



الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
وزارة التعليم العالي والبحث العلمي
Ministère de l'Enseignement Supérieure et de la Recherche Scientifique

جامعة وهران 2 محمد بن أحمد
Université d'Oran 2 Mohamed Ben Ahmed
معهد الصيانة و الأمان الصناعي
Institut de Maintenance et de Sécurité Industrielle

Département de Maintenance en Instrumentation

MÉMOIRE

Pour l'obtention du diplôme de Master

Filière : Génie Industriel

Spécialité : Ingénierie de la Maintenance en Instrumentation

Thème

La domotique en utilisant Arduino

Présenté et soutenu publiquement par :

Bouamra mohammed Amine

et

Benmana Reda

Devant le jury composé de :

Nom et Prénom	Grade	Etablissement	Qualité
Mekki Brahim	MCB	IMSI-Univ. D'Oran2	Président
Aissani Nassima	MCA	IMSI-Univ. D'Oran2	Encadreur
Belkhoudja Leila	MCB	IMSI-Univ. D'Oran2	Examineur

Année 2019/2020

Remerciement

Je remercie vont premièrement à Dieu tout puissant pour la volonté, la santé, et la patience, qu'il nous a données durant toutes ces années d'étude.

Nous tiens à remercier grandement nos Encadreur

Madame AISSANI NASIMA

pour sa grande disponibilité et ses précieux conseils.

Nous remerciant également tous les enseignants du département maintenance en instrumentation

d'université d'oran 02

plus spécialement les membres de jury de notre travail.

Enfin, nous adressons nos plus sincères

remerciements à tous

nos amis et collègues qui nous ont toujours

soutenue et encouragée au cours de

la simulation de ce mémoire.

Merci à tous.

Dédicaces

Je dédie ce modeste travail :

Aux êtres qui me sont les plus chers ma mère et mon père. Que Dieu préserve bonne santé et longue vie.

Qui ont tous fait pour m'encourager durant les années de mes études

A mes Grands Parents que Dieu les protège et à toute ma famille

*A mes chers frères et mes chères soeurs
A mes chers amis*

Ainsi que la promotion Master IMI sans oublier tous mes enseignants durant tout mon cursus

Merci à tous.

Bouamra Mohammed Amine

Dédicaces

Je dédie ce modeste travail :

Aux êtres qui me sont les plus chers ma mère et mon père. Que Dieu préserve bonne santé et longue vie.

Qui ont tous fait pour m'encourager durant les années de mes études

A mes Grands Parents que Dieu les protège et à toute ma famille

*A mes chers frères et mes chères soeurs
A mes chers amis*

Ainsi que la promotion Master IMI sans oublier tous mes enseignants durant tout mon cursus

Merci à tous.

Benmana Reda

Sommaire

Introduction générale	1
Chapitre I: Domotique et maison intelligente	
I.1. Introduction	3
I.2. Définitions de la domotique	3
I.3. Historique	4
I.4. Objectif de la domotique	5
I.5. fonctionnement de la domotique	6
I.6. Différents domaines d'application de la domotique	9
I.6.1. Domotique pour le confort	9
I.6.2. Domotique pour l'énergie	10
I.6.3. Domotique pour la sécurité	10
I.6.4. Domotique pour la santé	12
I.7. Recherche de la domotique	12
I.8. Maison intelligente 1	13
I.9. Maison intelligente et la domotique	14
I.9.1. Sécurité et contrôle	14
I.9.2. Appareils électroménagers	15
I.9.3. Éclairage	16
I.9.4. Énergie	16
I.10. Avantages et inconvénients de la domotique	17
I.10. 1. Avantages	17
I.10. 2. Inconvénients	17
I.11. Dernières innovation	18
I.11.1. Détection domestique active avec alerte à distance	18
I.11.2. Actions simultanées détection intrusion	19
I.11.3. Identifier vos visiteurs par interphone ou vidéophone	21
I.12. Conclusion	21

Chapitre II: Arduino et Shields

II.1 Introduction	22
II.2 Domaine d'utilisation et ces applications	22
II.3 Avantages d'Arduino	23
II.4Présentation de la carte	23
II.4.1 Qu'est-ce qu'un microcontrôleur	23
II.4.2 Microcontrôleur ATMEL ATMega328	24
II.4.3 Caractéristiques techniques de l'Arduino UNO	25
II.4.4 Les différents éléments de la carte de commande	27
II.5 Capteurs	27
II.5.1. Présentation des capteurs	28
II.5.2 Communication	31
II.5.3 Module Arduino Bluetooth	31
II.5.4Module composant Arduino wifi	31
II.6 Présentation du logiciel	32
II.6.1. IDE Arduino	32
II.6.2 Les boutons	33
II.6.3 Langage Arduino	33
II.6.4 Fonctionnalité de base	36
II.6.5Les entrées/sorti	36
II.6.6 Gestion du temps	38
II.6.7 Les interruptions	38
II.6.8 Quelques librairies	41
II.7. Conclusion	42

Chapitre III : conception et simulation du système

III.1	Introduction	43
III.2	Conception du système	44
III.3	Simulation	45
III.4	Schéma électronique du montage sur proteus	46
III.5	Description des shields utilisés	47
III.5.1	Capteur de gaz (MQ2)	47
III.5.1.1	Description du capteur	47
III.5.2	Capteur de mouvement (PIR)	48
III.5.2.1	Description du capteur	48
III.5.3	Détecteur de l'éclairage (capteur LDR)	50
III.5.3.1	Description du capteur	50
III.5.4	Le buzzer	52
III.5.4.1	Description du buzzer	52
III.5.5	Led	53
III.5.5.1	Description du led	53
III.5.6	Relais électrique	53
III.5.6.1	Description du relais	53
III.5.7.	Resistance	54
III.5.7.1	Description de résistance	54
III.5.8	Moteur (courant continue)	54
III.5.8.1	Description du moteur	54
III.6.	Conclusion	55

Liste des figures

Figure I.1 : illustre du confort	9
Figure I.2 : détecteur de fumée	10
Figure I.3 : alarme piscine	10
Figure I.4 : alarmes anti-intrusion	11
Figure I.5 : détecteur de gaz butane et propane	12
Figure I.6 : schéma représentatif de fonctionnement général des équipements d'un maison intelligente	14
Figure I.7 : Kit Alarme sans fil Vidéo Domonial Standard HONEYWELL	15
Figure I.8 : Aspirateur robot Wi-Fi Roomba 695 d'iRobot	15
Figure I.9 : Led ampoule sans fil	16
Figure I.10 : moniteur d'énergie connecté	16
Figure I.11 : Détection domestique active avec alerte à distance	19
Figure I.12 : détection intrusion : alarme, éclairage, volets	20
Figure II.1 : Datasheet ATmega328.	24
Figure II.2 : La carte Arduino UNO.	26
Figure II.3 : Schéma simplifié de la carte Arduino UNO.	26
Figure II.4 : Type de modules Bluetooth	31
Figure II.5 : Module shield wifi	31
Figure II.6 : IDE Arduino.	32
Figure II.7 : Présentation des boutons.	33
Figure II.8 : Un code minimal.	33
Figure II.9 : Signal PWM.	37
Figure III.1 : diagramme de cas d'utilisation	44
Figure III.2 : les constituants du système	45
Figure III.3 : logiciel proteus 8 professionnel	46
Figure III.4 : schéma électronique du montage	46
Figure III.5 : Image réelle du capteur de gaz/fumée MQ-2.	47
Figure III.6 : le schéma de simulation de détecteur fuite de gaz	48
Figure III.7 : Image réelle du capteur de mouvement PIR.	49
Figure III.8 : le schéma de la simulation de détecteur de présence	50

Figure III.9 : Image réelle du capteur LDR	50
Figure III.10 : le schéma de la simulation de détecteur de l'éclairage	52
Figure III.11 : Image réelle du buzzer	52
Figure III.12 : Image réelle du led	53
Figure III.13 : Image réelle d'un relais électrique	53
Figure III.14 : Image réelle d'une résistance	54
Figure III.15 : Image réelle d'un moteur	55

Liste des tableaux

Tableau II.1: Différents éléments de la carte de commande.	27
Tableau II.2: Représente différents shields qui connectée avec arduino.	30
Tableau II.3: Les types de base.	34
Tableau II.4: Les structures de contrôle.	35

Introduction générale

Aujourd'hui, l'électronique est de plus en plus remplacée par l'électronique programmée. On parle aussi de système embarquée ou d'informatique embarquée. Son but est de simplifier les schémas électroniques et par conséquent réduire l'utilisation des composants électroniques, réduisant ainsi le coût de fabrication d'un produit. Il en résulte des systèmes plus complexes et performants pour un espace réduit. [1]

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. En effet, La domotique regroupe les technologies de l'électronique, de l'automatique, de l'informatique et des télécommunications permettant d'améliorer le confort, la sécurité, la communication et la gestion d'énergie d'une maison. [2]

D'autre part, la forte augmentation des ventes de smart phone et de tablettes électronique se fait en même temps qu'une adoption rapide par le grand public des technologies de la domotique ainsi que l'autopilotage. Au fond, le smart phone devient une télécommande universelle pour toute la maison et les équipements électriques. Les utilisateurs pourront à terme contrôler à distance un très grand nombre de fonctions sans avoir à tenir compte de la marque ou de l'origine du produit qu'ils pilotent.

