



Figure I.4 : alarmes anti-intrusion

I.5.4.Domotique pour la santé

La domotique trouve aujourd'hui de nouvelles applications dans le domaine de la santé. En installant des systèmes domotiques dans les maisons des personnes en situation de handicap, atteintes de maladies neurodégénératives telles que la maladie d'Alzheimer ou encore des personnes âgées, il est possible de les aider dans leur quotidien en automatisant le plus possible des tâches considérées comme complexes.

Cela permet également à la personne de rester à son domicile plus longtemps et d'être suivie à distance. Par exemple, grâce à la domotique, on peut détecter quand une personne ne boit pas assez d'eau ou quand elle oublie de se nourrir. Si le comportement est considéré comme « préoccupant », il est alors possible d'alerter la famille ou les secours selon les scénarii programmés dans l'interface de commande. [5]

I.6.Recherche de la domotique

Depuis quelques années, avec la démocratisation des Smartphones, des tablettes et des objets communicants, les maisons connectées se sont développées à des coûts plus raisonnables que par le passé. Les recherches menées actuellement dans le domaine de la domotique consistent à proposer aux consommateurs des solutions domotiques meilleur marché, plus facile à installer et à développer des applications et interfaces de gestion plus ergonomiques et simples à utiliser, adaptées aux particularités des marchés locaux.

La gestion de l'énergie est un enjeu historique de la domotique. Profitant de l'essor des énergies renouvelables telles que le solaire et l'éolien, les technologies domotiques permettent une maîtrise des consommations tout en respectant le confort des usagers. L'intégration de ces énergies dépasse le cadre de la maison et s'étend également au domaine des transports. Par exemple, aujourd'hui, en cas de panne électrique, les véhicules électriques branchés peuvent prendre le relais et alimenter la maison.

Par ailleurs, de nombreuses recherches en domotique portent sur l'axe « santé ». En lien étroit avec les usagers (personnes âgées, personnes en situation de handicap) et les fournisseurs de produits et services, les ingénieurs et chercheurs développent des solutions fiables et sécurisées pour renforcer l'autonomie de ces personnes : domotique innovante, systèmes de surveillance et d'alerte adaptés (capteurs de chutes), aides

techniques connectées (robotique, automatique), solutions de gestion énergétique efficaces, solutions pour la mobilité et le lien social. L'objectif est de permettre à ces individus de rester le plus longtemps possible à domicile en toute sécurité. [5]

I.7.Maison intelligente

Pour faire simple, une maison intelligente est une maison dans laquelle plusieurs objets et appareils sont connectés à votre Smartphone. Du thermostat à l'éclairage, en passant par le système d'alarme ou le réfrigérateur, tous ces appareils intelligents (« smart devices ») communiquent entre eux par le biais d'une connexion internet sans fil. Une maison intelligente est une maison partiellement ou totalement automatisée. Dans une maison intelligente, plusieurs petites applications peuvent être connectées. La transformation de votre habitation en maison intelligente peut donc se faire progressivement. Le grand intérêt d'une maison connectée est que la communication se fait de manière bilatérale. À l'aide de votre Smartphone, vous gérez vos appareils à distance et ces derniers vous envoient des informations pratiques. [7]

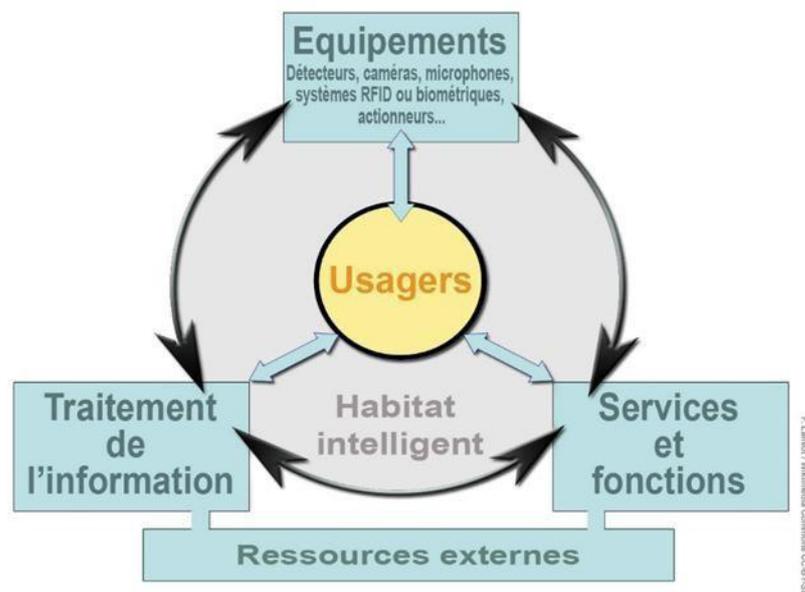


Figure I.5 : schéma représentatif de fonctionnement général des équipements d'un maison intelligente

I.8.L'internet des objets (IOT)

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. 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. [8]

I.8.1.Objet connecté

Un objet connecté est un objet électronique relié à internet et capable de communiquer des informations, apportant ainsi un service ou une valeur ajoutée. Le premier objet connecté était la lampe DAL (figure I.6), lancée en 2003 par RAFI Haladjan. 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. Les fonctions proposées aujourd'hui vont beaucoup plus loin que la simple annonce de la météo. Prenons l'exemple du thermostat Qivivo (figure I.7), 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 afin d'étudier les possibilités d'amélioration de son logement.

Les usages ont donc pu se développer et, aujourd'hui, les objets connectés sont partout. Ils sont particulièrement appropriés dans certains domaines. [8]



Figure I.6 : La lampe DAL (premier objet connecté).



Figure I.7 : Thermostat Qivivo.

I.9.Maison communicante

L'homme avait imaginé qu'il est impossible de se communiquer avec son habitat, le contrôler à distance. Maintenant, la communication tient une place de plus en plus importante dans le logement. Une installation domotique adaptée, avec les appareils de la maison montés en réseau, satisfait aux besoins et aux loisirs de chaque personne du foyer.

La centralisation des commandes est le corps du système domotique. Les appareils mis en réseau se reconnaissent et dialoguent entre eux, se déclenchant par simple appui sur une touche. Par le biais d'un interrupteur centralisé, les éclairages et volets motorisés peuvent être actionnés individuellement, par groupes de pièces ou simultanément.

En communiquant avec l'habitat, il est possible de régler le chauffage par zones, de simuler à distance une présence, etc. En couplant l'installation avec une télécommande universelle ou avec un simple appui sur une touche sur son Smart phone, le pilotage s'effectue de n'importe où dans la maison, en fonction des besoins.

Afin d'adapter la domotique à chaque logement et utilisation, plusieurs configurations sont à disposition :

- En domotique sans fil (Wifi, ondes radio,...).
- Par domotique CPL ou à courant porteur (appelé X10).
- Avec un câblage domotique bien pensé. [2]

I.9.1. Domotique sans fil

La domotique sans fil utilise plusieurs supports technologiques : les ondes radio ou RF (sur des fréquences en MHz) et l'infrarouge ou IR, qui a pour inconvénient de ne pas traverser les murs. Il est conseillé, pour une meilleure stabilité du système, de ne pas mixer le sans fil avec un autre type de technologie, le CPL par exemple. Cela peut nuire à l'installation et à la qualité de la communication entre les équipements. [2]

Les ondes radio sont employées par de multiples protocoles comme le X10 RF, le HomeEasy, le X2D, le Zigbee, le Zwave, ou encore le Bluetooth.

Les principales fréquences utilisées dans la domotique sont le 433 MHz et le 868MHz. On trouve parmi les protocoles sans fil :

- Le protocole radio Zwave, fréquence 868,42 MHz en Europe, répercute un ordre reçu vers les modules voisins. La portée du contrôleur Zwave peut équiper toute la maison sans risquer de problèmes de transmission.



Figure I.8 : Le protocole radio Zwave.

- Le HomeEasy, lui, utilise la fréquence 433 MHz qui est règlementée par l'UIT (Union internationale des télécommunications).



Figure I.9 : Le HomeEasy.

- Le X2D est mixte (courant porteur ou radio 868 MHz) convient à la domotique de sécurité et la domotique du chauffage.



Figure I.10: Le X2D (courant porteur).

- L'io-Home Control utilise les fréquences allant de 868 MHz à 870MHz, il possède un véritable retour d'informations grâce à son protocole bidirectionnel. Cette technologie est ouverte à différents fabricants leaders dans l'habitat.



Figure I.11: L'io-Home.

- Le réseau Zigbee, basé sur le standard 802.15.4, ratifié par l'IEEE (Institute of Electrical and Electronics Engineers), a de plus en plus de fidèles. Il fonctionne avec des piles très longues durées d'autonomie, sur 866 MHz (bande libre en Europe) et 915 MHz (aux États-Unis).



Figure I.12: Le réseau Zigbee.

I.9.2.Domotique à courant porteur CPL

L'utilisation de la domotique à courant porteur revient à transformer son habitat en maison communicante par le biais d'une installation domotique; c'est-à-dire que l'on utilise le réseau électrique déjà existant.

La domotique CPL est aussi connue sous l'acronyme de X10, qui est un protocole de communication et de contrôle de plusieurs appareils domotiques.



Figure I.13: courant porteur CPL

Les CPL, c'est la possibilité de faire passer de l'information numérique (voix, donnée, image) sur le réseau électrique ordinaire. Ils s'avèrent très utiles en cas de rénovation.

L'intérêt de cette technologie porte sur l'utilisation d'un réseau filaire structuré déjà existant et parfaitement distribué dans toute la maison ou le bâtiment : le réseau électrique et ses points d'accès constitués par les prises électriques.

Toutefois, la fiabilité de la domotique CPL est contestable. Cette technologie peut parasiter le réseau et perturber les autres transmissions. De plus, cet équipement est encore coûteux. À performances équivalentes, il est en effet plus cher que le sans fil. Enfin, ce système est aussi moins rapide, et il n'a pas de mobilité par construction.

I.9.3.Domotique câblée

Certains professionnels ne sont pas favorables, au sein d'une installation domotique, aux approches sans fil ou CPL. Ils leur préfèrent une domotique par câbles. Le pré-câblage doit être souple et évolutif, car la technologie ne cesse d'évoluer.

Il faut ainsi prévoir un local technique, le «local de répartition», qui centralise les points d'arrivée de toutes les liaisons externes (électricité, téléphone, Internet, télévision, fibre optique ...).

Dans les logements, le Bus de terrain KNX est une excellente solution domotique. Ce Bus est constitué d'un câble fait de conducteurs torsadés par paires (deux au minimum) alimenté en très basse tension (courant faible).



Figure I.13: Le Bus de terrain KNX

Le réseau a pour but d'empêcher les interférences électriques reprochées au CPL. Cependant, tout repose sur la qualité des câbles choisis.

Trois types de câbles sont fréquemment rencontrés, le câble UTP, le câble STP et le câble FTP. Les meilleurs câbles sont blindés ou écrantés, de type STP ou FTP.

Il est recommandable de choisir un réseau électrique, car c'est le plus simple à installer (et le mieux connu par les artisans et les architectes). Il doit respecter toute fois la norme NFC15-100. Il est aussi préférable d'installer un panneau de brassage équipé de prises RJ45.

Ensuite, il faut prévoir un onduleur pour les équipements du réseau (modem ADSL, routeur, switches) et les équipements de la domotique de sécurité. [5]

I.10. Maison intelligente et la domotique

Bien que foncièrement liés, les concepts de maison intelligente et de domotique présentent des différences notoires. Un système domotique constitue un ensemble intégré reprenant tous les composants de votre habitation. Un tel système exige une installation spécifique effectuée par un installateur qualifié. C'est là que se situe la principale différence avec les nouvelles applications de la maison intelligente. Celles-ci sont faciles à utiliser et ne nécessitent aucune installation sophistiquée. Il suffit de posséder un Smartphone, une connexion internet et, bien entendu, une... maison !

Aujourd'hui, tous les appareils et tous les éléments d'habitation sont proposés en version connectée. Exemples : [5]

- **Sécurité et contrôle** : systèmes d'alarme intelligents, détecteurs de mouvements et caméras connectés, volets roulants commandés à partir de votre Smartphone, détecteurs de fumée et d'incendie qui envoient un message d'alerte sur votre Smartphone...



Figure I.14 : Kit Alarme sans fil Vidéo Domonial Standard HONEYWELL

Appareils électroménagers : aspirateurs-robots, lave-linge, réfrigérateurs intelligents...



Figure I.15 : Aspirateur robot Wi-Fi Roomba 695 d'iRobot

- **Éclairage** : lampes intelligentes qui, via votre Smartphone, s'allument et s'éteignent automatiquement lorsque vous pénétrez dans votre habitation ou que vous la quittez.



Figure I.16: Led ampoule sans fil

- **Énergie** : des moniteurs d'énergie permettant de contrôler à tout moment et à distance votre consommation énergétique, ainsi que celle de vos appareils électroménagers



Figure I.17 : moniteur d'énergie connecté

I.11. Avantages et inconvénients de la domotique

I.11. 1. Avantages :

- Le principal avantage de la domotique est l'amélioration du quotidien au sein de la maison, du point de vue du confort, de la sécurité et de la gestion de l'énergie.
- Ce type d'équipement vous simplifie la vie et optimise votre confort en adaptant votre maison à différents scénarios de la vie quotidienne.
- Il vous permet notamment d'éteindre tous vos appareils électriques et de mettre l'alarme quand vous quittez votre domicile, de régler des ambiances lumineuses

(ambiance lecture, ambiance relaxation avec lumières tamisées), de vous réveiller dans un habitat chauffé où le café est déjà prêt, d'enclencher automatiquement l'arrosage ou l'ouverture des volets chaque matin.

- La domotique permet aussi de réaliser des économies d'énergie grâce à la gestion automatique du chauffage, de la climatisation et de l'éclairage et à la programmation des appareils électroménagers en heures creuses.
- Elle a pour avantage d'améliorer la sécurité grâce à des alarmes, des systèmes d'ouverture automatique de la porte (reconnaissance vocale, carte magnétique...)
- En cas de tentative d'intrusion dans la maison, un appel téléphonique automatique peut contacter le propriétaire ou une entreprise de sécurité.
- Enfin, ces différentes technologies constituent une aide précieuse pour les personnes dépendantes et handicapées. [6]

I.11.2. Inconvénients

Le principal inconvénient est le prix d'achat et d'installation. Le prix est beaucoup plus élevé mais vos factures d'énergie baisseront. Il faut donc le prendre en compte dans le budget initial. Le deuxième inconvénient est le verrouillage qu'offrent certaines marques dans leurs produits ne permettant pas d'avoir un logiciel ouvert. [6]

II.12. Conclusion

Dans ce chapitre nous avons commencé par une présentation générale de la domotique ainsi ses secteurs d'application et les différents types de technologies utilisées. Ensuite on a mis la lumière sur la maison intelligente et ses avantages/inconvénients.

Dans le chapitre qui suit on va mettre le point sur la présentation de la carte Arduino UNO, ses caractéristiques et présentation de quelques shields.

Sommaire

I.1.Introduction.....	3
I.2.Définitions de la domotique.....	3
I.3.Historique	4
I.4.Fonctionnement de la domotique	5
I.5.Différents domaines d’application de la domotique	6
I.5.1. Domotique pour le confort.....	6
I.5.2.Domotique pour l’énergie.....	7
I.5.3.Domotique pour la sécurité.....	7
I.5.4.Domotique pour la santé.....	9
I.6.Recherche de la domotique	9
I.7.Maison intelligente	10
I.8.L’internet des objets (IOT)	11
I.8.1.Objet connecté.....	11
I.9.Maison communicante.....	12
I.9.1. Domotique sans fil.....	13
I.9.2. Domotique à courant porteur CPL.....	14
I.9.3. Domotique câblée.....	15
I.10.Maison intelligente et la domotique	16
I.11.Avantages et inconvénients de la domotique	17
I.11. 1.Avantages.....	17
I.11. 2.Inconvénients.....	18
II.12.Conclusion	18



Chapitre II :

**Le Dispositif
Programmable
Arduino.**

II. Introduction

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 coût de la réalisation, mais aussi la charge de travail à la conception d'une carte électronique

Arduino est un circuit imprimé en matériel libre sur lequel se trouve un microcontrôleur qui peut être programmé pour analyser et produire des signaux électriques, de manière à effectuer des tâches très diverses comme la domotique (le contrôle des appareils domestiques - éclairage, chauffage...), le pilotage d'un robot [1].

Le langage Arduino se distingue des langages utilisés dans l'industrie de l'informatique embarquée de par sa simplicité. En effet, beaucoup de bibliothèques et de fonctionnalités de base occulte certains aspects de la programmation de logiciel embarquée afin de gagner en simplicité. Cela en fait un langage parfait pour réaliser des prototypes ou des petites applications.

Les possibilités des cartes Arduino sont énormes, un grand nombre d'applications ont déjà été réalisées et testées par bon nombre d'internautes [9].

Dans ce chapitre nous allons voir une présentation générale du système Arduino UNO, ses caractéristiques et présentation des quelques shields et on mettra la lumière sur le logiciel IDE Arduino.

