

الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

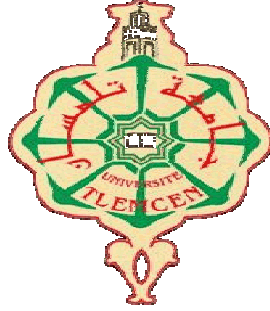
وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة أبي بكر بلقايد- تلمسان

Université Aboubakr Belkaïd- Tlemcen –

Faculté de TECHNOLOGIE



MEMOIRE

Présenté pour l'obtention du **diplôme** de **MASTER**

En : (Télécommunication)

Spécialité : (système de télécommunication)

Par :

BENACHOURA Chouaib

TAHRI Belkhir

Sujet

Contrôle à distance d'appareils électriques d'une maison communicante

Soutenu publiquement, le 25 / 06 / 2019, devant le jury composé de :

M/BENADDA Belkacem

Professeur

Univ. Tlemcen

Président

M/BORSALI Ahmed

Professeur

Univ. Tlemcen

Directeur de mémoire

Riad

M/HADJILA Mourad

MCB

Univ. Tlemcen

Examineur

Remerciements

Nous remercions tout d'abord le grand Allah pour l'achèvement de ce mémoire.

*Nous exprimons nos gratitudees à Monsieur le président de jury d'avoir accepté
examiné ce mémoire.*

*Nous remercions Messieurs les membres de jury, **BENADDA Belkacem** et
HADJILAJ Mourad d'avoir accepté de prendre part à ce jury ainsi que pour
l'intérêt qu'ils l'ont portés à ce travail.*

*Nous remercions Monsieur **BORSALI Ahmed Riad**, notre encadreur, pour ses
conseils et suggestions avisés qui nous aidés à mener à bien ce travail, et d'avoir
rapporté à ce mémoire ces remarques et conseils.*

*Nous remercions notre jury, et tous les professeurs et tous les travailleurs de
département de Télécommunication.*

**BELKHIR
CHOVAIB**

Dédicace

Je dédie ce travail à :

*Mes très chers parents qui ont tous sacrifié pour que je puisse arriver
Jusque-là et qui m'a beaucoup soutenu durant mon cursus.*

*Mes adorables frères et sœurs Qui sont toujours à mes coté et n'ont
jamais cessé de me soutenir.*

*HOUSSAM, MEKHLLOUF, MOHAMMED, MAROUAN,
CHOVAIB, AMMAR, OUAFA et
FAYZA*

Celui qui est toujours à mes coté et qui m'a aidé de toutes les moyennes

Toute ma famille.

Tous mes amis (es).

BELKHIR

Je dédie du fond du cœur ce modeste travail :

*Mes très chers parents qui ont tous sacrifié pour que je puisse arriver
Jusque-là et qui m'a beaucoup soutenu durant mon cursus.*

MOHAMMED, BELKHIR, AMOR, MAROUAN, et SALAH

Et mes plus proche amis et collègues.

Toute ma famille.

Tous mes amis (es).

CHOVAIB

| Sommaire | |
|---|----|
| Introduction générale..... | 1 |
| Partie théorique | |
| Chapitre I : Généralité sur la domotique | |
| I.1 Introduction..... | 3 |
| I.2 Origine et historique de la domotique..... | 3 |
| I.3 Définitions..... | 3 |
| I.4 Critères de la domotique..... | 4 |
| I.4.1 Sécurité..... | 5 |
| I.4.2 Confort..... | 5 |
| I.4.3 Economies d'énergie..... | 5 |
| I.4.4 santé..... | 6 |
| I.4.5 Communication..... | 6 |
| I.5 Les Technologies de la Domotique..... | 7 |
| I.5.1 Domotique sans fil..... | 7 |
| I.5.2 Courants Porteurs..... | 8 |
| I.5.3 Domotique câble..... | 8 |
| I.6 Conclusion..... | 9 |
| Chapitre II : Généralité sur la carte Arduino | |
| II.1 Introduction..... | 10 |
| II.2 Définition du module Arduino..... | 10 |
| II.3 Les gammes de la carte Arduino..... | 10 |
| II.4 Pourquoi Arduino UNO ? | 12 |
| II. 5 La constitution de la carte Arduino UNO..... | 13 |
| II.5.1 Partie matérielle..... | 13 |
| II.5.1.1 Le Microcontrôleur ATmega328 | 13 |
| II.5.1.2 Les sources de l'alimentation de la carte..... | 14 |
| II.5.1.3 Les entrées & sorties..... | 15 |
| II.5.1.4 Les ports de communications..... | 16 |
| II.5.2 Partie programme..... | 17 |
| II.5.2.1 L'environnement de la programmation..... | 17 |
| II.5.2.2 Structure générale du programme (IDE Arduino)..... | 18 |
| II.5.2.3 Injection du programme..... | 18 |
| II.5.2.4 Structure d'un programme en Arduino..... | 19 |
| II.5.2.5 Les étapes de téléchargement du programme | 20 |
| II.6 Les Accessoires de la carte Arduino..... | 21 |
| II.6.1 Communication | 21 |
| II.6.1.1 Le module Arduino Bluetooth..... | 21 |
| II.6.1.2 Le module shield Arduino Wifi..... | 22 |
| II.6.1.3 Le Module XBee..... | 22 |
| II.6.2 Les capteurs | 23 |
| II.6.3 Les drivers(les cartes auxiliaires)..... | 23 |
| II.6.3.1 Des moteurs électriques..... | 23 |
| II.6.3.2 Les afficheurs LCD..... | 24 |
| II.6.3.3 Le relais | 24 |
| II.7 Conclusion | 25 |

| Partie pratique | |
|--|----|
| Chapitre III : Commande d'un appareil électrique par relais | |
| III.1 Introduction..... | 26 |
| III.2 Matérielle et logiciel requis..... | 26 |
| III.3 Configurer le matérielle..... | 27 |
| III.4 Piloter le relais..... | 27 |
| III.5 Conclusion | 29 |
| Chapitre IV : Commande d'une lampe par Wifi | |
| IV.1 Introduction..... | 30 |
| IV.2 Conception et réalisation de la maison intelligente..... | 30 |
| IV.2.1 Outils utilisé..... | 30 |
| IV.2.1.1 Hardware (Matériel utilisé) | 30 |
| IV.2.1.2 Software..... | 31 |
| IV.3 Contrôle de la maison intelligente..... | 32 |
| IV.3.1 Application web..... | 32 |
| IV.3.2 Application Androïde..... | 32 |
| IV.3.2.1 Ses caractéristiques..... | 33 |
| IV.3.3 Commande Vocale | 33 |
| IV.4 Implémentation de circuit | 34 |
| IV.5 Configuration du software..... | 34 |
| IV.5.1 Configuration d'application androïde..... | 34 |
| IV.5.1.1 Téléchargement de code | 36 |
| IV.5.1.2 Exécution finale..... | 37 |
| IV.5.2 Configuration de Web site (<i>Adafruit.IO</i>)..... | 37 |
| IV.5.3 Connexion à Google Assistant..... | 38 |
| IV.6 Conclusion..... | 40 |
| Conclusion générale..... | 41 |
| Annexe..... | 43 |
| Référence Bibliographiques..... | 49 |

Liste des figures

Chapitre I

| | |
|---|---|
| Figure I.1: Concept de la domotique..... | 4 |
|---|---|

Chapitre II

| | |
|--|----|
| Figure II.1: La carte Arduino UNO..... | 12 |
| Figure II.2: Microcontrôleur ATmega328..... | 14 |
| Figure II.3 : Constitution de la carte Arduino UNO..... | 17 |
| Figure II.4 : Interface IDE Arduino..... | 18 |
| Figure II.5: Paramétrage de la carte..... | 19 |
| Figure II.6: Structure d'un programme en Arduino..... | 20 |
| Figure II.7: Les étapes de téléchargement du code..... | 21 |
| Figure II.8: Type de modules Bluetooth HC-06 HC06..... | 22 |
| Figure II.9: Module shield wifi..... | 22 |
| Figure II.10: Module XBee..... | 23 |
| Figure II.11: Capteur Arduino..... | 23 |
| Figure II.12: Moteurs électriques..... | 24 |
| Figure II.13: Afficheurs LCD..... | 24 |
| Figure II.14: Relais..... | 25 |

Chapitre III

| | |
|--|----|
| Figure III.1: Module relais(SONGLE)..... | 26 |
| Figure III.2: Image simplificateur de la configuration d'un relais..... | 27 |
| Figure III.3: Photo réelle de notre circuit..... | 29 |

Chapitre IV

| | |
|---|----|
| Figure IV.1: Le module wifi NodeMCU..... | 31 |
| Figure IV.2 : Module de relais à 4 canaux..... | 31 |
| Figure IV.3: Vue globale de fonctionnement de l'application Blynk..... | 33 |
| Figure IV.4 : Réalisation de notre circuit | 34 |
| Figure IV.5 : Création de nouveau projet avec Blynk..... | 35 |
| Figure IV.6 : Configuration des paramètres de bouton..... | 35 |
| Figure IV.7: Capture de tableau de bord de l'application Androïde..... | 36 |
| Figure IV.8 : Téléchargement le code de l'application Blynk..... | 36 |
| Figure IV.9 : Réalisation réel de notre projet..... | 37 |
| Figure IV.10 : Capture de tableau de bord de l'application WEB..... | 38 |
| Figure IV.11 : Configuration de l'applet via IFTTT..... | 39 |

Introduction générale

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.[1]

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 travail porte sur la réalisation d'un contrôleur intelligent permettant de piloter une maison intelligente à partir d'une application à distance en créant un Web Server entre les différents équipements du système domotique.

Ce travail est structuré en quatre chapitres :

Dans le premier chapitre est une présentation générale de la domotique ainsi que des secteurs d'application et les différents types de technologies utilisées.

Le deuxième chapitre est consacré à la description de la carte Arduino, son langage de programmation, sa construction, son principe de fonctionnement.

Le troisième chapitre est basé sur la commande d'une lampe à l'aide d'un relais.

Le quatrième chapitre est divisé en deux parties, la première est basée sur la commande d'une lampe par wifi à l'aide d'une application androïde installée sur Smartphone et la deuxième est basée sur la reconnaissance vocale pour allumer ou éteindre un appareil électronique.

Chapitre I

Généralité sur la domotique

I.1 Introduction

La domotique regroupe les technologies de l'informatique et l'électronique et des télécommunications pour améliorer le confort, la gestion d'énergie et la sécurité d'une maison ou d'un endroit public, la domotique permet par exemple d'optimiser l'utilisation des lumières et des équipements pour réduire la consommation de l'énergie.

I.2 Origine et historique de la domotique

L'ajout de la domotique dans des bâtiments a débuté dans la deuxième moitié des années 1980. À cette époque, de nouvelles technologies voient le jour et prennent de l'ampleur, telles que l'électronique, l'informatique et les télécommunications. De plus en plus d'éléments de la maison deviennent automatisés, tels que les appareils de chauffage ou les portes. La domotique fait alors son arrivée dans une optique d'économie d'énergie et de contrôle des actions dans les bâtiments. Toutefois, les coûts de la domotique dans les années 80 ne sont pas accessibles à la majorité de la population et seulement les plus nantis profitent de cette technologie.

En 1990, l'arrivée de l'Internet dans les domiciles, ainsi que les débuts du Wifi quelques années plus tard, viennent aussi donner un coup de pouce au développement de la domotique, qui commence à inclure ces technologies.

À partir des années 2000, la domotique croît de manière très rapide. De nombreuses compagnies se spécialisent dans ces technologies et de plus en plus de professionnels les utilisent dans leur travail, que ce soit les architectes, les ingénieurs.

I.3 Définitions

Le mot domotique a été introduit dans le dictionnaire « le petit Larousse » en 1988. Ce mot a été construit à partir de « Domus », qui signifie « maison » ou « domicile », associé au suffixe « tique » couramment employé pour évoquer le terme des technologies (automatique, électronique, électrique, informatique...). [2]

Pour une définition plus précise, on peut dire que l'on regroupe sous l'appellation « Domotique » l'ensemble des technologies de l'électronique, de l'informatique et des télécommunications utilisées dans les domiciles. La domotique vise à assurer des fonctions de sécurité, de confort, de gestion d'énergie et de communications, qu'on peut retrouver dans la maison. [3]

La domotique offre une simplification qui peut alléger le poids des actions quotidiennes pour les personnes âgées ou handicapées.

