

الجمهورية الجزائرية الديمقراطية الشعبية

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي و البحث 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 : Réseau et télécommunication.

Par :

Benladghem Djazia

Benyahia Nabila

Sujet :

Le système de verrouillage de porte et système d'alerte en utilisant Arduino RFID

Soutenu publiquement, le 14/09/2020, devant le jury composé de :

M BENADDA .B

Professeur

Univ. Tlemcen

Président.

M BAHRI .M

Maitre de conférences

Univ. Tlemcen

Examinateur.

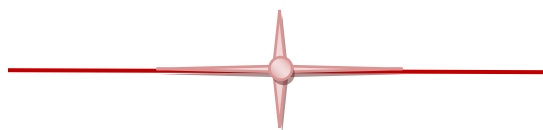
M Borsali Riad

Professeur

Univ. Tlemcen

Encadreur.

Dédicace



Aucune dédicace ne pourrait assez exprimer notre gratitude et notre profond respect pour toutes les personnes qui nous ont entourés, avec grande joie nous dédions ce travail :

A la sensibilité de nos mères,

A la justice de nos pères,

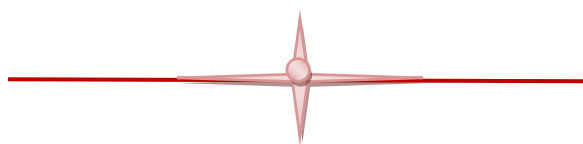
A l'amour de nos frères et sœurs,

A la fidélité de nos amis,

A toute la famille Benladghem et Benyahia,

A nos généreux formateurs qui n'ont pas ménagés aucun effort à notre compte,

A toute personne qui a contribué de près ou de loin pour la réalisation de ce travail.



Djazia et Nabila

Remerciement

En préambule à ce mémoire nous remercions **ALLAH** qui nous aide et nous donne la patience et le courage durant ces longues années d'étude.

Nous souhaitons adresser nos remerciements les plus sincères aux personnes qui nous ont apporté leur aide et qui ont contribué à l'élaboration de ce mémoire ainsi qu'à la réussite de cette formidable année universitaire.

Nous tenant à remercier sincèrement Mr **BORSALI Riad**, en tant qu'encadreur, qui à toujours su montré à l'écoute et très disponible tout au long de la réalisation de ce mémoire, nos vifs remerciements vont aux membres du jury de nous avoir honoré en acceptant d'évaluer ce travail.

Nos remerciements vont aussi à tous les enseignants du département de télécommunication qui ont contribué à notre formation.

Enfin, nous adressons nos plus sincères remerciements à tous nos proches et amis, qui nous ont toujours soutenue et encouragée au cours de la réalisation de ce mémoire.

Merci à tous.

Djazia BENLADGHEM Nabila BENYAHIA

Table des matières

Liste des figures :	vii
Liste des tableaux :	ix
Introduction général :	1

Chapitre I : Généralité sur la domotique.

1. Introduction sur la domotique	4
2. Généralité sur La domotique	4
2.1. Définition :	4
2.2. Les types de maisons développées par le système domotique	5
2.2.1. Maison communicante	5
2.2.2. Maison intelligente	6
2.2.3. Maison connectée	6
2.3. Historique	6
3. Structure d'un système domotique	7
3.1. le cerveau :	7
3.2. Les capteurs :	8
3.3. Les actionneurs :	8
4. Les protocoles de la domotique	8
4.1. Domotique sans fils	8
4.1.1. Les protocoles à courte portée	9
4.1.2. Les protocoles à longue portée	11
4.2. Domotique à Courant porteur CPL	12
4.3. Domotique câblé	13
5. Avantages et inconvénients de la domotique	15
5.1. Avantage	15
5.2. Inconvénient :	15
6. Le cout de la domotique	16
7. Différent domaines de la domotique	16
7.1. Confort	17
7.2. L'économie d'énergies	17
7.3. La santé	17
7.4. La sécurité	18
7.4.1. Les alarmes intelligentes	18
7.4.2. Les tags	20

7.4.3.	Sonnette connectée associée à une caméra.....	20
7.4.4.	Les caméras connectées.....	20
7.4.5.	Serrure connectée.....	21
8.	Conclusion :	22

Chapitre II : Plateforme Arduino.

1.	Introduction.....	24
2.	Présentation d'Arduino	24
2.1.	Définition.....	24
2.2.	Historique	25
2.3.	Domaine d'utilisation et ses applications	25
2.4.	Les atouts de l'Arduino	25
2.5.	Différents types de cartes	26
2.6.	Différent type de carte Arduino.....	27
3.	Présentation de la carte Arduino UNO :	27
3.1.	Définition carte Arduino UNO.....	28
3.2.	Différents éléments de la carte Arduino Uno	28
3.3.	Caractéristique de la carte Arduino UNO :	29
3.4.	Description générale d'Arduino Uno.....	29
3.4.1.	Partie matérielle :	30
3.4.2.	Partie logicielle.....	35
4.	Conclusion.....	42

Chapitre III : Description du projet.

1.	Introduction.....	44
2.	Définition du système	44
3.	Le schéma théorique du montage	44
4.	Principe de fonctionnement.....	45
5.	Description des composants utilisés.....	46
5.1.	Arduino UNO :	46
5.2.	Les serrures connectées RFID.....	46
5.2.1.	Définition :	46
5.2.2.	Les composants d'un système RFID.....	47
5.2.3.	Les gammes de fréquences RFID.....	48
5.2.4.	Module RFID RC522.....	48
5.2.5.	Connectique du module RFID-RC522	48
5.3.	Module GSM SIM900 :	49

5.3.1.	Les caractéristiques principales :	50
5.3.2.	Vue sur l'ensemble du module GSM :	50
5.4.	L'écran LCD I2C	52
5.4.1.	Connecteur de l'afficheur LCD	52
5.4.2.	Communication avec le LCD	52
5.4.3.	Pilotage via le bus I2C	53
5.4.4.	Caractéristiques :	53
5.4.5.	Connectivité avec Arduino :	53
5.5.	Servomoteur	54
5.5.1.	Composition d'un servomoteur	54
5.5.2.	Type des servomoteurs	55
5.5.3.	Fonctionnement du système	56
5.5.4.	La connectivité avec l'Arduino	56
5.6.	Clavier 4X4	57
5.6.1.	Branchement avec la carte Arduino	58
5.7.	Buzzer	58
5.7.1.	Caractéristique du Buzzer	59
5.7.2.	Branchement avec l'Arduino :	59
5.8.	Les LEDs	59
5.8.1.	Branchement avec l'Arduino :	60
6.	Résultats pratiques	61
7.	Conclusion	62
	Conclusion générale	64
	Résumé :	67
	Référence :	69
	Annexes	72

Liste des figures :

Figure 1: La maison domotique.....	5
Figure 2 : logo du protocole KNX.....	14
Figure 3: les domaines de la Smart House.....	16
Figure 4: Modèle d'une alarme d'incendie.....	19
Figure 5: Modèle d'une alarme à gaz.....	19
Figure 6 : Modèle d'une alarme anti-intrusion.....	19
Figure 7 : Modèle d'une sonnette connectée.....	20
Figure 8: 3 différents modèle d'une serrure connectée.....	22
Figure 9: L'interface de la carte Arduino.....	24
Figure 10 : Arduino équipée d'un Shield Ethernet.....	26
Figure 11 : modèle Arduino Uno.....	28
Figure 12 : Schéma d'une carte Arduino Uno.....	29
Figure 13 : Deux modèles d'Arduino Uno.....	30
Figure 14 : Schéma simplifié du contenu type d'un microcontrôleur.....	31
Figure 15 : Les broches numériques (digitales).....	33
Figure 16 : Les broches analogiques.....	34
Figure 17 : Le branchement de la carte Arduino avec l'ordinateur via un câble USB.....	36
Figure 18 : Interface du logiciel Arduino.....	37
Figure 19 : Le menu du logiciel Arduino.....	37
Figure 20 : Quelques boutons du logiciel.....	38
Figure 21 : le logiciel IDE.....	39
Figure 22 : Exemple d'un programme (sous IDE).....	39
Figure 23 : Les différents types utilisés.....	40
Figure 24 : menu pour importation des bibliothèques.....	41
Figure 25 : menu pour la sélection de la bonne carte.....	41
Figure 26 : Montage en plaque d'essai.....	44
Figure 27 : modèle RFID RC522.....	47
Figure 28 : Les composants d'un système RFID.....	47
Figure 29 : câblage RFID RC522 avec Arduino UNO.....	49
Figure 30 : les photos de module GSM.....	49
Figure 31 : Vue d'ensemble du module GSM.....	51
Figure 32 : Branchement du sim900 avec l'Arduino.....	51
Figure 33: Nomenclature du connecteur de l'afficheur LCD.....	52
Figure 34 : Afficheur LCD 16x2 avec I2C.....	53
Figure 35 : Branchement écran LCD I2C avec Arduino.....	54
Figure 36 : schéma représentatif des composants de servomoteur à l'intérieur et extérieur.....	55
Figure 37 : les fils de servomoteur (alimentation, masse et l'entrée du signal).....	55
Figure 38 : Caractéristiques des servomoteurs.....	56

Figure 39 : Le câblage du servomoteur SG90 avec l'Arduino UNO.	57
Figure 40 : modèle d'un clavier matricielle 4x4.....	58
Figure 41 : Câblage clavier 4x4 avec Arduino UNO.....	58
Figure 42 : Le câblage du buzzer dans la plaque d'essai et Arduino Uno.	59
Figure 43 : LEDs de différentes couleurs et le symbole de la diode électroluminescente	60
Figure 44 : branchement des LEDs avec l'Arduino.	61
Figure 45 : la réalisation de la fonction d'accès sécurisé à l'habitat.....	62

Liste des tableaux :

Tableau 1 : Comparaison des différents modèles des cartes Arduino	27
Tableau 2 : Caractéristiques principales de la carte Uno.....	29
Tableau 3: Rôle des interfère (broches) dans le microprocesseur.....	33
Tableau 4 : principales caractéristique du SIM900.....	50



Introduction général :

Introduction général :

Le développement technologique et l'évolution du mode de vie permet aujourd'hui de prévoir des logements en progression qui suit le mouvement. De même, de développer la qualité de vie au sein de l'habitat en améliorant tous les aspects qui permettent d'augmenter le niveau de confort et le sentiment de sûreté et sérénité car nos foyers sont dépourvus de système de sécurité, le taux de cambriolage et d'intrusion des habitats sont élevé et le nombre de victime est important, c'est là où la Domotique apparait pour améliorer la vie quotidienne.