Dans ce travail , nous essayons de réaliser une maison intelligente à base d'un arduino UNO pour commander des installations électriques pour la domotique comprenant plusieurs capteurs et actionneurs tels que: Capteur de Gaz, capteur d'éclairage, capteur de présence ... ,et nous travaillons également sur le système de sécurité , un système de sécurité est un ensemble des techniques d'électronique, de mécanique, d'automatismes et d'informatique qui sont utilisées dans les propriétés publiques ou privés tel que les entreprises et les habitations. Les systèmes de sécurité visent à apporter des fonctions de sécurité (comme les alarmes).

Ainsi, ce mémoire sera organisé comme suit : la carte d'acquisition à base d'Arduino UNO est pour but de transférer les données de ces capteurs pour commander des actionneurs.

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, couteuse. 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 événements. [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 :

1. 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.
2. 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.L'Objectif de la domotique :

Pour rentrer dans le vif du sujet, L'idée derrière ce projet est de gagner en confort, L'objectif premier de la domotique est d'apporter des solutions techniques pour répondre aux besoins de confort de l'utilisateur (gestion d'énergie, optimisation de l'éclairage et du chauffage...), de sécurité (alarmes, vidéo surveillance...) et de communication (commandes à distance, signaux visuels ou sonores...) que l'on .

Avec la domotique, votre maison devient intelligente !

Grâce à la domotique, votre habitation devient plus confortable, sécurisée, plus économique et facile à vivre au quotidien... vous pouvez piloter votre maison du bout des doigts !

Contrôle des éclairages, des équipements multimédias, du chauffage... d'une simple pression sur un interrupteur ou grâce à une télécommande.

Est possible aussi de **centraliser** et gérer à distance les fonctions des équipements électriques d'une maison grâce à une tablette ou encore un smartphone connecté à internet, que ce soit de l'intérieur ou, à quelques kilomètres ou même à l'autre bout du monde !

Il est possible aussi d'**automatiser** tout ou partie des équipements et programmer des scénarios d'usage selon le mode de vie et les habitudes.

Optimiser une installation domotique peut également faire réaliser des **économies d'énergie** (jusqu'à 25% par an).

I.5. Le fonctionnement de la domotique :

La **domotique** désigne toutes les technologies électroniques et de l'information appliquées à la gestion des bâtiments. Le **fonctionnement de la domotique** est simple à comprendre.

1. **La domotique** permet de centraliser les commandes, de créer des scénarios de vie et d'ambiance, de programmer à distance le chauffage, les éclairages, la fermeture des portes et des fenêtres.
2. Toutes les commandes sont réunies sur une **interface domotique** qui devient en quelques sortes le cerveau de votre maison.
3. Cette interface peut être une télécommande, un écran tactile, un ordinateur, ou un smartphone .
4. **Le fonctionnement d'une solution domotique** vous permet également accéder aux commandes directement par Internet, pour une gestion à distance.

Le fonctionnement de la domotique améliore le confort et la sécurité du bâtiment

Grâce à la domotique, vous pouvez réduire l'effort humain pour certaines tâches et **augmenter le niveau de sécurité** du bâtiment.

La domotique vous procure plus de confort et de simplicité au quotidien :

1. La fermeture des portes et des stores peut être centralisée : un seul geste suffit à sécuriser l'ensemble du bâtiment.
2. Vous pouvez programmer l'éclairage et le chauffage pour obtenir un meilleur confort de vie.
3. Des systèmes d'assistance à domicile peuvent être installés (notamment pour les personnes âgées ou les personnes à mobilité réduite).

La sécurité du bâtiment peut aussi être assurée par la domotique.

1. Une **alarme domotique** anti-intrusion peut être installée, et pilotée directement depuis **l'interface de gestion de la domotique**.

2. Des détecteurs d'incendie ou de fuite de gaz peuvent vous prévenir automatiquement en cas d'incident.
3. Il est possible de relier ces systèmes de sécurité à un commissariat de police ou à une caserne de pompiers.

La domotique comprend plusieurs éléments indispensables à son fonctionnement :

Le fonctionnement de la domotique repose sur la communication entre plusieurs appareils électriques. Pour assurer cette communication et obtenir les effets souhaités, une maison domotique doit comporter plusieurs éléments.

1. Les automatismes :

Les automatismes ont pour objectif d'éviter d'effectuer des gestes qui pourraient être automatisés. Il peut s'agir par exemple d'appuyer sur un bouton pour allumer une lumière, ou bien d'éteindre la télévision après une heure de veille. Le fonctionnement des automatismes est directement lié aux capteurs, au système de programmation, et à la **centrale domotique**.

2. Le système de programmation domotique:

Un système de programmation est indispensable pour profiter des avantages d'une installation domotique. C'est grâce à lui qu'il est possible choisir comment programmer des appareils électriques, en fonction des habitudes et des besoins.

3. Les appareils domotiques :

Il existe beaucoup d'appareils domotiques, puisqu'en fait tous les appareils électriques peuvent être domotique s'ils sont intégrés au fonctionnement de l'installation. Les lampes, les écrans, les machines à laver, les radiateurs... autant d'appareils qui peuvent être contrôlés par la domotique. Des appareils spéciaux ont été développés pour répondre aux besoins d'automatisation. On pense notamment à l'aspirateur domotique et à la tondeuse domotique, ou encore aux installations Hi-fi comme les homes cinéma.

4. Les capteurs domotique :

Les capteurs sont indispensables pour adapter le fonctionnement d'un objet domotique selon un critère défini. Par exemple, il existe des capteurs qui détectent le mouvement, et donc la présence humaine. D'autres détectent la luminosité, d'autres la température...

Ces capteurs convertissent une donnée en un signal. Ainsi, il est possible, par exemple, de régler un appareil électrique pour qu'il ne fonctionne que lorsque le capteur qui lui est associé constate un mouvement.

5. Le réseau domotique:

Pour transformer un logement en maison intelligente, il est nécessaire d'installer un réseau pour que les appareils électriques, les capteurs et le système de programmation puissent communiquer entre eux.

Il existe plusieurs types de réseau domotique :

1. **Le câblage domotique** : chaque appareil est relié à la centrale domotique via un câble (souvent Ethernet RJ45).
2. **La domotique par courant porteur CPL** (Courant Porteur en Ligne) : le réseau électrique fait circuler les données à travers les câbles électriques existants.
3. **La domotique sans fil** : plus pratique d'installation et d'utilisation, le réseau s'appuie sur l'échange d'informations par ondes radio ou par infrarouge.

L'interface domotique:

L'**interface domotique** permet de paramétrer en temps réel les réglages de fonctionnement de des appareils électriques. En fonction des habitudes et du rythme de vie de l'utilisateur,. Voici les principales interfaces possibles :

1. Une télécommande domotique;
2. Un écran de contrôle tactile.
3. Un ordinateur ou une tablette.
4. Un smartphone .

5. Directement sur Internet.

1. La centrale domotique :

Tous les équipements ci-dessus doivent être connectés entre eux pour pouvoir communiquer. C'est pour cela qu'une centrale domotique est indispensable au fonctionnement d'un équipement domotique. Il existe des centrales domotiques pour chaque type d'installation : câblage domotique, domotique par courant porteur, et domotique sans fil.

I.6. Différents domaines d'application de la domotique :

I.6.1. 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.6.2.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.6.3.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

L'alarme gaz :

Une alarme gaz est constituée d'un détecteur de gaz munie d'une sirène sonore intégrée. Le détecteur de gaz prévient les occupants de la maison en cas de fuite de gaz. C'est un petit appareil qui détecte la présence de gaz avant que sa concentration n'atteigne la Limite Inférieure d'Explosion (L.I.E) et prévient les occupants de l'habitation avec un puissant signal sonore. Selon les modèles, le détecteur de gaz peut être raccordé à un système d'alarme filaire ou sans fil.