Ou tout simplement apporter un confort majeur. Il peut s'agir, par exemple d'automatiser les thermostats des maisons afin que la température puisse être préréglée pour se modifier à certaines heures précises. Il peut aussi s'agir de l'utilisation d'un bouton d'urgence qui permettra d'appeler automatiquement les services d'urgence dès que ce bouton est activé. La figure 1 permet de visualiser ce concept.[4]

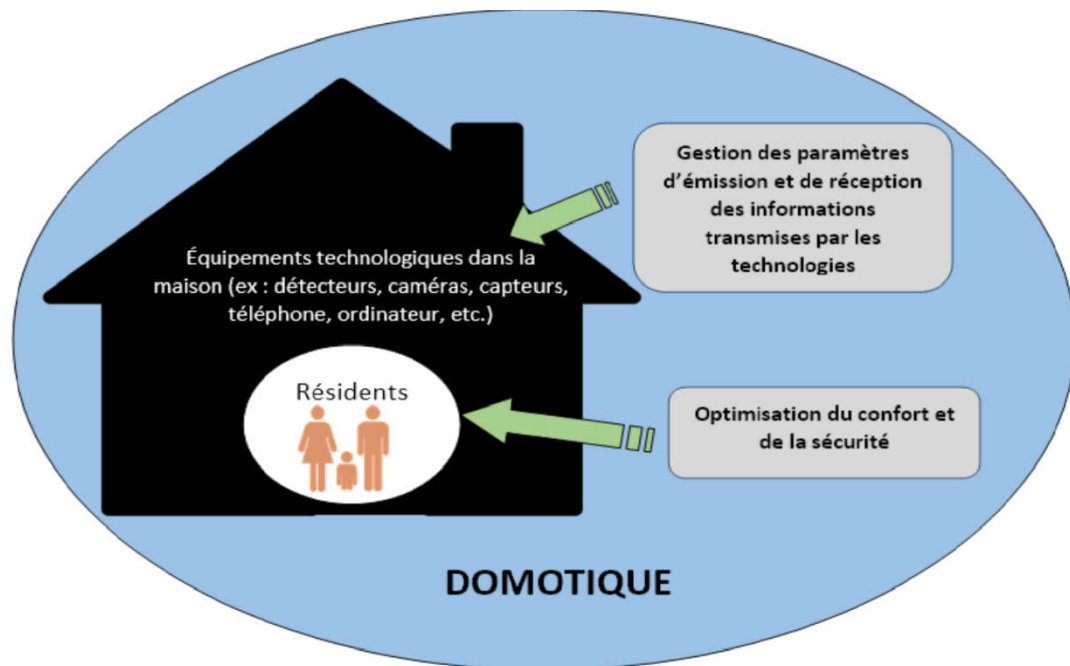


Figure I.1 : Concept de la domotique [4]

I.4 Critères de la domotique

Comprendre la définition de la domotique permet de saisir quels sont ses trois objectifs majeurs et les outils utilisés pour y parvenir :

- Assurer la protection des personnes et des biens en domotique de sécurité.
- Veiller au confort de vie quotidien des personnes âgées, entre autres, en installant une domotique pour les personnes à mobilité réduite.
- Faciliter les économies d'énergie grâce à la réactivité maîtrisée d'une maison intelligente.
- Surveiller la santé (télésanté, télémédecine) et établir la communication (avec un réseau, wifi Bluetooth etc...). [5]

I.4.1 Sécurité

Les détecteurs de fumées et de CO (le monoxyde de carbone, un gaz mortel) préviennent les incendies et les accidents dus à un dysfonctionnement des appareils de chauffage (poêle, cheminée, chaudière). D'un autre côté, des détecteurs de présence ou des caméras permettent de détecter des intrusions dans votre domicile.

La domotique vous permet de sécuriser encore plus votre habitation : alarme, Contrôle d'accès, vidéosurveillance. Combine intelligemment avec le reste de L'installation, votre maison est encore plus sûre. Vous pouvez sécuriser encore plus l'ensemble en ajoutant des alarmes techniques pour être averti en cas de: fuites d'eau, fuite de gaz, surconsommation inhabituelle, etc.... [6]

I.4.2 Confort

Avec une installation domotique, on pourra aujourd'hui avoir une maison vivante et économe. Le fait de rendre la maison intelligente assurera un résultat basse-consommation évident. L'habitat offre aussi un bien-être sur-mesure, avec un confort en permanence.

Ces équipements et cette programmation offrent également et avant tout un confort de vie. Outre la régulation optimale de température, vous pouvez créer des ambiances sonores et lumineuses, automatiser les lumières, les volets, contrôler plusieurs prises électriques d'un seul clic, etc.

La maison connectée, c'est aussi les listes des denrées manquantes dans le réfrigérateur, un pense-bête centralisé sur l'écran central de votre installation domotique et/ou sur votre Smartphone sans oublier la commande de votre cafetière ou de votre TV.

I.4.3 Economies d'énergie

La domotique permet de diminuer jusqu'à 10 % des factures d'énergie. Grâce aux automatismes et à des capteurs, les équipements électriques inter-reliés pilotent au plus juste la consommation énergétique (chauffage, éclairage, eau, ventilation, etc.), tout en gardant sous contrôle le confort des zones occupées.

Le but principal de la domotique est d'éviter le gaspillage en supprimant les dépenses inutiles. Les systèmes de régulation permettent de maîtriser la consommation d'électricité, de gérer le chauffage et la production d'eau chaude sanitaire, avec un

niveau de confort optimal. Un détecteur de présence placé dans chaque pièce, par exemple, commande instantanément l'allumage ou l'extinction des éclairages, la mise en route ou l'arrêt du chauffage, etc.

Au jardin par exemple, l'arrosage s'automatise et le détecteur crépusculaire se charge d'allumer les lumières dès la tombée de la nuit et ainsi de lancer l'irrigation des plantes.

La maison intelligente utilise la programmation domotique via des scénarios qu'on peut déterminer en fonction des besoins spécifiques, évitant les pertes thermiques inutiles et palliant les risques d'oubli ou de sécurité.

I.4.4 Santé

La première utilisation concerne la santé. Cette application de la domotique est essentiellement prévue pour le suivi des personnes fragiles (Personnes âgées, Handicapé, Malentendants ou sourds...). On peut imaginer qu'un équipement installé sur une personne ou dans son domicile contrôle un certain nombre de paramètres comme : son rythme cardiaque, sa température corporelle, son taux de glycémie (pour le cas d'un diabétique), la qualité de l'air... On peut également imaginer un capteur sensoriel qui permet de détecter la perte de connaissance. En fonction des résultats mesurés par ces détecteurs, il va être possible de remonter des alertes vers des organismes compétents : services d'urgence, ... ou les proches.

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 scénarios programmés dans l'interface de commande.[7]

I.4.5 Communication

Un système domotique permet la communication non seulement à l'intérieur de la maison, mais aussi à l'extérieur. La technologie Internet interviendra de plus en plus pour la commande à distance par La certains utilisateurs. Vous ne devez même pas être à la maison pour commander vos appareils. Un simple coup de fil ou un sms vous permettra par exemple de régler le chauffage à distance, d'activer une simulation de présence ou de lancer le lave-vaisselle ou le lave-linge. [8]

I.5 Les technologies de la Domotique :

L'essence même d'une installation domotique est la communication entre ses différents éléments.

Pour cela, beaucoup de protocoles sont nés, car chaque constructeur a réalisé son protocole de communication, ce qui a conduit à une situation très complexe. Les protocoles présentés ci-dessous ne sont pas des protocoles propriétaires : la plupart sont standardisés et/ou ouverts.

I.5.1 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.

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 :

- **Zwave:** 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.
- **HomeEasy:** Le HomeEasy, lui, utilise la fréquence 433 MHz qui est réglementée par l'UIT (Union internationale des télécommunications).
- **X2D:** Le X2D est mixte (courant porteur ou radio 868 MHz) convient à la domotique de sécurité et la domotique du chauffage.
- **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.
- **Zigbee :** 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).

I.5.2 Courants Porteurs

Les protocoles utilisant les courants porteurs sont très prisés actuellement, car ils permettent de réduire le câblage et n'ont supposément pas recours aux radiofréquences. Ils présentent néanmoins des désavantages, ils sont très rapidement perturbés par l'environnement électrique (radiateurs, gradateurs...), ils ne franchissent pas, ou très mal, les transformateurs électriques, et le rayonnement électromagnétique des câbles dans lesquels ils passent en font de très bons émetteurs radio.[9]

X10 est un protocole de communication ouvert pour la domotique, surtout utilisé sur le continent présent du côté ouest de l'océan Atlantique. Ce protocole est né en 1975 et utilise le principe du courant porteur. Ce protocole est peu recommandable, à l'heure actuelle, pour une nouvelle installation ; il offre des débits très faibles qui occasionnent des latences fortes (de l'ordre d'une seconde pour l'envoi d'un ordre). Beaucoup d'autres limitations sont présentes et détaillées sur le Web.

I.5.3 Domotique câble

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).

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 conseillé 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.

I.6 Conclusion

Malgré les progrès techniques et dispositifs récents en domotique, ces installations ne sont pas à la portée de tout le monde, à cause des prix, qui restent relativement élevées. Il semble donc qu'il faille attendre encore un certain temps avant que les "maisons intelligentes " soient à la portée de tous. [10]

Actuellement, un nombre réduit de personnes accèdent aux avantages de la domotique. La simplification des circuits et la réduction du prix doit être le principal objectif, pour que la domotique soit large public.

Chapitre II

Généralité sur la carte Arduino

II.1 Introduction

Aujourd'hui, l'électronique programmée remplace l'électronique. On parle aussi d'informatique embarquée ou de système embarquée. Son but est de réduire l'utilisation de composants électroniques et de simplifier les schémas électroniques, réduisant ainsi le coût de fabrication d'un produit. Il en résulte des systèmes performants pour un espace réduit et des systèmes plus complexes.

Depuis que l'électronique existe, sa croissance est fulgurante et continue encore aujourd'hui. L'électronique est devenue accessible à toutes les personnes désireuses : ce que nous allons apprendre dans ce travail est un mélange de programmation et d'électronique. On va en effet parler d'électronique embarquée qui est un sous-domaine de l'électronique et qui a l'habileté d'unir la puissance de la programmation à la puissance de l'électronique.

II.2 Définition du module Arduino

Le module Arduino est un circuit imprimé en matériel libre (plateforme de contrôle) dont les plans de la carte elle-même sont publiés en licence libre dont certains composants de la carte : comme le microcontrôleur et les composants complémentaires qui ne sont pas en licence libre. Un microcontrôleur 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 domestique-éclairage, chauffage...), le pilotage d'un robot, commande des moteurs et faire des jeux de lumières, communiquer avec l'ordinateur, commander des appareils mobiles (modélisme). Chaque module d'Arduino possède au moins un régulateur linéaire +5 V et un oscillateur à quartz 16 MHz (ou un résonateur céramique dans certains modèles). La programmation de cette carte est basée sur l'logiciel IDE Arduino.

II.3 Les gammes de la carte Arduino

Actuellement, il existe plusieurs versions de module Arduino, nous citons quelques un afin d'éclaircir l'évaluation de ce produit scientifique et académique:

- L'Arduino UNO, utilisations microcontrôleur ATmega328 et une puce ATmega8u2 programmée comme un convertisseur série.
- L'Arduino Esplora : ressemblant à un contrôleur visuel de jeu, avec un manche et des sondes intégrées pour le bruit, la lumière, la température, et l'accélération.
- L'extrémité d'Arduino, avec une interface d'USB pour la programmation et l'utilisation d'un Microcontrôleur ATmega8.