La domotique regroupe les technologies de l'électronique, de l'automatique, de l'informatique et des télécommunications, ou encore un terme plus courant, la maison intelligente ou communicante est l'ensemble des objets connectés qui devient une résidence équipée de plusieurs technologies 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 (ouvrir/fermer les portes, allumée/éteindre la lumière à distance lors d'une absence imprévue...) et surtout fortifier le niveau de sécurité.

L'être humain cherche toujours à mettre en place un système de sécurité et de surveillance fiable afin d'augmenter la protection de ses biens immobilier et les locaux collectifs contre les intrusions et cambriolage et les prévenir contre le vol. Les serrures intégrer dans le système de verrouillage des porte ont pour but d'assurer cette tache depuis longtemps et ne cesse pas à évoluer jusqu'à nos jours dont on trouve des serrures dites intelligentes permet de gérer l'accès aux endroits privé d'une manière très pratique et en ajoutant un système d'alarme en cas d'intrusions. L'évolution technologique a permis le développement des systèmes de sécurité qui deviennent de plus en plus performants. Cette évolution est due essentiellement à l'utilisation des applications de l'électronique moderne du point de vue communication entre les périphériques de commande (Bluetooth, WIFI, Infra rouge...) et coté composants (microcontrôleurs programmables, carte ARDUINO...).

Dans ce cadre, notre projet qui s'intitule « système de verrouillage de la porte et système d'alarme à base de Arduino et RFID » consiste en la réalisation d'une

serrure électronique qui sera notre système d'accès et aussi d'un système d'alarme codée et programmée à l'aide d'une carte ARDUINO Uno.

Ce projet sera divisé en trois chapitres. Dans le premier chapitre nous allons élaborer des généralités sur la domotique et étudier ces différents domaines d'application notamment dans la sécurité.

Le deuxième chapitre est consacré à la mise en place de la domotique, qui est l'interaction entre l'ordinateur et le monde extérieur grâce à une carte Arduino, avec une étude approfondie sur cette carte et sa programmation pour cela nous allons la voir en détail et voir ses caractéristiques, ses modèles et spécifier la carte d'Arduino Uno.

Enfin, pour le dernier chapitre, nous allons aborder la partie pratique du projet et pour augmenter et fortifier le niveau de sécurité de l'accès qui consiste à réaliser un système de verrouillage de porte et d'alarme, en présentant les différentes parties matérielle et logiciel. Une description générale du système est présenté et une conclusion générale pour finir.

.

Chapitre I : généralité sur la domotique.

Sommaire

1.	Introduction sur la domotique	4
2.	Généralité sur La domotique	4
2.1.	Définition :	4
2.2.	Les types de maisons développées par le système domotique	5
2.3.	Historique	6
3.	Structure d'un système domotique	7
3.1.	le cerveau :	7
3.2.	Les capteurs :	8
3.3.	Les actionneurs :	8
4.	Les protocoles de la domotique	8
4.1.	Domotique sans fils	8
4.2.	Domotique à Courant porteur CPL	12
4.3.	Domotique câblé.....	13
5.	Avantages et inconvénients de la domotique	15
5.1.	Avantage.....	15
5.2.	Inconvénient :	15
6.	Le cout de la domotique	16
7.	Différent domaines de la domotique	16
7.1.	Confort.....	17
7.2.	L'économie d'énergies.....	17
7.3.	La santé	17
7.4.	La sécurité	18
8.	Conclusion :	22

1. Introduction sur la domotique

La domotique, maison intelligente, maison communicante ou maison de futur ce sont des différents termes pour exprimer une maison entièrement automatique et autonome.

Dans ce chapitre, nous allons présenter en premier lieux généralité sur la domotique et les types de maison qui existe qui sont les maisons communicantes, intelligentes et connectée d'une manière générale, et d'une manière détaillée on expliquera le point commun entre eux qui est la domotique ainsi que ces domaines d'application spécialement au niveau de la sécurité.

2. Généralité sur La domotique

2.1. Définition :

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.

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 elle était 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. [1]



Figure 1: La maison domotique.

2.2. Les types de maisons développées par le système domotique

La domotique a surtout elle-même évolué, si bien que les termes peuvent être différents mais reflète au même sens qui est la domotique. La domotique servait à automatiser les maisons ; aujourd'hui on parle de plusieurs types de maisons développées comme : domotique 2.0 qui est la « maison intelligente », maison communicante ou maison connectée. Les différents domaines de la maison ne se contentent plus d'être automatisés domotisée, robotisée, technologisée et pilotables, ils communiquent ensemble, permettant à la maison de réagir selon différents événements.

2.2.1. Maison communicante

La maison communicante est l'évolution des maisons normales dites traditionnelles. Elle permet de passer à des paramétrages automatiques, elle s'appuie sur l'utilisation de la domotique et du multimédia au sein de l'habitat pour constituer un réseau domestique numérique intelligent ; puis elle maintient le fonctionnement parfait pour répondre aux moindres besoins et demandes des occupants de la maison.

Elle est reliée à une centralisation de commandes. Les appareils mis en réseau se reconnaissent et dialoguent entre eux, et en couplant l'installation avec une télécommande universelle ou avec un simple appui sur une touche sur son Smartphone ou tablette, le pilotage s'effectue de n'importe où dans la maison.

2.2.2. Maison intelligente

Une maison intelligente est une résidence qui utilise des appareils connectés à Internet pour permettre la surveillance et la gestion à distance d'appareils et de systèmes, tels que l'éclairage et le chauffage.

Une maison intelligente est une installation à domicile pratique où les appareils peuvent être commandés automatiquement à distance depuis n'importe où dans le monde en utilisant un appareil mobile ou un autre appareil en réseau. Une maison intelligente a ses dispositifs interconnectés par Internet, qui contrôlent des fonctions telles que l'accès de sécurité à la maison, la température, l'éclairage et le cinéma maison. Les termes connexes comprennent «domotique» et «bâtiment intelligent». Les appareils d'une maison intelligente sont tous reliés entre eux et accessibles par un point central : un Smartphone, une tablette, un ordinateur portable ou même une console de jeux. Les serrures de porte, les téléviseurs, les thermostats, les moniteurs de maison, les appareils-photo, les lumières et même les appareils tels que le réfrigérateur peuvent être commandés par l'intermédiaire d'un système domotique.

2.2.3. Maison connectée

C'est une maison dotée de capteurs en tous genres, quasiment invisibles mais permettant la connexion avec votre Smartphone ou votre tablette via internet. Ces capteurs vous permettront de réaliser des économies sans contraintes et sans perte de temps. Grâce à votre Smartphone, on peut recevoir des alertes lorsque des intrus tentent de faire une intrusion, ou lorsqu'une personne âgée maintenue à domicile est en situation de détresse. Gérez les accès internet, l'enregistrement de vos émissions tv, la température de la maison, les programmes des machines à laver sans modération, où que vous soyez et en toute simplicité.

2.3. 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 20^e siècle, que la domotique va se démocratiser.

Deux raisons expliquent ce développement :

L'arrivée de l'ordinateur et des technologies de communication dans la maison au début des années 1990 ; notamment, le déploiement d'Internet qui permet aux ordinateurs de communiquer entre eux.

Le coût de l'énergie qui augmente suite aux deux crises pétrolières survenues dans les années 70. Désormais, de nouvelles normes forcent les constructeurs privilégiés 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]

3. Structure d'un système domotique

Pour domotiser sa maison, il faudra mettre en place un système domotique. Un tel système est toujours constitué des mêmes équipements, quelque soit le système et la technologie utilisés :

3.1. le cerveau :

Que ce soit un automate, un ordinateur, ou plus communément aujourd'hui une "box domotique". C'est lui qui va centraliser toutes les informations de votre maison et déclencher des actions.

Il est dit le cerveau, car c'est réellement un équipement qui va donner de l'intelligence aux maisons. [3]

3.2. Les capteurs :

Les capteurs sont des périphériques qui vont donner des sens à votre maison :

Relever la température, l'humidité, la luminosité, le niveau de CO₂, le niveau de bruit, détecter une présence, de la fumée, une fuite de gaz, suivre la consommation électrique des appareils, etc ...