Figure I.5: détecteur de gaz butane et propane

I.6.4.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 neuro dégénératives 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.7.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.8.Maison intelligente :

Pour faire simple, une maison intelligente est une maison dans laquelle plusieurs objets et appareils sont connectés au 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 d'une 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 d'un Smartphone, il est possible de gérer les appareils à distance et ces derniers peuvent envoyer des informations pratiques. [7]

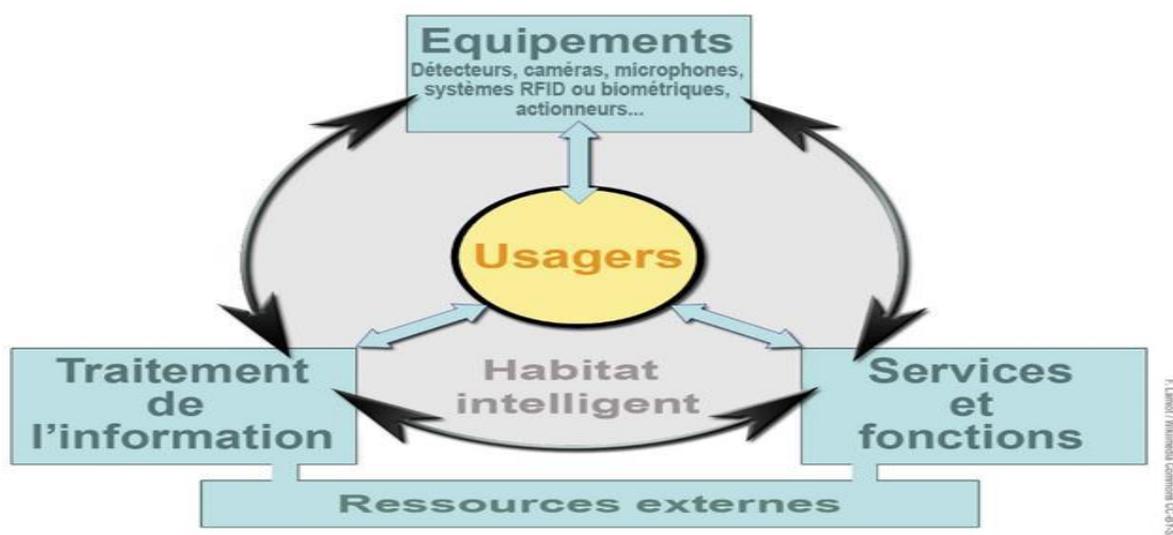


Figure I.6 : schéma représentatif de fonctionnement général des équipements d'une maison intelligente

I.9. Maison intelligente et la domotique :

Bien que extrêmement liés, les concepts de maison intelligente et de domotique présentent des différences notoires. Un système domotique constitue un ensemble intégré reprenant tous les composants d'une 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]

I.9.1. 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.7 : Kit Alarme sans fil Vidéo Do monial Standard HONEYWELL

I.9.2.Appareils électroménagers : aspirateurs-robots lave-linge, réfrigérateurs intelligents...



Figure I.8 : Aspirateur robot Wi-Fi Roomba 695 d'iRobot

I.9.3.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.9: Led ampoule sans fil

I.9.4.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.10 : moniteur d'énergie connecté

I.10. Avantages et inconvénients de la domotique**I.10. 1. Avantages :**

1. 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.
2. Ce type d'équipement vous simplifie la vie et optimise votre confort en adaptant votre maison à différents scénarios de la vie quotidienne.
3. 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.
4. 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.
5. Elle a pour avantage d'améliorer la sécurité grâce à des alarmes, des systèmes d'ouverture automatique de la porte (reconnaissance vocale, carte magnétique...)
6. En cas de tentative d'intrusion dans la maison, un appel téléphonique automatique peut contacter le propriétaire ou une entreprise de sécurité.
7. Enfin, ces différentes technologies constituent une aide précieuse pour les personnes dépendantes et handicapées. [6]

I.10. 2. Inconvénients :

Le principal inconvénient est le prix d'achat et d'installation. Le prix est beaucoup plus élevé mais il y'a toujours un retour sur investissement notamment les factures d'énergie qui 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. Il y'a aussi un problème de sécurité car ces installations peuvent être assujettie à des cybers. [6]

I.11.Dernières innovation :

1. Item 1: Alarme anti intrusion
2. Item 2: Détecteur techniques
3. Item 3: Gestions de l'accès au logement

I.11.1.Détection domestique active avec alerte à distance :

Quels avantages au quotidien ?

Dès que l'incident survient, les détecteurs reliés à la centrale d'alarme alertent par e-mail ou par téléphone et, selon le cas, déclenchent des actions automatiques de sécurité.

1. Détection des principaux incidents : fuite d'eau, fumée, émanation de gaz, coupure de courant, élévation de température... à chacun de ces incidents correspond un détecteur spécifique.
2. Parades automatiques de sécurité : reliés à la centrale d'alarme, les détecteurs techniques peuvent simultanément alerter et déclencher des actions de sécurité telles que la coupure de l'eau ou du gaz au moyen d'une électrovanne.

Toutes les solutions possibles :

Eléments du système domotique, les détecteurs techniques bénéficient de toutes ses potentialités et offrent de nombreuses fonctions.

1. Déclenchement de scénarios : à l'alarme provoquée par la fumée ou une fuite de gaz, on peut associer, par exemple, signal sonore dans la chambre des parents.
2. Alerte à distance : le transmetteur envoie un sms en cas d'incident.



Figure I.11 Détection domestique active avec alerte à distance

I.11.2. Actions simultanées détection intrusion : alarme, éclairage, volets :

Quels avantages au quotidien ?

D'un seul geste et avec une seule commande, activer le scénario programmé : le système d'alarme associé à différents automatismes du logement.

1. Efficacité de dissuasion, simplicité d'utilisation : à la première tentative d'intrusion, tout se déroulera comme le scénario l'a prévu, sirène en action, volets fermés, logement en pleine lumière, par exemple. Une puissante invitation à faire demi-tour !
2. Sentiment de sécurité accru : une majorité de cambrioleurs abandonnent leur tentative au déclenchement d'une simple alarme ; en y associant d'autres moyens, l'effet dissuasif se trouve renforcé !



Figure I.12 détection intrusion : alarme, éclairage, volets.

Toutes les solutions possibles :

De multiples variations s'offrent à nous pour associer dans un même programme le système d'alarme et d'autres automatismes de la maison.

1. Alarme et confort : chaque matin avant de quitter la maison, il est possible simultanément commander l'alarme, le chauffage au degré voulu, les lumières et les volets, et tout ceci avec une seule commande !
2. Alarme et scénarios de dissuasion : à la demande, il est possible de scénario de dissuasion en choisissant les fonctions qui accompagneront le déclenchement du système d'alarme : lumières intérieures, lumières extérieures, diffusion de message, fermeture des volets... tous les scénarios sont possibles !
3. En option, la transmission vers l'extérieur : avec celle-ci, le déclenchement du système d'alarme entraîne l'envoi immédiat d'un message vocal, d'un sms ou d'un e-mail à tous les destinataires concernés : propriétaire ainsi que le centre de télésurveillance ou toute autre personne.

I.11.3. Identification des visiteurs par interphone ou vidéophone :

Quels avantages pour au quotidien ?

De l'intérieur communiquer avec les personnes qui sonnent à la porte. Pour ouvrir ou non de l'intérieur !

1. Pas de déplacements inutiles : si on décide d'accueillir la personne qui demande l'accès, on peut lui ouvrir depuis le poste intérieur de l'interphone ou vidéophone.
2. Pas d'inquiétudes inutiles : par l'image ou par le son, il est possible de reconnaître le visiteur ou connaître le motif de la visite.
3. Entrée libre si l'on souhaite : interphone et vidéophones peuvent être réglés pour permettre un accès libre au logement quand on le désire, pour des visites régulières à heures fixes, par exemple.

I.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.

II.1. 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, etc. [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 verrons une présentation générale du système Arduino UNO, ses caractéristiques et présentation des quelques shields et on mettra la lumière sur le logiciel IDE Arduino et proteus.

II.2. 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.

1. Contrôler les appareils domestiques.
2. Fabriquer votre propre robot.
3. Faire un jeu de lumières.
4. Communiquer avec l'ordinateur.
5. Télécommander un appareil mobile (modélisme).
6. 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.