- L'Arduino Mini, une version miniature de l'Arduino en utilisant un microcontrôleur ATmega168 de type CMS.
- L'Arduino Bluetooth, avec une interface de Bluetooth pour programmer en utilisant un microcontrôleur ATmega168.
- L'Arduino Nano, une petite carte programme à l'aide porte USB cette version utilisant un microcontrôleur ATmega168 (ATmega328 pour une plus nouvelle version).
- Le LilyPad Arduino, une conception de minimaliste pour permettre une application portable Utilisant un ATmega168.
- Le NG d'Arduino plus, avec une interface d'USB pour programmer et usage d'un ATmega168.
- L'Arduino Diecimila, avec une interface d'USB et utilise un microcontrôleur ATmega168.
- L'Arduino Mega, est équipé d'un ATmega 1280 de type CMS pour E/S additionnel et mémoire.
- L'Arduino Mega2560, utilisations un microcontrôleur ATmega2560, et possède toute la mémoire à 256 KBS. Elle incorpore également le nouvel ATmega8U2 (ATmega16U2 dans le jeu de puces d'USB de révision 3).
- L'Arduino Duemilanove ("2009"), en utilisant un microcontrôleur l'ATmega168 (ATmega328 pour une plus nouvelle version) et actionné par l'intermédiaire de la puissance d'USB/DC.

D'après ces types, nous avons choisi une carte Arduino UNO (carte Basique). L'avantage principal de cette carte est de faciliter la mise en œuvre d'une telle commande qui sera détaillée par la suite.

L'Arduino fournit un environnement de développement basé sur des outils open source en tant qu'interface de programmation. L'injection du programme déjà converti par l'environnement sous la forme d'un code « HEX » dans la mémoire du microcontrôleur se fait très simplement par la liaison USB. De plus, des bibliothèques de fonctions "clé en main" sont également fournies pour le fonctionnement d'entrées-sorties. Cette carte est basée sur un microcontrôleur ATmega 328 et des composants complémentaires. La carte Arduino contient une mémoire en lecture seul de 1 kilo. Il possède 14 entrées/sorties numériques (6 peuvent être utilisées en tant que sortie PWM), 6 entrées analogiques et un cristal à 16 MHz, une

connexion USB, ainsi qu'un bouton de réinitialisation et une prise d'alimentation. La carte est illustrée dans la figure ci-dessous. [11]

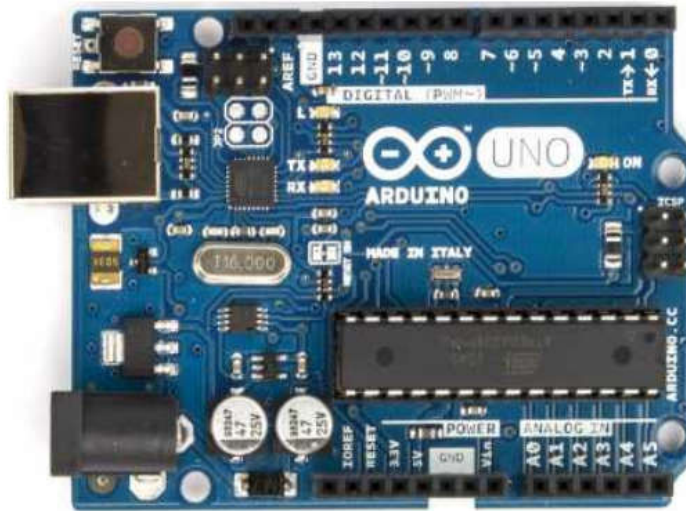


Figure II.1 La carte Arduino UNO [11]

II.4 Pourquoi Arduino UNO ?

De nombreuses cartes électroniques dotées de plates-formes basées sur un microcontrôleur sont disponibles pour l'électronique programmée. Tous ces outils prennent en charge les détails complexes de la programmation et les intègrent dans une présentation facile à utiliser. De la même manière, le système Arduino simplifie la manière de travailler avec les microcontrôleurs tout en offrant aux personnes intéressées plusieurs avantages mentionnés ci-dessous:

- **Le prix (réduits) :** pas trop chère.
- **Multi plateforme :** le logiciel Arduino, écrit en JAVA, fonctionne sur les systèmes d'exploitation Windows, Macintosh et Linux. La plupart des systèmes à microcontrôleurs sont limités à Windows.
- **Un environnement de programmation clair et simple :** l'environnement de programmation Arduino (le logiciel Arduino IDE) est facile à utiliser pour les débutants.
- **Logiciel Open Source et extensible :** le logiciel Arduino et le langage Arduino sont publiés sous licence open source, pouvant être complétés par des programmeurs expérimentés. Le logiciel de programmation de module Arduino est une application JAVA multiplateformes (fonctionnant sur n'importe quel système d'exploitation),

servant d'éditeur de code et de compilateur, et peuvent transférer le programme via la liaison série (RS232, Bluetooth ou USB selon le module).

- **Matériel Open source et extensible** : les cartes Arduino sont basées sur les Microcontrôleurs Atmel ATMEGA8, ATMEGA168, ATMEGA 328, les schémas des modules sont publiés sous une licence créative Commons, et les concepteurs des circuits expérimentés peuvent construire leur propre version des cartes Arduino, les compléter et les mettre à niveau. Même les utilisateurs relativement inexpérimentés peuvent construire la version d'essai de la carte Arduino, dont le but est de comprendre comment elle fonctionne pour économiser sur les coûts.

II.5 La constitution de la carte Arduino UNO

Un module Arduino est généralement basé sur un microcontrôleur ATmega328 cadencé à 16 MHz. Les connecteurs situés sur les bords extérieurs du circuit imprimé permettent de brancher une série de modules complémentaires facilitant la programmation et l'interfaçage avec d'autres circuits. Chaque module comporte au moins un régulateur linéaire 5V et un oscillateur à quartz 16 MHz (ou un résonateur en céramique sur certains modèles). Le microcontrôleur est préprogrammé avec un bootloader qui permet de modifier le programme sans passer par un programmeur.

II.5.1 Partie matérielle

En général, tout module électronique comportant une interface de programmation est toujours basé sur sa construction sur un ou plusieurs circuits programmable.

II.5.1.1 Le Microcontrôleur ATmega328

Un microcontrôleur ATmega328 est un microcontrôleur à une puce créé par Atmel dans la famille des méga AVR (une technologie plus récente, acquise par Atmel en 2016), avec une architecture RISC à 8 bits. Aujourd'hui, en soudant un grand nombre de composants volumineux ; tels que les transistors; les résistances et les condensateurs peuvent tous être logés dans un petit boîtier en plastique noir muni d'un certain nombre de broches dont la programmation peut être effectuée en langage C. la figure I.2 montre un microcontrôleur ATmega 328, qui se trouve sur la carte Arduino. [12]



Le composant CMS



Le composant classique

Figure II.2 Microcontrôleur ATmega328

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

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

II.5.1.2 Les sources de l'alimentation de la carte

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

La carte Arduino Uno peut-être alimentée Soit via la connexion USB (qui fournit 5V) ou à l'aide d'une alimentation externe. La source d'alimentation est sélectionnée automatiquement par la carte. L'alimentation externe (non-USB) peut être soit un adaptateur secteur (pouvant fournir typiquement de 3V à 12V sous 500mA) ou des piles (ou des accus). L'adaptateur secteur peut être connecté en branchant une prise 2.1mm positif au centre dans le connecteur jack de la carte. Les fils en provenance d'un bloc de piles ou d'accus peuvent être insérés dans les connecteurs des broches de la carte appelées Gnd (masse ou 0V) et Vin (Tension positive en entrée) du connecteur d'alimentation.

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

II.5.1.3 Les entrées & sorties

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

- **Interruptions Externes:** Broches 2 et 3. Ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, sur un front montant ou descendant, ou sur un changement de valeur. -Impulsion PWM (largeur d'impulsion modulée): Broches 3, 5, 6, 9, 10, et 11. Fournissent une impulsion PWM 8-bits à l'aide de l'instruction `analogWrite()`.
- **SPI (Interface Série Périphérique):** Broches 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ces broches supportent la communication SPI (Interface Série Périphérique) disponible avec la bibliothèque pour communication SPI. Les broches SPI sont également connectées sur le connecteur ICSP qui est mécaniquement compatible avec les cartes Mega.
- **I2C:** Broches 4 (SDA) et 5 (SCL). Supportent les communications de protocole I2C (ou interface TWI (Two Wire Interface - Interface "2 fils"), disponible en utilisant la bibliothèque `Wire/I2C` (ou TWI - Two-Wire interface - interface "2 fils").
- **LED:** Broche 13. Il y a une LED incluse dans la carte connectée à la broche 13. Lorsque la broche est au niveau HAUT, la LED est allumée, lorsque la broche est au niveau BAS, la LED est éteinte.

Par défaut et contrairement aux entrées/sorties numériques qui ne peuvent prendre que deux états HAUT et BAS, ces six entrées peuvent admettre toute tension analogique comprise entre 0 et 5 Volts.

Pour pouvoir être traitées par le microcontrôleur, ces entrées analogiques sont prises en charge par un CAN (Convertisseur Analogique Numérique ou ADC pour Analogue Digital Convertir) dont le rôle est de convertir l'échantillon de tension V_E en une grandeur numérique binaire sur 10 bits.

La carte Arduino UNO intègre un fusible qui protège le port USB de l'ordinateur contre les surcharges en intensité (le port USB est généralement limité à 500mA en intensité). Bien que la plupart des ordinateurs aient leur propre protection interne, le fusible de la carte fournit une couche supplémentaire de protection. Si plus de 500mA sont appliqués au port USB, le fusible de la carte coupera automatiquement la connexion jusqu'à ce que le court-circuit ou la surcharge soit stoppé. [13]

II.5.1.4 Les ports de communications

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

On utilise (RX) pour recevoir et (TX) transmettre (les données séries de niveau TTL). Ces broches sont connectées aux broches correspondantes du circuit intégré ATmega328 programmé en convertisseur USB – vers – série de la carte, composant qui assure l'interface entre les niveaux TTL et le port USB de l'ordinateur.

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

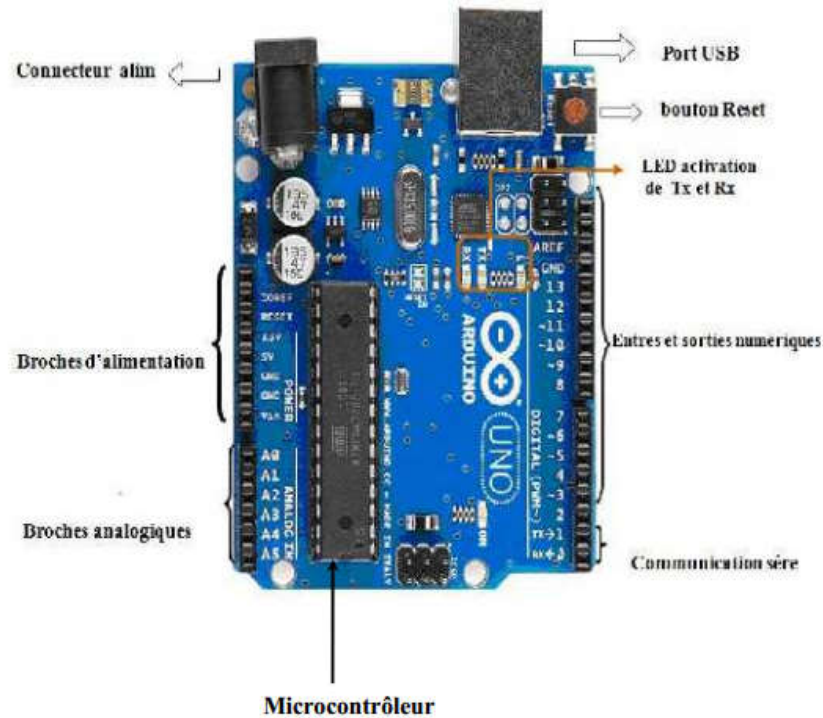


Figure II.3 : Constitution de la carte Arduino UNO [11]

II.5.2 Partie programme

Une telle carte d'acquisition qui se base sur sa construction sur un microcontrôleur doit être dotée d'une interface de programmation comme est le cas de notre carte. L'environnement de programmation open-source pour Arduino peut être téléchargé gratuitement (pour Mac OS X, Windows, et Linux).

II.5.2.1 L'environnement de programmation

Le langage de programmation utilisé est le C++, compilé avec `avr-g++ 3`, et lié à la bibliothèque de développement Arduino, permettant l'utilisation de la carte et de ses entrées/sorties. La mise en place de ce langage standard rend aisé le développement de programmes sur les plates-formes Arduino, à toute personne maîtrisant le C ou le C++.