Grâce à eux la maison saura tout ce qui se passe. [3]

3.3. Les actionneurs :

Les actionneurs sont donc des périphériques qui vont permettre de piloter des appareils (radiateurs, chaudière, TV, machine à laver...), des lampes, ou encore des automatismes (volets, porte de garage, store banne, etc...). Le « cerveau », grâce aux informations recueillies par les différents capteurs disséminés à travers la maison, va pouvoir déclencher des actions en conséquence.

Par exemple, si aucun détecteur de présence dans la maison ne détecte personne, le « cerveau » pourra demander aux radiateurs de passer en mode éco et aux lampes de s'éteindre. [3]

4. Les protocoles 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.

4.1. Domotique sans fils

La domotique sans fils 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. [2]

Les principales fréquences utilisées dans la domotique sont le 433 MHz et le 868MHz. On trouve parmi les protocoles sans fil ceux qui ont une courte portée et d'autre qui sont de longue portée :

4.1.1. Les protocoles à courte portée

- **Z-Wave**

Z-wave est un protocole de communication dédié à la domotique. Ce protocole sans fil est facile à installer et offre une portée de 30 mètres. Z-Wave est un réseau maillé, impliquant que tous les objets connectés au système émettent des données et peut relayer celles émises par ses voisins. Le réseau maillé va permettre d'élargir la portée des appareils. Le Z-Wave est une technologie bidirectionnelle à retour d'état permettant de garantir la bonne exécution des ordres. Très rapide et beaucoup plus sûre par l'utilisation d'un code sur 32 bits qui bloque tout équipement ne disposant pas du bon code. Z-Wave utilise une technologie radio de faible puissance dans la bande de fréquence de 868,42 MHz.

Ce protocole permet de connecter jusqu'à 232 appareils. Le Z-Wave est utilisé par un large éventail de produit et permet d'équiper sa maison pour un budget bien inférieur à certaines technologies filaires tout en garantissant fiabilité et sécurité. De plus, Z-wave utilise un protocole bien plus simple que les autres permettant un développement plus simple et plus rapide.

- **Zig Bee**

Comme le Z-wave, Zigbee est un réseau maillé. Zigbee permet de transférer un nombre de données plus important que le Z-wave et est moins chère et plus facile d'utilisation pour les fabricants d'objets connectés que le Z-wave ou Bluetooth. Cependant, la portée moyenne pour le Zigbee n'est que de 100 mètres.

Zigbee est basé sur le protocole IEEE802.15.4, un réseau sans fil industriel qui opère à 2.4Hz ciblant les applications qui reçoivent une faible quantité de données sur une zone limitée comme une maison ou un bâtiment.

Zigbee présente de nombreux avantages tel qu'un fonctionnement basse consommation, sécurité, robustesse, une grande évolutivité et se retrouve en bonne position pour les systèmes machine to machine et applications IoT.

- **WI-FI**

Le Wi-Fi permet de transférer rapidement un grand nombre de données. Le Wi-Fi est une technologie de transmission haut-débit sans fil qui utilise les ondes radio. Il s'agit de la dénomination de la norme IEEE 802.11 qui est le standard international décrivant les caractéristiques d'un réseau local sans fil (WLAN). Le protocole 802.11b autorise un débit de 11 Mbits à 22 Mbits par seconde, alors que le protocole 802.11g permet d'atteindre un débit théorique de 54 Mbits/s. La technologie utilise la radio fréquence pour permettre à deux appareils de communiquer entre eux. Cette technologie est communément utilisée pour connecter des routeurs aux ordinateurs, tablettes et smartphones. Le Wi-Fi peut également être utilisé pour connecter des objets. La technologie Wi-Fi est la technologie la plus répandue. Cette technologie utilise une large bande passante et requiert une importante quantité d'énergie. Cette solution consomme donc beaucoup d'énergie et les composants nécessaires sont coûteux ce qui fait que la plupart des concepteurs d'objets connectés évitent cette technologie.

- **Bluetooth**

Le Bluetooth établit une connexion sécurisée de proximité. L'unique problème de cette technologie est que les objets connectés doivent être proche et à distance réduite. Cependant, le fait que la connexion se fasse sur une courte distance va réduire les risques d'interception du signal.

Le système Bluetooth opère dans les bandes de fréquences ISM (Industrial, Scientific and Medical) 2,4 GHz dont l'exploitation ne nécessite pas de licence vu la faible puissance d'émission et le risque faible d'interférences. Cette bande de fréquences est comprise entre 2 400 et 2 483,5 MHz.

Le Bluetooth est une technologie utilisée pour envoyer des informations sur de courtes distances. Cette technologie est en général utilisée sur des appareils connectés aux téléphones et tablette. Le Bluetooth Low-energy utilise lui moins d'énergie que le Bluetooth standard et est utilisé en général sur des objets connectés dans le but de transmettre des données sans compromettre la batterie. Le Bluetooth Lowenergy est très largement utilisé dans le monde. Pratiquement tous les smartphones sont équipés de cette technologie qui offre une portée d'environ 60 mètres et qui consomme moins d'énergie que le Wifi. Aujourd'hui le Bluetooth lowenergy, appelé Bluetooth Smart, est supporté par la majorité des smartphones, ordinateurs et systèmes d'exploitation. Cela

lui offre un avantage notable dans le cadre des objets connectés étant donné son intégration répandue dans les smartphones et autres appareils mobile.

4.1.2. Les protocoles à longue portée

- **Lora**

Lora est un protocole basse consommation, à faible coût et qui permet un transfert des données sur des distances de 2 à 5 kilomètres en milieu urbain et jusqu'à 45 kilomètres en milieu rural.

Lora cible une large gamme d'applications et est conçu pour offrir une connexion de faible puissance avec des caractéristiques favorables pour supporter une communication bidirectionnelle sécurisée mobile pour les objets connectés, le machine to machine et autres. Lora est optimisé pour une faible consommation et supporte de larges réseaux avec des millions d'appareils et de données. Ce réseau est intéressant pour faire transférer une faible quantité de données mais ne peut faire transiter que 0,3 à 0,5 kilobits par secondes.

- **Sigfox**

Sigfox en termes de portée se positionne entre la Wi-Fi et la connexion cellulaire. En ville, Sigfox a une portée de 10Km et peut atteindre jusqu'à 50km en campagne. Il utilise des bandes ISM, lesquelles sont libres d'utilisation, pour transmettre des données d'un spectre très étroit à ou depuis des **objets connectés**. Avec une grande quantité d'applications machine to machine qui fonctionnent avec une batterie faible et qui ne peuvent faire transiter qu'un faible volume de données Sigfox apparaît comme une solution intéressante. En effet, la portée de la Wi-Fi est trop courte, la technologie cellulaire trop chère et les deux consomment trop d'énergie. Sigfox utilise une technologie appelée Ultra Narrow Band et est uniquement conçu pour gérer une faible vitesse de transfert des données. Il consomme seulement 50 microwatts comparé à 5000 microwatts pour la communication cellulaire. En contrepartie de sa faible consommation d'énergie, Sigfox ne permet de transporter que de très faibles quantités de données, entre 10 et 100 bits par seconde maximum (bps). Monodirectionnel au départ, il permet désormais d'envoyer des informations à ses **objets connectés**, même s'il est impossible de réaliser rapidement d'importantes mises à jour. Déjà déployé dans plus de 10 000 **objets connectés**, ce réseau est utilisé dans les principales villes d'Europe. Ce réseau est robuste et peu communiquer avec des millions

d'appareils fonctionnant sur batteries sur des distances de plusieurs kilomètres carré, en faisant un réseau intéressant pour les applications machine to machine.

- **Cellulaire**

Le principal avantage des réseaux cellulaires est qu'ils permettent de transférer d'importantes quantités de données. Très gourmands en énergie ces réseaux sont à réserver à des appareils branchés au secteur. Le prix des abonnements (qui dépend de l'opérateur choisi) est nettement plus élevé que ceux de Sigfox et Lora.

- **EnOcean**

EnOcean est basée sur un protocole de communication demandant très peu d'énergie pour envoyer ses trames radio sans fil et sans pile. EnOcean utilise la bande des 868MhZ. La technologie EnOcean est autonome et puise l'énergie de son environnement pour émettre un signal et communiquer avec d'autres produits EnOcean. Il s'agit d'un protocole de communication sans fil et donc facile à installer. La maintenance des périphériques équipés d'un protocole EnOcean est proche de zéro car ils n'utilisent aucune pile contrairement aux capteurs Z-wave, très gourmand en énergie. La durée de vie des **objets connectés** est donc rallongée. Son principe de fonctionnement est extrêmement simple et ne nécessite pas obligatoirement de contrôleur domotique. EnOcean est 100% Standalone. Les nouveaux produits EnOcean sont pour la plupart bidirectionnel et peuvent même faire office de répéteur wifi.