II.1. Domaine d'utilisation et ces applications

Le système Arduino nous permet de réaliser un grand nombre de choses, qui ont une application dans tous les domaines, l'étendue de l'utilisation de l'Arduino est gigantesque. Pour vous donner quelques exemples, vous pouvez :

- Électronique industrielle et embarquée.
- Contrôler les appareils domestiques.
- Fabriquer votre propre robot.
- Faire un jeu de lumières.
- Communiquer avec l'ordinateur.

- Télécommander un appareil mobile (modélisme).
- Physical computing: Au sens large, construire des systèmes physiques interactifs qui utilisent des logiciels et du matériel pouvant s'interfacer avec des capteurs et des actionneurs.
- Hacker, Prototypage, Education [10].

II.2. Avantages d'Arduino

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 :

- Simplifie grandement les schémas électroniques.
- Diminué le coût de la réalisation.
- La charge de travail à la conception d'une carte électronique.
- Environnement de programmation clair et simple.
- Multiplateforme : tourne sous Windows, Macintosh et Linux.
- Nombreuses bibliothèques disponibles avec diverses fonctions implémentées.
- Logiciel et matériel open source et extensible.
- Nombreux conseils, tutoriaux et exemples en ligne (forums, site perso, etc.).
- Existence de « shield » (boucliers en français).

II.3. Présentation de la carte

Qu'est ce qu'un microcontrôleur ?

Les cartes Arduino font partie de la famille des microcontrôleurs. Un microcontrôleur est une petite unité de calcul accompagné de mémoire, de ports d'entrée/sortie et de périphériques permettant d'interagir avec son environnement. Parmi les périphériques, on recense généralement des Timers, des convertisseurs analogique-numérique, des liaisons Séries, ... etc. On peut comparer un micro contrôleur à un ordinateur classique, mais avec un autre système d'exploitation et avec une puissance de calcul considérablement plus faible.

Les microcontrôleurs sont inévitables dans les domaines de l'informatique embarquée, de l'automatique et de l'informatique industrielle. Ils permettent de réduire

le nombre de composant et de simplifier la création de cartes électroniques logiques [11].

II.3.1. Microcontrôleur ATMEL ATmega328

Le microcontrôleur utilise sur la carte Arduino UNO est un microcontrôleur ATmega328. C'est un microcontrôleur ATMEL de la famille AVR 8bits.

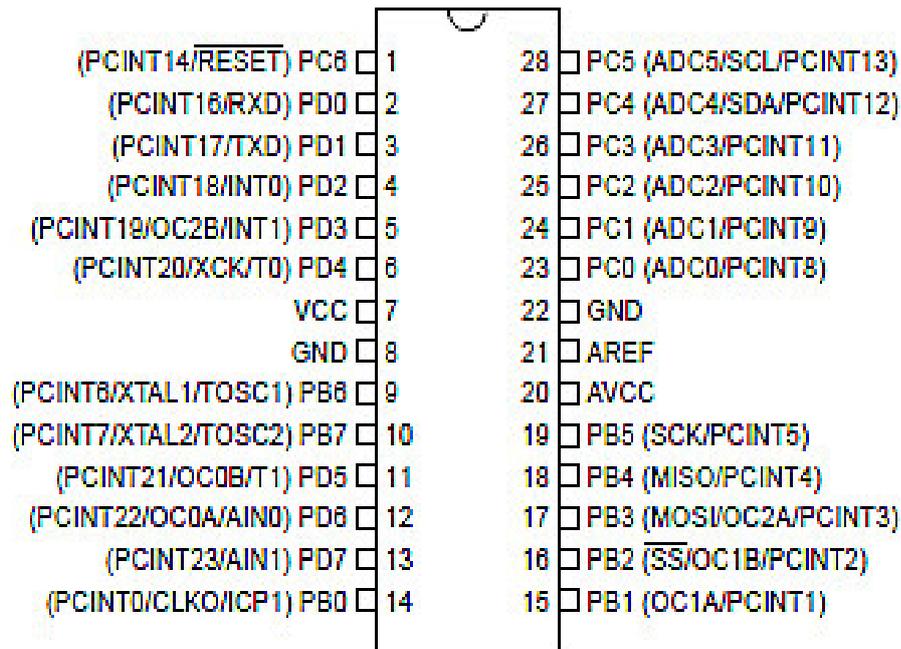


Figure II.1: Datasheet ATmega328.

Les principales caractéristiques d'ATmega328 sont :

- **FLASH** = mémoire programme de 32Ko.
- **SRAM** = données (volatiles) 2Ko.
- **EEPROM** = données (non volatiles) 1Ko.
- **Digital I/O (entrées-sorties Tout Ou Rien)** = 3 ports Port B, Port C, Port D (soit 23 broches en tout I/O).
- **Timers/Counters** : Timer0 et Timer2 (comptage 8 bits), Timer1 (comptage 16 bits) Chaque timer peut être utilisé pour générer deux signaux PWM. (6 broches OCxA/OCxB).
- **Plusieurs broches multi-fonctions:** certaines broches peuvent avoir plusieurs fonctions différentes choisies par programmation.
- **PWM** = 6 broches OC0A(PD6), OC0B(PD5), OC1A(PB1), OC1B(PB3), OC2A(PB3), OC2B(PD3)

- **Analog to Digital Converter** (resolution 10 bits) = 6 entrees multiplexes ADC0(PC0) à ADC5(PC5).
- **Gestion bus I2C** (TWI Two Wire Interface) = le bus est exploite via les broches SDA(PC5)/SCL(PC4).
- **Port série (USART)** = émission/réception série via les broches TXD(PD1)/RXD(PD0)
- **Comparateur Analogique** = broches AIN0(PD6) et AIN1 (PD7) peut déclencher interruption
- **Watch dog Trimer programmable.**
- **Gestion d'interruptions (24 sources possibles (cf. interruptif vecteurs))** : en résumé
 - Interruptions liées aux entrées INT0 (PD2) et INT1 (PD3).
 - Interruptions sur changement d'état des broches PCINT0 a PCINT23.
 - Interruptions liées aux Timers 0, 1 et 2 (plusieurs causes configurables).
 - Interruption liée au comparateur analogique.
 - Interruption de fin de conversion ADC.
 - Interruptions du port série USART.
 - Interruption du bus TWI (I2C). [12]

II.3.2.Caractéristiques techniques de l'Arduino UNO

Un des modèles les plus répandu de carte Arduino est l'Arduino UNO (voir Fig. II.2). C'est la première version stable de carte Arduino. Elle possède toutes les fonctionnalités d'un microcontrôleur classique en plus de sa simplicité d'utilisation. Elle utilise une puce ATmega328P (1) cadencée à 16 Mhz. Elle possède 32 ko de mémoire flash destinée à recevoir le programme, 2 ko de SRAM (mémoire vive) et 1 ko d'EEPROM (mémoire morte destinée aux données). Elle offre 14 pins (broches) d'entrée/sortie numérique (données acceptée 0 ou 1) (2) dont 6 pouvant générer des PWM (Pulse Width Modulation, détaillé plus tard). Elle permet aussi de mesurer des grandeurs analogiques grâce à ces 6 entrées analogiques (3). Chaque broche est capable de délivrer un courant de 40 mA pour une tension de 5 V.

La carte Arduino peut aussi s'alimenter et communiquer avec un ordinateur grâce à son port USB (4). On peut aussi l'alimenter avec une alimentation comprise en 7V et 12V grâce à sa connecteur Power Jack (5) [11].

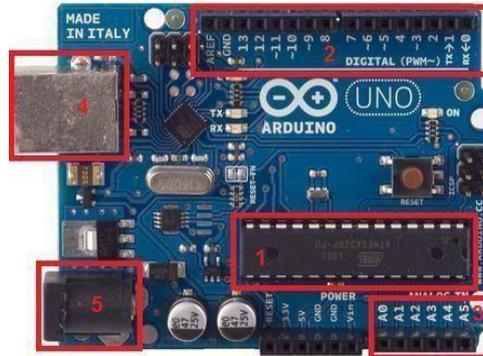


Figure II.2: La carte Arduino UNO.

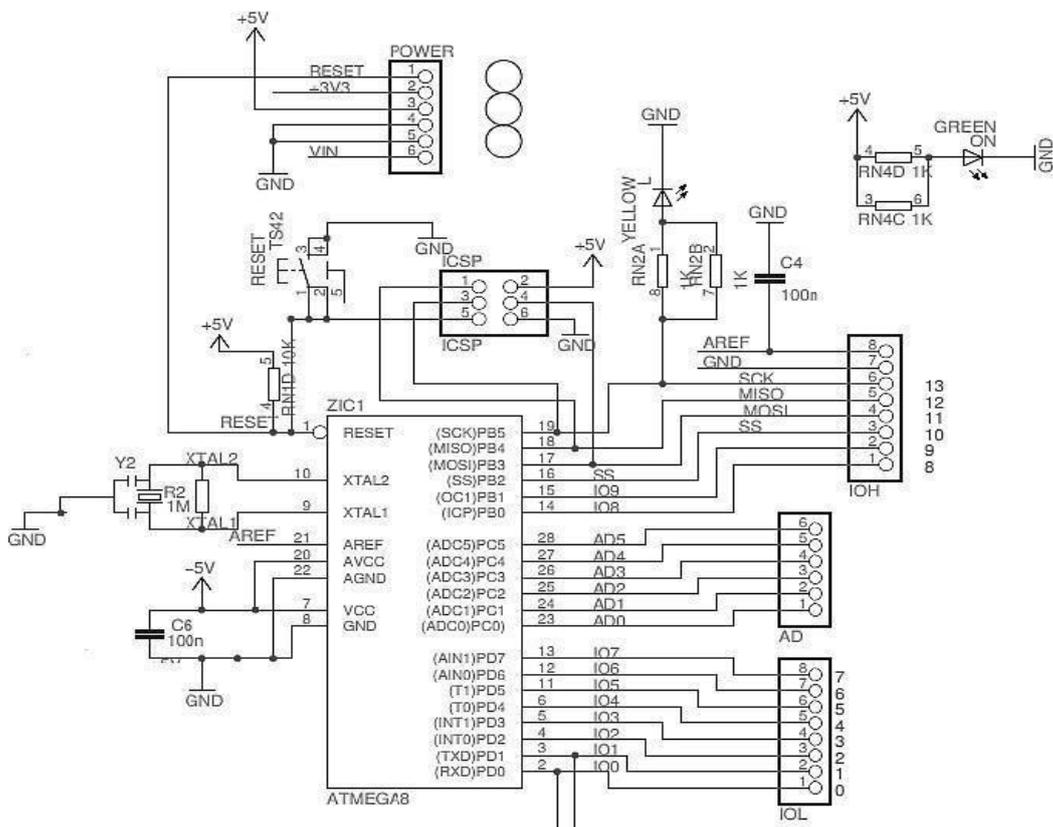


Figure II.3: Schéma simplifié de la carte Arduino UNO.

II.3.3. Les différents éléments de la carte de commande sont regroupés dans le Tableau

Tension de fonctionnement	5 V
Tension d'alimentation (recommandée)	7-16 V
Tension d'alimentation (limites)	6-20 V
Broches E/S numériques	14(dont 6 disposent d'une sortie PWM).
Broches d'entrées analogiques	6 (utilisables en broches E/S numériques).
Intensité maxi disponible par broche E/S (5 V)	40 mA (attention : 200 mA cumulé pour l'ensemble des broches E/S).
Intensité maxi disponible pour la sortie 3.3 V	50 mA
Intensité maxi disponible pour la sortie 5 V	Fonction de l'alimentation utilisée - 500 mA max si port USB utilisée seul

Mémoire programme flash	32 KB (ATmega328) dont 0.5 KB sont utilisés par le boot loader (programme de base préprogrammé conçu pour établir la communication entre l' ATmega et le logiciel Arduino).
Mémoire SRAM (mémoire volatile)	2 KB (ATmega).
Mémoire EEPROM (mémoire non volatile)	1 KB (ATmega).
Vitesse d'horloge	16 Mhz

Tableau II.1: Différents éléments de la carte de commande.

II.4. Capteurs

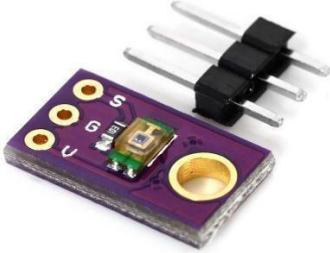
Un capteur est un dispositif ayant pour tâche de transformer une mesure physique observée en une mesure généralement électrique qui sera à son tour traduite en une donnée binaire exploitable et compréhensible par un système d'information.

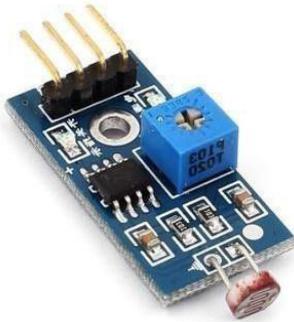
II.4.1. Présentation des capteurs

Pour la plupart des projets, il est souvent nécessaire d'ajouter des fonctionnalités aux cartes Arduino. Plutôt que d'ajouter soit même des composants extérieurs (sur une platine d'essai, circuit imprimé, etc.), il est possible d'ajouter des shields.

Un shield est une carte que l'on connecte directement sur la carte Arduino qui a pour but d'ajouter des composants sur la carte. Ces shields viennent généralement avec une librairie permettant de les contrôler. On retrouve par exemple, des shields Ethernet, de contrôle de moteur, lecteur de carte SD, etc.

Le principal avantage de ces shields est leurs simplicités d'utilisation. Il suffit des les emboîter sur la carte Arduino pour les connecter, les circuits électronique et les logiciels sont déjà faits et on peut en empiler plusieurs. C'est un atout majeur pour ces cartes pour pouvoir tester facilement de nouvelles fonctionnalités. Cependant il faut bien garder à l'esprit que les shields ont un prix. Suivant les composants qu'ils apportent, leurs prix peuvent aller de 2 à 100\$.

Nom de capteurs	Image de capteurs	Principales caractéristiques
capteur de Pression BMP180		<ul style="list-style-type: none"> ✓ V_{in}: 3 V à 5 V. ✓ Plage de détection de pression: 30-110 KPa (9000 m à -500 m au-dessus du niveau de la mer). ✓ Jusqu'à 0.003 KPa / 0.25 m de résolution. ✓ plage de mesure de -40 à + 85 ° C de, +2 ° C de précision de température [13].
capteur d'ensoleillement TEMT6000		<ul style="list-style-type: none"> ✓ Alimentation: 3 à 5 V ✓ Adapté à la réactivité des yeux humains. ✓ Température de service: -40°C à +85°C ✓ Composant sans plomb [13].

<p>Capteur de distance a ultrason SRF05</p>		<ul style="list-style-type: none"> ✓ tension d'entrée : 5v dc ✓ Courant de repos: < 2ma ✓ niveau (élevé) de sortie: 5v ✓ niveau de sortie (de bas): 0v ✓ angle induction: < 15 ° ✓ distance de détection: 2cm - 450cm ✓ Précision: jusqu'à 0.3cm ✓ taches aveugles: 2cm. [14]
<p>Photosensible (module de capteur de luminosité résistance)</p>		<ul style="list-style-type: none"> ✓ en utilisant capteur photosensible résistance sensible type; ✓ tension de Fonctionnement: 3.5 V, 5 V; ✓ forme de sortie: numérique de commutation spectacle (0 et 1); ✓ trou de vis fixe, installation facile; ✓ petit PCB Taille: 3.2 cm * 1.7 cm; ✓ la sortie du comparateur, le signal propre, bonne vague, la capacité de conduire est forte, pour plus que 15 mA; ✓ avec réglable potentiomètre pour régler la luminosité de la lumière. [15]
<p>Détecteur de niveau d'eau Grove 101020018</p>		<ul style="list-style-type: none"> ✓ Interface: compatible Grove ✓ Signal: digital ou analogique ✓ Faible consommation ✓ Dimensions: 40 x 20 x 12 mm ✓ Température de service: 0°C à +40°C ✓ Connectique non compatible avec Tinker Kit ✓ Référence Speede studio: 101020018 (remplace SEN11304P). [16]

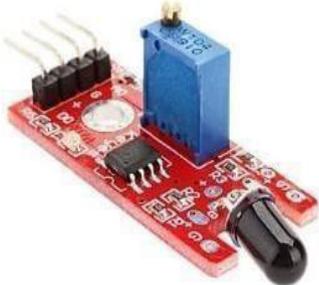
<p>capteur de pluie double face</p>		<ul style="list-style-type: none"> ✓ Tension : 3,3V-5V ✓ Dimension du capteur : ~ 40 x 54 mm ✓ Pistes sur capteur : Double face. Les deux faces du capteur sont actives. ✓ Potentiomètre de sensibilité et 2 LED de contrôle intégré au module ✓ Module livré avec deux câbles Femelle / Femelle. [13]
<p>capteur infrarouge d'évitement d'obstacles</p>		<ul style="list-style-type: none"> ✓ VCC: alimentation 3v-12v (pouvant se connecter directement à 5V ou 3,3V microcontrôleur) ✓ GND: Masse ✓ OUT: interface de sortie numérique de la carte (0 et 1). [13]
<p>capteur de flamme</p>		<ul style="list-style-type: none"> ✓ haute sensibilité du récepteur IR ✓ extrêmement sensibles aux ondes entre 760-1100nm ✓ voyant d'alimentation ✓ voyant comparateur de sortie ✓ sortie analogique quantité ✓ seuil de renversement du niveau de sortie électrique ✓ seuil réglé par potentiomètre ✓ Gamme de détection d'angle: environ 60 degrés ✓ Alimentation: 0-15 V DC. [13]

Tableau II.2: Représente différents shields qui connectent avec arduino.