II.3 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 :

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

II.4 Présentation de la carte

II.4.1. 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.4.2. Microcontrôleur ATMEL ATmega328

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

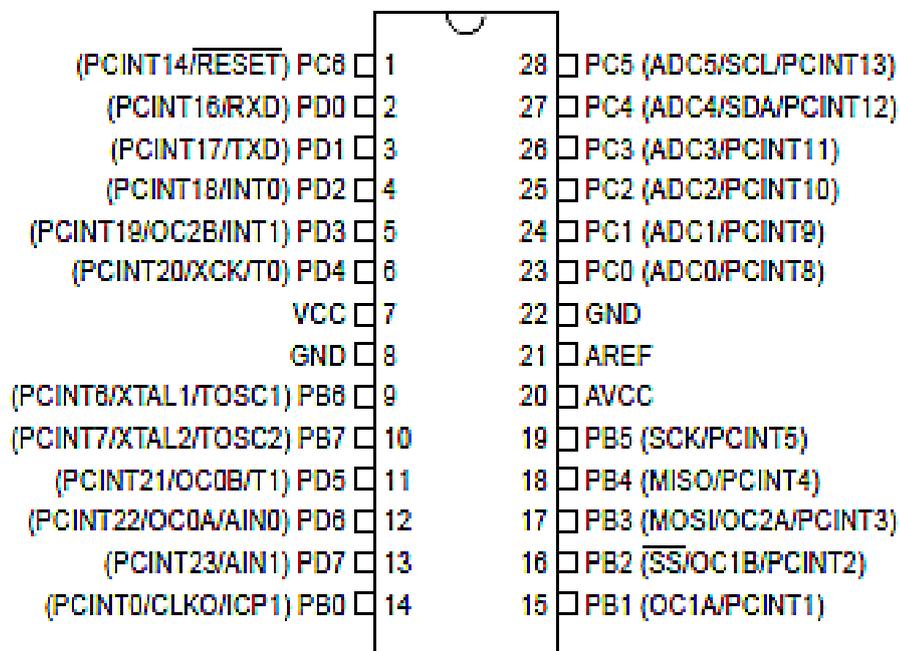


Figure II.1: Datasheet ATmega328.

Les principales caractéristiques d'ATmega328 sont :

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

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

II.4.3. 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 délivré 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 connection Power Jack (5). [11]

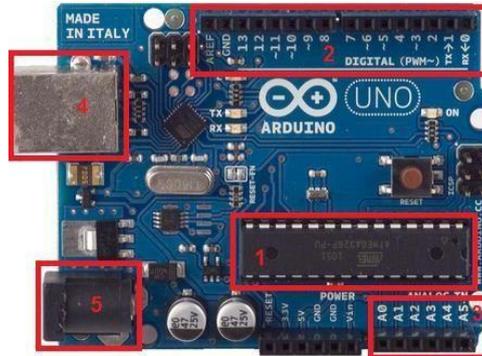


Figure II.2: La carte Arduino UNO.

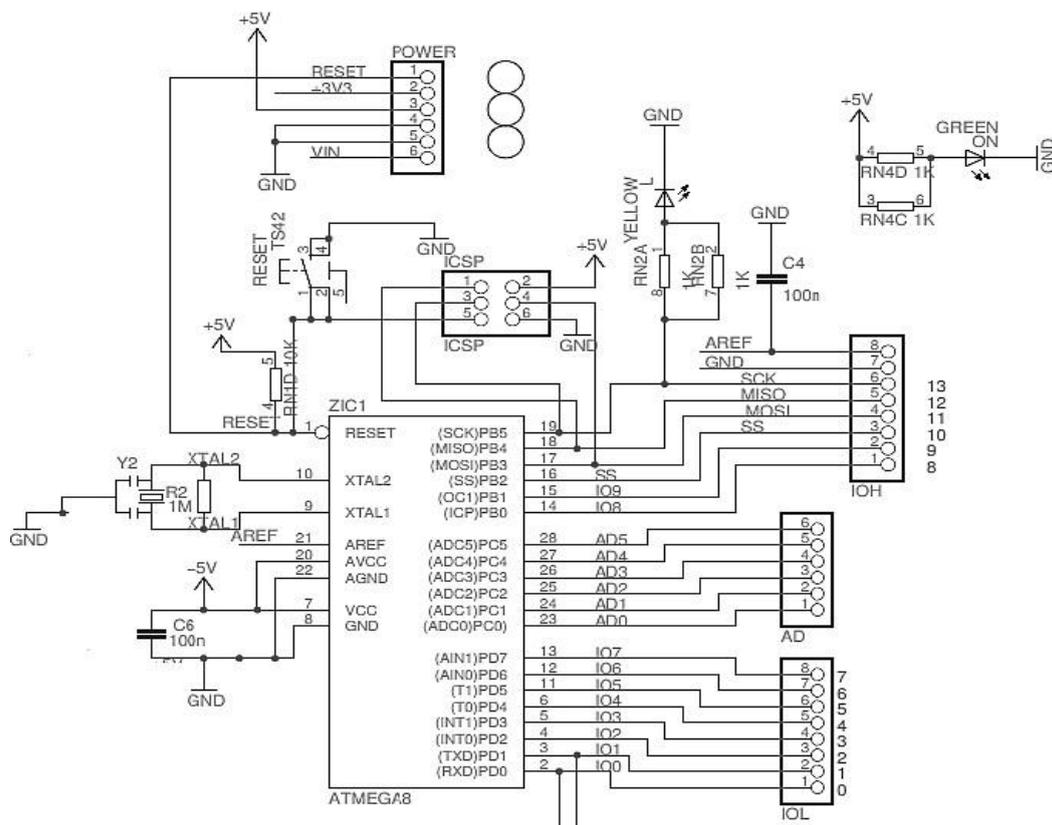


Figure II.3: Schéma simplifié de la carte Arduino UNO.

II.4.4. 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.5. Capteurs

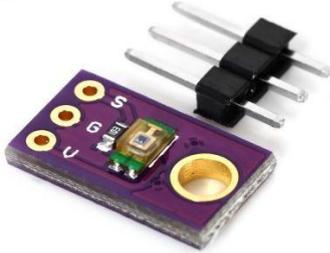
Un capteur est un dispositif ayant pour tâche de transformer une mesure physique observée en une mesure généralement électrique qui sera à son tour traduite en une donnée binaire exploitable et compréhensible par un système d'information.

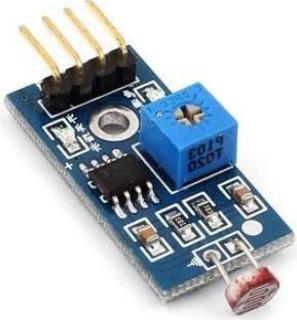
II.5.1.Présentation des capteurs

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	<p>TEMT6000 Light Sensor</p> 	<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]

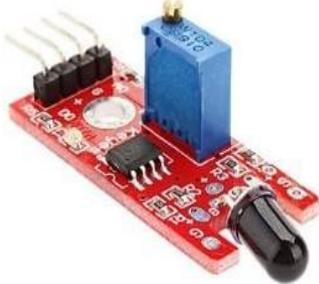
<p>capteur de pluie double face</p>		<ul style="list-style-type: none"> ✓ Tension : 3,3V-5V ✓ Dimension du capteur : ~ 40 x 54 mm ✓ Pistes sur capteur : Double face. Les deux faces du capteur sont actives. ✓ Potentiomètre de sensibilité et 2 LED de contrôle intégré au module ✓ Module livré avec deux 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 connectée avec arduino

II.5.2.Communication

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

II.5.2.1. 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 : Type de modules Bluetooth

II.5.2.2. 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.6. Présentation du logiciel

II.6.1. 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 et d'une barre d'outils rapide. 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 plus classique qui est utilisé pour accéder aux fonctions avancées de l'IDE. Enfin, une console affichant les résultats de la compilation du code source, des opérations sur la carte, etc.