Il a une portée sans fil pouvant aller de 30 mètres en intérieur à 300 mètres en extérieur suivant l'environnement et les conditions d'emploi. Mais aussi et surtout, tout produit EnOcean est défini par un Profil d'Équipement « EEP » développé par l'Alliance EnOcean, une norme internationale entièrement ouverte et interopérable. Une sécurité des données renforcée avec le code de roulement (Rolling Code) et le cryptage AES 128. Contrairement au Wi-Fi ou GSM, la technologie EnOcean ne dialogue que très peu, pas d'émission radio en continu ce qui en fait également un atout pour les réticents aux ondes... Il ne subit pas d'interférences liées à l'environnement Wi-Fi, 4G, DECT,[4]

4.2. Domotique à Courant porteur CPL

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

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

renovation. L'intérêt de cette technologie porte sur l'utilisation d'un réseau filaire structuré déjà existant et parfaitement distribué dans toute la maison ou le bâtiment : le réseau électrique et ses points d'accès constitués par les prises électriques. Toutefois, la fiabilité de la domotique CPL est contestable. Cette technologie peut parasiter le réseau et perturber les autres transmissions. De plus, cet équipement est encore coûteux. À performances équivalentes, il est en effet plus cher que le sans fil. Enfin, ce système est aussi moins rapide, et il n'a pas de mobilité par construction. [2]

4.3. Domotique câblé

La technologie bus filaire, est souvent utilisée dans la construction ou la rénovation de bâtiments en raison de l'installation d'un bus filaire. Cette technologie veille à ce que tous les composants communiquent entre eux avec le même langage afin qu'ils puissent échanger des informations, les analyser et les traiter.

L'information circule dans les deux sens : une unité d'entrée envoie des informations aux récepteurs de sortie chargés de faire effectuer une tâche précise à des équipements de l'installation électrique (éclairage, ouvrants, chauffage, alarmes...). Ces derniers envoient ensuite des informations concernant leur état vers la ou les unités d'entrée.

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

- Un réseau bus filaire reliant les capteurs (détecteurs, interrupteurs, sondes) aux actionneurs (éclairage, ouvrants, chauffage, produits de puissance) ;
- Un réseau d'alimentation reliant les actionneurs au courant fort ;

Ce type d'installation présente pour les utilisateurs plusieurs avantages :

- La réduction massive du câblage : un seul câble en général pour tous les équipements au lieu d'un par équipement ;
- Une meilleure fiabilité de la transmission des informations et de l'installation ;
- Une supervision en local ou à distance ;
- Une évolutivité de l'installation à tout moment

Exemple de réseau bus filaire (Technologie KNX) :



Figure 2 : logo du protocole KNX

La technologie KNX se compose de deux couches techniques principales :

- **La couche BCU [Bus Coupler Unit]**

La couche BCU est composée d'un micro-processeur et d'une mémoire de stockage du programme destiné au produit qui lui est attaché [interrupteur, détecteur, sonde...]. Ce programme est défini -comme indiqué ci-avant - par le concepteur/fabricant du produit.

- **La couche KNX**

Il s'agit du système de communication normalisé (filaire, radio) qui permet à tous les composants de se connecter entre eux, d'échanger des informations, de les analyser et de les traiter.

Dans le cas d'une communication filaire, le terme de BUS KNX est employé pour désigner le câble qui sert aux transmissions. KNX utilise pour cela un câble contenant une paire torsadée : il est donc insensible aux perturbations électriques et électromagnétiques.

Ce bus (KNX) est un système à intelligence répartie ne nécessitant pas d'ordinateur ou d'automate central. Chaque composant [ou participant] KNX relié au bus, disposant des couches BCU et KNX, est capable d'émettre et de recevoir des messages en toute autonomie.

L'information circule dans les deux sens sur le réseau : Une unité d'entrée envoie des informations aux récepteurs de sortie chargés de faire effectuer une tâche précise à des appareils qui eux-mêmes envoient vers la ou les unités d'entrée des informations concernant leur état. [5]

5. Avantages et inconvénients de la domotique

La domotique présente de nombreux atouts mais aussi des inconvénients, parmi lesquels :

5.1. Avantage

- **Le confort ménager** : est un des principaux avantages de la domotique, il permet de programmer la maison comme l'utilisateur le souhaite.
- **Sécurité** : une maison domotisée est souvent mieux protégée contre les cambriolages. Le système envoie les informations en temps réel s'il détecte une intrusion.
- **Gain de temps** : La programmation à distance permet de gagner du temps et en s'éloignant de la routine habituelle.
- **Economie d'argent** : La domotique permet de faire diminuer jusqu'à 40 % des factures d'énergie avec un système de réglage automatique qui dépend du temps et le moment de la journée.
- **Solution écologique** : La domotique réduit la consommation d'énergie et par conséquent les émissions de dioxyde de carbone.

5.2. Inconvénient :

- **Frais d'équipement et d'installation onéreux** : une installation partielle ou totale d'appareils et d'accessoires domotiques nécessite un investissement important.
- **Système bloqué** : le système peut se bloquer suite à des dommages au niveau du réseau central. Une rupture des câbles ou des fibres, provoque un plantage dans la plupart des systèmes. En outre, s'il existe plusieurs données à transmettre à travers le réseau, cela peut également ralentir les diverses programmations.
- **Accès difficile** : les connaissances en informatique est une nécessité, car les sites spécialisés dans le domaine sont difficiles d'accès pour les débutants. Les fabricants doivent simplifier les applications au maximum pour que chacun puisse profiter totalement des avantages de ce système.
- **Risques de piratage** : le fonctionnement de la domotique est uniquement via un réseau informatique, donc le risque du piratage augmente. En effet des individus malveillants peuvent exploiter les failles de votre système et les utiliser pour de mauvaises intentions.

6. Le cout de la domotique

Pour les marques connus dans le domaine des maisons connectées, l'installation domotique ça commence par un budget de 10000€ au minimum. Après une recherche sur internet on a trouvé des chercheurs dans ce domaine qui font des efforts pour la réalisation des maisons intelligentes avec un budget moins cher.

Exemple :<< Extrait d'un ouvrage >>

Pour commencer il faudra une box domotique pour piloter la maison. On trouve des box allant de 150 à 600€ environ, en restant dans le « grand public », car il y a beaucoup plus cher. Disons que pour 200€ on trouve des box très corrects, le prix n'étant pas spécialement un gage de qualité.

Ensuite, le budget va surtout dépendre du nombre de périphériques qu'on va mettre en place, et de la technologie utilisée, certaines étant plus couteuses que d'autres.[6]

7. Différent domaines de la domotique

La domotique utilise plusieurs critères clés : la sécurité (alarmes, caméras et télésurveillance), le confort de vie (automatisation et programmation des tâches quotidiennes), les économies d'énergies (chauffage, lumière), la santé (télésanté, télémedecine) et la communication (avec un réseau, wifi Bluetooth).



Figure 3: les domaines de la Smart House

7.1. Confort

La domotique a tellement évolué qu'on l'appelle aujourd'hui, la robotique domestique. Ces robots ménagers qui existent sont des appareils conçus pour faciliter la vie. Ils améliorent notre confort et notre bien-être.

Pour l'amélioration des maisons il faut donc installer trois fonctions qui rendent la vie plus meilleure :

- **Contrôle des appareils à distance** : grâce à la domotique vous pouvez désormais avoir le contrôle sur votre maison. En cas d'avoir oublié votre volet où vos lumières allumée, par exemple, vous pouvez régler cela via une application sur votre téléphone portable ou tablette.
- **Configuration de l'ambiance** : après une dure journée vous avez envie de vous relaxer et vous détendre, la domotique s'occupe de tout, de l'éclairage d'ambiance à la chaleur agréable. Elle vous offre tout ce que vous allez besoin.
- **Éclairage, ventilation et chauffage** : la possibilité de régler la température de la maison selon nos besoins, activer la lumière et renouveler l'air grâce au ventilateur mis en marche.

7.2. L'économie d'énergies

Le système domotique vous permet d'économiser des factures d'énergie et donc de l'argent. Le simple fait d'activer l'alarme en partant va passer le chauffage en mode éco par exemple (il est possible de gérer les apports naturels en fonction de l'enveloppe thermique du bâtiment). Et éteindre toutes les lampes et les appareils reste en veille, de l'ouverture et de la fermeture des volets (en fonction de l'ensoleillement ou de l'heure de la journée, par exemple), de l'eau (le remplissage de la baignoire peut s'arrêter automatiquement grâce à un capteur, les robinets de lavabos peuvent ouvrir l'eau à l'approche des mains, etc.). Réduisant ainsi votre consommation d'énergie en votre absence et ceci sans aucune action de votre part, c'est cela la domotique.

7.3. La santé

La domotique trouve aujourd'hui de nouvelles applications dans le domaine de la santé. En installant des systèmes 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 scénarios programmés dans l'interface de commande. [7]

7.4. La sécurité

La sécurité dans une maison est un des critères important à trouver. La domotique est un système qui permet de mieux sécuriser l'habitat avec l'installation de divers équipements au sein d'un foyer commandé par une centrale dédiée.

7.4.1. Les alarmes intelligentes

Une alarme est principalement composée des éléments suivants :

- **Un ou plusieurs détecteurs, selon les besoins** : détecteur de mouvement, détecteur d'ouverture, détecteur de fuite, détecteur de fumée, détecteur de coupure de courant...
- **Une centrale**, qui commande l'ensemble du **système d'alarme**. C'est elle qui reçoit les informations et agit en fonction des paramètres que vous lui demandez.
- Une **sirène** et, éventuellement en plus, un **transmetteur téléphonique** qui envoie une alerte
- Une commande fixe ou mobile qui permet d'activer ou de désactiver la centrale d'alarme.