II.4.2.Communication

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

II.4.2.1.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.



Figure II.4 : Module shield Bluetooth.

II.4.2.2.Module shield Arduino Wifi

Le module Shield Arduino Wifi permet de connecter une carte Arduino à un réseau internet sans fil Wifi.



Figure II.5: Module shield wifi.

II.4.3. Capteur de Température et d'Humidité (DHT22)

II.4.3.1. Description du capteur

Le DHT22 est un capteur de température et d'humidité numérique basique à faible coût. Il utilise un capteur d'humidité capacitif et une thermistance pour mesurer l'air environnant et émet un signal numérique sur la broche de données (aucune broche d'entrée analogique n'est requise). Il est assez simple à utiliser, mais nécessite un temps minutieux pour saisir les données. Le seul inconvénient réel de ce capteur est que vous ne pouvez obtenir de nouvelles données qu'une fois toutes les 2 secondes, alors lorsqu'on déclare la bibliothèque de ce capteur dans le programme, la lecture de ce capteur peut durer jusqu'à 2 secondes.



. **Figure II.6:** Capteur de Température et d'Humidité DHT22.

Il suffit de connecter la première broche à gauche à l'alimentation de 3 à 5 V, la deuxième broche doit brancher à la carte Arduino, juste au pin 2 numérique. La troisième broche est montée à la masse GND [13].

II.4.3.2. Principales caractéristiques de DHT22

- ❖ À bas prix.
- ❖ Puissance et E / S de 3 à 5 V.
- ❖ 2.5 mA max courant d'utilisation pendant la conversion (lors de la demande de données).
- ❖ Bon pour 0-100% de lectures d'humidité avec 2-5% de précision.
- ❖ Bon pour les lectures de température de -40 à 80 ° ± Précision ± 0,5 ° C.
- ❖ Pas plus de 0,5 Hz de fréquence d'échantillonnage (une fois toutes les 2 secondes).
- ❖ Taille du corps 27 mm × 59 mm × 13.5 mm (1.05 "× 2.32" × 0.53 ").
- ❖ Poids (juste le DHT22): 2,4 g [15].

II.4.4. Capteur de gaz (MQ2)

II.4.4.1. Description du capteur

Le capteur de gaz MQ2 est utilisé pour la détection des fuites de gaz pour les équipements des marchés de grandes consommations et industriel. Ce capteur est conçu pour détecter le LPG, i-butane, propane, méthane, alcool, hydrogène et la fumée. Il a une grande sensibilité et un temps de réponse rapide. Sa sensibilité peut d'ailleurs être ajustée par potentiomètre [19].



Figure II.7: Capteur de gaz/fumée MQ-2.

II.4.4.2. Principales caractéristiques de gaz/fumée MQ-2

- ❖ Alimentation: 5V
- ❖ Type d'Interface: Analogique
- ❖ Connectique: 1-Sortie 2-GND 3-VCC
- ❖ Large panel de détection
- ❖ Réponse rapide et haute sensibilité
- ❖ Circuit de Contrôle Simple
- ❖ Système stable à longue durée de vie
- ❖ Dimensions: 40x20mm

II.4.5. Capteur de mouvement (PIR)

II.4.5.1. Description du capteur

Les capteurs PIR (« passive infra éd », « pyroélectriques » ou « à mouvement infrarouge »), sont utilisés comme des capteurs de mouvement. Ils permettent de déterminer si un être humain (et, dans le cas du capteur PIR d'Ada fruit, un animal) est entrée ou sorti du champ de détection du module [20].



Figure II.8.: Capteur de mouvement PIR.

II.4.5.2. Principales caractéristiques de mouvement PIR.

- ❖ Alimentation : 5 V
- ❖ Signal de sortie numérique : 3,3V
- ❖ Portée : 7m
- ❖ Cône de détection : 120°
- ❖ Sensibilité et délai de réponse (2-4 s) ajustables
- ❖ Longueur : 24,03 mm
- ❖ Profondeur : 32,34 mm
- ❖ Distance des trous de vissage : 28 mm
- ❖ Diamètre des trous de vissage : 2 mm
- ❖ Hauteur (avec lentille) : 24,66 mm
- ❖ Poids : 5,87 g

II.5. Module de Relais à 4 canaux

Il s'agit d'une carte d'interface de relais, qui peuvent être contrôlé directement par un large éventail de microcontrôleurs comme Arduino, AVR, PIC, ARM, API, etc.

Ce module de relais est bas actif 5V. Il est également capable de contrôler

appareils divers et autres équipements avec le grand courant. Cette interface standard peut être connectée directement avec les microcontrôleurs. Le voyant rouge qui indique l'état de travail est propice à l'utilisation de sécurité. Le module de relais est largement utilisé pour tout contrôle MCU, le secteur industriel, contrôle PLC, contrôle de la maison intelligente [13].



Figure II.9 : Module de relais à 4 canaux.

II.6.Servomoteurs

Pour motoriser nos volets des fenêtres ainsi de la porte principale et le garage, on a pensé à utiliser des servomoteurs vue de leur souplesse, simplicité de commande et de leur couple acceptable.

Les servomoteurs servent en principe à actionner les parties mobiles d'un système. Ils sont prévus pour être commandés facilement en position ou en vitesse. En effet, ils sont équipés d'un système d'asservissement basé sur un potentiomètre rotatif qui sert de capteur de rotation [22].

C'est un ensemble mécanique et électronique comprenant :

- Un moteur à courant continu de petite taille.
- Un réducteur en sortie de ce moteur diminuant la vitesse mais augmentant le couple.
- Un potentiomètre (faisant fonction de diviseur résistif) qui génère une tension variable, proportionnelle à l'angle de l'axe de sortie.
- Un dispositif électronique d'asservissement.
- Un axe dépassant hors du boîtier avec différents bras ou roues de fixation.

Ils sont faciles à utiliser car ils ne nécessitent que :

- Une source de tension continue.
- Une sortie PWM (Pulse Width Modulation) du microcontrôleur de notre robot.

II.6.1. Fonctionnement des servomoteurs

Les servomoteurs sont commandés par l'intermédiaire d'un câble électrique à 3 fils qui permettent d'alimenter le moteur et de lui transmettre des ordres de positions sous forme d'un signal codé en largeur d'impulsion plus communément appelés PWM (Pulse Width Modulation ou Modulation de Largeur d'Impulsion) ou RCO (Rapport Cyclique d'Ouverture). Cela signifie que c'est la durée des impulsions qui détermine l'angle absolu de l'axe de sortie et donc la position du bras de commande du servomoteur. Le signal est répété périodiquement, en général toutes les 20 millisecondes, ce qui permet à l'électronique de contrôler et de corriger continuellement la position angulaire de l'axe de sortie, cette dernière étant mesurée par le potentiomètre.

Servomoteur à rotation continue	Servomoteur à rotation angulaire (Micro-Servo)
	
<p>Caractéristiques :</p> <p>Alimentation : 4,8 à 6V</p> <p>Angle de rotation : 360°</p> <p>Couple : 3,3 à 4,8 kg.cm</p> <p>Vitesse : 60 à 70 tr/min</p> <p>Dimensions : 41 x 20 x 40 mm</p> <p>Poids : 44 g</p>	<p>Caractéristiques :</p> <p>Alimentation : 4,8 à 6V</p> <p>Angle de rotation : 180°</p> <p>Couple : 1,3 kg.cm</p> <p>Vitesse : 0,12 sec/ 60°</p> <p>Dimensions : 23,2 X 12,5 X 22 mm</p> <p>Poids : 9g</p>

Tableau II.3: Servomoteurs utilisés dans notre projet.

II.7.Ventilateur

Une installation domotique dédiée permettra de programmer les aspirations et extractions d'air, notamment à des endroits stratégiques et nécessaires comme la salle de bains, connue pour sa grande humidité. Le tout grâce à de simple bouton sur l'application que nous avons l'adopter derrière lequel se trouve le module d'interface de ventilation.



Figure II.10 : Ventilateur 5V.

II .8.Description du module de connexion Wifi ESP8266

Les cartes wifi basées sur le microcontrôleur ESP8266 sont programmables comme les cartes Arduino et peuvent communiquer par wifi avec d'autres appareils (ordinateurs, Smartphones, etc.). Il existe plusieurs modèles : l'ESP-D1, l'ESP-03, l'ESP-12 ... etc. L'ESP-01 que nous allons utiliser dans notre travail est présenté dans la figure III.10.

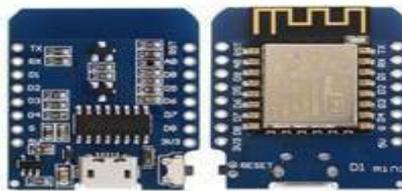


Figure II.11: Module Wifi ESP8266 – ESP-D1.

Le module Wifi ESP8266 est un réseau autonome avec une pile de protocoles TCP/IP intégrée qui peut donner accès au réseau Wifi. L'ESP8266 est capable d'héberger une application ou de décharger toutes les fonctions de réseau Wifi d'un autre processeur d'application. Chaque module ESP8266 est préprogrammé avec un microprogramme de configuration de commande AT (voir l'annexe pour la liste des commandes de l'ESP8266), ce qui signifie que vous pouvez simplement le brancher sur votre appareil Arduino et obtenir autant de fonctionnalités Wifi. Le module ESP8266 est un tableau extrêmement rentable avec un ensemble des commandes [23].

II.8.1.Mise sous tension

- Alimenter le module en reliant V_{cc} et CH PD, et câbler GND;
- La LED rouge doit être allumée et le rester;
- La LED bleue peut clignoter au démarrage;
- Vérifiez si votre module ne présente pas son réseau dans la liste des réseaux WIFI (par exemple un nouveau réseau ESP_99b22 [24]).

II.8.2.Principales caractéristiques d'ESP8266

- ❖ Norme sans fil: IEEE 802.11.
- ❖ Gamme de fréquence: 2.4 GHz ~ 2.5 GHz.
- ❖ Tension de fonctionnement: 3.3v ou 5v.
- ❖ Courant de fonctionnement moyenne 80MA.
- ❖ Taille de paquet: 14.3 mm ×24.8 mm × 3 mm.
- ❖ Sans fil mode réseau: station/softAP/SoftAP + station.

II.9. Les afficheurs LCD

Les afficheurs à cristaux liquides, autrement appelés afficheurs LCD (Liquide Crystal Display), sont des modules compacts intelligents et nécessitent peu de composants externes pour un bon fonctionnement. Ils consomment relativement peu (de 1 à 5 mA), sont relativement bons marchés. Plusieurs afficheurs sont disponibles sur le marché et diffèrent les uns des

autres, non seulement par leurs dimensions, (de 1 à 4 lignes de 6 à 80 caractères), gourmand en intensité (de 80 à 250 mA).



Figure II.12: Afficheur LCD.

II.10. Module pour carte SD / Micro SD

II.10.1. Description :

Le module (adaptateur de carte microSD) est une carte micro SD module de lecteur, et l'interface de spi via le pilote de système de fichiers, système de microcontrôleur pour compléter la carte microSD lire et écrire des fichiers.

II.10.2. Caractéristiques :

- Le circuit de conversion de niveau conseil qui peut interfacer niveau est 5V ou 3,3V.
- Alimentation est 4.5v ~ 5.5v, 3.3v de tension régulateur de circuit.
- Interface de communication est une interface de spi norme.
- 4 M2 trous de positionnement de vis pour une installation facile.
- Taille:4.1*2.4cm.
- Interface dé de commande.
- Un total de six broches (GND, VCC, miso, Mosi, SCK, cs), GND à la terre, VCC

est l'alimentation, le miso, Mosi, sck est le bus de spi, cs est la broche de sélection de signal de puce.

-Régulateur LDO sortie 3.3v comme puce de conversion de niveau, carte ---
Micro SD approvisionnement.

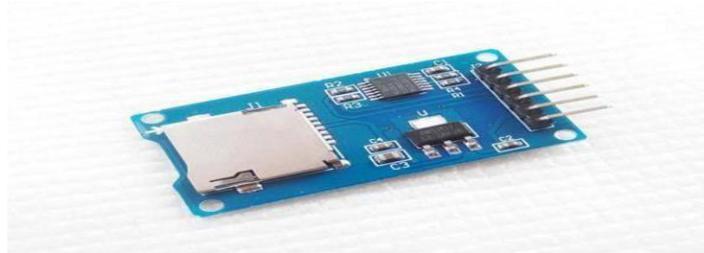


Figure II.13 : Le module carte SD / Micro SD.

II.11.woofer :

Les boomers sont généralement des haut-parleurs électrodynamique mais, contrairement à une opinion très répandue, ils ne sont pas forcément de grand diamètre. Rien n'interdit de reproduire du grave avec un petit haut-parleur adapté à cet usage. Ce sont ses caractéristiques techniques qui déterminent si un haut-parleur peut reproduire du grave, pas son diamètre. En revanche, toutes choses égales, dans le grave un petit haut-parleur ne pourra fournir qu'un niveau sonore limité. Ce niveau est toutefois suffisant dans de nombreux usages. Les haut- parleurs de fort diamètre (200 à 460 mm pour les productions courantes, plus avec certaines réalisations exceptionnelles) permettent d'atteindre des niveaux très élevés, indispensables dans nombre d'utilisations [23].



Figure II.14. Woofer.

II.12. Clavier matriciel

Afin d'assurer l'accès sécurisé à l'habitat, on a pensé à introduire dans notre système domotique, un sous-système permettant à l'utilisateur d'accéder chez lui tout en introduisant un code sécurisé. Pour cela on a besoin d'un clavier matriciel pour entrer le mot de passe d'ouverture de la porte [20].

II.12.1. Principe du clavier matriciel :

Un clavier matriciel (dans notre cas 16 touches) dispose uniquement de 8 broches pour la gestion de ses touches. L'organisation est de 4 colonnes et 4 lignes. Les lignes ont un état de repos imposé (ici l'état bas).

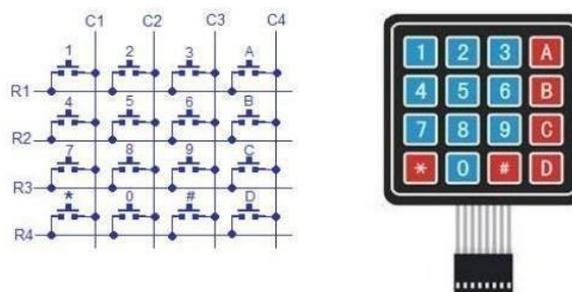


Figure II.1.5 : Clavier matriciel (4*4) et son schéma de principe.

II.12.2. Détection des touches :

Il faut balayer les colonnes une à une par un état haut. Ainsi dès lors qu'une touche est pressée, l'état de la colonne est transmis sur la ligne. Nous pouvons ainsi détecter un état haut. Pour déterminer la touche pressée, il faut se synchroniser avec le signal envoyé sur les colonnes.

II.12.3. Branchement avec Arduino UNO :

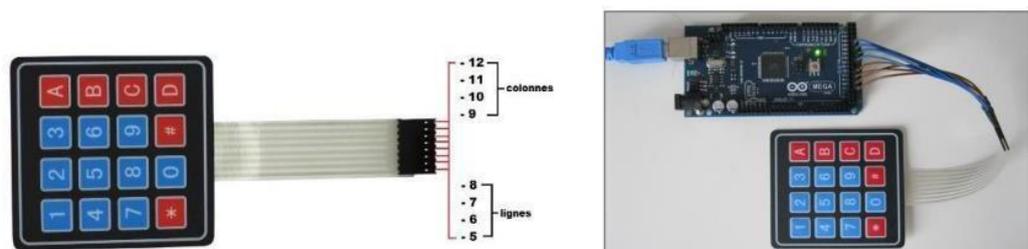


Figure II.16 : Branchement du clavier avec l'Arduino UNO.

II.13.Présentation du logiciel

II.13.1.IDE Arduino

Un IDE (environnement de développement) libre et gratuit est distribué sur le site d'Arduino (compatible Windows, Linux et Mac) à l'adresse <http://arduino.cc/en/main/software>. D'autres alternatives existent pour développer l'Arduino (extensions pour Code Blocks, Visual Studio, Eclipse, XCode, etc.) mais nous n'aborderons dans ce chapitre que l'IDE officiel.

L'interface de l'IDE Arduino est plutôt simple (Figure II.6), il offre une interface minimale et épurée pour développer un programme sur les cartes Arduino. Il est doté d'un éditeur de code avec coloration syntaxique (1) et d'une barre d'outils rapide (2). Ce sont les deux éléments les plus importants de l'interface, c'est ceux que l'on utilise le plus souvent. On retrouve aussi une barre de menus (3) plus classique qui est utilisé pour accéder aux fonctions avancées de l'IDE. Enfin, une console (4) affichant les résultats de la compilation du code source, des opérations sur la carte, etc.