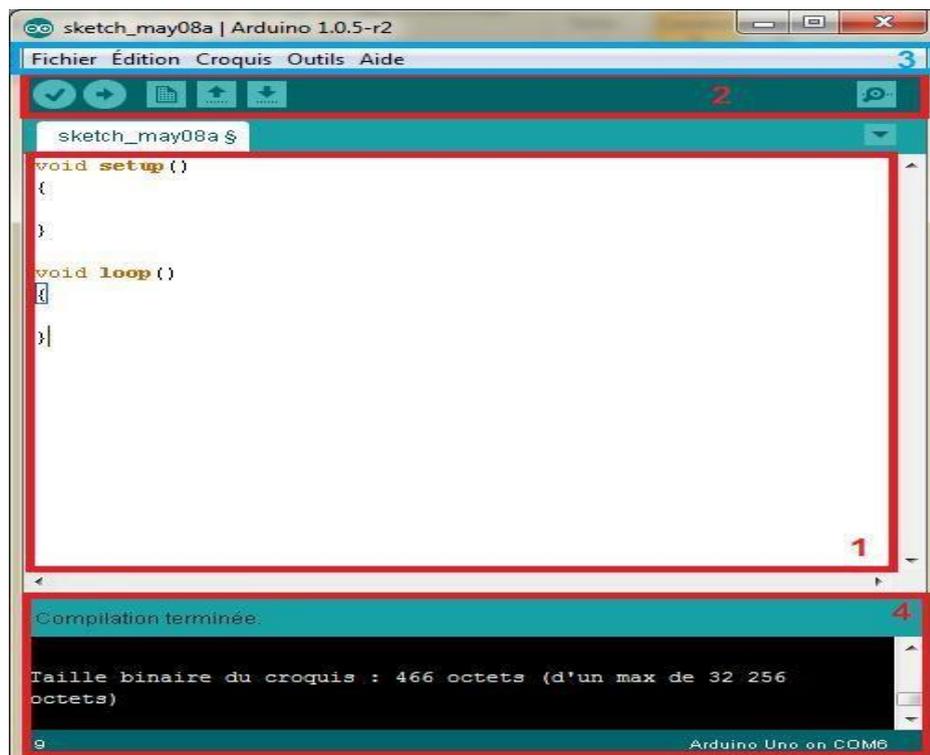


Figure II.6 : IDE Arduino.

II.6.1.1. Les boutons

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



Figure II.7 : Présentation des boutons.

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

II.6.2. 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.8 : 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>	Chaine de caractères (Objet)	variable	Aucune
<i>boolean</i>	Booléen	1	True / False

Tableau II.3: 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 :

No m	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.4: 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.6.3. Fonctionnalité de base

II.6.3.1. 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 `pinMode` 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 s'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 :

1. **digitalRead(pin)**: mesure une donnée numérique sur une des broches, la broche en question doit être réglée en entrée.
2. **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).
3. **analogRead(pin)**: mesure une donnée analogique sur une des broches (compatible seulement), la broche doit être réglée sur entrée.
4. **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 $[0;255]$.

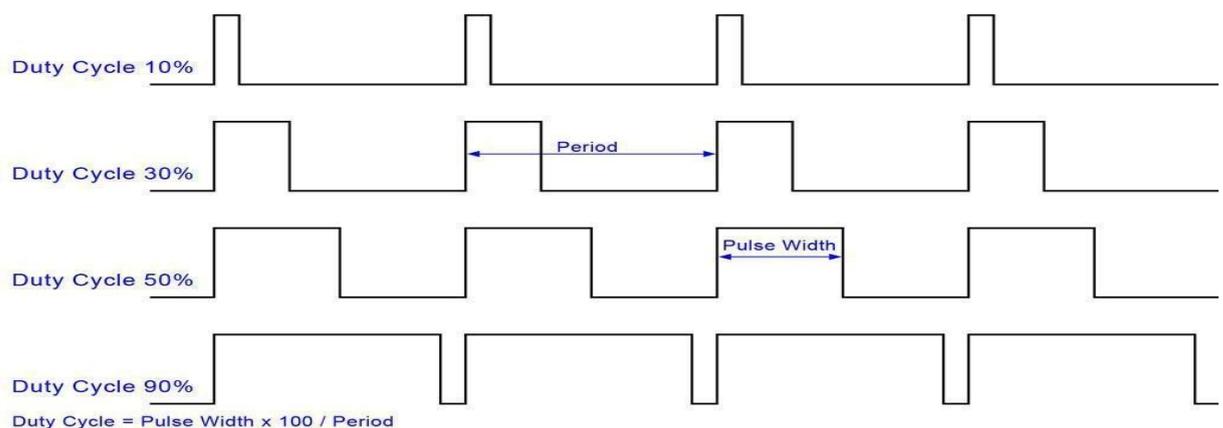


Figure II.9: Signal PWM.

II.6.3.2. 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.6.3.3. 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 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 :

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

Voici un exemple d'utilisation :

```
volatile boolean etat = false ; void appuiBouton() {  
    etat = !etat ; // Changement d'état } void setup() {  
pinMode(2,INPUT) ; // Broche 2 en entrée  
attachInterrupt(0,appuiBouton,RISING) ; // On attache à l'interruption 0 (broche 2) la  
fonction appui Bouton sur un front montant  
}
```

On remarque l'apparition du mot clef `volatile` avant la déclaration de la variable `etat`. Ce mot clef est nécessaire pour toutes les variables qui sont modifiée dans une interruption. Cela à une incidence sur la manière dont le compilateur traite l'accès à la variable.

Il est parfois nécessaire de désactiver temporairement les interruptions par exemple lorsque l'on exécute du code critique (activation d'un moteur, etc.). Deux fonctions permettent de changer l'activation des interruptions `interrupts` et `noInterrupts` pour activer (respectivement désactiver) les interruptions.

Exemple

Nous allons maintenant passer à un exemple. Il s'agit de faire clignoter une LED de façon régulière. On doit utiliser une LED avec une résistance en série sur une breadboard. Ou en peut utiliser la LED directement sur l'Arduino (broche 13 sur la carte Arduino UNO).

Code avec delay :

```
const int pinLed = 13; void setup()

{ pinMode(pinLed, OUTPUT); // Broche 13 en sortie }

void loop()

{

delay(500); // Attente d'une demi seconde digitalWrite(pinLed,
HIGH); // Allumage de laLED delay(500);

digitalWrite(pinLed, LOW); // Eteignage de laLED

}
```

Code avec millis :

```
const      int pinLed=13;

int temps; int etat; void setup(){

pinMode(pinLed, OUTPUT); // Broche 13 en sortie etat = LOW; //
LED éteinte }

void loop()

{ int present = millis();

if (temps+500 < present) // Vérification du chronomètre

{ temps = present; // Actualisation du chronomètre etat =
!etat; //          Changement d'état
```

```
digitalWrite (pinLed,etat); // Changement d'état de la LED  
}  
}
```

II.6.3.4. Quelques librairies

En plus de la simplicité du langage et des nombreuses fonctionnalités qu'offre. L'IDE vient avec un nombre important de librairies évitant ainsi d'implémenter des fonctions courantes dans l'informatique embarquée.

Une des librairies les plus utilisée est celle implémentant la communication série. La majorité des cartes Arduino possède un émulateur de connexion série pour communiquer au travers de l'USB. Ainsi, on peut communiquer avec l'ordinateur sur lequel la carte Arduino est connectée. Cela permet, par exemple, de déboguer un programme en affichant la valeur des variables ou simplement afficher la valeur des capteurs. Cette librairie a été directement implémentée dans le langage Arduino. On peut accéder à la communication série (au travers l'USB) grâce à l'objet Serial. Une autre librairie existe pour communiquer par liaison série via une des broches de la carte.

Il est parfois nécessaire de stocker des informations même après l'arrêt de la carte. Il est possible de stocker une petite quantité d'information sur la mémoire EEPROM (Electrically Erasable Programmable Read Only Memory) intégrée. Une librairie est aussi fournie pour dialoguer avec cette mémoire. Il est possible de lire et d'écrire sur cette mémoire sans rien ajouter sur la carte. Cette mémoire est accessible via l'objet EEPROM et en ajoutant la librairie du même nom. Attention, la mémoire EEPROM à une durée de vie limitée (environ 100 000 cycles d'écritures). Il faut donc veiller à ne pas écrire répétitivement les données. D'autres alternatives existent pour ajouter de plus grandes quantités de mémoires mortes à l'Arduino (carte SD notamment) mais cela demande l'ajout de composants externes.

Pour terminer, il existe aussi des librairies permettant de contrôler des composants externes. Parmi ces librairies, une des plus populaires est celle contrôlant des servomoteurs.

Un servomoteur est un composant électronique composé d'un moteur et d'une carte d'asservissement. On peut le contrôler en position (c'est-à-dire choisir l'angle du moteur) grâce aux PWM. Une librairie Arduino est implémentée pour contrôler simplement ces moteurs : la librairie servo. Il faut toutes fois faire attention à éviter

d'alimenter les servomoteurs directement sur une des broches d'alimentation de la carte. En effet, un servomoteur consomme une quantité important de courant (suivant le couple exercé par le moteur). Il est donc fortement conseillé de relier les servomoteurs sur des alimentations externes (batteries, piles de 9V, etc.). [17]

II.7. Conclusion

Dans ce chapitre nous avons vu une présentation générale de la carte Arduino UNO, ses caractéristiques et présentation des quelques shields mettant 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 leurs 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 utilisera 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

III.1 Introduction

La phase de Conception implémentation est la plus importante étape dans le cycle de vie de notre système, car à la fin de cette dernière, nous aurons le fruit de toutes les étapes précédentes. Pour la réalisation de notre travail, plusieurs outils de développement sont disponibles et vu les contraintes matérielles et logicielles on a opté sur la carte "ARDUINO".