Voici quelques alarmes qui permettent d'augmenter la sécurité dans les maisons :

1. **L'alarme à incendie** : permet de sauver des vies. En effet, dans la majorité des cas, les personnes qui décèdent à leur domicile suite à un incendie étaient tout simplement endormies lors du départ du feu. Dans l'idéal, il faut un détecteur de fumée pour chaque étage, notamment dans les chambres ou l'escalier.



Figure 4: Modèle d'une alarme d'incendie.

2. **L'alarme à gaz** : est un détecteur pour le gaz naturel ou le monoxyde de carbone qui trouve sa place à proximité des appareils à gaz ou à combustion dans la cuisine, le salon ou le cellier.



Figure 5: Modèle d'une alarme à gaz.

3. **Alarmes anti-intrusion** : se compose d'une centrale et de détecteurs de mouvement ou d'ouverture à placer de préférence dans chaque pièce, ainsi que d'une sirène intérieure et extérieure si votre logement le permet. Parfois, elle est également équipée d'un transmetteur GSM pour prévenir les propriétaires en déplacement par téléphone.

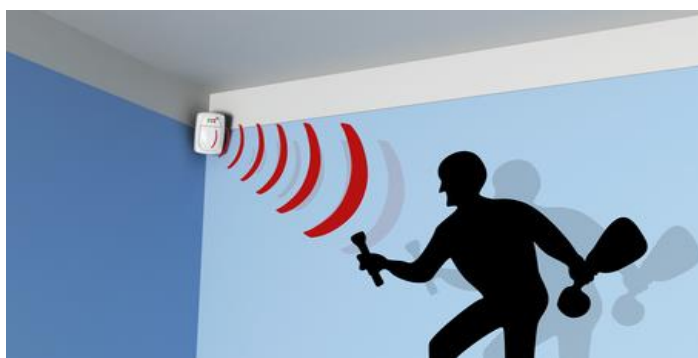


Figure 6 : Modèle d'une alarme anti-intrusion.

7.4.2. Les tags

Capteurs de sécurité à installer sur les fenêtres et les portes, les tags sont des dispositifs intelligents d'une grande efficacité en matière de détection de vibrations et de mouvements. Ce sont des objets connectés qui se caractérisent par leur principe de fonctionnement assez simple. Dès lors qu'une vibration ou un mouvement est détecté, l'envoi d'alertes se déclenche automatiquement sur vos appareils comme les tablettes et les smartphones. Les tags séduisent par leur praticité surtout pour des personnes qui oublient de fermer les portes et fenêtres par exemple avant de sortir. Malgré leur efficacité, ce sont des équipements de sécurité de la maison qui se démarquent par leur facilité d'installation. Pour preuve, il suffit de les coller simplement sur la porte ou la fenêtre pour qu'ils soient fonctionnels. En les combinant à une caméra, vous renforcerez sa performance.

7.4.3. Sonnette connectée associée à une caméra

Une autre combinaison à laquelle vous devez penser lorsque vous comptez sécuriser votre logement au moyen d'objets connectés est celle d'une sonnette intelligente et une caméra de surveillance. Il s'agit d'une solution qui vous permet de bénéficier d'une vision nocturne et d'un angle de vue plus grand. Avec un tel dispositif, vous pouvez assurer une surveillance permanente des abords de votre habitation en toute sérénité. Certains modèles de sonnettes disponibles dans le commerce vous permettent de voir tout ce qui se passe autour de votre maison sur une distance de près de 10 mètres.



Figure 7 : Modèle d'une sonnette connectée.

7.4.4. Les caméras connectées

Les caméras connectées offrent une meilleure sécurité pour pouvoir surveiller votre habitation et ce qui s'y passe. Ces appareils se montrent particulièrement performants pour protéger votre demeure des intrusions ou effractions. Avec le même

principe du fonctionnement d'une webcam, elles assurent une transmission en temps réel des images puis l'envoi de notification en cas de détection d'une situation suspecte.

7.4.5. Serrure connectée

7.4.5.1 Définition :

Les serrures connectées (ou serrures intelligentes), permettent d'ouvrir les portes sans utiliser de clé physique. Par un des protocoles de communication (Bluetooth, Wifi...) elles se déverrouillent à l'aide d'un simple Smartphone.

7.4.5.2 Le fonctionnement d'une serrure connectée :

Une serrure connectée s'ouvre lorsque sont connecteur détecte la proximité d'une clé électronique, telle qu'un Smartphone ou une carte magnétique. Les clés électroniques et les droits qui leurs sont associés sont définies par un administrateur à distance, qui n'est autre que le principal utilisateur.

Les clés électroniques fonctionnent grâce à différents protocoles de communication, les principaux protocoles utilisés actuellement sont le Bluetooth, la NFC (Near Field Communication), la RFID ou directement via internet.

7.4.5.3 Les avantages des serrures connectées :

Les serrures connectées s'adapte à toutes les besoins des entreprises et des particuliers, et répond ainsi à toutes les problématiques du contrôle d'accès, ils permettent :

- Améliorer la sécurité des locaux.
- Contrôler les sites sensibles.
- Moderniser les infrastructures.
- Faciliter l'accès aux sous-traitants.
- Simplifier la gestion des plannings d'accès.
- Garantir les temps d'intervention ... [8]

7.4.5.4 Les types de serrure connectée :

On distingue plusieurs modèles de serrure électronique, en voici une liste :

- **Les modèles à clavier**, aussi connus sous le terme de « digicode », nécessitent de composer un code de chiffres et de lettres afin d'actionner l'ouverture d'une porte close.
- **Les modèles à carte magnétique** remplacent la clé par les cartes, dont le passage à travers des fentes ou devant des détecteurs permet de déverrouiller le mécanisme de sécurité. On retrouve ce type d'appareils dans le milieu professionnel.
- **Les serrures à télécommande** sont utilisées par les automobilistes pour déverrouiller les parkings.

- **Les serrures biométriques** utilisent un système de reconnaissance digitale autorise l'accès uniquement aux personnes enregistrées. On retrouve ce type de dispositif dans des zones de haute sécurité. [9]

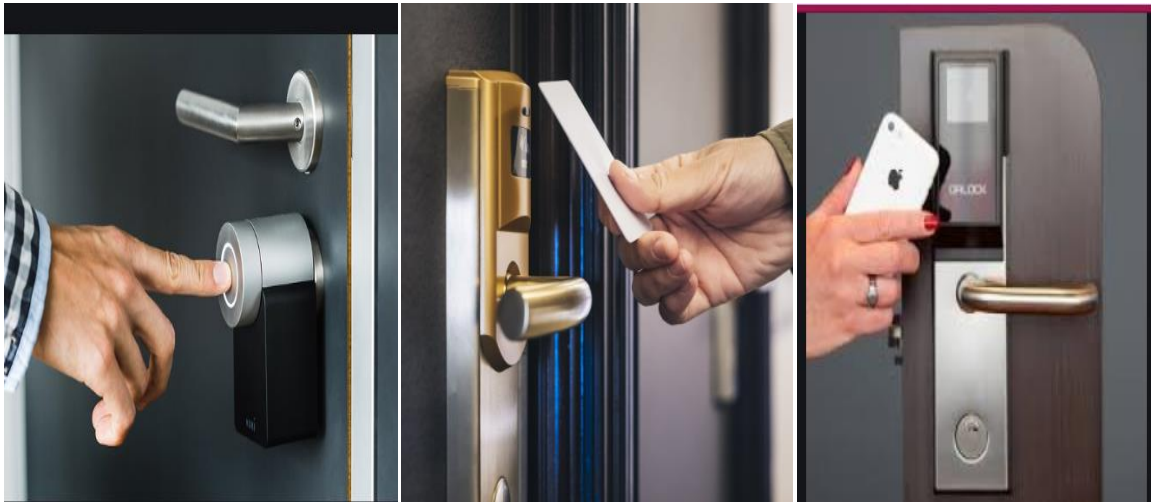


Figure 8: 3 différents modèle d'une serrure connectée.

8. Conclusion :

Ce premier chapitre est consacré à la technologie domotique, nous avons parlé dans un premier temps de généralités sur la domotique. D'abord une définition générale, puis présentation des maisons développées par le système domotique ainsi que son historique. En second lieu on a expliqué son principe de fonctionnement, des différents protocoles de communication utilisé pour les trois types de câblage, ces avantages et inconvénients et en dernier les domaines d'utilisations avec plus de détails au niveau de la sécurité.

Dans le chapitre qui suit on va détailler le système Arduino qui est un des systèmes qui permet de domotiser les maisons et représenter l'électronique programmée ou embarquée des maisons.

Sommaire

1. Introduction	24
2. Présentation d'Arduino.....	24
2.1. Définition	24
2.2. Historique.....	25
2.3. Domaine d'utilisation et ses applications	25
2.4. Les atouts de l'Arduino	25
2.5. Différents types de cartes.....	26
2.6. Différent type de carte Arduino.....	27
3. Présentation de la carte Arduino UNO :.....	27
3.1. Définition carte Arduino UNO.....	28
Figure 12 : Arduino Uno.....	28
3.2. Différents éléments de la carte Arduino Uno	28
3.3. Caractéristique de la carte Arduino UNO :.....	29
3.4. Description générale d'Arduino Uno.....	29
3.4.1. Partie matérielle :.....	30
3.4.2. Partie logicielle	35
4. Conclusion.....	42

1. Introduction

Ce chapitre est consacré à la description de la partie matérielle du projet, en identifiant, le choix du noyau de notre système domotique : la carte Arduino, ainsi que ses caractéristiques et les différents types de cette carte. On mettra la lumière sur la carte Arduino-Uno, dans la seconde partie.