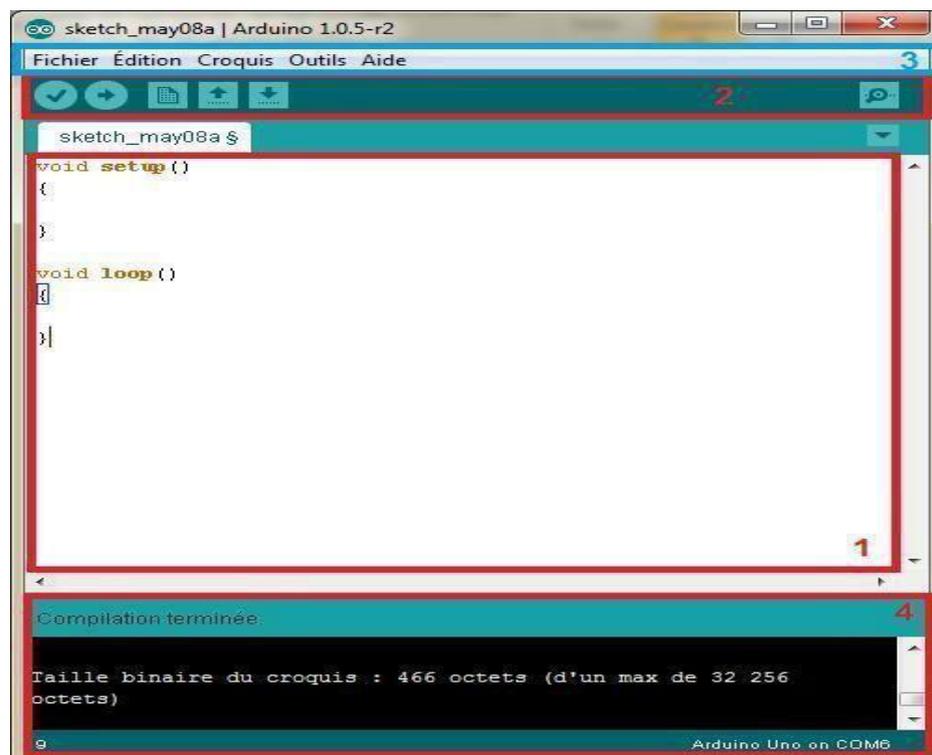


Figure II.17 : IDE Arduino.

II.13.2. Les boutons :

Voyons à présent à quoi servent les boutons numérotés en rouge.



Figure II.18 : Présentation des boutons.

- Bouton 1 (verify): Ce bouton permet de vérifier le programme, il actionne un module qui cherche les erreurs dans votre programme.
- Bouton 2 (upload): Compiler et envoyer le programme vers la carte.
- Bouton 3 (new): Créer un nouveau fichier.
- Bouton 4 (open): Charger un programme existant.
- Bouton 5 (save): Sauvegarder le programme en cours.
- Bouton 6 (serial monitor): Permet d'accéder au port série (en RX/TX).

II.13.3. Langage Arduino

Le langage Arduino est inspiré de plusieurs langages. On retrouve notamment des similarités avec le C, le C++, le Java et le Processing. Le langage impose une structure particulière typique de l'informatique embarquée. La fonction `setup` (voir Fig.8) contiendra toutes les opérations nécessaires à la configuration de la carte (directions des entrées sorties, débits de communications série, etc.). La fonction `loop`, elle est exécutée en boucle après l'exécution de la fonction `setup`. Elle continuera de boucler tant que la carte n'est pas mise hors tension, redémarrée (par le bouton *reset*). Cette boucle est absolument nécessaire sur les microcontrôleurs étant donné qu'il n'on pas de système d'exploitation. En effet, si l'on omettait cette boucle, à la fin du code produit, il sera impossible de reprendre la main sur la carte Arduino qu'exécuterait alors du code aléatoire.

```
void setup()           //fonction d'initialisation de la carte
{
    //contenu de l'initialisation
}

void loop()           //fonction principale, elle se répète
                     // (s'exécute) à l'infini
{
    //contenu de votre programme
}
```

Figure II.19 : Un code minimal.

Au niveau de la syntaxe, on retrouve des similarités avec les langages précédemment cités. La déclaration des variables se fait généralement dans l'espace global (de façon à partager les variables les plus importantes entre les deux fonctions principales). On retrouve les types de base suivant (voir Tableau II.3):

Nom	Contenu	Taille (en octet)	Plage de valeurs
<i>(unsigned) char</i>	Entier ou caractère	1	(0->255) -128 -> 127
<i>(unsigned) int</i>	Entier	2	(0->65 535) -32 768 -> 32 767
<i>(unsigned) long</i>	Entier	4	(0 -> 4 294 967 295) -2 147 483 648 -> 2 147 483 647
<i>float/double</i>	Nombre à virgule flottante	4	-3,4028235E+38 -> 3,4028235E+38
<i>String</i>	Chaîne de caractères (Objet)	variable	Aucune
<i>boolean</i>	Booléen	1	True / False

Tableau II.4: Les types de base.

Il existe d'autres types de base mais ils ne sont qu'un alias de ceux cités précédemment, la liste des types est disponible sur la page des références du site Arduino (<http://arduino.cc/en/Reference/HomePage>). La déclaration des variables suit cette syntaxe:

(const) <type> <nom>([<longueur du tableau>]) (= valeur);

Exemples

const int constante = 12 ;

float univers = 42.0 ; *char* lettre = 'b' ;

String chaine = "Hello World ";

long tableau[12] ;

boolean vrai = true ;

On retrouve les opérateurs les plus courants pour les types de bases. Parmi eux, = (affectation), == (comparaison), != (différence), <, >, <=, >=, && (et logique), || (ou logique), ! (non logique). On retrouve aussi les opérateurs mathématiques (+, -, *, /, %) et les opérateurs logiques bit à bit (^ (XOR), & (et), | (ou), ~(non), << (décalage logique à gauche), >> (décalage logique à droite)).

Les structures de contrôle sont elles aussi similaires aux langages de références. On y retrouve toutes les structures de contrôle standard, conditions, boucle, switch,

fonctions, etc. On peut aussi écrire des structures et des classes. Chaque structure de contrôle est suivie d'un bloc d'instructions délimitées par des accolades. Voici une liste des structures de contrôles les plus utilisées :

Nom	Utilité	Syntaxe
If-else	Condition logique	If (<valeur booléenne>) { <instruction> } else { <instruction> }
If-else if – else	Condition logique multiples	If (<valeur booléenne>) { <instruction> } else if (<valeur booléenne>) { <instruction> } else { <instruction> }
Switch	Sélecteur	Switch (<variable>) { case <valeur> : <instruction> break ; default : <instruction> }
While	Boucle	While (<valeur booléenne>) { <instruction> }
For	Boucle itérative	For (<initialisation> ; <valeur booléenne> ; <évolution>) { <instruction> }

Tableau II.5 : Les structures de contrôle.

Les fonctions, les structures et les classes se déclarent de la même façon qu'en C++. Elles se déclarent sous cette forme :

Structur:

```
struct  
{ <nom><type><nomduchp>;
```

Fonction :

```
<type de reto <nom>(<paramètre>)  
{ <instruction> }
```

Classe : **class** <nom> { **public** : <attributs et champ publics>

private :<attributs et champ privés>

protected :<attribut et champ privés> ;}

II.14.Fonctionnalité de base

II.14.1.Les entrées/sorties

Le langage Arduino vient avec un nombre important de fonction de base permettant d'interagir avec son environnement. Les fonctions les plus utilisées sont les fonctions d'entrée/sorties. Ce sont elles qui permettent d'envoyer ou de mesurer une tension sur une des broches de la carte.

Dans un premier temps, avant d'effectuer une mesure ou d'envoyer une commande. Il est nécessaire de définir la direction des broches utilisées. Pour cela on fait appel à la fonction `pin Mode` en lui donnant d'une part, la broche concernée, et d'autre part, la direction :

```
Void setup() {  
pinMode(1,OUTPUT) ; // Broche 1 en sortie  
pinMode(2,INPUT) ; // Broche 2 en entrée  
}
```

Une fois cette configuration faite, on peut procéder à l'utilisation des broches. Toutes les broches sont capables d'écrire et de lire des données numériques (c'est-à-dire des 0 (0V) ou des 1 (5V)). Mais, certaines disposent de fonctionnalité supplémentaire.

Tout d'abord, toutes les cartes Arduino possèdent des entrées analogiques. Ce sont les broches A0-A1-A2 etc. Elles permettent de lire des tensions analogiques (comprise entre 0 et 5V) et de le convertir en entier (compris entre 0 et 1023) proportionnellement à la tension mesurée. Certaines cartes Arduino possède des sorties analogique faisant l'opération inverse (met une tension sur la broche proportionnellement à l'entier donné), mais ce n'est pas le cas pour l'Arduino UNO.

Pour pouvoir tout de même contrôler des composants autrement qu'en « tout ou rien » il est possible d'utiliser des broches PWM. Ce sont les broches annotés par un tilde ~ sur la carte. Les PWM (Pulse Width Modulation) sont utilisées pour synthétiser

des signaux analogiques en modulant le temps passé à l'état 1 (5V). Le signal obtenu est représenté (figure II.9). En utilisant une fréquence relativement élevée, les PWM permettent de commander certains composants comme si il recevait une tension analogique. Cela provient du fait que les composants utilisés dans l'électronique analogique, ne changes pas d'états instantanément. Par exemple, une ampoule à incandescence reste chaude et éclaire un court instant après avoir été éteinte. Ce phénomène est généralement invisible à l'œil nu. Grâce à elles, on pourra par exemple faire varier l'intensité d'une LED. La plupart des cartes Arduino utilisent des PWM cadencées à 490 Hz environ.

Toutes ces fonctionnalités sur les broches d'entrées sorties sont utilisables par le biais de quatre fonctions :

- **digitalRead(pin)**: mesure une donnée numérique sur une des broches, la broche en question doit être réglée en entrée.
- **digitalWrite(pin, value)**: écrit une donnée numérique sur une des broches, la broche concernée doit être réglée en sortie. Le paramètre *value* doit être égal à *HIGH* (état 1 soit 5V) ou *LOW* (état 0 soit 0V).
- **analogRead(pin)**: mesure une donnée analogique sur une des broches (compatible seulement), la broche doit être réglée sur entrée.
- **analogWrite(pin, value)**: écrit une donnée sous forme de PWM sur une des broches (compatible uniquement), la broche doit être réglée en sortie. Le paramètre *value* doit être compris dans l'intervalle $[0;255]$.

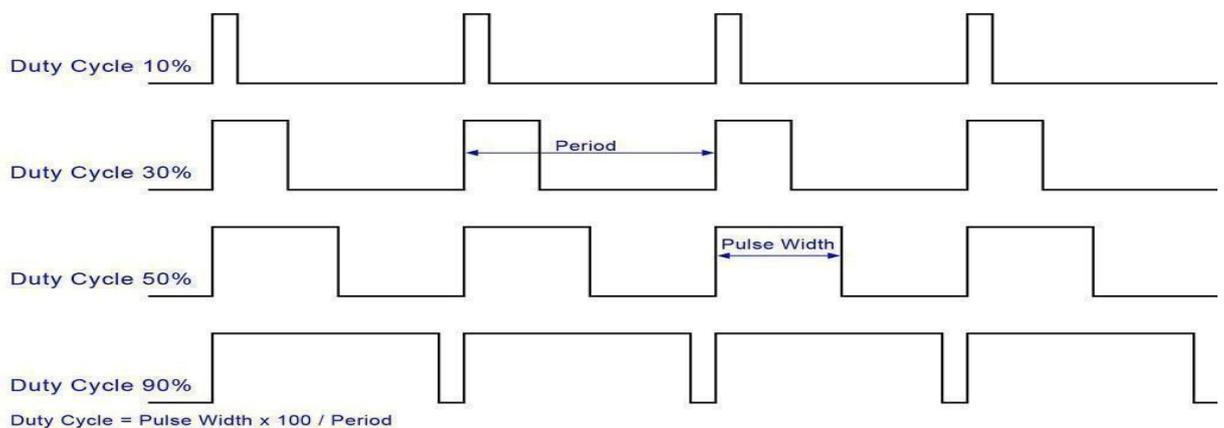


Figure II.20: Signal PWM.

II.15. Gestion du temps

Pour la plupart des applications de domotique, il est nécessaire de faire intervenir des intervalles de temps. Par exemple, pour gérer le temps d'appui sur un bouton ou pour faire une sonnerie qui se répète un certain nombre de fois. Le langage Arduino fournit quelques fonctions permettant de gérer le temps.

Il est possible d'insérer une pause dans son programme pendant un instant. Pour cela, on utilise les fonctions `delay` et `delayMicroseconds` qui insère une pause suivant le paramètre passé (en milliseconde pour l'un, en microseconde pour l'autre). Cependant ces fonctions bloquent le microcontrôleur, on ne peut alors plus effectuer aucune action.

En plus d'insérer une pause, il est possible de mesurer le temps. De la même manière que les fonctions de délai, on utilise les fonctions `millis` et `micros` qui donnent le nombre de milliseconde (respectivement microseconde) depuis le lancement de la carte. Attention, ces fonctions incrémentent une variable (interne). Ces variables se remettent à zéro une fois le maximum atteint (overflow). La variable utilisée pour les millisecondes atteint son maximum au bout de 49 jours et 17 heures et la variable utilisée pour les microsecondes au bout de 71 minutes et 34 secondes environ. Il faut donc faire attention lors de l'utilisation de ces fonctions pour des utilisations longues durées.

II.16. Les interruptions

Il est parfois nécessaire en informatique embarquée, d'attendre un événement externe (appui sur un bouton, données d'un capteur, etc.) pour effectuer une action. Pour ce type de problème, on utilise les interruptions. Les interruptions sont des portions de code (fonctions) appelés lorsque qu'un événement (interne ou externe) survient et a besoin d'être traité sur le champ. Il faut cependant faire attention, ce mécanisme interrompt le code exécuté, il est prioritaire par rapport au reste du code. Vu qu'il est possible de mesurer les événements ponctuellement (via les fonctions d'entrées/sorties) on utilise généralement les interruptions pour du code critique (arrêt d'urgence par exemple) ou des événements non-ponctuels (transmissions de données depuis un ordinateur par exemple).

Aussi, le nombre d'interruption externe est limité sur à 2 sur la plupart des cartes Arduino. Les interruptions sont utilisables sur les broches compatibles seulement (broches 2 et 3 sur l'Arduino UNO). Pour choisir la fonction et la broche utilisée pour

l'interruption, on utilise la fonction `attachInterrupt`. On peut utiliser `detachInterrupt` pour supprimer l'interruption. Il est possible de partir en interruptions sur 4 types d'événements :

- **LOW** : Lorsque la broche est à l'état 0 (0V)
- **RISING** : Lorsque la broche passe de l'état 0 (0V) à l'état 1 (5V) (front montant).
- **FALLING** : Lorsque la broche passe de l'état 1 (5V) à l'état 0 (0V) (front descendant).
- **CHANGE** : Lorsque la broche change d'état (front montant et front descendant).

II.17. Description du programme

Le programme est structuré en 4 parties distinctes :

La première partie est une zone de commentaire (qui ne sera pas interprété par le microcontrôleur) dans lequel on renseigne en général, le nom du créateur du programme, la date de création, d'éventuels liens pointant vers une page web, les révisions (s'il y en a) et la description du programme. Cette partie n'a pas une grande importance mais elle peut vous aider à comprendre ce qu'à voulu faire le codeur de ce programme. Elle est donc aussi utile pour les autres. Cette zone de commentaire (pour ne pas être lue par le microcontrôleur et donc générer des erreurs) commencera par un « `/*` » et se terminera par un « `*/` ».

La deuxième partie est faite pour déclarer des variables et les représentées par la fonction « `int` ». Les variables sont des expressions que vous pouvez utiliser dans les programmes pour stocker des valeurs numériques. Les constantes prédéfinies du langage Arduino sont des valeurs particulières ayant une signification spécifique.

La troisième partie permet de configurer les entrées et sorties. La syntaxe employée est

« `void setup ()` » dans laquelle on donnera le détail entre accolade par exemple: `{ fonction (nom, état) }`.

La quatrième et dernière partie concerne la programmation des interactions et des comportements que l'on souhaite donner à un composant électronique par exemple.

La syntaxe employée est « `void loop()` » dans laquelle on donnera le détail entre accolade comme pour la configuration des entrées/sorties.

Les fonctions de base « `void setup ()` » et « `void loop ()` » sont obligatoires dans tout programme en langage Arduino.

II.17.1. Les étapes de téléchargement du programme

Une simple manipulation enchaînée doit être suivie afin d'injecter un code vers la carte arduino via USB.

- On conçoit ou on ouvre un programme existant avec le logiciel IDE Arduino. On vérifie ce programme avec le logiciel Arduino (compilation).
- Si des erreurs sont signalées, on modifie le programme. On charge le programme sur la carte.
- On câble le montage électronique.
- L'exécution du programme est automatique après quelques secondes.
- On alimente la carte soit par le port USB, soit par une source d'alimentation autonome (Pile 9 volts par exemple).
- On vérifie que notre montage fonctionne.