Une fois, le programme tapé ou modifié au clavier, il sera transféré et mémorisé dans la carte à travers de la liaison USB. Le câble USB alimente à la fois en énergie la carte et transporte aussi l'information ce programme appelé IDE Arduino. [14]

II.5.2.2 Structure générale du programme (IDE Arduino)

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

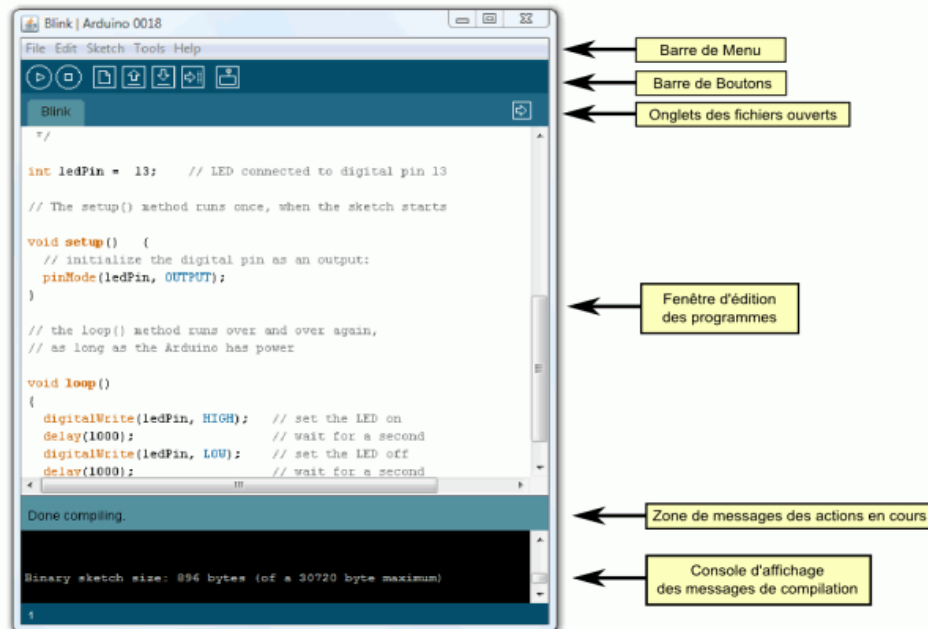


Figure II.4 Interface IDE Arduino [12]

II.5.2.3 Injection du programme

Avant d'envoyer un programme dans la carte, il est nécessaire de sélectionner le type de la carte (Arduino UNO) et le numéro de port USB (COM 3) comme à titre d'exemple cette figure suivante.

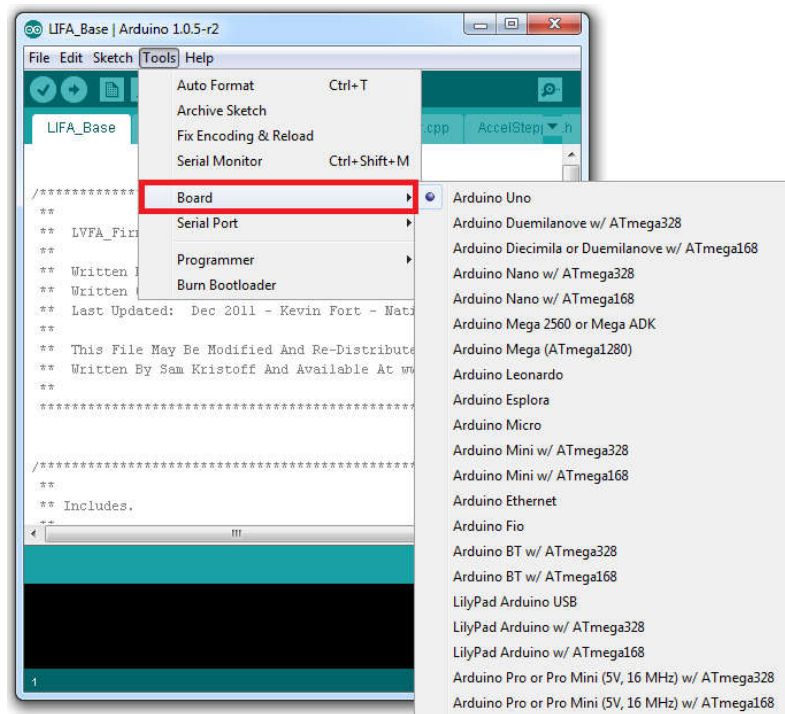


Figure II.5 Paramétrage de la carte

II.5.2.4 Structure d'un programme en Arduino

Un programme utilisateur Arduino est une suite d'instructions élémentaires sous forme textuelle, ligne par ligne. La carte lit puis effectue les instructions les unes après les autres, dans l'ordre défini par les lignes de code. La structure d'écriture d'un programme sous Arduino est de la forme suivante :

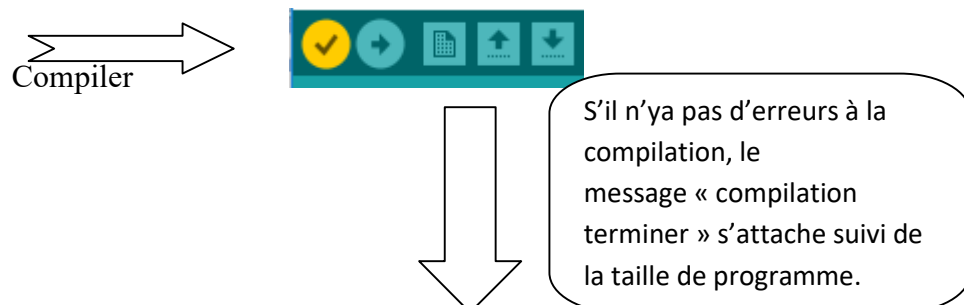


Figure II.6 Structure d'un programme en Arduino [12]

II.5.2.5 Les étapes de téléchargement du programme

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

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



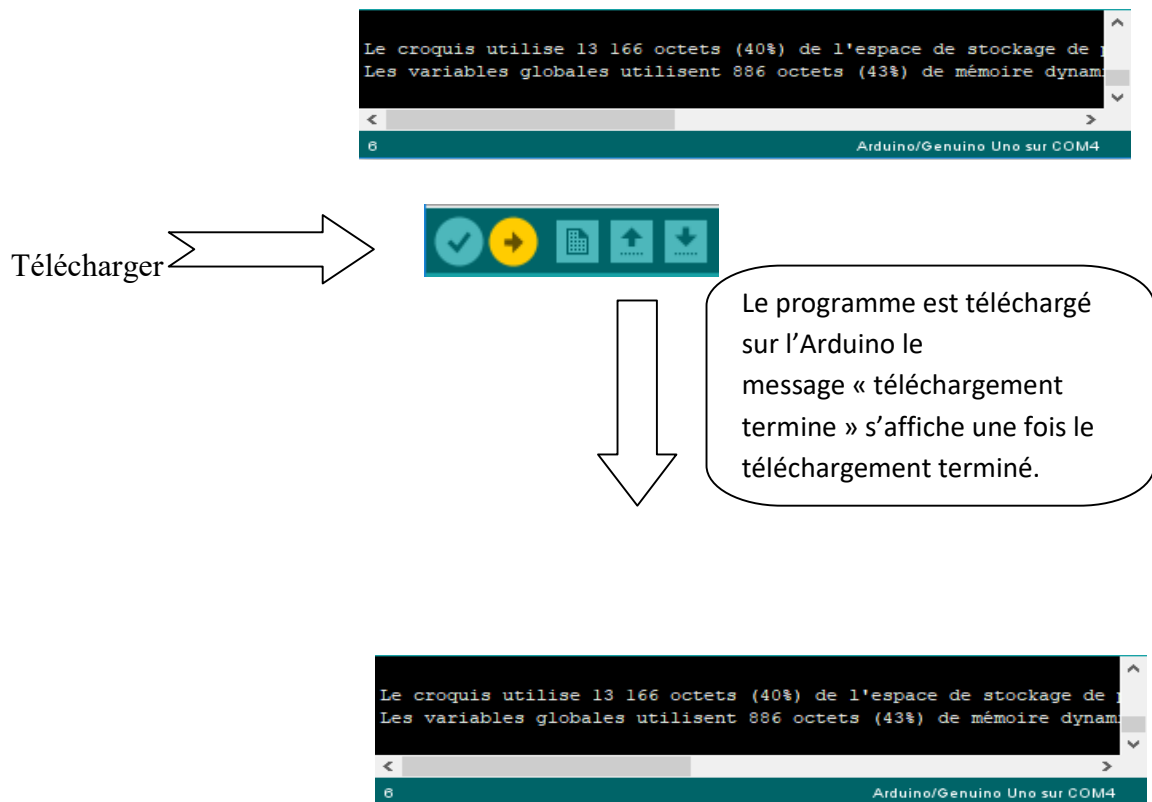


Figure II.7 Les étapes de téléchargement du code

II.6 Les Accessoires de la carte Arduino

La carte Arduino est généralement associée aux accessoires qui simplifient les réalisations.

II.6.1 Communication

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

II.6.1.1 Le module Arduino Bluetooth

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

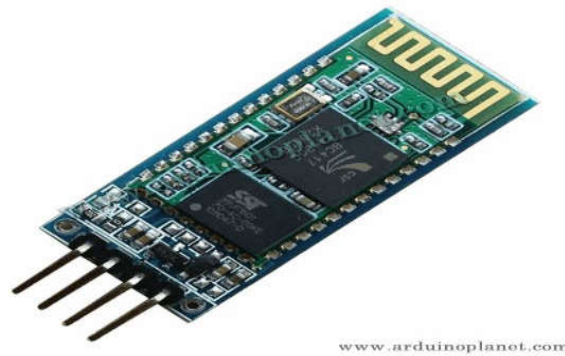


Figure II.8 : Type de modules Bluetooth HC-06 HC06 [11]

II.6.1.2 Le module shield Arduino Wifi

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

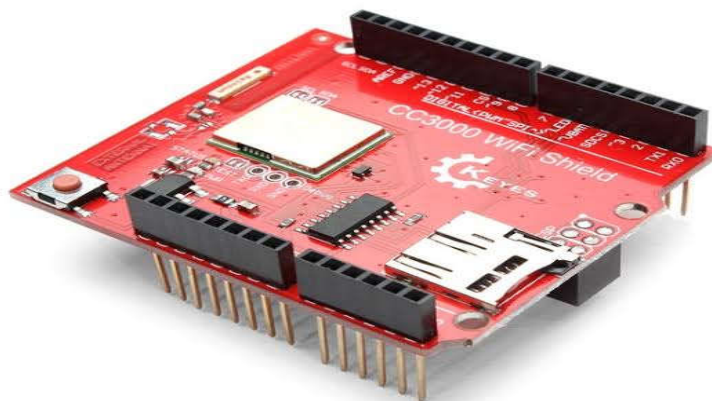


Figure II.9 Module shield wifi [11]

II.6.1.3 Le Module XBee

Technologie basée sur le protocole ZigBee, permettant de communiquer par ondes radio de faible puissance en s'appuyant sur la norme IEEE 802.15.



Figure II.10 Module XBee [11]

II.6.2 Les capteurs

Un capteur est une interface entre un processus physique et des informations manipulables. Il ne mesure rien, mais fournit des informations en fonction de la sollicitation à laquelle il est soumis. Il fournit ces informations grâce à un système électronique auquel il est associé.

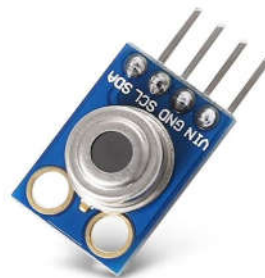


Figure II.11 Shield Arduino [11]

II.6.3 Les drivers (les cartes auxiliaires)

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

II.6.3.1 Des moteurs électriques

Un moteur est un convertisseur électromécanique permettant la conversion bidirectionnelle d'énergie entre une installation électrique parcourue par un courant continu et un dispositif mécanique. Les moteurs ont ainsi la particularité de pouvoir fonctionner dans les 2 sens, suivant la manière dont le courant lui est soumis.



Figure II.12 Moteurs électriques [11]

II.6.3.2 Les afficheurs LCD

Les afficheurs LCD sont devenus indispensables dans les systèmes techniques qui nécessitent l'affichage des paramètres de fonctionnement. Ces Afficheurs permettent d'afficher des lettres, des chiffres et quelques caractères spéciaux. Les caractères sont prédéfinis.