2. Présentation d'Arduino

2.1. Définition

Le système Arduino est une carte électronique basée sur un microcontrôleur et de différents composants pour réaliser des applications plus ou moins évoluées à bas coût et dotée d'une interface USB pour la programmer. C'est une plateforme open-source que constituée d'une simple carte de la famille AVR, un logiciel et un environnement de développement intégré, pour écrire, compiler et transférer le programme de la carte au microcontrôleur.

Arduino peut être utilisée pour développer des applications matérielles industrielles légères ou des objets interactifs (créations artistiques par exemple) et peut recevoir en entrée une très grande variété de capteurs. Arduino peut aussi contrôler plusieurs actionneurs, lumières, moteurs ou toutes autres sorties matériels. Les projets Arduino peuvent être autonomes, ou communiquer avec des logiciels à travers un ordinateur (Flash, Processing ou MaxMSP). Les cartes électroniques peuvent être fabriquées manuellement ou bien achetées pré-assemblées (figure 10). Le logiciel de développement open-source est téléchargeable gratuitement. [10]

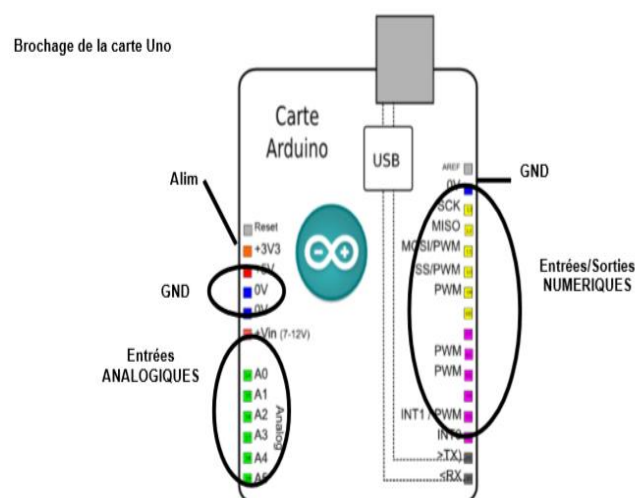


Figure 9: L'interface de la carte Arduino

2.2. Historique

Le projet Arduino fut réalisé en hiver 2005. Il emprunte son nom au bar Arduino lieu de réunion des concepteurs de la carte, à Ivree en Italie du Nord. Massimo Banzi enseigna dans une école de Design à Ivrea en Italie et souvent ses étudiants se plaignirent de ne pas pouvoir accéder à des solutions bas prix pour accomplir leurs projets de robotique. Banzi a discuté avec David Cuartielles, un ingénieur espagnol spécialisé dans les systèmes embarqués. Ainsi, ils décidèrent de créer leur propre carte en embarquée et un des étudiants de Banzi, David Melliss en chargea de la création du langage de programmation de la carte. En deux jours, David écrivit le code ! Trois jours de plus et la carte fut créée.

Ça devient un hit tout de suite auprès des étudiants. Tout le monde pouvait réaliser des applications très rapidement sans même avoir de connaissances particulières ni en électronique ni en informatique comme la manipulation des capteurs, faire clignoter des leds, contrôler des moteurs. Ils publient les schémas, investissent 3000 euros pour créer les premiers lots de cartes : Les 50 premières partent directement à des élèves de l'école. En 2006, 5 000 cartes vendues. Puis en 2007, plus de 30 000 et en 2011 >120 000, sans compter les clones et les modules équivalents.

2.3. Domaine d'utilisation et ses applications

La plate-forme Arduino sert à réaliser de nombreux projets, et créer des applications dans tous les domaines, l'étendue de l'utilisation de l'Arduino est gigantesque.

Pour cela, on propose quelques exemples d'utilisation :

- Domaine de télécom : liaison des projets avec l'IOT ou les modules GSM.
- Domaine de santé : contrôle des personnes âgées ou des bébés.
- Domaine d'agriculture : serre intelligente.
- Domotique : maison communicante, contrôle des appareils.
- Fabrication des robots.
- Électronique industrielle et embarquée.

2.4. Les atouts de l'Arduino

- **Matériel peu onéreux** : Les cartes Arduino sont relativement peu coûteuses comparativement aux autres plateformes de développement.

- **Multiplateforme** : compatible sous les plateformes les plus courantes Windows, Macintosh et Linux, contrairement aux autres outils de programmation.
- **Environnement de programmation clair et simple** : l'environnement de programmation Arduino facile à utiliser, tout en étant assez flexible pour que les utilisateurs avancés puissent en tirer profit également.
- **Logiciel et matériel open source et extensible** : le logiciel Arduino et son langage sont publiés sous licence open source, disponible pour être complété par des programmeurs expérimentés.
- **La communauté** : Nombreuses bibliothèques disponibles avec diverses fonctions implémentées, de plus on peut trouver divers conseils, tutoriaux et exemples en ligne (forums, site perso etc...).
- **Existence de « Shield » (boucliers en français)** : ce sont des cartes supplémentaires qui se connectent sur le module Arduino pour augmenter les possibilités comme par exemple : afficheur graphique couleur, interface Ethernet, GPS, etc... (Figure 11). [11]

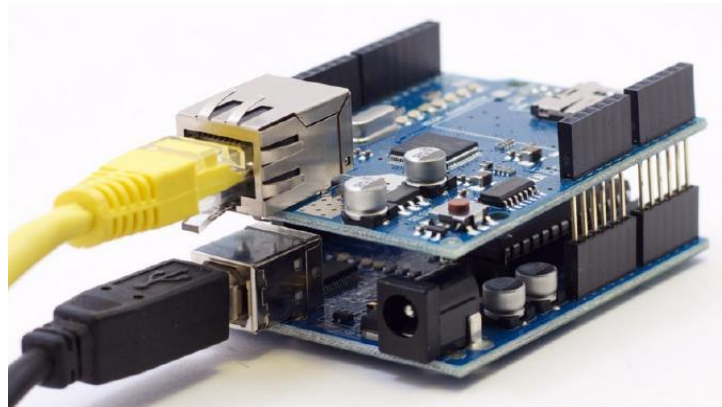


Figure 10 : Arduino équipée d'un Shield Ethernet

2.5. Différents types de cartes

Arduino est Open source, mais est aussi une marque protégée. Pour garantir la qualité de ses produits, les nouvelles cartes doivent être approuvées par l'équipe Arduino avant d'être officiellement reconnues et de pouvoir prendre le nom d'Arduino. Vous pouvez reconnaître une carte « **officielle** » avant tout par son nom Arduino Pro, Fio ou Lilypad, par exemple.

Par ailleurs, il existe des cartes qui ne sont pas officielles, elles sont souvent nommées en mentionnant « compatible Arduino » ou « pour Arduino ». On peut trouver d'autres types

fabriqués par diverse entreprise et commercialisées sous un nom différent (freeduino, Seeduino, Femtoduino...). [12]

2.6. Différent type de carte Arduino

Il existe plus de 12 cartes Arduino, le tableau suivant récapitule la comparaison entre elles :

Carte / caractère	UNO	LEONARDO	MEGA 2560	NANO		DUE	YUN
Micro-contrôleur	ATmega 328P	ATmega 32U4	ATmega 2560	• ATmega 168	• ATmega3 28P	ATSAM3X8 E	- ATmega3 2U4 - AR9331 Linux
Fréquence d'horloge	16 MHz	16 MHz	16MHz	16 MHz		84 MHz	- 16 MHz - 400 MHz
Tension Service / entrée (V)	3.3 ; 5 / 7-12	5 / 7-12	5 / 7-12	5 / 7- 9		3.3 / 7-12	- 5 / 5 - 3.5 / 5
Ports numériques ES	14	20	54	14		54	20
Ports analogiques ES	6/0	12/0	16/0	8/0		12/2	12/0
Courant max par broche d'E/S (cc)	40 mA	40 mA	40 mA	40 mA		800 mA	40 mA
Courant max par broche 3.3V	50 mA	50 mA	50 mA	-		800 mA	50 mA
Mémoire (Ko) Flash/ SRAM /EEPROM	21/ 2/1	32/2.5/1	256/8/4	ATmega 168 :16/1/0.512	ATmega3 28 : 32/2 /1	512/94/0.512	32 / 2.5 / 1 64 Mo / 16 Mo
Dimension (cm)	6.86 x5.3	6.86 x 5.3	10.16 x5.3	1.9 x 4.3		10.2 x 5.3	7 x 5.3
Prix (euro)	24	20	47	40		47	62

Tableau 1 : Comparaison des différents modèles des cartes Arduino. [1]

3. Présentation de la carte Arduino UNO :

Parmi toutes les cartes de la famille Arduino, nous avons choisi la carte Arduino UNO (carte Basique). L'intérêt principal de cette carte est de faciliter la mise en œuvre d'une telle commande qui sera détaillée par la suite.