II.18. Conclusion

Dans ce chapitre nous avons vu une présentation générale de la carte Arduino UNO, ses caractéristiques et présentation des quelques shields et on mettra la lumière sur le logiciel IDE Arduino.

On peut conclure sur le fait que les cartes Arduino sont des outils puissants pour les cartes électroniques. Mais aussi, elles permettent un accès facile et intuitif à l'informatique embarqué. On pourra ainsi enrichir tout ces projets à base d'un microcontrôleur pour leur donner une valeur plus importante.

La carte d'Arduino UNO est la plus simple carte et la plus courante. Il existe d'autres versions de cartes Arduino plus adaptées pour certains projets. Dans notre travail on va utiliser la carte d'Arduino UNO pour réaliser un système de commande à distance des installations électriques pour la domotique connectée par wifi. Pour certains projets plus complexes, il va falloir d'utiliser l'Arduino Méga.



Chapitre III :
**Développement
d'une application**

III.1.Introduction

L'Androïde est parmi les derniers systèmes d'exploitation qui développent les exigences des téléphones intelligents. La plateforme androïde de smart phone devient de plus en plus importante pour les réalisateurs de logiciel, en raison de ses puissantes possibilités et open source.

Lors des années précédentes, le traitement des données informatiques se fait par des ordinateurs ; en revanche le smart phone a des avantages qui ont les mêmes fonctions que l'outil informatique ce dernier porte l'intérêt de l'ordinateur grâce à l'androïde.

La téléphonie mobile a connu une explosion dans les années 2000 mais aucune révolution n'a semblé arriver depuis que les appareils se ressemblent. Les innovations n'avaient plus vraiment de saveur ; les applications étaient difficiles d'accès de par leur mode de distribution et Souvent peu performantes à cause des faibles capacités des appareils.

Depuis quelques mois, les smart phones sont dotés d'une puissance plus importante et D'espaces de stockages conséquents. Les téléphones tendent à devenir des objets artistiques, presque de reconnaissance sociale, et possèdent des fonctionnalités qu'aucun téléphone ne pouvait espérer auparavant : connexion haut débit, localisation GPS, boussole, accéléromètre, écran tactile souvent multipoint, marché d'applications en ligne. Autant de qualités permettant de créer des applications innovantes et de les distribuer en toute simplicité.

La plate-forme Androïde apporte tout cela au consommateur, mais surtout, elle affranchit le développeur de nombreuses contraintes. Par son ouverture ; elle permet à n'importe quel développeur de créer ses applications avec un ticket d'entrée quasi nul. Le framework et le système d'exploitation et outils associés ont un code source ouvert, leur accès est gratuit et illimité. Plus besoin de négocier avec le constructeur du téléphone pour qu'il vous laisse développer sur sa plate-forme. Tous les développeurs sont ainsi sur un même pied d'égalité, tous peuvent ajouter de la mobilité à des applications existantes.

Cette partie de notre étude ne nous donnera pas de bons résultats si on néglige certains paramètres ; donc le bon fonctionnement de notre système se base essentiellement sur une bonne démarche et une bonne réflexion de notre programme.

III.2. le système Androïde

Androïde est un système d'exploitation développé initialement pour les Smart phones. Il utilise un noyau Linux qui est un système d'exploitation libre pour PC et intègre tous utilitaires et les périphériques nécessaires à un smart phone. Il est optimisé pour les outils Gmail. Aussi, l'androïde est libre et gratuit et a été ainsi rapidement adopté par des fabricants.

La société Androïde a été rachetée en 2007 par Google. Mais aujourd'hui, l'Androïde est utilisé dans de nombreux appareils mobiles (smart phones). Les applications sont exécutées par un processeur de type ARM à travers un interpréteur JAVA. En plus de cela, l'androïde concurrence l'opérateur système d'Apple qu'il tend à dépasser en nombre d'utilisateurs. Androïde évolue pour mieux gérer l'hétérogénéité des appareils qu'il utilise. [25]

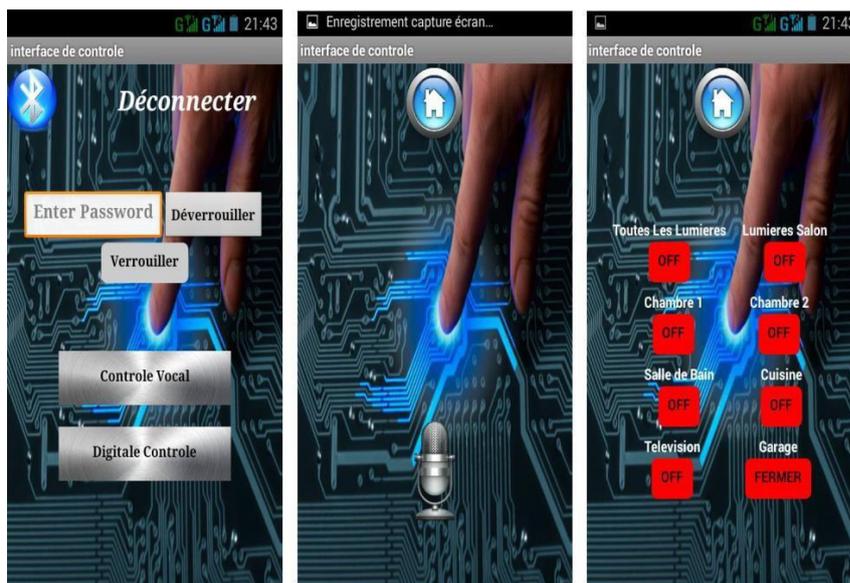


Figure.III.1 : Image de l'application sur Mobile.

III.3. Application Android dans les Smartphones

Pour faciliter l'accès aux résultats sur un Smartphone nous avons ajouté une application sur Android qui s'appelle " **App Inventor**".

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

La plateforme de développement est offerte à tous les utilisateurs possédant un compte Gmail. Elle rappelle certains langages de programmation simplifiés basés sur une interface graphique similaire à Scratch. Les informations des applications sont stockées sur des serveurs distants [25].

Elles sont actuellement entretenues par le Massachusetts Institute of Technologie (MIT).

L'environnement d'App Inventor contient trois fenêtres qui sont proposées pendant le développement :

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

La connexion d'un terminal réel sous Androïde permettra ensuite d'y télécharger le programme pour un test réel. Ce terminal pourra aussi bien être un téléphone qu'une tablette ; le comportement du programme sera identique [25].

III.3.1. Structure d'une application App Inventor

Une application développée sous App Inventor est constituée de deux parties distinctes mais étroitement liées.

- A. **L'interface graphique** : Pour créer l'application sous App Inventor l'interface graphique contient nos propriétés (taille, couleurs, position, textes).

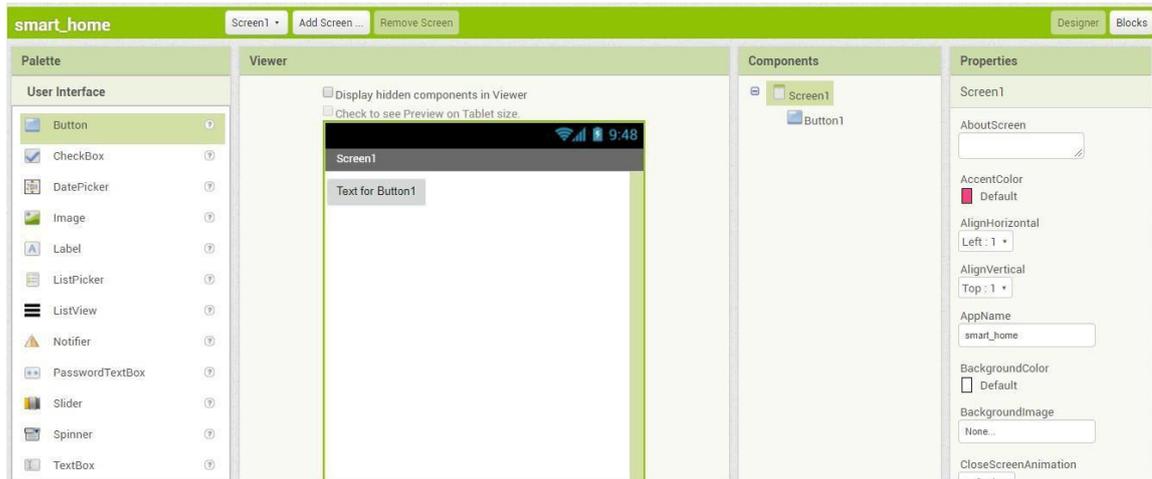


Figure III.2 : Première interface de la création App Inventor

Cette partie contient des composants (visibles ou invisibles) [25].

Cette interface graphique contient quatre parties :

Partie 1 : Une palette sous App Inventor contenant tous les éléments qui peuvent être positionnés sur l'écran du téléphone.

Partie 2 : C'est la surface du téléphone ajusté automatiquement par App Inventor ou manuellement par nous-même en utilisant le composant « Screen arrangement ».

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

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

Après l'assemblage des différents composants qui constituent notre application, on peut résumer cette phase « Scratch » par la figure qui suit.

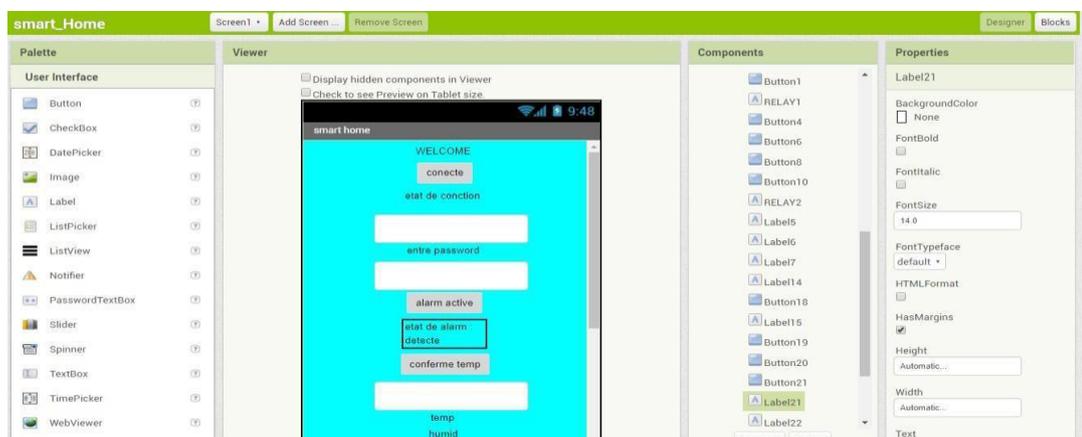


Figure III.3 : Schéma global du Screen 1

B. Editeur de blocs (Fenêtre Scratch)

Une fois les composants de l'écran de téléphone mis en place et désigné, nous passons à la deuxième phase de développement d'une application via App Inventor : l'interface Scratch, pour cela, il faut cliquer sur «Open the Blocks Editor» en haut à droite de la page. L'interface Scratch permet d'imbriquer des éléments graphiques entre eux pour effectuer la partie programmation de l'application à développer. De cette partie, on peut assembler les différents blocs de l'application et indiquer comment les composants doivent se comporter et qui s'affichent dans l'émulateur virtuel [25].

Pour ouvrir ce Block, on a cliqué en haut à gauche sur « Open the Blocks Editor ». App Inventor, lancé le Blocks Editor depuis le serveur du MIT et proposé l'exécution du fichier App Inventor For Android Code blocks.

Comme indiqué sur la figure (III.2) App Inventor, éditeur de blocks, cette interface est très et épurée. En effet, en haut ; on retrouve des éléments classiques «Save», «Undo» et «Redo» ainsi qu'un bouton de test pour lancer l'application sur le mobile ou sur l'émulateur.

Cette interface contient des parties Built-in et blocks.



Figure III.4: En-tête d'éditeur de blocks App Inventor

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

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

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

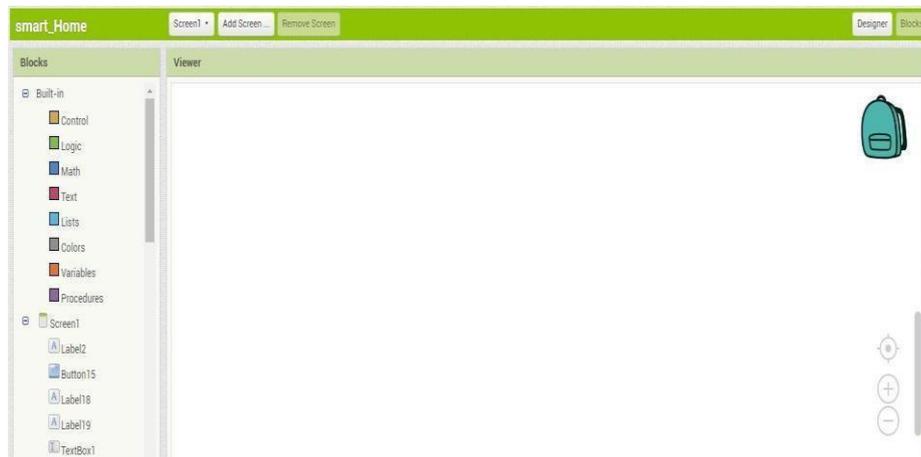


Figure III.5 : Editeurs de blocks App Inventor

La palette des variables et fonctions est à gauche, l'onglet My blocks propose les fonctions associées aux éléments déposés sur notre écran au préalable. Dans l'onglet «My Blocks», Lorsque on clique sur l'élément « Button1 » on obtient la figure suivante est cela comme échantillon pour n'importe quel composant associé dans notre application.



Figure III.6 : Echantillon d'un composant sous App Inventor

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

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

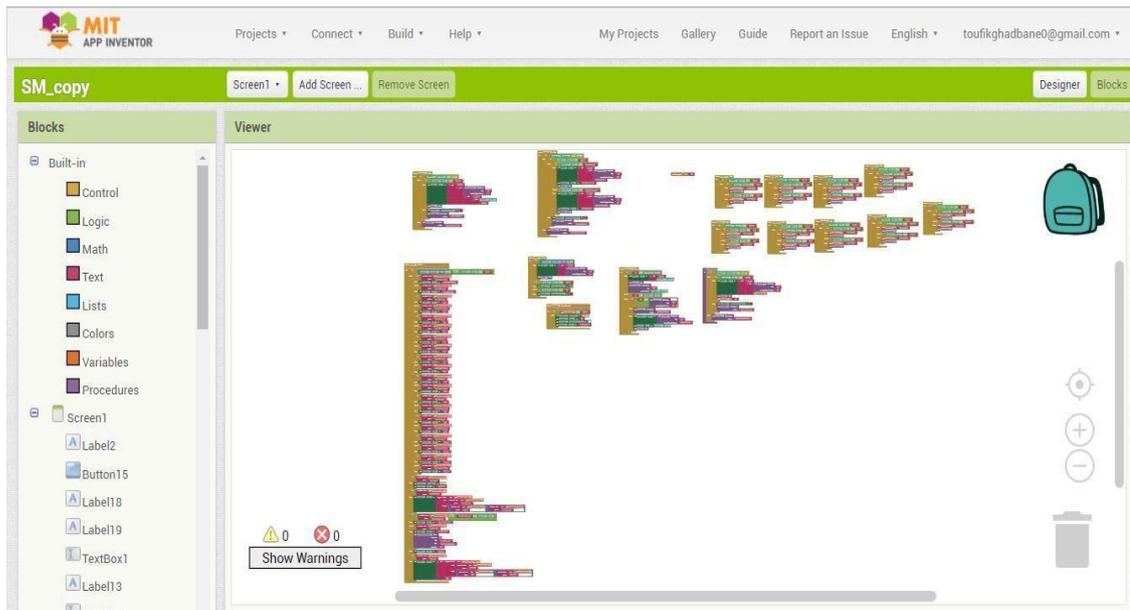


Figure III.7 : Schéma global du Scratch

III.4. Explication et démarche

Notre interface contient un bouton afficher l'état de connexion donc il faut d'abord s'assurer que le smart phone est connecté. Un mot de passe pour raison de protection a été programmé afin d'exécuter un ordre quelconque comme contrôler l'éclairage de chaque pièce, ouvrir le garage et la porte tout par un simple appui sur un bouton. En peut lire la température, humidité et l'état de l'alarme (la détection de gaz et le mouvement). Cette figure montre que notre application nommée «smart home» peut être installée sur smart phone.