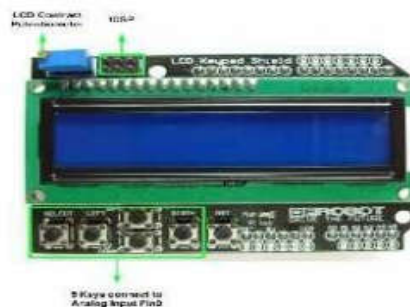


Figure II.13 Afficheurs LCD [11]

II.6.3.3 Le relais

C'est un élément clé en domotique étant donné qu'il permet de commander des appareils consommant beaucoup d'énergie (lampes) à partir d'une carte Arduino.



Figure II.14 Relais [11]

II.7 Conclusion

Dans ce chapitre, nous avons bien détaillée la carte d'acquisition : Arduino, puis nous avons cité des différents types de cette dernière. Ensuite, nous avons expliqué les deux parties essentielles de l'Arduino : la partie matérielle et la partie de programmation ; plus précisément. Nous avons également expliqué le principe de fonctionnement de la carte Arduino sans oublier ses caractéristiques.

Chapitre III

Commande d'un appareil électrique par relais

III.1 Introduction

Dans ce troisième chapitre, nous allons voir comment envoyer une commande vers un relais via la carte Arduino. Le relais est un élément clé en domotique étant donné qu'il permet de commander des appareils consommant beaucoup d'énergie (lampes) à partir d'une carte Arduino. Ce travail d'initiation à base d'une carte Arduino UNO permet de commander une maison intelligente (la domotique).

Après avoir donné dans le chapitre précédent une description théorique sur le module Arduino et son environnement de développement, on va procéder à l'application expérimentale.

III.2 Matérielle et logiciel requis

Premièrement, munissons-nous d'une carte Arduino Uno. En ce qui concerne le relais, notre choix s'est porté sur un modèle 5 V de la marque SONGLE (figure II.1).

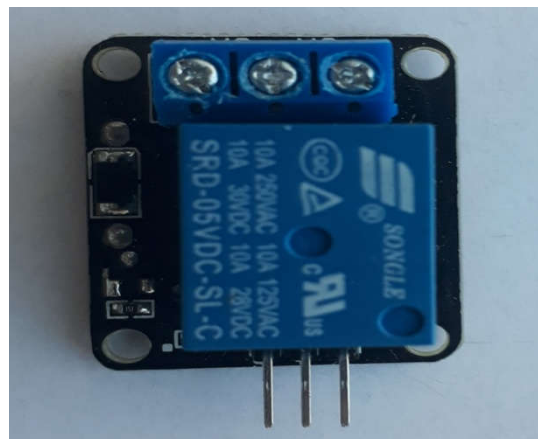


Figure III.1: Module relais (SONGLE)

La marque du relais n'est pas un facteur très important. Il est cependant essentiel de choisir un relais 5 V pour qu'il puisse être piloté directement par la carte Arduino. Nous voulons aussi nous assurer que la puissance maximale, le courant nominal et la tension nominale de notre relais correspondent à ce que nous voulons en faire. Ensuite, nous aurons besoin de fils de raccordement mâle/femelle pour établir les différentes connexions entre la carte Arduino et le relais. La liste des composants est la suivante :

- Carte ArduinoUno.
- Module relais.
- Fils de raccordement mâle/femelle.
- Une lampe.

III.3 Configuration du matériel

Assemblons maintenant notre projet. Les connexions de ce projet sont très simples puisqu'aucune plaque d'essai n'est nécessaire. Voici le schéma du projet (la plaque d'essai apparaît dans l'unique but de visualiser les différentes connexions) :

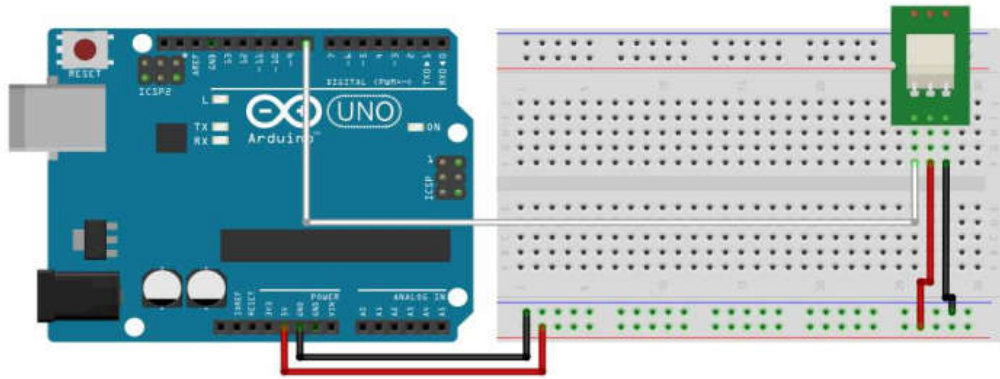


Figure III.2: Image simplificateur de la configuration d'un relais (créé par logiciel fritzing)

Un module relais dispose en principe de trois broches : deux broches pour l'alimentation électrique et la dernière pour le signal de commande permettant de l'activer ou de le désactiver. Connectons tout d'abord la broche VCC du relais à la broche 5 V de la carte Arduino. Puis, connectons la broche GND du relais à la broche GND de la carte.

Enfin, connectons la broche signal du relais (appelée SIG en général) à la broche n° 12 de la carte. À ce stade, nous pouvons également connecter aux sorties du module relais l'appareil que nous désirons commander (ex: une lampe). Cette opération n'est cependant pas recommandée ici car nous voulons vérifier en premier lieu que le relais fonctionne. [15]

III.4 Piloter le relais

Nous allons maintenant écrire un sketch simple pour commander le relais. En guise de test, nous allons simplement activer et désactiver le relais toutes les 5 secondes. Vous trouverez ci-dessous le sketch complet de cette opération :

```
// Simple sketch to control the relay
// Relay pin
const int relay_pin = 12;
void setup() {
  pinMode(relay_pin, OUTPUT);
}
```

```
}  
void loop() {  
  // Activate relay  
  digitalWrite(relay_pin, HIGH);  
  // Wait for 5 seconds  
  delay(5000);  
  // Deactivate relay  
  digitalWrite(relay_pin, LOW);  
  // Wait for 5 seconds  
  delay(5000);  
}
```

Étudions maintenant ce code. Premièrement, on indique sur quelle broche le relais est connecté :

```
const relay_pin = 12;
```

Ensuite, dans la fonction setup() du sketch, on configure cette broche en tant que sortie afin qu'elle puisse recevoir nos commandes :

```
pinMode(relay_pin, OUTPUT);
```

Après ça, dans la fonction loop() du sketch, nous activons dans un premier temps le relais grâce à la commande digitalWrite() :

```
digitalWrite(relay_pin, HIGH);
```

Il nous faut maintenant instaurer un délai de 5 s :

```
delay(5000);
```

Passé ce délai, on désactive le relais :

```
digitalWrite(relay_pin, LOW);
```

Il nous faut de nouveau insérer un délai de 5s.

Nous pouvons désormais charger le sketch sur la carte Arduino. Le relais émettra alors un clic caractéristique et nous devrions voir le voyant LED du module relais s'allumer. Après 5 s, nous devrions entendre ce même clic marquant la désactivation du relais.



Figure III.3 : Photo réelle de notre circuit

Après connexion de l'Arduino à l'ordinateur par câble USB et téléversement du programme on note que la lumière s'allume pendant Cinq secondes et s'éteint pendant Cinq secondes. C'est littéralement une grande LED qui s'allume et s'éteint, contrôlée par l'Arduino.

Nous allons utiliser le relais pour allumer et éteindre une lumière (ampoule ou lampe à tube). Cette étape va nous guider pour le chapitre suivant, en se basant sur ce travail, nous allons introduire dans un premier un application Androïde pour contrôler cette lampe, puis un application vocal.

III.5 Conclusion

Dans ce chapitre, nous avons réalisé un circuit de contrôle d'une lampe à l'aide d'un relai et d'une carte Arduino. Le chapitre suivant est consacré au contrôle à distance d'un appareil électronique à l'aide d'un Smartphone.

Chapitre IV

Commande d'une lampe par Wifi

IV.1 Introduction

Ces dernières années, la technologie a dominé le monde et a imprégné tous les domaines de notre vie quotidienne, jour après jour. Il contrôle nos vies. Elle a connu un développement terrible au fil du temps, ce qui l'a traduit est l'apparition des maisons intelligentes, des téléphones intelligents et des voitures intelligentes....

L'intelligence qui distingue ces choses est la possibilité de communication entre elles via Internet. Nous pouvons donc l'utiliser à tout moment et n'importe où, on parle alors d'Internet des objets.

Smart House (maison intelligent) 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).

Dans ce chapitre, on a développé une application web qui communique avec un serveur web qui est appelé dans ce domaine « serveur domotique » afin de contrôler la maison même hors réseau interne; ainsi qu'une application Androïde qui supporte la commande vocale.

IV.2 Conception et réalisation de la maison intelligente

IV.2.1 Outils utilisé

IV.2.1.1 Hardware (Matériel utilisé)

Pour la partie hardware de ce projet nous avons besoin de :

a. Module wifi (NodeMCU)

Il existe de nombreux modules wifi pour Arduino. Nous avons utilisé le module wifi NodeMCU qui est le seul à avoir fonctionné sans problème dans notre cas. Ce module est basé sur un microcontrôleur ESP8266 cadencé à 80 MHz et exécutant le firmware open source NodeMCU. Le microcontrôleur ESP8266 dispose d'une interface Wifi idéale pour les objets connectés. Des connecteurs latéraux mâles et femelles permettent d'enficher le module sur une plaque de montage rapide voir la annexe (A).

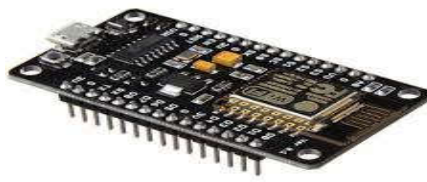


Figure IV.1 : Le module wifi NodeMCU

b. Relais

Nous avons choisi un modèle 5 V de la marque signal qui intègre un relais à une carte. Ce module comporte tous les composants nécessaires au pilotage du relais à partir de la carte Arduino. Voici un aperçu de ce modèle :



Figure IV.2 : Module de relais à 4 canaux

c. Appareil électrique

Ce travail consiste à commander n'importe quel appareil électronique à distance (Téléviseur, poste radio ...), dans notre cas, on va utiliser une ampoule qui produit la lumière.

IV.2.1.2 Software

a. Arduino logiciel

Le logiciel open source Arduino (IDE) facilite l'écriture du code et le téléversement sur la carte. Il fonctionne sous Windows, Mac OS X et Linux. L'environnement est écrit en Java.. Il peut être utilisé avec n'importe quelle carte Arduino. Pour notre projet, on utilise la dernière version v1.8.9.

b. Bibliothèques

❖ ESP8266WiFi.h

La bibliothèque ESP8266WiFi.h comme l'indique son nom est dédiée spécialement pour le circuit intégré ESP 8266, cette dernière développée sur la base du SDK ESP8266, en utilisant la convention de dénomination et la philosophie de fonctionnalité globale de la bibliothèque Wifi Arduino, elle va nous permettre de configurer et programmer nos cartes NodeMCU.

❖ BlynkSimpleEsp8266.h

Pour toutes les plateformes hardware populaires - active la communication avec le serveur et traite toutes les commandes entrantes et sortantes.

❖ Adafruit_MQTT.h

Bibliothèque Arduino pour la prise en charge le protocole MQTT, y compris l'accès au site *Adafruit.IO*. Fonctionne avec les plates-formes ESP8266 Arduino et tout ce qui prend en charge l'interface client d'Arduino voir annexe(B).

IV.3 Contrôle de la maison intelligente

IV.3.1 Application web

Dans notre projet on va utiliser l'une des applications Web les plus connus (site *io.adafruit.com*) qui est un environnement WEB permettant de collecter les données de nos objets connectés (dans des feeds) et leur représentation dans des tableaux (dashboard). Nous pouvons ainsi les visualiser sur notre Ordinateur via un navigateur WEB ou même sur notre Tablette ou Smartphone. [16]

IV.3.2 Application Androïde

Notre Smartphone dans lequel nous exécutons notre application Androïde est connecté au même point d'accès que le box domotique (La carte NodeMCU), l'utilisateur fait appel aux actions en cliquant sur des boutons qui sont des liens exécutant des fonctions spécifiques. On va utiliser l'application Blynk qui a été conçu pour l'Internet des Objets. Il peut contrôler un hardware à distance, il peut afficher des données de capteur, il peut stocker des données, les visualiser et faire beaucoup d'autres trucs cools. La figure IV.3. Donne une vue globale de fonctionnement de l'application Androïde.