Au cours de ce chapitre, nous présenterons la conception générale de notre système ensuite par la programmation de chaque capteur puis on passera à la simulation de notre système.

Une fois la réalisation est achevée, une phase de validation sera nécessaire. Nous testerons la fonctionnalité du prototype en vérifiant sa réponse aux détections de la fuite de gaz, détection de présence , détecteur de l'éclairage

III.2 Conception du système

Le système que nous sommes en vu de conception doit être au service de l'homme pour contrôler et exploiter au mieux son logement dans le cadre de la domotique. Pour ce faire, il doit avoir deux modes d'accès à ce système :

D'abord comme utilisateur, activer ou désactiver des fonctions comme la sécurité « System Activation/Stop »

Mais aussi comme manager, qui pourra avoir la main pour changer les paramètres du système « Setting arameters » et créer des profils et scénarios qu'il pourra activer selon le besoin « Creating profils ».

L'accès au système doit être bien sûr authentifié, n'importe qui ne pourra pas accéder au système. L'accès doit être authentifié par un code ou mot de passe « Login system ».

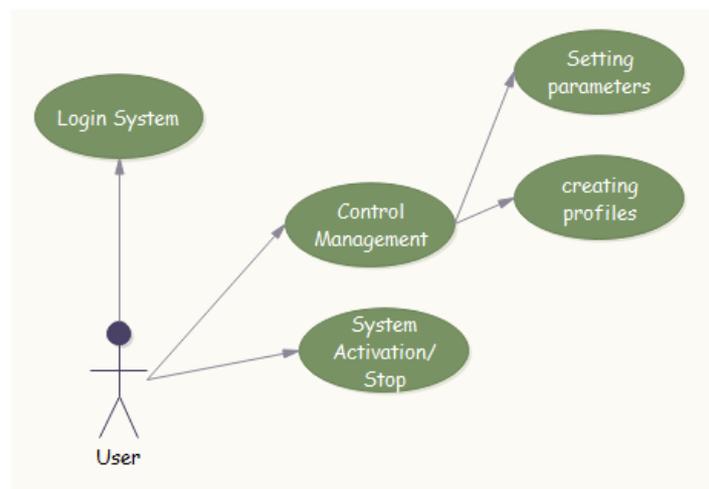


Figure III.1 : diagramme de cas d'utilisation

Ce système sera composé d'une partie opérative contenant les différents capteurs et actionneurs et la partie commande qui sera constituée du micro-contrôleur ATM de la carte ARDUINO.

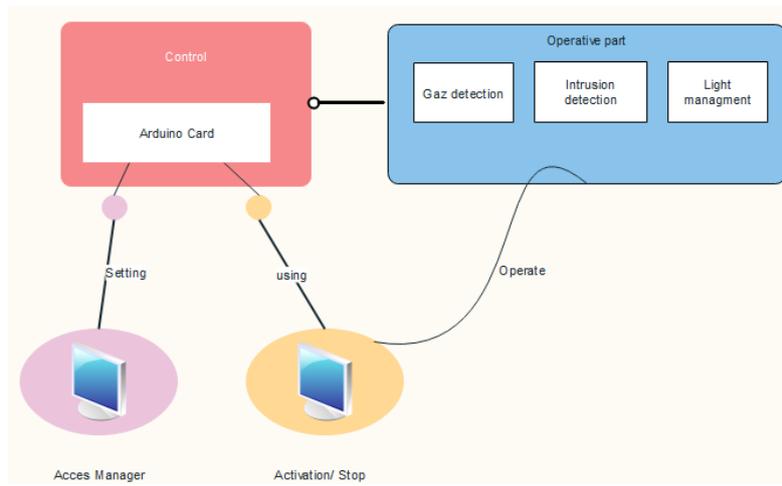


Figure III.2 : les constituants du système

La partie opérative est ainsi composée de 3 parties :

Gaz détection : capteur de gaz et buzzer.

Intrusion détection : capteur de présence et une led.

Light management : capteur de l'éclairage et un moteur.

Ce système sera réalisé, programmé et simulé sur PROTEUS.

III.3 Simulation

La simulation permet d'assurer le fonctionnement du circuit comme si on manipulait un montage réel ceci permet d'accélérer le prototypage et de réduire son cout .

Il faut toujours prendre en considération que les résultats obtenus de la simulation sont un peu différents de celles du monde réel .

Les outils de conception et de simulation utilisés :

Grace à PROTEUS de LABCENTER nous simulant notre circuit , c'est un logiciel très populaire pour la simulation , pour cela la figure ci-dessous représente l'interface du logiciel PROTEUS 8 professionnel.

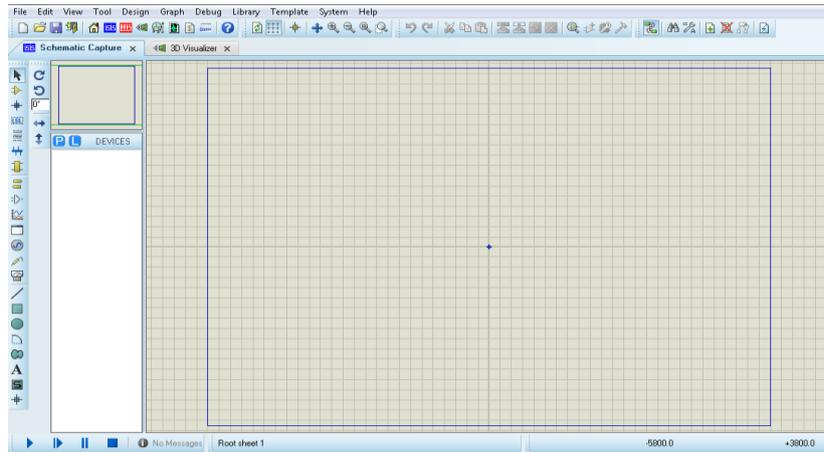


Figure III.3 logiciel proteus 8 professionnel

III.4 Schéma électronique du montage sur proteus

Le schéma électronique de l'Arduino avec les actionneurs (Alarme, led, Motors) .

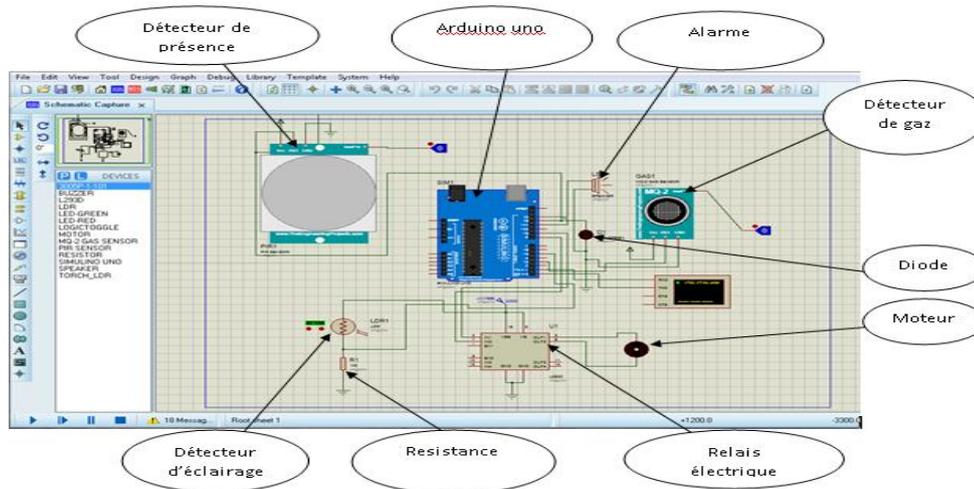


Figure III.4 : schéma électronique du montage

Au niveau de ce schéma électronique qui contient plusieurs composants, parmi ces composants l'arduino qui scrute les entrées des capteurs comme capteur de gaz, capteur de présence et capteur de l'éclairage et dans notre projet on a 3 cas, premièrement lorsqu'il y a un cas de fuite de gaz, l'arduino commande l'actionneur (sortie) , il va activer le système d'alarme (buzzer) , deuxièmement lorsqu'il y a un cas de présence d'une personne, l'arduino

commande la led pour s'allumer, troisièmement lorsqu'il y a un cas de changement d'éclairage, l'arduino commande le démarrage de moteur pour ouvrir ou fermer la fenêtre

III.5 Description des shields utilisés :

Dans ce qui suit, on donnera une description détaillée des différents shields utilisés et ils sont comme suit :

III.5.1 Capteur de gaz (MQ2)

III.5.1.1 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. [10]



Figure III.5 Image réelle du capteur de gaz/fumée MQ-2.