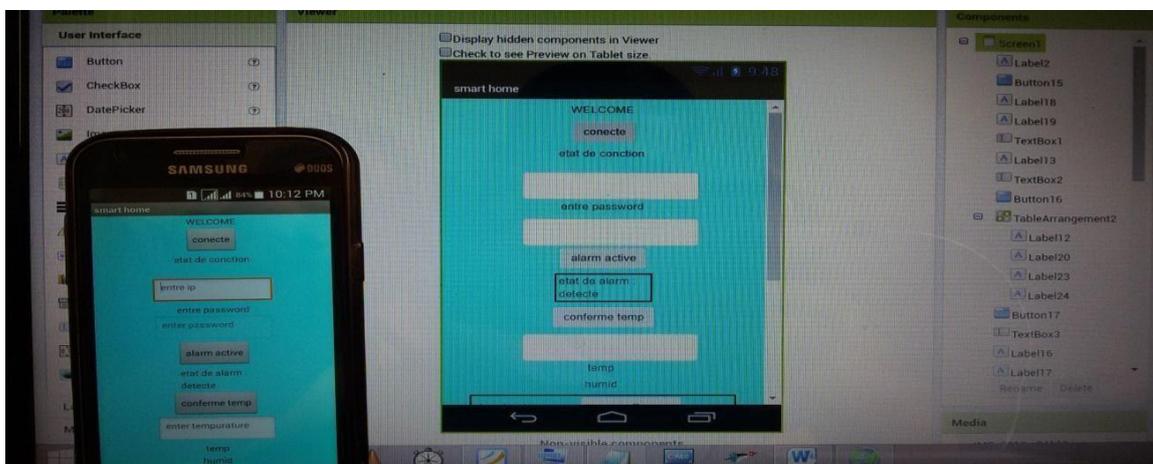


Figure III.8 : Photo réelle de notre application.

III.5.Présentation des fonctions de système

III.5.1Fonction de point d'accès (Passerelle)

Pour se connecter au réseau local du système domotique via le module Wifi ESP8266, on a besoin d'une passerelle pour que l'utilisateur puisse accéder à l'application de commande.

III.5.2.Fonction de la sécurité

Cette fonction permet de détecter s'il y a une intrusion à l'aide du capteur de mouvement (PIR) qui lancer une alarme pour informer l'utilisateur, en affichant sur l'application de commande la détection de mouvement et aussi envoyé un appelle d'alerte sur la Smartphone à l'aide du module GSM SIM800L. L'accès sécurisé à l'habitat est assuré en introduisant le code correct par le smart phone.

III.5.3.Fonction d'ouverture du garage et la porte principale

La commande d'ouverture du portail du garage et la porte principale est réalisée à distance via l'application de commande en agissant sur le contrôle du servomoteur pour faire ouvrir/fermer le portail du garage et la porte principale.

III.5.4.Fonction d'éclairage

La fonction d'éclairage est assurée via l'application par les boutons ON/OFF qui va être commandée à travers le module relais qui commander les lampes de l'éclairage.

III.5.5.Fonction de l'acquisition de la température et d'humidité

La fonction de l'acquisition de la température et d'humidité est réalisée via le capteur DHT22 par la suite les valeurs seront affichées dans le smart phone.

III.5.6.Fonction de ventilation

La ventilation de l'intérieur de l'habitat est assurée par l'intermédiaire de l'application ou automatiquement vis-à-vis la variation de la température (déclaration d'un seuil de température pour lancer la ventilation)

III.5.7. Fonction de détection de gaz/fumée

Cette fonction permet de détecter s'il y a des fuites de gaz dans la cuisine à l'aide du capteur MQ-2 et lancer une alarme pour informer l'utilisateur en affichant sur l'application de commande la détection de gaz et aussi envoient un message d'alerte sur la Smartphone.

III.6. Présentation de l'interface Arduino-PC

L'interface permet de faire une connexion entre notre carte Arduino UNO et le PC à travers le module Wifi ESP8266 et commander par le smart phone.

Le programme de gestion des données permet d'afficher toutes les valeurs en temps réel et chaque 5 secondes sur une fenêtre du moniteur sérié (Voir figure III.21 et III.4). Dans les annexes vous allez trouver le programme exécuté dans Arduino uno et le module Wifi ESP8266 (voir l'annexe programme 01 et 2).

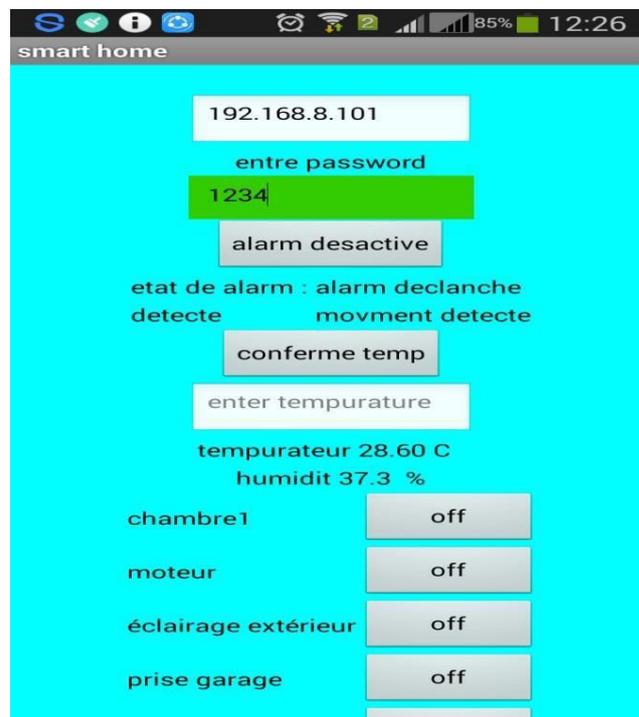


Figure III.9: Image réel de l'interface de l'application de commande.

```
mov detecte
Call #
Calling 0669422800
----> ATD0669422800;
<--- OK
Sent!
alarm-declanche
tempurateur 28.60
humidity 37.00
tempurateur 28.60
humidity 37.00
mow acune tempurateur 28.60
humidity 37.10
tempurateur 28.60
humidity 36.80

BUSY
alarm repose mow acune tempurateur 28.60
humidity 37.10
tempurateur 28.60
humidity 37.30
tempurateur 28.60
humidity 37.50
alarm repose ----> ATH0
<--- OK
OK!
tempurateur 28.60
humidity 37.40
alarm-active
alarm repose ----> ATH0
<--- OK
```

Figure III.10: Fenêtre d'affichage des résultats de système sur le moniteur sérié.

III.7.Conclusion

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



Chapitre VI :
Réalisation du
système

IV. Introduction

La domotique est basée sur la mise en technologie des différents appareils électriques de la maison, contrôlés par une « intelligence » centralisée. L'intelligence qui gère ces commandes est une centrale programmable, des modules (intelligence décentralisée), ou bien une interface micro-informatique (un serveur). Nous avons décidé, afin de faciliter nos échanges, de diviser l'installation en 6 domaines d'application : serrure codée, positionnement de caméra, détecteur de présence, commande Bluetooth, garage et l'application, tous ces domaines peuvent être contrôlés individuellement ou ensemble grâce à des échanges d'information.

IV.1.Réalisation pratique

Après les tests en simulation il a fallu passer par la réalisation en réelle car il y'a une différence entre la théorie et la pratique.

IV.1.1.Montage sur plaque d'essai

Pour tester le bon fonctionnement de notre système il a fallu qu'on le monte sur une plaque d'essai pour voir s'il n'y a pas de problème avant de réaliser le circuit imprimé.

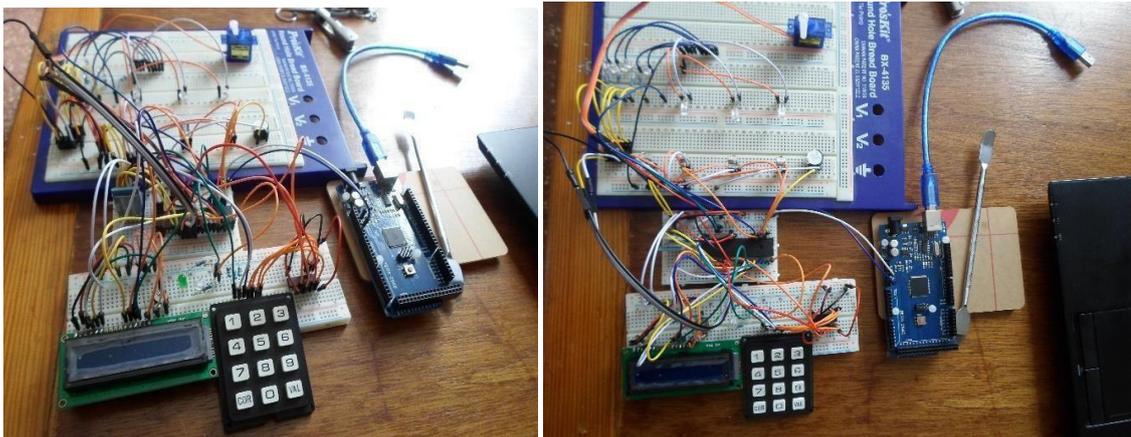


Figure IV.1: Montage sur plaque d'essai.

IV.1.2.Les problèmes de la réalisation pratique

Tout d'abord pour commencer, notre projet exige beaucoup de temps car il est un peu complexe à réaliser, il exige beaucoup de matériels et de tests pour les différents composant du circuit car il ne suffit pas de rassembler tous les composants dont nous avons besoin pour que le circuit fonctionne il faut qu'ils doivent fonctionner ensemble comme par exemple on a essayé de mettre un registre a décalage pour faire fonctionner le moteur pas à pas et pour minimiser l'utilisation des ports du microcontrôleur ; mais ça n'a pas fonctionné nous avons eu des chutes de courant provoqués par le registre à décalage.

Pour rester sur le moteur pas à pas nous avons eu un autre souci sur celui-ci pour le faire tourner car entre la théorie et la pratique il y a un grand espace il faut prendre en considération les

Caractéristiques du moteur et sa fréquence de rotation car c'est ce point qui nous a posé problème, Nous avons envoyé les donnés à une fréquence faible pour voir la progression des nombre de pas et en théorie c'est correcte même en simulation ça a fonctionné mais dans la pratique le moteur reste figé alors après quelques tests et de nombreuses recherches il a fallu adapter notre programme à partir des caractéristiques du moteur qui se trouve sur son data-sheet malgré que le test en simulation ne fonctionne pas on a tenté le test en réelle et ça a marché très bien.

Nous avons aussi rencontré quelques problèmes avec un autre type de moteur qui est le servomoteur, normalement pour faire tourner ce moteur il faut injecter un signal carré d'une durée de 20ms mais le temps de sa position haute change en fonction de l'angle voulu alors nous avons essayé d'envoyer ces signaux en utilisant le signal sortait correctement mais le servomoteur ne réagisse pas comme en voulait alors nous avons changer de méthode en utilisant la PWM mais ça n'a pas marché aussi alors on a utilisé une autre méthode qui consiste à utiliser une boucle avec un délais bien précis mais la variable de la boucle change en fonction de l'angle voulu.

Nous avons eu aussi un autre souci concernant la partie de saisie du mot de passe car sur Simulation ça fonctionnait très bien mais arrivé au test réel il a fallu corriger certaines parties entre autre celle du clavier coté mécanique. Pour remédier à ce problème il a fallu ajouter en parallèle des résistances de 1K reliées à la masse sur chaque sortie du clavier. Et enfin on a pu résoudre chaque problème individuellement pour chaque étage et on a réussi à faire fonctionner chaque dispositif correctement et selon sa tâche bien précise .Il nous restait juste à tout rassembler et les faire marcher en totale synchronisation et pour cela il a fallu réaliser le circuit imprimé final.

IV.2.Circuit Imprimé Final

Pour réaliser le Circuit final il a fallu passer par l'autre partie du logiciel « ISIS 7 » qui est « ARES » ce dernier est conçu spécialement pour la réalisation des « PCB » c'est-à-dire les circuits imprimés.

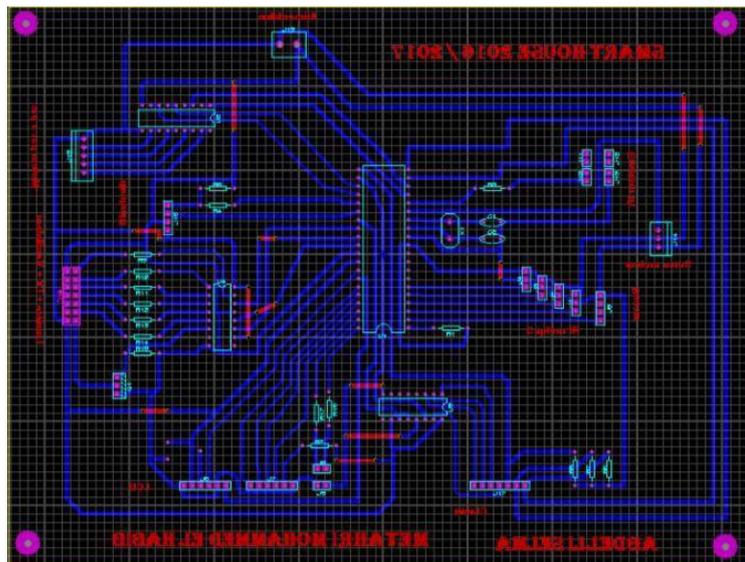


Figure IV.2: Circuit sur ARES.

IV.3.Réalisation du système



Figure IV.3 : Photos de la maquette réalisée (vue externe).



Figure IV.4: Photos d'intérieur de maquette réalisée.

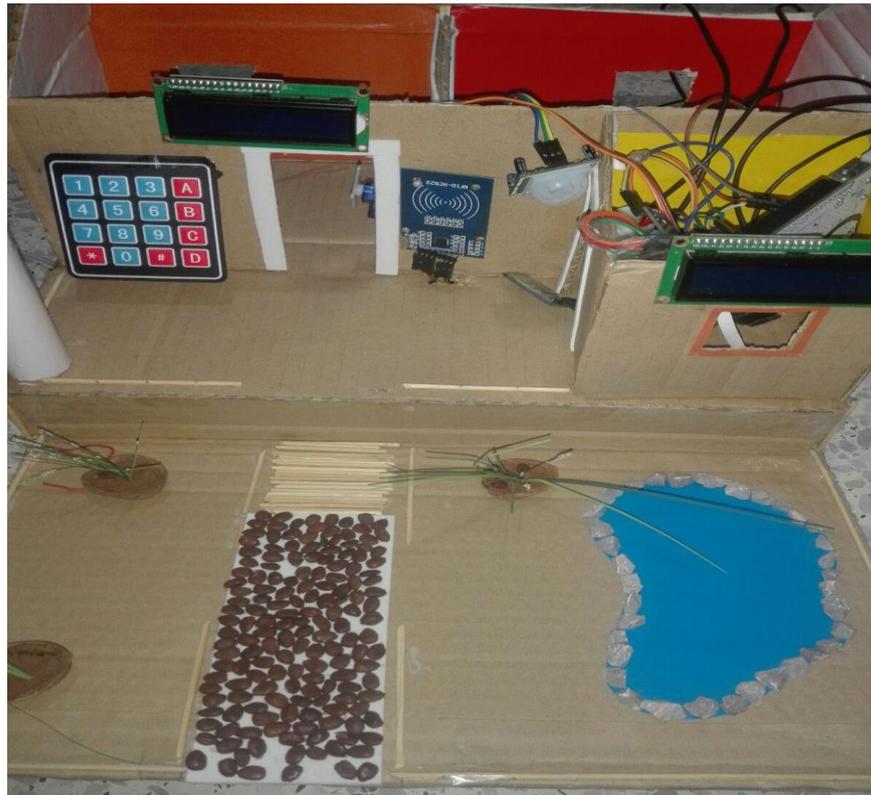


Figure IV.5 : Photos de la réalisation finale.

IV.4.Conclusion

Nous avons pu concevoir une maison intelligente avec un système simple. Grâce à des matériaux électroniques et des outils, nous avons créé une maquette intégrant des composants électroniques, puis nous avons relié ces composants au microcontrôleur. Nous avons utilisé un logiciel permettant de contrôler la maquette. En conclusion, on peut dire que les systèmes de la domotique sont des systèmes révolutionnaires voués à évoluer encore plus dans le futur. Ces systèmes permettent de contrôler une maison depuis de simples petits boîtiers disposés un peu partout dans la maison. Le fait de pouvoir relier ces boîtiers à un Smartphone ou autre réseau de communication (Bluetooth, wifi...) est très intéressant car cela permet de contrôler sa maison à distance. Ce système pourra donc s'avérer très utile pour le confort des personnes.

Conclusion générale

Dans ce projet nous nous sommes concentrés à l'étude et la réalisation d'un système de commande à distance pour commander des installations électriques pour la domotique, utilisant un protocole de communication sans fil avec un Smart- phone.

Nous avons réalisé un système de mesure en temps réel de l'ensemble des phénomènes physiques à base d'une carte Arduino UNO comme une unité de commande, le rôle de la carte Arduino UNO est de traiter les données délivrées par les capteurs et commander des différents actionneurs utilisés.

Au début on a essayé de relier le système au support d'affichage (PC) par un câble USB pour assurés le bon fonctionnement des capteurs. Le programme écrit sur IDE Arduino permet d'afficher les résultats sur le moniteur série.

En deuxième lieu, nous avons développé une interface de commande sous Androïde (une application sur Smart-phone) avec l'environnement MIT App Inventor. Nous avons préparé les icônes et les labelles des commandes pour permettre au système de superviser les résultats attendus programmés dans la carte Arduino.

La troisième partie de ce travail consiste à faire la réalisation pratique de notre système. Pour connecter le système au réseau internet on a utilisé la connexion sans fil (par wifi) à travers le module ESP8266. Pour la visualisation des grandeurs physiques mesurées on a exécuté un programme capable d'afficher les résultats en temps réel sur l'application de Smart-phone.