3.1. Définition carte Arduino UNO

La carte Arduino Uno est le produit populaire parmi les cartes Arduino. Parfaite pour débiter la programmation Arduino (figure 12), elle est constituée de tous les éléments de base pour construire des objets d'une complexité relativement faible. La carte Arduino Uno, comme son nom l'indique, a été la première à utiliser la version de programmation Arduino 1.0, et elle est devenue le symbole de l'univers Arduino.

La carte Arduino Uno est constituée de 14 broches d'entrées/sorties digitales, dont six sont utilisables en PWM, de 6 broches d'entrées analogiques, d'une connectique USB, d'une connectique d'alimentation, d'un port ICSP et d'un bouton RESET. [13]

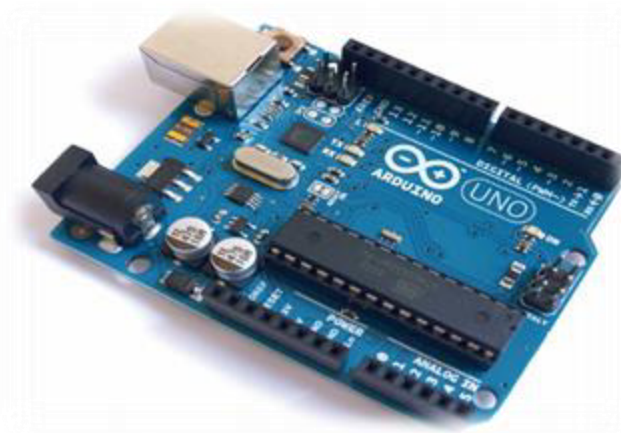


Figure 11 : modèle Arduino Uno

3.2. Différents éléments de la carte Arduino Uno

Une carte Arduino est un circuit imprimé sur lequel sont soudés et reliés différents composants électroniques. (Figure 13).

Le composant principal est un microcontrôleur, circuit intégré comprenant dans un seul boîtier :

- Microprocesseur pour effectuer les traitements,
- Mémoire vive pour stocker les informations temporaires,
- Mémoire Flash pour stocker le programme, les ordres à exécuter (conservés hors alimentation).
- Interfaces d'entrées / sorties pour acquérir et produire les signaux électriques désirés.

Autour du microcontrôleur on trouve entre autres :

1. Prise d'alimentation auxiliaire.

2. Connecteur USB pour programmer la carte à partir d'un ordinateur.
3. Connecteurs d'entrées / sorties numériques, analogiques et d'alimentations. [14]

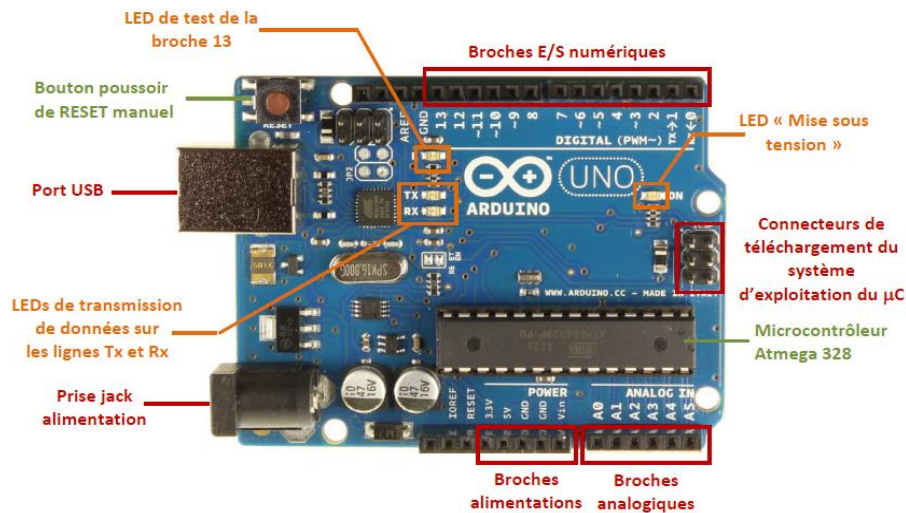


Figure 12 : Schéma d'une carte Arduino Uno.

3.3. Caractéristique de la carte Arduino UNO :

Microcontrôleur	ATmega 328
Tension de fonctionnement	5 V
Tension d'alimentation (recommandée)	7 - 12 V
Tension d'alimentation (limites)	6 - 20 V
Broches E/S numériques	14 (dont 6 disposent d'une sortie PWM)
Broches Entrée analogiques	6 (utilisables en broches E/S numériques)
Intensité max disponible par broche E/S (5V)	40 mA (200 mA cumulé pour l'ensemble des broches)
Mémoire programme Flash	32 Ko dont 0.5 Ko sont utilisés par bootloader
Mémoire SRAM (mémoire volatile)	2 KB
Mémoire EEPROM (mémoire non volatile)	1 KB
Fréquence d'horloge	16 MHz
Dimensions	68.6 mm x 53.3 mm

Tableau 2 : Caractéristiques principales de la carte Uno. [11]

3.4. Description générale d'Arduino Uno

Le système Arduino est composé de deux parties : un coté matérielle qui a une relation avec les composant électronique de la carte Arduino et le second c'est le côté

logiciel qui représente la programmation du système et pour plus de précision on va les détailler par la suite :

3.4.1. Partie matérielle :

Le système Arduino représente plusieurs composants électroniques qui sont :

3.4.1.1. Microcontrôleur ATmega328

C'est le cerveau de notre carte. Il va recevoir le programme que nous allons créer et va le stocker dans sa mémoire avant de l'exécuter. Grâce à ce programme, il va savoir faire des choses, qui peuvent être : faire clignoter une LED, afficher des caractères sur un écran, envoyer des données à un ordinateur, mettre en route ou arrêter un moteur...

Il existe deux modèles d'Arduino Uno : l'un avec un microcontrôleur de grande taille, et un autre avec un microcontrôleur dit SMD (SMD: Surface Mounted Device, soit composants montés en surface, en opposition aux composants qui traversent la carte électronique et qui sont soudés du côté opposé). D'un point de vue utilisation, il n'y a pas de différence entre les deux types de microcontrôleurs.

Les couleurs de l'Arduino peuvent varier du bleu au bleu-vert, en fonction des modèles et années de production (Figure 14). [15]

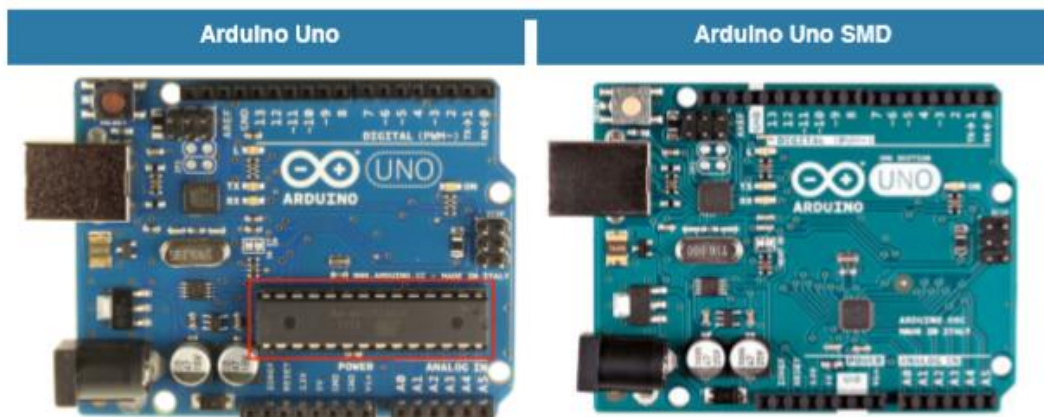


Figure 13 : Deux modèles d'Arduino Uno.

Un microcontrôleur est constitué par un ensemble d'éléments qui ont chacun une fonction bien déterminée, on les citera par la suite :

➤ **Le processeur** : c'est le composant principal du microcontrôleur. C'est lui qui va exécuter le programme que nous lui donnerons à traiter. On le nomme souvent le CPU. Il

a une largeur du chemin de données allant de 4 bits pour les modèles les plus basiques à 32 ou 64 bits pour les modèles les plus évolués.

- **La mémoire Flash :** C'est celle qui contiendra le programme à exécuter (celui que vous allez créer). Cette mémoire est effaçable et réinscriptible (c'est la même qu'une clé USB par exemple)
- **La mémoire vive (RAM) :** elle va contenir les variables de votre programme. Elle est dite "volatile" car elle s'efface si on coupe l'alimentation du microcontrôleur (comme sur un ordinateur).
- **La mémoire morte (ROM) :** nommée aussi EPROM, EEPROM. C'est le "disque dur" du microcontrôleur. Vous pourrez y enregistrer des infos qui ont besoin de survivre dans le temps, même si la carte doit être arrêtée et coupée de son alimentation. Cette mémoire ne s'efface pas lorsqu'on l'éteint le microcontrôleur ou lorsqu'on le reprogramme. (Figure 15)

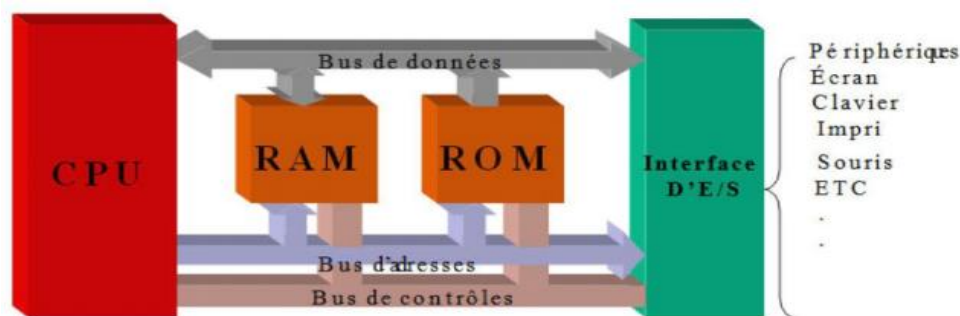


Figure 14 : Schéma simplifié du contenu type d'un microcontrôleur.