IV.3.2.1 Ses caractéristiques

- ❖ Prend en charge la majorité des cartes de développement comme Arduino, esp8266.
- ❖ Facile à utiliser. Et pas limité au réseau Wifi local.
- ❖ Manipulation directe des broches sans écriture de code.
- ❖ Facile à intégrer et ajouter de nouvelles fonctionnalités en utilisant des épingles virtuelles.
- ❖ Nous supposons que cela devrait être plus que suffisant pour construire le projet. [17]

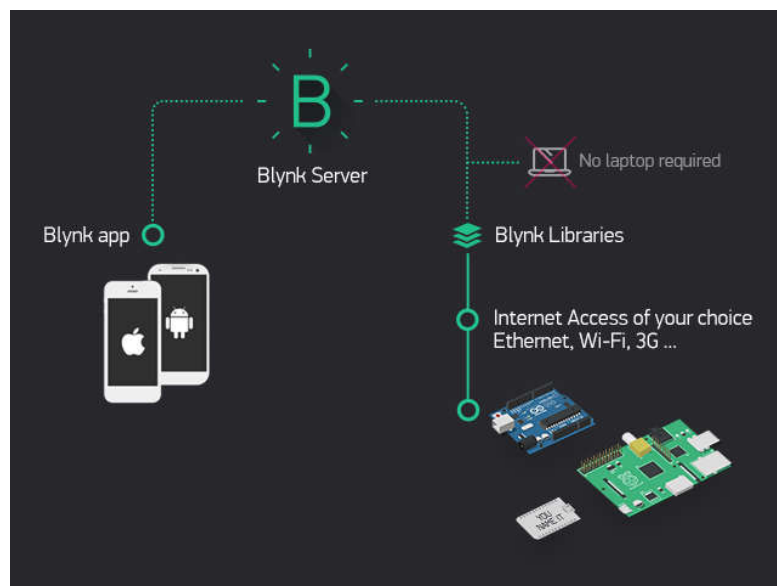


Figure IV.3: Vue globale de fonctionnement de l'application Blynk

IV.3.3 Commande Vocale

Nous avons utilisé la reconnaissance vocale offerte par Google, qui permet de traduire une commande vocale en une phrase (chaîne de caractères), cette phrase va être comparée avec la liste des chaînes de caractères pour chaque action une fois que la correspondance est établie, l'appareil envoie directement la requête d'action.

Dans cette partie, nous nous sommes concentrés sur la reconnaissance vocale car l'intérêt majeur de ce travail est d'offrir une aide nécessaire aux personnes handicapées et les personnes âgées pour assurer le confort et faciliter leur vie quotidienne, où ils peuvent contrôler les appareils électroménagers de n'importe où, uniquement par le son via Smartphone et sans accéder à un interrupteur ou une télécommande.

IV.4 Implémentation de circuit

Une fois les parties Hardware et Software développées, nous avons réalisé notre circuit, qui est le suivant :

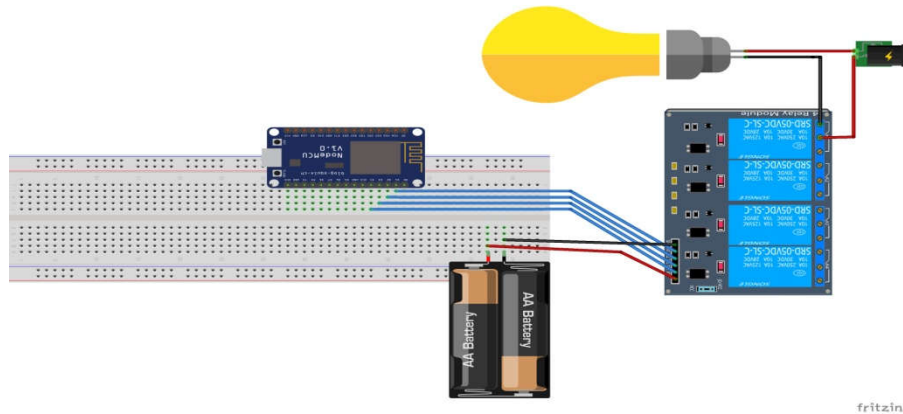


Figure IV.4 : Réalisation de notre circuit (créée avec le logiciel Fritzing)

Connecter les composants à la plaque d'essai. Ensuite relier notre carte NodeMCU à la plaque d'essai comme le montre la figure IV.4.

Les connexions de la carte NodeMCU au relais à 4 canaux sont les suivantes:

- GND → GND(de la batterie)
- D1 → IN1
- 5V → VCC (5V de la batterie)
- D2 → IN2
- D3 → IN3
- D4 → IN4

IV.5 Configuration du software

IV.5.1 Configuration d'application androïde

Installons d'abord l'application Blynk à partir de Google Play Store, puis connectons-nous. Ensuite, cliquons sur Nouveau projet et nous obtiendrons un écran (voir la figure IV.5). [18]

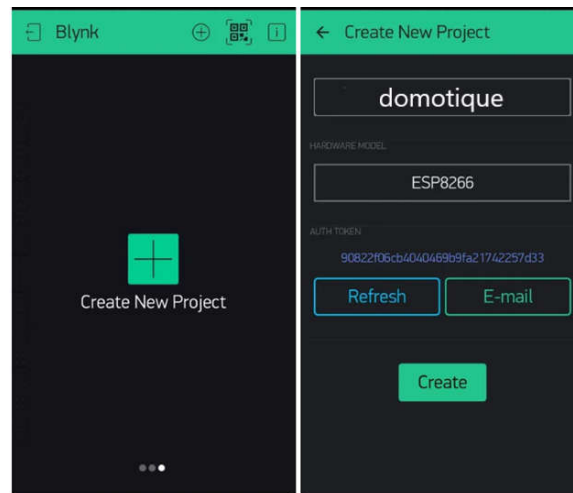


Figure IV.5 : Création de nouveau projet avec Blynk

On fait entrer le nom de notre projet, puis on sélectionne le tableau ESP8266. Ensuite nous allons voir ci-dessous le jeton d'authentification.

Maintenant, nous aurons notre écran de tableau de bord. Cliquons simplement sur le bouton le plus en haut « + » dans le coin droit pour ajouter des widgets à notre projet.

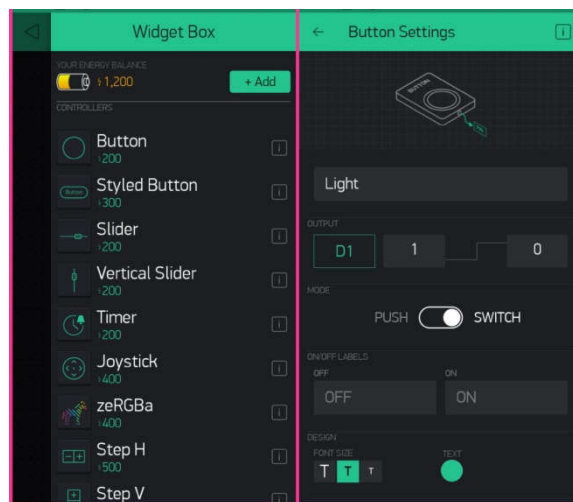


Figure IV.6 : Configuration des paramètres de bouton

Dans ce projet, nous ajoutons un simple bouton, puis configurons ses paramètres en tant que broche Digital (voir la figure IV.7).

Nous pouvons configurer le bouton comme type de bouton poussoir ou comme commutateur. Puis indiquons ensuite le bouton sur ON et OFF dans les paramètres.

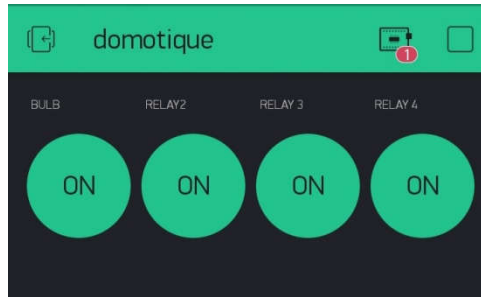


Figure IV.7: Capture de tableau de bord de l'application Androide

IV.5.1.1 Téléchargement de code

Connectons notre Esp8266 Wifi à notre PC et Ouvrons Arduino IDE Puis allons aux *Fichier/ Eamples / Blynk-Boards_Wifi / Esp8266Standalone*.

Ensuit Sélectionnons la carte appropriée (*NodeMCU 1.0*) et la port « com » dans le menu Outils. Nous extrayons le code comme ci-dessous. Enfin, enregistrons le fichier et cliquons sur « Upload ». (Voir la figure IV.8).

```
Serial.begin (9600) ; // Change le débit en bauds à 115200
```

```
Blynk.begin (auth, "ssid", "pass"); // Entrons notre SSID et notre mot de passe Wifi, à l'intérieur de la double citation.
```

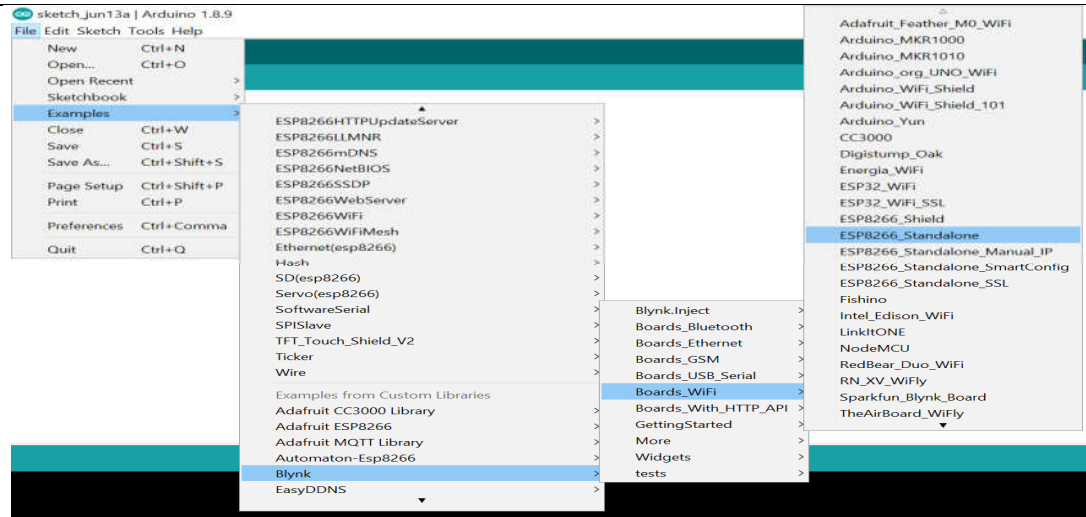


Figure IV.8 : Téléchargement le code de l'application Blynk

IV.5.1.2 Exécution finale

Après téléchargement du code, nous ouvrons l'application Blynk dans le Smartphone, qui se connecte à Internet. Ensuite, nous verrions notre tableau de bord avec un bouton. Et appuyons sur le bouton lecture dans le coin supérieur droit de l'application. Puis appuyons sur le bouton et nous verrions l'ampoule s'allumer.



Figure IV.9 : Réalisation réel de notre projet

IV.5.2 Configuration de Web site (*Adafruit.IO*)

Il existe de multiples technologies et protocoles pour l'internet des objets. *Adafruit.IO* est une plate-forme *IOT* construite autour du protocole *MQTT* (Message Queuing Telemetry Transport). *MQTT* est un protocole de messagerie *publish-subscribe* basé sur le protocole *TCP/IP*, ce protocole permettant aux appareils de publier vers un serveur partagé, appelé *MQTT* Broker. Les clients se connectent à ce broker qui joue ensuite le rôle de médiateur entre les deux appareils. Chaque appareil peut souscrire à des topics particuliers. Lorsqu'un autre client publie un message sur un topic, le broker transfère le message à tous les autres clients qui y ont souscrit.[19]

En plus de fournir le service Broker *MQTT*, *Adafruit.IO* nous permet également de configurer des tableaux de bord nous permettant de manipuler ou d'afficher directement la valeur actuelle de chaque sujet. Comme il est accessible depuis un navigateur Web, il s'agit du hub idéal pour surveiller et contrôler tous nos projets *IOT*.