Programme de détection de fuite de gaz :

Au niveau de cette exemple, l'ARDUINO UNO scrute les entrées des capteurs (gaz), lorsque il y a un cas d'incident (fuite de gaz par exemple), l'Arduino commande les actionneurs. Il va activer le système d'alarme.

Explication :

Le programme de ce montage comporte les fonctions suivantes :

```
int gaz_volume; // volume de gaz

int gaz_input = 3; // détecteur de gaz à position 3 d'arduino

void setup() {

  pinMode(gaz_input, INPUT); // mettre gaz_input comme entrée
```

```

gaz_volume = digitalRead(gaz_input);           // lire le gaz

pinMode(buzzer, OUTPUT);                       // mettre buzzer comme sortie
}

void setup() {

  if (gaz_volume == HIGH );                    // condition de la disponibilité de gaz
}

tone(buzzer, 1000);                             // active l'alarme

```

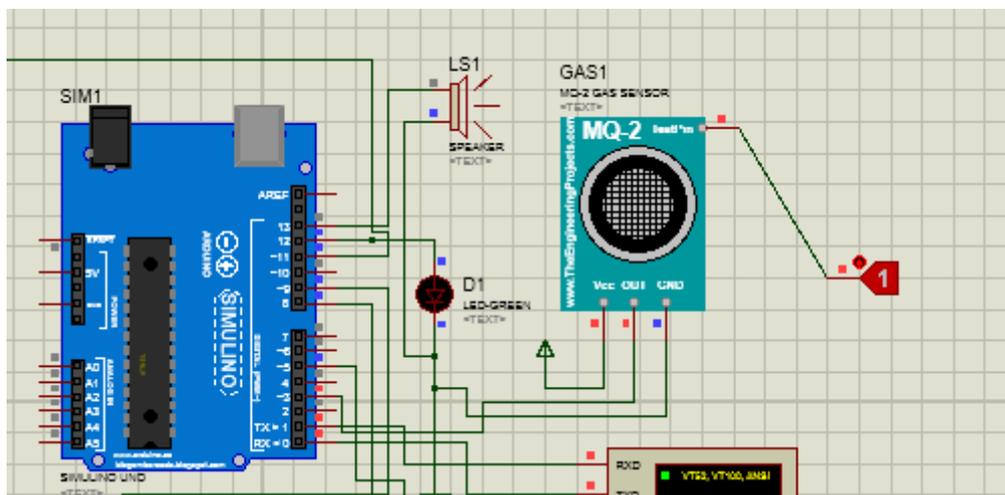


Figure III.6 : le schéma de simulation de détecteur fuite de gaz

III.5.2 Capteur de mouvement (PIR)

III.5.2.1 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. [11]



Figure III.7 Image réelle du capteur de mouvement PIR.

Programme de la fonction de détecteur de présence (PIR)

Au niveau de cette exemple, l'ARDUINO UNO scrute les entrées des capteurs (présence), lorsque il y a un cas de mouvement (présence d'une personne par exemple), l'Arduino commande les actionneurs. Il va activer l'éclairage.

Explication :

Le programme de ce montage comporte les fonctions suivantes :

```
int pir_input = 11;           // détecteur de présence à la 11eme position d'arduino.

int pir_presence;           // la présence de la personne.

void setup() {
  pinMode(pir_input, INPUT); // mettre pir_input comme entrée.

  pir_presence = digitalRead(pir_input); // lire présence d'un homme
}

void loop() {
  if (pir_presence == HIGH) // condition de la présence de la personne.

  digitalWrite(led, HIGH); // allumer une led en cas de présence d'une personne.
}
```

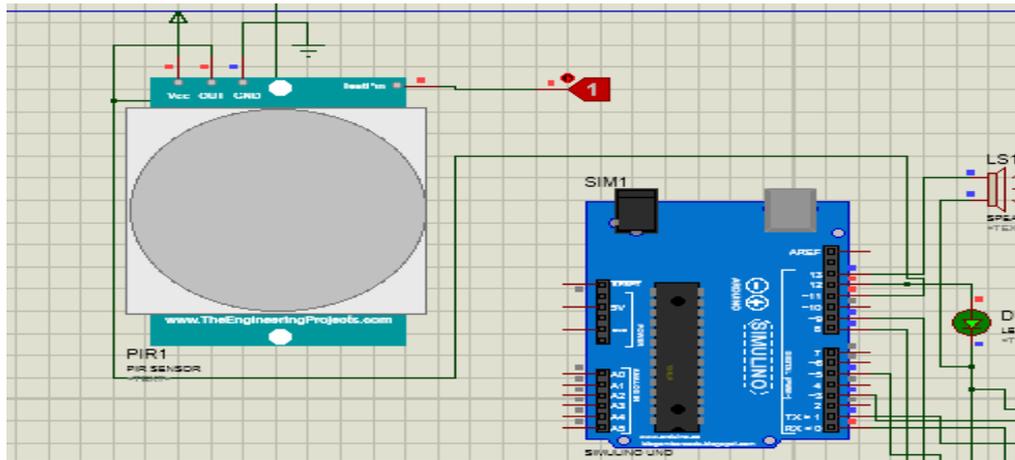


Figure III.8 : le schéma de la simulation de détecteur de présence

III.5.3 Détecteur de l'éclairage (capteur LDR)

III.5.3.1 Description du capteur

Une photorésistance est un composant électronique dont la résistivité varie en Fonction de la quantité de lumière incidente. On peut également la nommer résistance photo dépendante (Light dépendent resistor LDR) ou cellule photoconductrice.[12]



Figure III.9 Image réelle du capteur LDR

Fonction de gestion de l'éclairage (LDR)

Au niveau de cette exemple, l'ARDUINO UNO scrute les entrées des capteurs (l'éclairage), lorsque il y a un cas de changement de quantité d'éclairage (l'obscurité ou la lumière par exemple), l'Arduino commande les actionneurs. Il va fermer ou ouvrir les fenêtres.

Explication

Le programme de ce montage comporte les fonctions suivantes :

```
int ldr_volume;

bool firstTimeAvancer = true;

bool firstTimeReculer = true;

int avancer = 9;

int reculer = 8;

pinMode(avancer, OUTPUT);

pinMode(reculer, OUTPUT);

void setup() {

}

void loop() {

  ldr_volume = analogRead(5); // lire l'intensite de l'eclairage

  Serial.println(ldr_volume);

  else if (gaz_volume == LOW && pir_presence == LOW && ldr_volume > 10) {

    digitalWrite(reculer, HIGH);

    delay(1000);

    digitalWrite(reculer, LOW);

    firstTimeReculer = false;

    firstTimeAvancer = true;

  }

  if (firstTimeAvancer) {

delay(1000);

    digitalWrite(avancer, LOW);

    firstTimeAvancer = false;

    firstTimeReculer = true;

  }

  else if (gaz_volume == LOW && pir_presence == HIGH && ldr_volume <= 10) {

    if (firstTimeReculer) {
```

```
digitalWrite(avancer, HIGH);
```

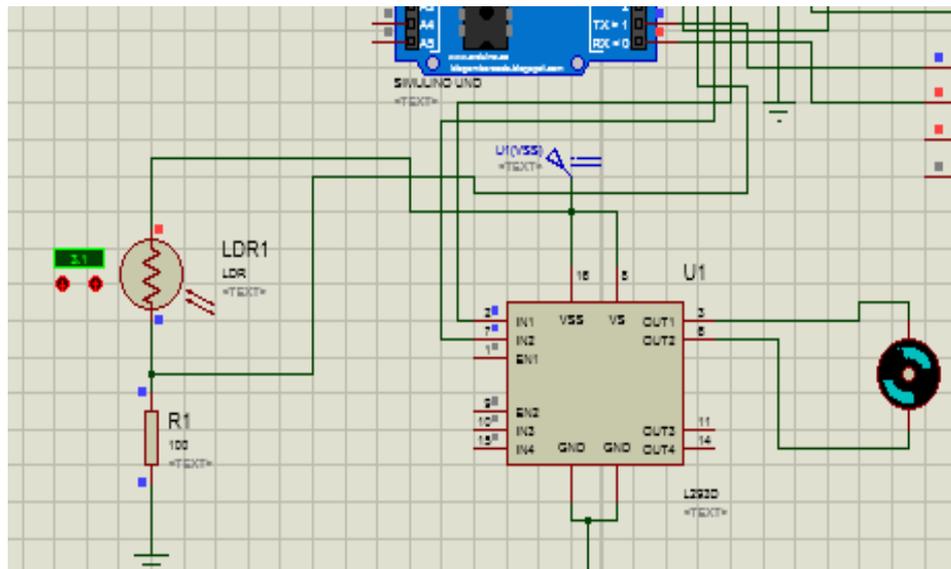


Figure III.10 le schéma de la simulation de détecteur de l'éclairage

III.5.4 Le buzzer

III.5.4.1 Description du buzzer

Le Buzzer est une structure intégrée de transducteurs électroniques, alimentation en courant continu, largement utilisé dans les ordinateurs, les imprimantes, les photocopieurs, les alarmes, jouets électroniques, matériel électronique automobile et d'autres produits électroniques pour les appareils sonores. Le Buzzer passif module d'alarme utilisé pour l'Arduino.[13]



Figure III.11 Image réelle du buzzer

III.5.5 Led :

III.5.5.1 Description du led

Le mot LED est l'acronyme de Light Emitting Diode (Diode Electroluminescente en français). Le symbole de la LED ressemble à celui de la diode mais on y a ajouté deux flèches sortantes pour représenter le rayonnement lumineux émis. [14]



Symbole de la LED.