Une telle réalisation n'est pas dénuée de difficultés. Il est à noter que nous nous sommes confrontés à plusieurs problèmes surtout dans la partie de la connexion sans fil. Cependant, on peut dire que malgré ces difficultés, les résultats obtenus à travers cette étude qu'ils soient pratiques ou théoriques, permettent d'ouvrir la porte à d'autres études. Nous espérons que ce mémoire sera une référence aux personnes désirant développer et réaliser des projets et systèmes à base d'une carte Arduino.

Bibliographie

- [1] Krama. A, Gougui. A, "Etude et réalisation d'une carte de contrôle par Arduino via le système Androïde", Mémoire Master Académique, Université Kasdi Merbah Ouargla, Algérie, 2015.
- [2] HAMOUCHI. H, "Conception et réalisation d'une centrale embarquée de la domotique «Smart Home »", Mémoire Master, Université Mohammed V de Rabat, Rabat, 2015.
- [3] Yacine. Y, "Minimisation d'énergie dans un réseau de capteur", Mémoire de Magister, Université Mouloud Mammeri de Tizi Ouzou, Algérie, 2012.
- [4] Serge Darrieumerlou, " le guide de la maison et des objets connectés", Edition Eyrolles, 2016.
- [5] <http://www.cea.fr/comprendre/Pages/nouvelles-technologies/essentiel-sur-domotique-maison-connectee>.
- [6] Mr, METAHRI.M, Melle ABDELLI.S, " Smart House ", Mémoire Master, Université ABOU BEKR BELKAID de Tlemcen, Algérie, 2017.
- [7] Boudellal. M, "Smart home - Habitat connecté, 361 installations domotiques et multimédia". Dunod, 2014.
- [8] http://www.cisco.com/c/dam/global/en_ca/solutions/executive/assets/pdf/internet-of-things-fr.pdf, Avril 2017.
- [9] Ming. Y, "Cobalt oxide nano sheet humidity sensor integrated with circuit on chip" Microelectronic Engineering, 2011.
- [10] Astalaseven, Eskimon et Olyte, "Arduino pour bien commencer en électronique et en programmation", Licence Créative Commons BY-NC-SA 2.0.
- [11] Lechalupé. J, "cours d'initiation à Arduino", Université Paul Sabatier, Mai 2014.
- [12] Cottenceau. B, "Carte ARDUINO UNO Microcontrôleur ATmega328".
- [13] moleins, j.-p. (s.d). maison communicante. Récupéré sur ooreka: <https://domotique.ooreka.fr/comprendre/maison-communicante>
- [14] P.Rey, P. &. (2008). Les afficheurs à cristaux liquides. Récupéré sur aurel32: <https://www.aurel32.net/elec/lcd.php>

-
- [15] Party, K. B. (s.d.). Servomoteur Fonctionnement. Récupéré sur interface-z:
<http://www.interface-z.fr/conseils/servomoteur.htm>
- [16] Quartz. (s.d.). Récupéré sur Génie électrique:
http://forge.blogspot.com/2015/11/quartz-electronique_17.html
- [17] Lechalupé, J, "cours d'initiation à Arduino", Université Paul Sabatier, Mai 2014.
- [18] SECURITY, B. (s.d.). *Clavier*. Récupéré sur Alarme Optique Domotique:
<http://www.aod-alarmes.com/Clavier.html>
- [19] Stargazer. (2016, mars 13). *Bluetooth (HC-06 ou HC-05)*. Récupéré sur informatique et electronique DIY:
<https://knowledge.parcours-performance.com/arduino-bluetooth-hc-05-hc-06/>
- [20] Tarrière, M. (s.d.). *Clavier à code : fonctionnement, mise en service et budget*. Récupéré sur companeo:
<http://www.companeo.com/securite-electronique/guide/clavier-acode-fonctionnement-mise-en-service-budget#0>
- [21] Tavernier, C. (1995). *Les microcontrôleurs PIC*. Dunod.
- [22] Thiébaud-Brodier, M. (2011). *Domotique - Sécurité, confort, économies*. Publitronic-Elektor.
- [23] WordPress. (2014). *Composition du Servomoteur*. Récupéré sur Le blog d'Eskimon:
<http://eskimon.fr/287-arduino-602-un-moteur-qui-de-la-tete-le-servo-moteur>
- [24] WordPress. (s.d.). *Définitions: Domotique / Smart Home*. Récupéré sur Domosens:
<http://domosens.com/definitions-domotique-gtb-gtc-ergonomie/>
- [25] Yvraut, G. (s.d.). Les Moteur pas a pas . *Séminaire Bellegarde Nov 99* (pp. 6,9,12). Copyright 99 G.

Annexes A

Proteus (ISIS et ARES)

Proteus est une suite logicielle destinée à l'électronique. Développé par la société LabcenterElectronics, les logiciels inclus dans Proteus permettent la CAO dans le domaine électronique. Deux logiciels principaux composent cette suite logicielle : ISIS, ARES, PROSPICE et VSM.

Présentation générale

Cette suite logicielle est très connue dans le domaine de l'électronique. De nombreuses entreprises et organismes de formation (incluant lycée et université) utilisent cette suite logicielle. Outre la popularité de l'outil, Proteus possède d'autres avantages

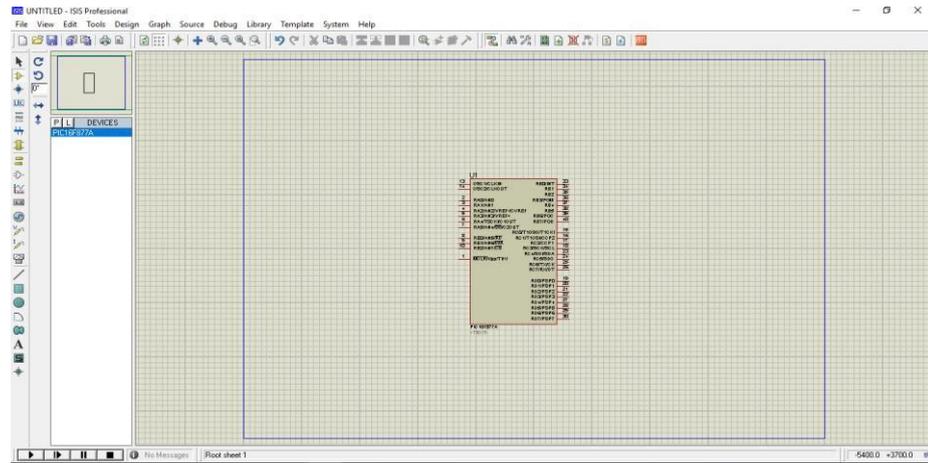
- Pack contenant des logiciels facile et rapide à comprendre et utiliser
- Le support technique est performant
- L'outil de création de prototype virtuel permet de réduire les coûts matériel et logiciel lors de la conception d'un projet

ISIS

Le logiciel ISIS de Proteus est principalement connu pour éditer des schémas électriques. Par ailleurs, le logiciel permet également de simuler ces schémas ce qui permet de déceler certaines erreurs dès l'étape de conception. Indirectement, les circuits électriques conçus grâce à ce logiciel peuvent être utilisés dans des documentations car le logiciel permet de contrôler la majorité de l'aspect graphique des circuits.

- Voici ci-dessous l'interface du logiciel.

Annexes

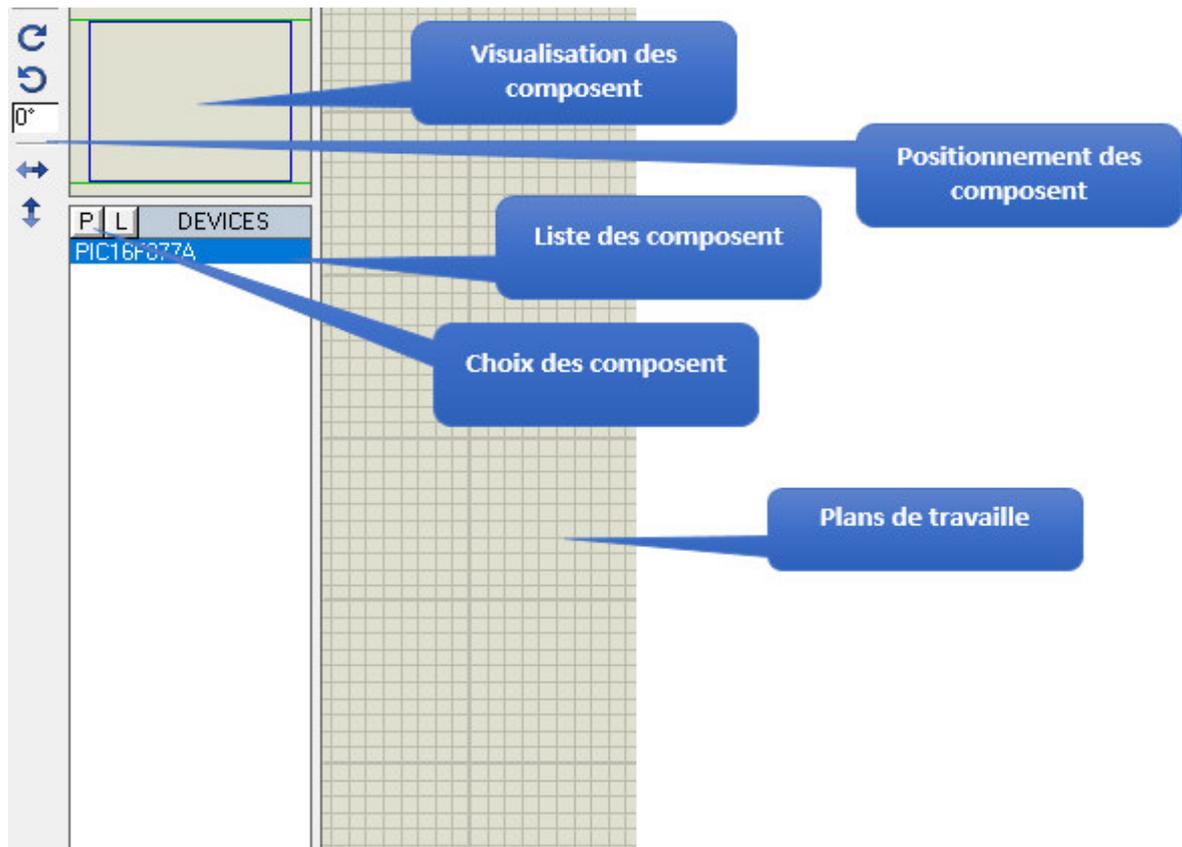


- La barre des tache contient plusieurs raccourcis pour vous aider à g eris votre projet.

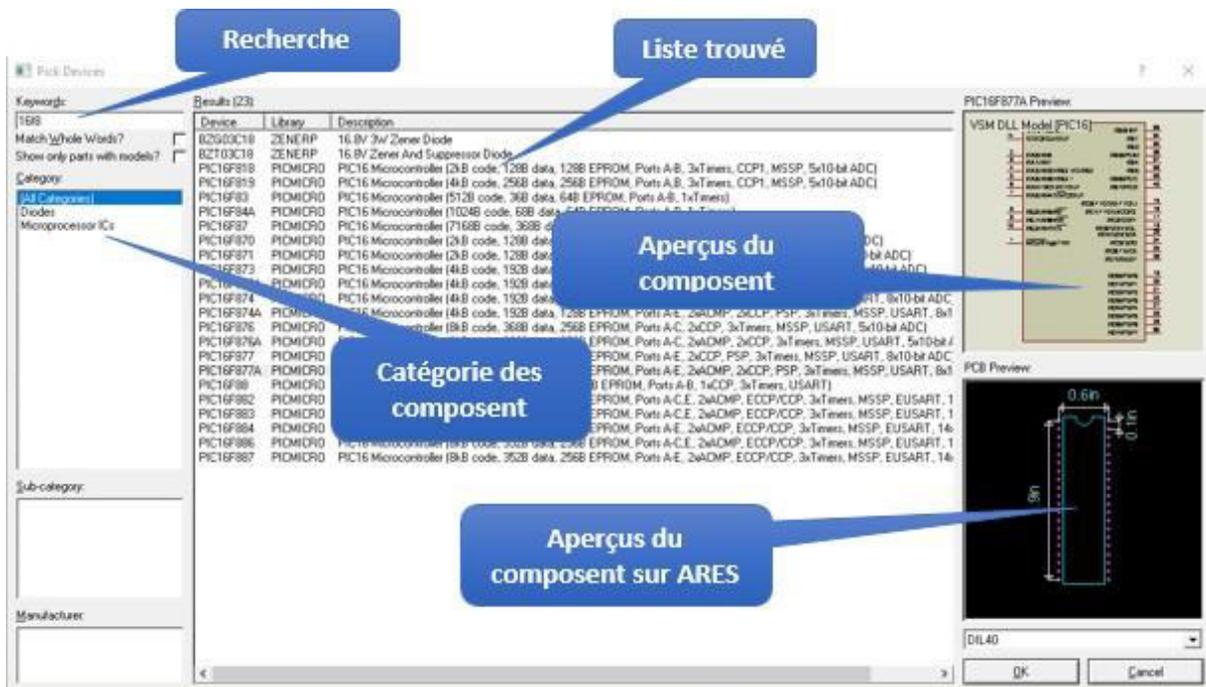


- La barre d'outils est tr s utile pour la conception et la simulation des circuits car elle contient tous les outils susceptibles d' tre utilis s dans la r alit  comme le voltm tre, oscilloscope, source de tension, de courant.....etc, et elle a aussi plusieurs fonctions comme ajouter du texte   notre conception

Annexes



- Après qu'on clique sur l'icône **P** il y'a une fenêtre qui apparais .Cette dernière est destinée à choisir les composants qu'il nous faut.

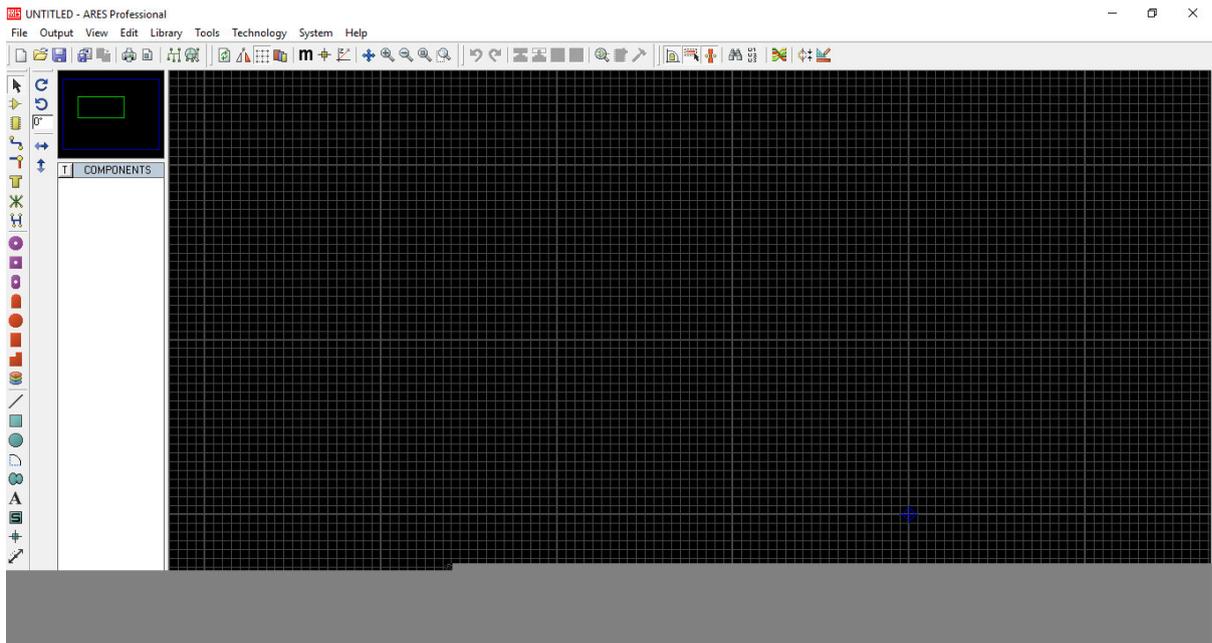


Annexes

- La partie la plus importante de tout le logiciel est le compilateur et l'exécuteur de la simulation

ARES

Le logiciel ARES est un outil d'édition et de routage qui complète parfaitement ISIS. Un schéma électrique réalisé sur ISIS peut alors être importé facilement sur ARES pour réaliser le PCB de la carte électronique. Bien que l'édition d'un circuit imprimé soit plus efficace lorsqu'elle est réalisée manuellement, ce logiciel permet de placer automatiquement les composants et de réaliser le routage automatiquement.