Après avoir créé notre compte *Adafruit.IO*, nous devrions être redirigés vers l'écran d'accueil. Sélectionnons "*Feed*" dans le menu de gauche. Cliquons sur le menu déroulant Actions et créez un nouveau flux « *Feed* ». Nous allons appeler bouton "*on off*".

Ensuite, nous allons à Tableaux de bord dans le menu de gauche. Cliquons sur le menu déroulant Actions et créons un nouveau tableau de bord. Nous allons appeler "*LightSwitch*". Ouvrons le nouveau tableau de bord et nous devrions être redirigés vers une page presque vierge. En appuyant sur le bouton bleu « + », nous pourrions ajouter de nouveaux composants d'interface utilisateur au tableau de bord. Pour le moment, tout ce dont nous aurons besoin est un bouton bascule, ce qui devrait constituer la première option. Lorsque nous sommes invité à choisir un flux « *Feed* », sélectionnons celui que nous venons de créer et conservons les valeurs par défaut pour le reste des paramètres. C'est tout pour l'instant sur le Web site « *Adafruit.IO* ».

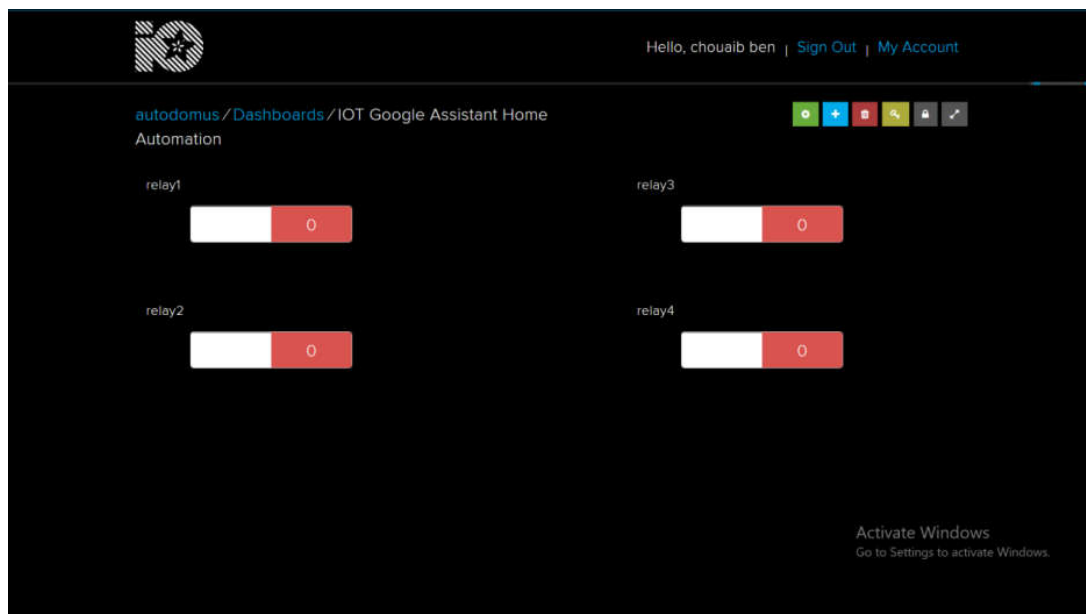


Figure IV.10 : Capture de tableau de bord de l'application WEB

IV.5.3 Connexion à Google Assistant

Dans cette dernière étape, nous allons connecter notre Google assistant à l'Adafruit.IO *MQTT* broker afin de nous permettre de contrôler les éclairages à l'aide de commandes vocales. Pour ce faire, nous utiliserons la plate-forme IFTTT (*If This Then That*), qui permet à des centaines de services différents de déclencher des actions dans divers autres services.

Une fois que nous avons configuré notre compte et que nous avons jeté un coup d'œil, nous devons nous rendre sur la plate-forme de l'éditeur pour créer nos propres applets. Sélectionnons **"Privé"** dans le menu de gauche, puis cliquons sur le bouton bleu **"Nouvel applet"**. Cela nous mènera à l'éditeur d'applet, où nous choisirons les déclencheurs **"If This"** et les actions suivantes **"Then That"**.

Pour notre déclencheur « *Trigger* », choisissons **"Google Assistant"** comme service, puis sélectionnons "Dites une phrase simple" dans le menu déroulant des déclencheurs « *Trigger* » spécifiques. Cela affichera une nouvelle liste de champs à renseigner, notamment des variantes de la phrase d'activation, la réponse de l'Assistant Google et la langue. Pour mes phrases d'activation, nous allons choisir "turne on the light " et " light on ".

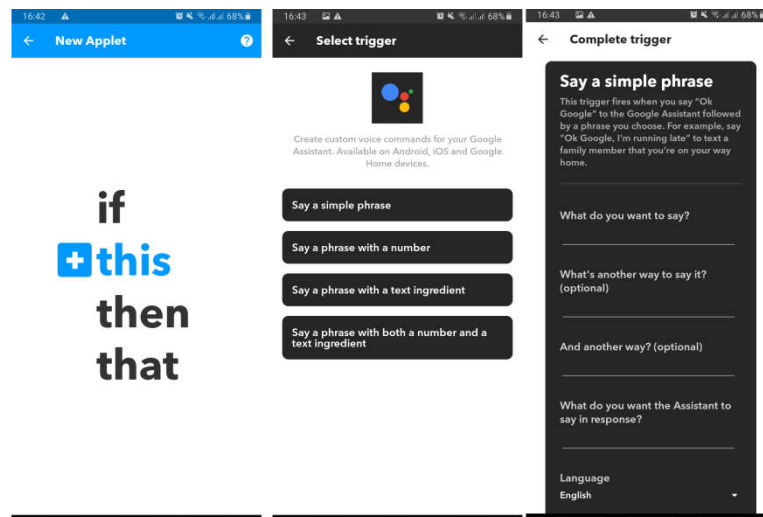


Figure IV.11 : Configuration de l'applet via IFTTT.

La dernière partie de notre applet est l'Action, ce que notre applet fait en réponse au déclencheur « *Trigger* ». Pour le service, choisissons **"Adafruit"**, et pour l'action spécifique, choisissons "Envoyer les données à **Adafruit.IO**". Cela fera apparaître deux champs que nous devons renseigner. Le premier doit être remplacé par le nom du flux « *Feed* » **Adafruit.IO** auquel nous souhaitons envoyer des données, dans ce cas, "on off ". Le deuxième champ est la donnée à envoyer. Pour cette applet, nous enverrons "ON", qui est la chaîne des caractères attendue par notre **ESP8266**.

Une fois que nous avons terminé cette applet, créons une seconde applet pour éteindre la lumière « *off* ». Nous devrions maintenant voir deux applets sur la page de notre plate-forme **IFTTT**. Pour les activer, accédons à la page « *My applets* » du site

principal *IFTTT*, cliquons sur la carte de l'applet, puis cliquons sur régler la bascule « on-off », et la mettre sur "ON".

IFTTT demandera de se connecter à nos comptes *Adafruit.IO* et Google Assistant. Autoriser les comptes à être liés, puis activer également le second applet. Une fois les deux applets activés, la configuration devrait être terminée.

Il ne reste plus qu'à tester le système. Pour ce faire, nous aurons besoin d'un appareil sur lequel Google Assistant est activé. Ceci est intégré aux dernières versions du système d'exploitation Android, ainsi que de la série d'appareils Google Home. Si nous n'avons ni l'une ni l'autre, l'application de messagerie Google Allo, disponible pour Android et iOS, inclut également l'Assistant Google. Lancions l'assistant, assurons-nous qu'il est connecté au compte approprié et prononçons la phrase d'activation que nous avons utilisée à l'étape précédente. Après un délai de 2 à 5 secondes, le voyant de notre carte ESP8266 doit s'allumer ou s'éteindre. Essayons l'autre phrase et assurons-nous que cela fonctionne aussi.

Finalement. Nous avons créé un appareil électrique (ampoule) connectée à Internet et nous le contrôlons avec commande vocale.

IV.6 Conclusion

Nous avons pu concevoir et réaliser notre propre petite maison intelligente grâce à l'utilisation de la programmation embarquée. Nous avons également réalisé différentes applications (web, mobile et vocale) pour pouvoir contrôler la maison à distance avec commande vocale.

Conclusion générale

Conclusion générale

La réalisation des maisons intelligentes a été au cœur de plusieurs projets de recherche lors de ces dernières années. D'énormes progrès ont été accomplis grâce aux avancées en intelligence artificielle, à la miniaturisation de dispositifs électroniques pour la domotique et au développement des réseaux de communication, malgré tout, il reste encore plusieurs défis à surmonter pour rendre possible l'implémentation des maisons intelligentes en situation réelle.

Le système Arduino est un outil pour fabriquer des dispositifs qui peuvent capter et contrôler davantage de choses du monde matériel que votre ordinateur. C'est une plateforme open-source d'électronique programmée qui est basée sur une simple carte à microcontrôleur (de la famille AVR), et un logiciel véritable environnement de développement intégré, pour écrire, compiler et transférer le programme vers la carte à microcontrôleur.

Grâce à ce projet, nous avons appris comment les relais fonctionnent et comment fonctionne le Node MCU. Nous les avons programmés pour fonctionner à l'unisson. Nous avons pu contrôler le relais et à son tour un appareil domestique en utilisant un terminal sur notre ordinateur et notre téléphone intelligent. De plus, nous avons utilisé une application pour contrôler directement le relais grâce à des interfaces graphiques esthétiques. Ensuite, nous avons appris la reconnaissance de la parole et compris son exactitude (ou son inexactitude). Nous l'avons ensuite utilisé pour contrôler le même relais. Enfin, nous avons rassemblé tout ce que nous avons appris pour contrôler l'éclairage de la maison à l'aide du terminal, du téléphone intelligent et de la reconnaissance vocale.

Dans le premier chapitre nous avons fait une présentation générale de la domotique ainsi que des secteurs d'application et les différents types de technologies utilisées.

Dans Le deuxième chapitre nous avons fait une étude approfondie sur la carte Arduino : son langage de programmation, sa construction, son principe de fonctionnement.

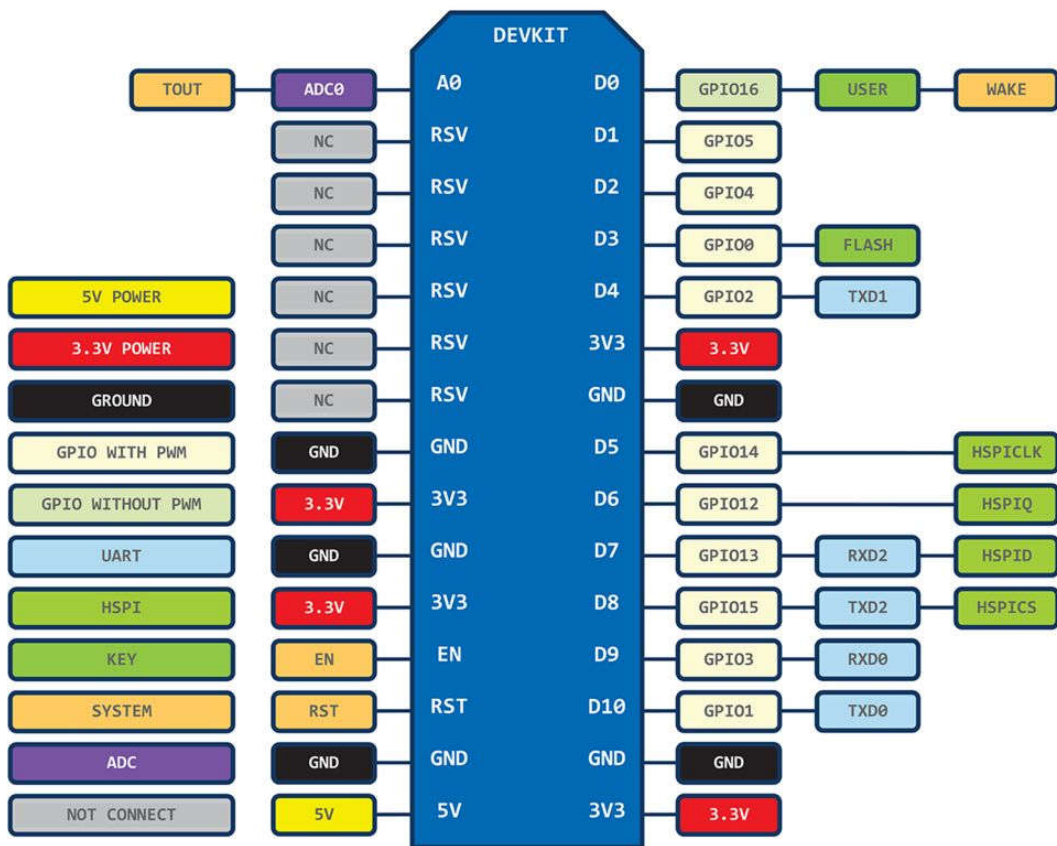
Au cours du troisième chapitre nous avons connectés un relai Arduino à une lampe, nous avons réalisé un programme qui allume et éteint la lampe avec une durée définie pour chaque cycle.