Figure III.12 Image réelle du led

III.5.6 Relais électrique :

III.5.6.1 Description du relais

Un relais est un circuit électrique permettant de contrôler le flot de courant, comme un interrupteur pour les lumières. La différence est qu'un interrupteur traditionnel fonctionne de façon mécanique, soit en l'activant physiquement avec votre doigt, alors qu'un relais fonctionne avec un courant électrique. L'avantage d'un relais est aussi qu'il isole complètement les deux circuits, permettant ainsi d'activer un circuit dans lequel circule un courant plus fort. [15]



Figure III.13 Image réelle d'un relais électrique

III.5.7 Resistance

III.5.7.1 Description de résistance

La résistance (nous devrions dire élément résistif ou résistor) est le composant électronique le plus utilisé.

La résistance limite le courant dans la branche du circuit électrique où elle est insérée.

Dans l'univers Arduino, elle est principalement utilisée pour :

1. limiter le courant dans une led
2. polariser correctement un transistor
3. former un pont diviseur de tension
4. imposer un niveau logique haut (résistance de tirage ou pull-up) ou un niveau logique bas (résistance de rappel ou pull-down) [16]



Figure III.14 Image réelle d'une résistance

III.5.8 Moteur (courant continu)

III.5.8.1 Description du moteur

Les moteurs à courant continu possèdent souvent une boîte de réduction afin d'augmenter leur couple pour un encombrement réduit. Le moteur CC est très simple d'utilisation. Pour le faire fonctionner, il suffit d'appliquer une tension électrique à ses bornes. Le signe et le niveau de cette tension vont imposer le sens et la vitesse de rotation.[17]



Figure III.15 Image réelle d'un moteur

III.6 Conclusion

L'arduino c'est un mélange d'électronique et de programmation donc c'est de l'électronique embarquée qui nous facilite la réalisation de beaucoup de projets.

Conclusion générale

Le système Arduino est un outil pour fabriquer des dispositifs qui peuvent capter et contrôler davantage de choses du monde matériel qu'un 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.

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 réalisation de la carte.

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 souhaitons vivement que ce projet puisse servir comme élément de base pour d'autres études plus approfondies pour le faire intégrer sous des systèmes plus complexes.

Bibliographie

- [1] Krama. A, Gougui. A, "Etude et réalisation d'une carte de contrôle par Arduino via le système Androïde", Mémoire Master Académique, Université KasdiMerbah Ouargla, Algérie, 2015.
- [2] HAMOUCHE. 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 TiziOuzou, 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-domotiquemaison-connectée>.
- [6] Mr: METAHRIM, Melle ABDELLIS, " 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] <https://www.gotronic.fr/art-detecteur-de-niveau-d-eau-grove-101020018-19048.htm>.
- [9] Lechalupé. J, "cours d'initiation à Arduino", Université Paul Sabatier, Mai 2014.
- [10] <http://www.arobose.com/shop/temperature-pression/217-capteur-analogique-de-gaz-mq2.html>.
- [11] <https://www.generationrobots.com/fr/402273-capteur-pir-de-mouvement.html>.

[12] <https://www.aranacorp.com/fr/mesure-de-luminosite-avec-une-photoresistance/>

[13] <https://www.aranacorp.com/fr/utilisation-dun-buzzer-avec-arduino/>.

[14] <https://www.positron-libre.com/cours/electronique/diode/led/led.php>

[15] <http://blog.ezoqc.com/comment-controler-un-relais-avec-arduino/>

[16] <https://pecquery.wixsite.com/arduino-passion/la-resistance>

[17] <https://www.aranacorp.com/fr/pilotez-un-moteur-cc-avec-arduino/>

Programme

```
int gaz_input = 3;

int led = 12;

int buzzer = 13;

int gaz_volume;

int pir_input = 11;

int pir_presence;

int ldr_volume;

int avancer = 9;

int reculer = 8;

bool firstTimeAvancer = true;

bool firstTimeReculer = true;

void setup() {

    pinMode(gaz_input, INPUT);

    pinMode(led, OUTPUT);

    pinMode(buzzer, OUTPUT);

    pinMode(pir_input, INPUT);

    pinMode(avancer, OUTPUT);

    pinMode(reculer, OUTPUT);

    Serial.begin(9600);

}

void loop() {

    gaz_volume = digitalRead(gaz_input); // lire le gaz
```

```
pir_presence = digitalRead(pir_input); // lire presence d'un homme

ldr_volume = analogRead(5); // lire l'intensite de l'eclairage

Serial.println(ldr_volume);

if (gaz_volume == LOW && pir_presence == LOW && ldr_volume <= 10) {

    digitalWrite(led, LOW);

    noTone(buzzer);

    if (firstTimeReculer) {

        digitalWrite(reculer, HIGH);

        delay(1000);

        digitalWrite(reculer, LOW);

        firstTimeReculer = false;

        firstTimeAvancer = true;

    }

}

else if (gaz_volume == LOW && pir_presence == LOW && ldr_volume > 10) {

    digitalWrite(led, LOW);

    noTone(buzzer);

    if (firstTimeAvancer) {

        digitalWrite(avancer, HIGH);

        delay(1000);

        digitalWrite(avancer, LOW);

        firstTimeAvancer = false;

        firstTimeReculer = true;

    }

}

else if (gaz_volume == LOW && pir_presence == HIGH && ldr_volume <= 10) {

    digitalWrite(led, HIGH);

    noTone(buzzer);
```

```
if (firstTimeReculer) {
    digitalWrite(reculer, HIGH);
    delay(1000);
    digitalWrite(reculer, LOW);
    firstTimeReculer = false;
    firstTimeAvancer = true;
}
}

else if (gaz_volume == LOW && pir_presence == HIGH && ldr_volume > 10) {
    digitalWrite(led, HIGH);
    noTone(buzzer);
    if (firstTimeAvancer) {
        digitalWrite(avancer, HIGH);
        delay(1000);
        digitalWrite(avancer, LOW);
        firstTimeAvancer = false;
        firstTimeReculer = true;
    }
}

else if (gaz_volume == HIGH && pir_presence == LOW && ldr_volume <= 10) {
    digitalWrite(led, LOW);
    tone(buzzer, 1000);
    delay(50);
    noTone(buzzer);
    delay(50);
    if (firstTimeReculer) {
        digitalWrite(reculer, HIGH);
        delay(1000);
```

```
    digitalWrite(reculer, LOW);

    firstTimeReculer = false;

    firstTimeAvancer = true;

}

}

else if (gaz_volume == HIGH && pir_presence == LOW && ldr_volume > 10) {

    digitalWrite(led, LOW);

    tone(buzzer, 1000);

    delay(50);

    noTone(buzzer);

    delay(50);

    if (firstTimeAvancer) {

        digitalWrite(avancer, HIGH);

        delay(1000);

        digitalWrite(avancer, LOW);

        firstTimeAvancer = false;

        firstTimeReculer = true;

    }

}

else if (gaz_volume == HIGH && pir_presence == HIGH && ldr_volume <= 10) {

    digitalWrite(led, HIGH);

    tone(buzzer, 1000);

    delay(50);

    noTone(buzzer);

    delay(50);

    if (firstTimeReculer) {

        digitalWrite(reculer, HIGH);

        delay(1000);
```

```
digitalWrite(reculer, LOW);

firstTimeReculer = false;

firstTimeAvancer = true;

}

}

else {

digitalWrite(led, HIGH);

tone(buzzer, 1000);

delay(50);

noTone(buzzer);

delay(50);

if (firstTimeAvancer) {

digitalWrite(avancer, HIGH);

delay(1000);

digitalWrite(avancer, LOW);

firstTimeAvancer = false;

firstTimeReculer = true;

}

}
```