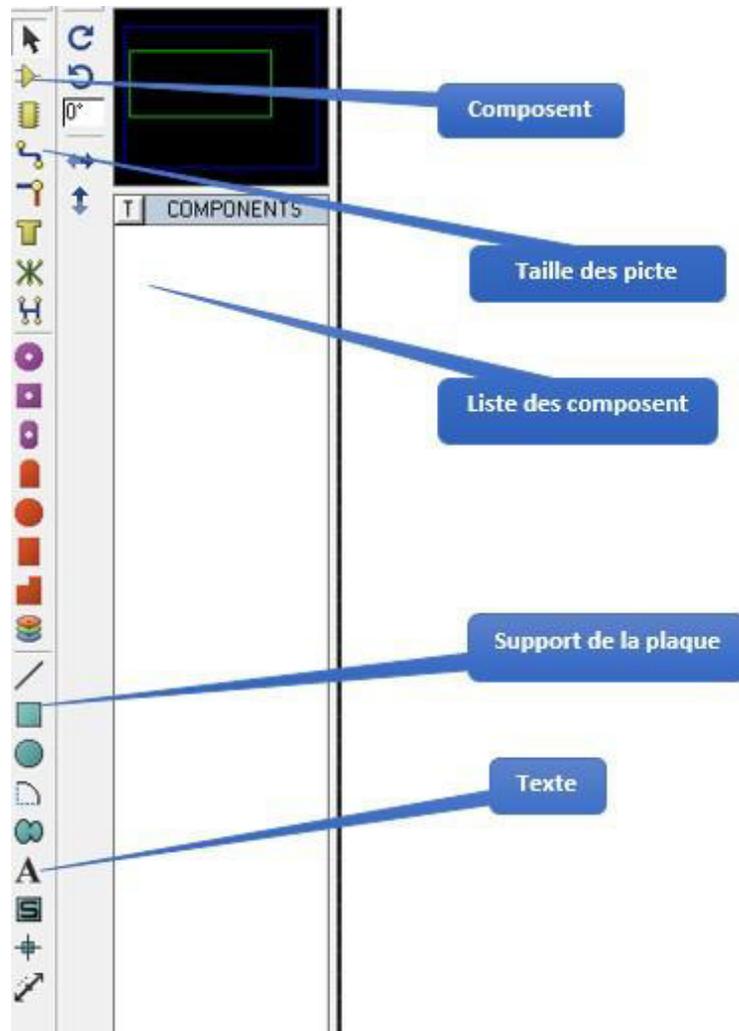


- Barre des taches là ou on choisi le genre de routage qu'on veut soit multicouche ou une seule couche, et plein d'autre chose.



Annexes

- Interface de conception.

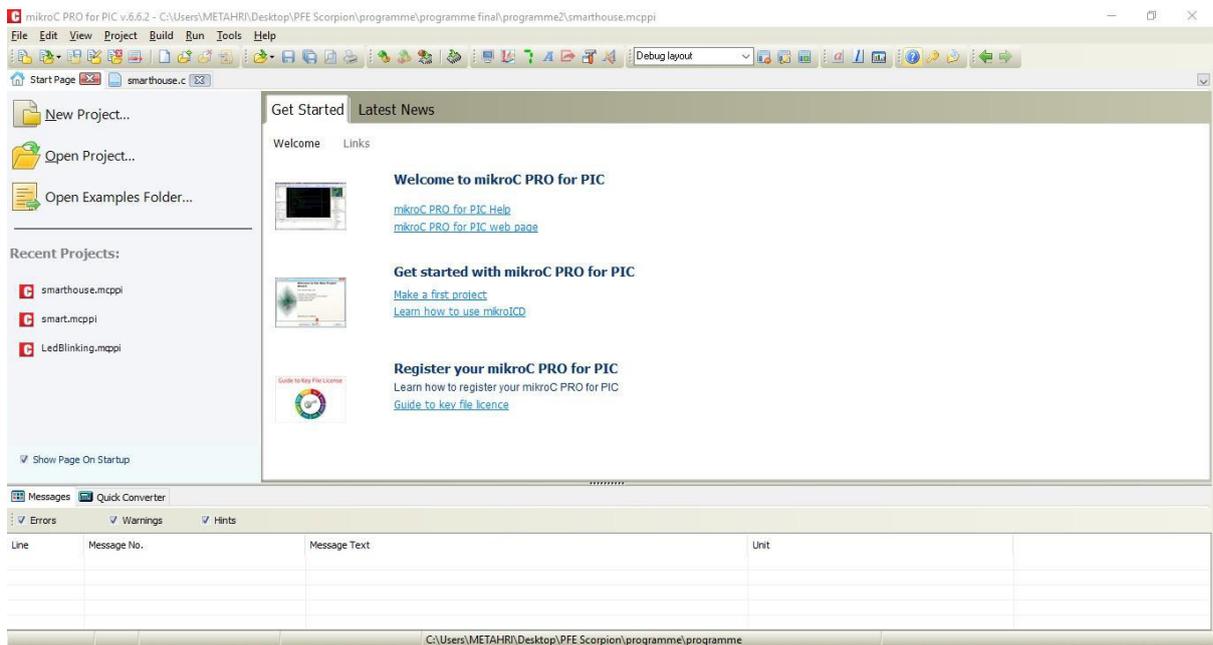


Annexes B

MikroC

Le mikroC PRO pour PIC est un puissant outil de développement riche en fonctionnalités pour les microcontrôleurs PIC. Il est conçu pour fournir au programmeur la solution la plus simple possible pour développer des applications pour les systèmes embarqués, sans compromettre les performances ni le contrôle.

PIC et langage C se combinent bien : PIC est la puce de 8 bits la plus populaire au monde, utilisée dans une grande variété d'applications, et le langage C, apprécié pour son efficacité, est le choix naturel pour le développement de systèmes embarqués. MikroC PRO pour PIC fournit une correspondance réussie avec un IDE hautement avancé, un compilateur compatible ANSI, un large ensemble de bibliothèques matérielles, une documentation complète et beaucoup d'exemples prêts à utiliser.



Barres d'outils

Cette section donne un aperçu des barres d'outils disponibles dans mikroC PRO pour PIC :

- Barre d'outils de fichier
- Modifier la barre d'outils

Annexes

- Advanced Edit Toolbar
- Trouver la barre d'outils
- Barre d'outils du projet
- Barre d'outils de construction
- Barre d'outils Debug
- Styles Toolbar
- Barre d'Outils
- Barre d'outils de visualisation
- Barre d'outils de mise en page
- Barre d'outils d'aide

Barre d'outils de fichier



La barre d'outils de fichier est une barre d'outils standard avec les options suivantes :

Icône	La description
	Ouvre une nouvelle fenêtre d'éditeur.
	Fichier source ouvert pour l'édition ou le fichier image à afficher.
	Enregistrez les modifications pour la fenêtre active.
	Enregistrez les modifications dans toutes les fenêtres ouvertes.
	Aperçu avant impression.
	Impression.

Modifier la barre d'outils



La barre d'outils d'édition est une barre d'outils standard avec les options suivantes :

Icône	La description
-------	----------------

Annexes

Icône	La description
	Annuler la dernière modification.
	Refaire la dernière modification.
	Couper le texte sélectionné dans le presse-papier.
	Copiez le texte sélectionné dans le presse-papier.
	Colle du texte dans le presse-papiers.

Advanced Edit Toolbar



La barre d'outils d'édition avancée comprend les options suivantes :

Icône	La description
	Commenter le code sélectionné ou mettre un commentaire sur une seule ligne s'il n'y a pas de sélection
	Décomment le code sélectionné ou supprimez un commentaire de ligne unique s'il n'y a pas de sélection.
	Sélectionnez le texte du délimiteur de départ vers le délimiteur final.
	Aller au délimiteur final.
	Aller à la ligne.
	Déclenchez les lignes de code sélectionnées.
	Outdent lignes de code sélectionnées.
	Générer un code HTML approprié pour publier le code source actuel sur le Web.

Trouver / Remplacer la barre d'outils



La barre d'outils Rechercher / Remplacer est une barre d'outils standard avec les options suivantes:

Annexes

Icône	La description
	Trouver du texte dans l'éditeur actuel.
	Trouvez l'apparition suivante.
	Trouvez une occurrence antérieure.
	Remplacer le texte.
	Trouver du texte dans les fichiers .

Barre d'outils du projet



La barre d'outils du projet comporte les options suivantes:

Icône	La description
	Nouveau projet.
	Projet ouvert
	Sauvegarder le projet
	Modifier les paramètres du projet.
	Fermer le projet en cours.
	Nettoyer le dossier du projet.
	Ajouter un fichier au projet
	Supprimer le fichier du projet

Barre d'outils de construction



La barre d'outils de construction comprend les options suivantes:

Icône	La description
-------	----------------

Annexes

Icône	La description
	Créer un projet en cours.
	Construisez tous les projets ouverts.
	Créer et programmer un projet actif.
	Démarrez le programmeur et chargez le fichier HEX actuel.

Barre d'outils Debug



La barre d'outils Debug contient les options suivantes:

Icône	La description
	Démarrez le logiciel Simulator ou mikroICD (débogueur in-circuit) .
	Run / Pause Debugger.
	Arrêtez le débogueur.
	Entrer dans.
	Enjamber.
	Sors.
	Run To Cursor.
	ToggleBreakpoint.
	Afficher la fenêtre des points d'arrêt
	Points d'arrêt clairs.
	Afficher la fenêtre de surveillance
	Afficher la fenêtre Chronomètre

Styles Toolbar

La barre d'outils Styles vous permet de modifier facilement les couleurs de votre espace de travail.



Barre d'Outils



La barre d'outils Outils comprend les options par défaut suivantes:

Icône	La description
	Exécuter le terminal USART
	EEPROM
	Graphique ASCII
	Seven Segment Editor.
	Ouvrez l' éditeur de commentaires actifs .
	Menu Options



Astuce : La barre d'outils Outils peut facilement être personnalisée en ajoutant de nouveaux outils dans la fenêtre du menu Options .

Barre d'outils de visualisation



La barre d'outils permet d'accéder au code d'assemblage, à la liste des fichiers et aux statistiques.

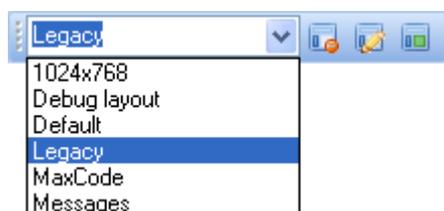
Icône	La description
-------	----------------

Annexes

Icône	La description
	Ouvrez le code d'assemblage dans l'éditeur.
	Ouvrez le fichier d'inscription dans l'éditeur.
	Afficher les statistiques pour le projet en cours.

Barre d'outils de mise en page

La barre d'outils Styles vous permet de personnaliser facilement l'espace de travail à travers plusieurs mises en page IDE différentes.



Icône	La description
	Supprimez la disposition sélectionnée.
	Enregistrez la disposition actuelle.
	Définissez la disposition sélectionnée.

Barre d'outils d'aide



La barre d'outils d'aide fournit l'accès à l'information sur l'utilisation et l'enregistrement des compilateurs:

Icône	La description
	Ouvrez le fichier d'aide.
	Comment s'inscrire.

Annexes C

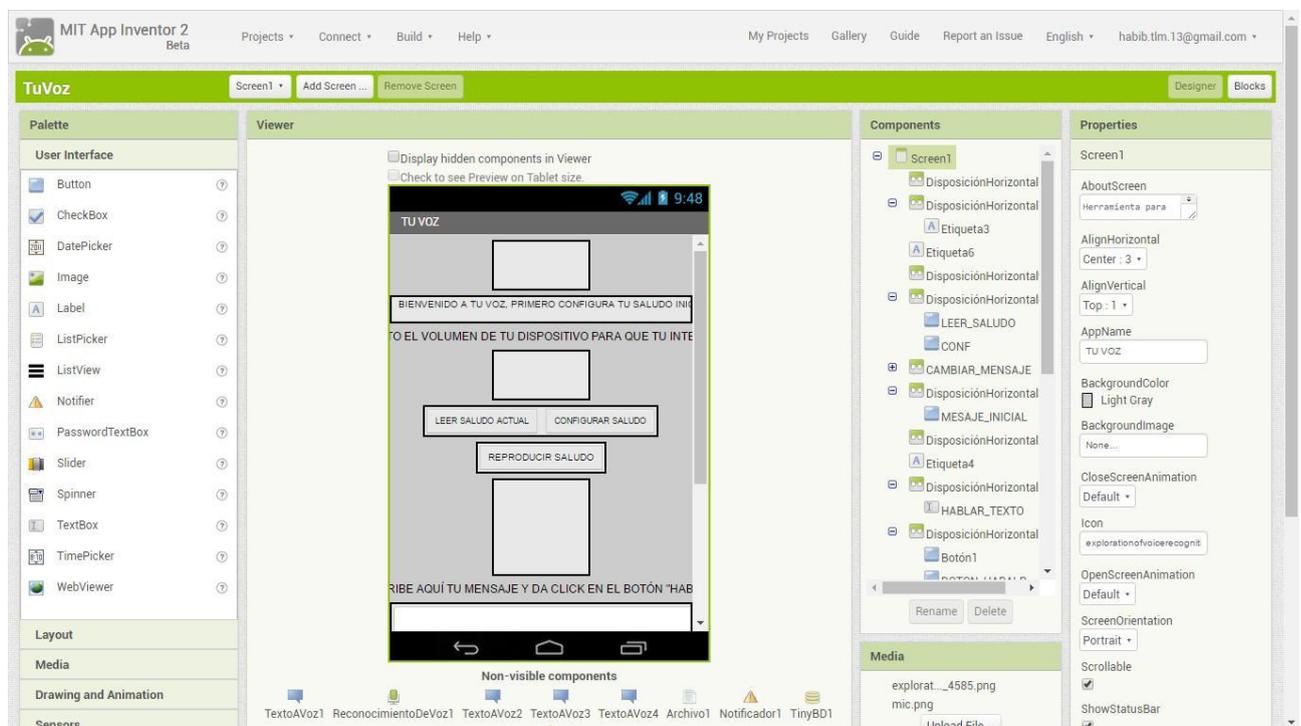
App Inventor

App Inventor pour Android est une application développée par Google. Elle est actuellement entretenue par le Massachusetts Institute of Technology (MIT).

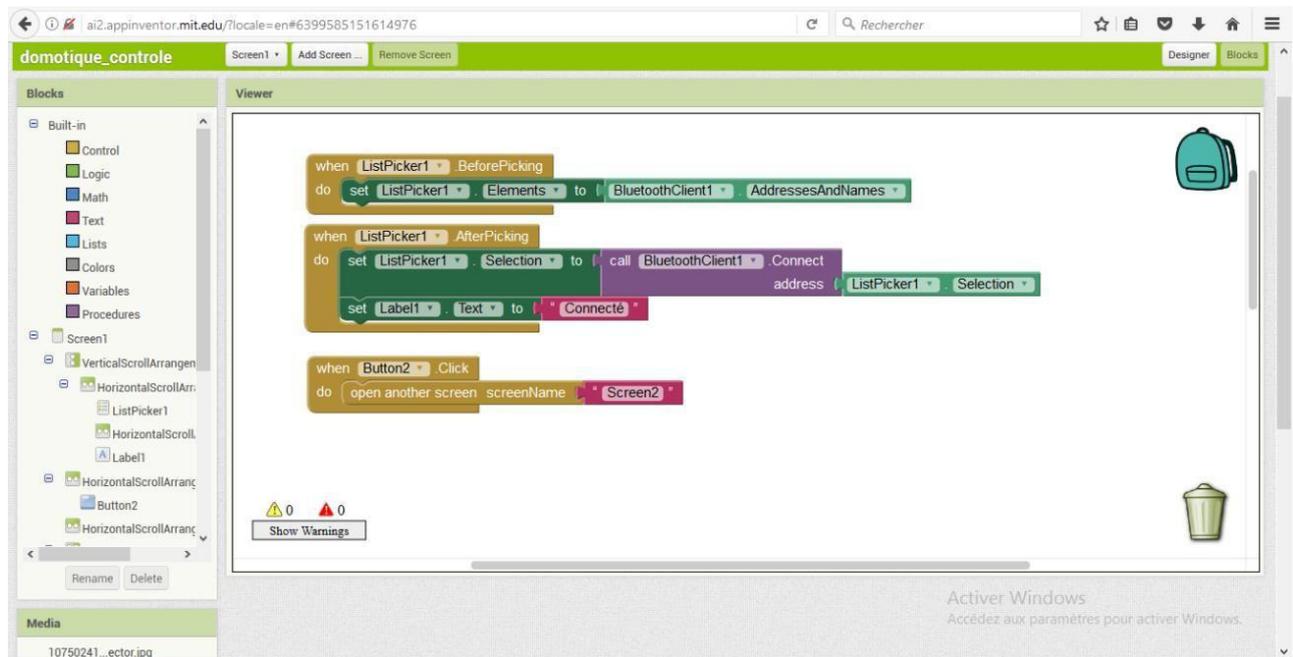
Elle simplifie le développement des applications sous Android et le rend accessible même pour les novices et ceux qui ne sont pas familiers avec les langages de programmation. Elle est basée sur une interface graphique similaire à Scratch et à celle de StarLogo TNG (en). Grâce à son interface entièrement graphique et à l'absence totale de ligne de code, elle est particulièrement adaptée à l'initiation des débutants à la programmation des Applications.

Google publie l'application le 15 décembre 2010 et met fin à son activité le 31 décembre 2011. Dès l'été 2011, Google travaille sur un projet similaire Blockly, développé cette fois en JavaScript. Depuis le retrait de Google, c'est le centre d'études mobiles au MIT qui gère le support technique de cette application sous le nouveau nom "MIT App Inventor".

- **Plateforme du design de l'application.**



Annexes



- **Coté de la programmation en Blocks.**