Le quatrième chapitre est consacré à la réalisation d'une commande d'une lampe (et éventuellement d'un autre appareil électronique) par wifi à l'aide d'une application android

installé sur le Smartphone permettant de contrôler l'éclairage via un simple panneau de commande dans un premier temps, ensuite par reconnaissance vocale permettant d'allumer et d'éteindre une lampe par une phrase simple.

Ce travail peut servir d'aide aux personnes handicapés et à mobilité réduite. Il ouvre la voie à plusieurs possibilités, en utilisant toujours la reconnaissance vocale comme élément de commande.

Annexes



NodeMCU ESP8266


```

*****/

#include <ESP8266WiFi.h>

#include "Adafruit_MQTT.h"

#include "Adafruit_MQTT_Client.h"

#define relay1  D1 // GPIO 0

#define relay2  D2// Tx Pin

#define relay3  D3 // GPIO 2

#define relay4  D4// Rx Pin

/***** WiFi Access Point *****/

#define WLAN_SSID  "D-Link"

#define WLAN_PASS  "saidelbah15051990"

/***** Adafruit.io Setup *****/

#define AIO_SERVER  "io.adafruit.com"

#define AIO_SERVERPORT  1883          // use 8883 for SSL

#define AIO_USERNAME  "autodomus"    //Your Username

#define AIO_KEY      "3872832f5f6e4105b6ee1478ac5a781d" // Your API Key

/***** Global State (you don't need to change this!) *****/

// Create an ESP8266 WiFiClient class to connect to the MQTT server.

WiFiClient client;

// or... use WiFiClientSecure for SSL

//WiFiClientSecure client;

// Setup the MQTT client class by passing in the WiFi client and MQTT server and login details.

Adafruit_MQTT_Clientmqtt(&client, AIO_SERVER, AIO_SERVERPORT, AIO_USERNAME, AIO_KEY);

/***** Feeds *****/

// Setup a feed called 'photocell' for publishing.

// Notice MQTT paths for AIO follow the form: <username>/feeds/<feedname>

//Adafruit_MQTT_Publish photocell = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/photocell");

// Setup a feed called 'onoff' for subscribing to changes.

Adafruit_MQTT_Subscribe onoff1 = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME "/feeds/relay1");

Adafruit_MQTT_Subscribe onoff2 = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME "/feeds/relay2");

Adafruit_MQTT_Subscribe onoff3 = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME "/feeds/relay3");

```

```
Adafruit_MQTT_Subscribe onoff4 = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME "/feeds/relay4");

/***** Sketch Code *****/

// Bug workaround for Arduino 1.6.6, it seems to need a function declaration
// for some reason (only affects ESP8266, likely an arduino-builder bug).

void MQTT_connect();

void setup() {

  //Serial.begin(115200);

  delay(10);

  pinMode(relay1, OUTPUT);
  pinMode(relay2, OUTPUT);
  pinMode(relay3, OUTPUT);
  pinMode(relay4, OUTPUT);

  digitalWrite(relay1, HIGH);
  digitalWrite(relay2, HIGH);
  digitalWrite(relay3, HIGH);
  digitalWrite(relay4, HIGH);

  //Serial.println(F("Adafruit MQTT demo"));

  // Connect to WiFi access point.

  // Serial.println(); Serial.println();
  // Serial.print("Connecting to ");
  // Serial.println(WLAN_SSID);
  WiFi.begin(WLAN_SSID, WLAN_PASS);

  while (WiFi.status() != WL_CONNECTED) {

    delay(500);

    // Serial.print(".");

  }

  // Serial.println();
  // Serial.println("WiFi connected");
  // Serial.println("IP address: "); Serial.println(WiFi.localIP());

  // Setup MQTT subscription for onoff feed.
```

```
mqtt.subscribe(&onoff1);

mqtt.subscribe(&onoff2);

mqtt.subscribe(&onoff3);

mqtt.subscribe(&onoff4);

}

uint32_t x=0;

void loop() {

  // Ensure the connection to the MQTT server is alive (this will make the first
  // connection and automatically reconnect when disconnected). See the MQTT_connect
  // function definition further below.

  MQTT_connect();

  // this is our 'wait for incoming subscription packets' busy subloop

  // try to spend your time here

  Adafruit_MQTT_Subscribe *subscription;

  while ((subscription = mqtt.readSubscription(5000))) {

    // to test the tx line data

    if (subscription == &onoff1){

      // Serial.print(F("Got: "));

      // Serial.println((char *)onoff1.lastread);

      uint16_t state1 = atoi((char *)onoff1.lastread);

      // Serial.println(state1);

      digitalWrite(relay1,state1);

    }

    if (subscription == &onoff2){

      // Serial.print(F("Got: "));

      // Serial.println((char *)onoff2.lastread);

      uint16_t state2 = atoi((char *)onoff2.lastread);

      //Serial.println(state2);

      digitalWrite(relay2,state2);

    }

    if (subscription == &onoff3){
```

```
//Serial.print(F("Got: "));

//Serial.println((char *)onoff3.lastread);

uint16_t state3 = atoi((char *)onoff3.lastread);

// Serial.println(state3);

digitalWrite(relay3,state3);

}

if (subscription == &onoff4){

// Serial.print(F("Got: "));

// Serial.println((char *)onoff4.lastread);

uint16_t state4 = atoi((char *)onoff4.lastread);

// Serial.println(state4);

digitalWrite(relay4,state4);

}

}

// ping the server to keep the mqtt connection alive

// NOT required if you are publishing once every KEEPALIVE seconds

/*

if(! mqtt.ping()) {

mqtt.disconnect();

}

*/

}

// Function to connect and reconnect as necessary to the MQTT server.

// Should be called in the loop function and it will take care if connecting.

voidMQTT_connect() {

int8_t ret;

// Stop if already connected.

if (mqtt.connected()) {

return;

}

// Serial.print("Connecting to MQTT... ");
```

```
uint8_t retries = 3;

while ((ret = mqtt.connect()) != 0) { // connect will return 0 for connected

    // Serial.println(mqtt.connectErrorString(ret));

    // Serial.println("Retrying MQTT connection in 5 seconds...");

    mqtt.disconnect();

    delay(5000); // wait 5 seconds

    retries--;

    if (retries == 0) {

        // basically die and wait for WDT to reset me

        while (1);

        }

    }

    // Serial.println("MQTT Connected!");

}
```

Référence
Bibliographiques

Référence Bibliographiques

- [1].HAMID HAMOUCHI ; (juillet 2015) - *Conception et réalisation d'une centrale embarquée de la domotique« Smart Home »*.Université Mohammed V de Rabat.
- [2]. Mathieu GALLISSOT ; (août 2006) - *Modéliser le concept de confort dans l'habitat intelligent : du multi sensoriel au comportement*. Université de Grenoble.
- [3]. Rachelle Duss ; Laure Salamolard ; (Mai 2005) - *La Domotique : La Maison du Futur*. Société de l'information.
- [4]. Marie-ÈVE DUCHARME ; (Décembre 2016) - *Domotique et Ergothérapie*. Université du Québec.
- [5]. MM. Alleguede et Eckert ; (2012) - *Le guide de la domotique*. Ooreka.fr.
- [6]. T.ZATLA ; *Conception et développement d'une application pour le contrôle a distance*. Université ABOU BEKR BELKAID Tlemcen.
- [7]. *La domotique* . Récupéré sur igm : [http://igm.univ-mlv.fr/~dr/XPOSE2007/aessaidi-ndiop_LA-DOMOTIQUE/intro.htm] (consulter le:2007).
- [8]. K.El Yahiaoui et A.Boukoutaya ; (2016) – *Réalisation d'une maison intelligente a base d'Arduino*. Université Mohammed V de Rabat.
- [9].M.T.Benabdallah ; *Maison Intelligente*. Université ABOU BEKR BELKAID.
- [10]. Mr METAHRI Mohammed et El Habib Melle ABDELLI Selma ; (juin 2017) *Smart House*. Université ABOU BEKR BELKAID.
- [11]. KRAMA Abdelbasset et GOUGUI Abdelmoumen ; (juin 2015) - *Etude et réalisation d'une carte de contrôle par Arduino via le système Androïde*. Université Kasdi Merbah Ouargla.
- [12]. C. Tavernier ; (Octobre 2012) - *Arduino applications avancées*. Version Dunod.
- [13]. *Arduino*. Récupéré sur génération robots : [<http://www.generationrobots.com/fr/152-arduino.>] (Consulter le: mars 2015)
- [14]. Jean- Noël ; (novembre 2006) - *livret Arduino en français*. Centre de ressources art sensitif.
- [15]. Marc-Olivier Schwartz ; (2015) - *Arduino pour la domotique*. Version Dunod.
- [16]. IO.ADAFRUIT – *L'internet des objets facile avec adafruit*. Récupéré sur : arduino103 : [<https://arduino103.blogspot.com/2015/05/ioadafruit-linternet-des-objet-facile.html>]. (consulter le : mai 2015).

[17]. *Documentation Français de Blynk*. Récupéré sur github :
[<https://booteille.github.io/blynk-docs-fr>].

[18]. [<https://www.instructables.com/id/Control-Home-Appliances-Using-NodeMCU-and-Blynk-Ap>] (consulter le 2019).

[19]. *Contrôle your project using google assistance*. Récupéré sur instructables
[<https://www.instructables.com/id/Control-Your-Projects-With-Google-Assistant-and-Ad>]
(consulté 2018).

ملخص

تأخذ أتمتة المنزل مكاناً مهماً في حياتنا اليومية، وتسعى التطبيقات الحالية إلى تسهيل الوصول إلى الأجهزة المختلفة. أداة مهمة تأتي في نطاق هذه التطبيقات هي اردوينو.

يتضمن هذا المشروع التحكم عن بُعد في الأجهزة الكهربائية، مثل المصباح والتلفزيون ومكيف الهواء، باستخدام لوحة اردوينو.

لقد أجرينا اتصالاً عن بُعد عبر تقنية واي فاي لأداء نوعين من الأوامر. أولاً، يمكنك التحكم في الأجهزة المنزلية الذكية من خلال تطبيق اندرويد المثبت على الهاتف الذكي وثانياً التحكم في الأجهزة الذكية عن طريق التعرف على الصوت.

الكلمات المفتاحية: أتمتة المنزل، اردوينو، التحكم عن بعد، واي فاي.

Résumé

La domotique prend une place importante dans notre quotidien, les applications actuelles cherchent à faciliter l'accès aux différents appareils. Un outil important rentre dans le cadre de ces applications, c'est Arduino.

Ce projet consiste à contrôler à distance des appareils électriques, tels qu'une lampe, une télévision et un climatiseur, à l'aide d'une carte Arduino.

Nous avons réalisé une connexion à distance via la technologie Wi-Fi afin de réaliser deux types de commandes. Premièrement, contrôler les équipements domestiques intelligents par une application Android installé sur le Smartphone et deuxièmement contrôler les équipements intelligents par reconnaissance vocal.

Mots Clés : Domotique, Arduino, commande à distance, wifi.

Abstract

Home automation takes an important place in our daily lives, current applications seek to facilitate access to different devices. An important tool comes within the scope of these applications is Arduino.

This project involves remote control of electrical devices, such as a lamp, a television and an air conditioner, using an Arduino board.

We made a remote connection via Wi-Fi technology to perform two types of commands. First, control smart home devices through an Android application installed on the Smartphone and secondly control smart devices by voice recognition

Key words: Home automation, Arduino, remote control, wifi.