3.4.1.2. Alimentation :

La carte Arduino UNO peut être alimentée par le câble USB, par un bloc secteur externe connecté grâce à une prise « jack » de 2,1 mm ou bien par un bloc de piles dont le raccordement est réalisé par l'intermédiaire des « GND » et « VIN » du connecteur d'alimentation (POWER). L'alimentation extérieure doit présenter une tension comprise entre 7 à 12 V.

La carte génère, par l'intermédiaire de régulateurs intégrés, deux tensions stabilisées : 5 V et 3,3V. Ces deux tensions permettent l'alimentation des composants électroniques de la carte Arduino. Etant disponibles sur-connecteurs placés sur le pourtour des cartes.

- **VIN** : La tension d'entrée positive lorsque la carte Arduino est utilisée avec une source de tension externe (à distinguer du 5V de la connexion USB ou autre source 5V régulée). Vous pouvez alimenter la carte à l'aide de cette broche, ou, si l'alimentation est fournie par le jack d'alimentation, accéder à la tension d'alimentation sur cette broche.
- **5V** : La tension régulée utilisée pour faire fonctionner le microcontrôleur et les autres composants de la carte (pour info : les circuits électroniques numériques nécessitent une tension d'alimentation parfaitement stable dite « tension régulée » obtenue à l'aide d'un composant appelé un régulateur et qui est intégré à la carte Arduino). Le 5V régulé fourni par cette broche peut donc provenir soit de la tension d'alimentation VIN via le régulateur de la carte, ou bien de la connexion USB (qui fournit du 5V régulé) ou de tout autre source d'alimentation régulée.
- **3V3** : Une alimentation de 3.3V fournie par le circuit intégré FTDI (circuit intégré faisant l'adaptation du signal entre le port USB de votre ordinateur et le port série de l'Atmega) de la carte est disponible : ceci est intéressant pour certains circuits externes nécessitant cette tension au lieu du 5V). L'intensité maximale disponible sur cette broche est de 50 mA GND. Broche de masse (ou 0V). [16]

3.4.1.3. Les entrées & sorties

La carte « Arduino Uno » dispose de 14 E/S numériques et de 6 entrées analogiques.

➤ **Broches numériques en entrée OU sortie :**

Chacune des 14 broches numériques de la carte UNO (numérotées des 0 à 13) peut être utilisée soit comme une entrée numérique, soit comme une sortie numérique. Il suffit de les déclarer en début de programme. Certaines interfaces les ont déjà programmées (S4A) d'autres vous demande de le faire (S2A, ardublock)

Ces broches fonctionnent en 5V. Chaque broche peut fournir ou recevoir un maximum de 40mA d'intensité.

De plus, certaines broches ont des fonctions spécialisées :

- **Broche 13.** Dans la carte est incluse une LED 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.
- **Broches PWM.** Elles pilotent les moteurs à courant continu en vitesse. On peut aussi les utiliser pour piloter une diode en luminosité. (Figure 16)



Figure 15 : Les broches numériques (digitales)

Ces lignes partagent leur rôle avec certaines interfaces spécialisées contenues dans le microcontrôleur (Tableau 3)

Numéro E/S	Numéro ligne de ports	Fonction
0	PD 0	Rx : Entrée liaison série synchrone
1	PD 1	Tx : Sortie liaison série synchrone
2	PD 2	INT0 : Entrée interruption externe
3	PD 3	INT1 : Entrée interruption externe
		OC2B : PWM modulation à largeur d'impulsion
4	PD 4	T0 : Entrée Timer/compteur 0
		XCK : Entrée horloge
5	PD 5	T1 : Entrée Timer/compteur 1
		OC0B : Sortie module PWM modulation à largeur d'impulsion
6	PD 6	OC0A : Sortie module PWM modulation à largeur d'impulsion
		AIN0 : Entrée comparateur analogique
7	PD 7	AIN1 : Entrée comparateur analogique
8	PB 0	CLKO : Sortie de l'horloge de fonctionnement
		ICP1 : Entrée de capture Timer/compteur 1
9	PB 1	OC1A : Sortie module PWM modulation à largeur d'impulsion
10	PB 2	SS : Sélect Slave liaison SPI
		OC1B : Sortie module PWM modulation à largeur d'impulsion
11	PB 3	MOSI : Sortie liaison SPI
		OC2A : Sortie module PWM modulation à largeur d'impulsion
12	PB 4	MISO : Entrée liaison SPI
13	PB 5	SCK : Horloge liaison SPI

Tableau 3: Rôle des interfère (broches) dans le microprocesseur. [11]

➤ **Broches analogiques en entrée** La carte Uno dispose de 6 entrées analogiques (numérotées de A0 à A5), chacune pouvant fournir une mesure d'une résolution de 10 bits (c'est à dire sur 1024 niveaux soit de 0 à 1023). En termes de tension la sensibilité est donc de $5/1024 = 4,88$ mV. [17]



Figure 16 : Les broches analogiques.

➤ **Différence entre entrée numérique (digital en anglais) et analogique** : Une entrée numérique ne détecte que deux valeurs : 0 (0 Volt) et 1 (5 Volts) Une entrée analogique peut détecter 1024 valeurs comprises entre 0 et 5 Volts par saut de 4,88 mV. [17]

3.4.1.4. Communications

La carte « Arduino Uno » a de nombreuses possibilités de communications avec l'extérieur. L'Atmega328 possède une interface de communication série UART accessible, grâce aux broches numériques 0 (Rx) et 1 (Tx). D'autre part elle supporte le bus I2C accessible, grâce aux broches analogiques 4 (SDA) et 5 (SCL) et la liaison série synchrone SPI grâce aux broches numériques 10 (SS), 11 (MOSI), 12 (MISO) et 13 (SCX). [11]

3.4.1.5. Reset

A la mise sous tension un reset automatique permet au programme contenu en mémoire du microcontrôleur de démarrer automatiquement dès que la carte Arduino est alimentée. La carte « Arduino Uno » est également équipée d'un bouton poussoir de reset manuel. Un appui sur celui-ci permet de relancer l'exécution d'un programme si nécessaire, soit parce qu'il s'est « planté » soit tout simplement parce que l'on souhaite le faire repartir de son début. [11]

3.4.1.6. Horloge

L'horloge détermine la fréquence ou la rapidité à laquelle les instructions seront exécutées par le microcontrôleur. Cette vitesse peut varier d'un microcontrôleur à

l'autre et est exprimée en hertz (Hz). Si les PC et Mac peuvent atteindre des fréquences de plusieurs gigahertz, la fréquence est en revanche beaucoup plus faible pour les microcontrôleurs, avec des fréquences de quelques mégahertz seulement. Cette fréquence est déterminée par un oscillateur à quartz ou résonateur céramique. [18]

3.4.1.7. Protection de surintensité USB

Par mesure de sécurité pour l'ordinateur auquel sera relié l'Arduino, un fusible réarmable ou « Polyfuse » est présent sur la connexion d'alimentation 5 V de la prise USB. Toute consommation supérieure à 500 mA, provoque le déclenchement de ce fusible, protégeant ainsi le port USB de l'ordinateur auquel la carte est reliée. Le fusible étant de type réarmable, il retrouvera son état normal quelques secondes après que la consommation excessive aura cessé. [11]

3.4.2. Partie logicielle

L'électronique d'aujourd'hui est remplacée de plus en plus par l'électronique programmée. On parle aussi d'électronique embarquée ou d'informatique embarquée.

3.4.2.1. Téléchargement et installation du logiciel Arduino

Pour l'exécution d'un programme Arduino, il faudra, en plus de la carte Arduino Uno, d'un ordinateur et d'un logiciel spécial pour la faire fonctionner.

En premier il faut écrire sur un ordinateur le code qui va s'exécuter sur l'Arduino Uno, puis le télécharger dans ta carte, avec l'utilisation d'un logiciel pour écrire le code de programmation. Pour cela il faut le télécharger.

Ce logiciel s'appelle l'environnement intègre Arduino, ou IntegratedDevelopmentEnvironment (IDE) en anglais. On pourra le trouver sur le site <http://arduino.cc/en/Main/Software>. C'est un open source (gratuit) et est disponible pour les trois principales plateformes d'ordinateurs : Windows, Mac OS, Linux. Utiliser celui qui correspond au système d'exploitation de votre ordinateur.

- **Installation du logiciel Arduino sur un PC (Windows)**

Afin d'installer le logiciel Arduino sous Windows il faudrait suivre un certain nombre d'étapes, qui sont expliquées ci-dessous :

1. Dans un navigateur Internet, se rendre sur la page de téléchargement Arduino "arduino.cc" et commencer par télécharger le fichier nommé "arduino-1.6.5-r2-windows.zip" ou une autre version approprié avec la version de votre système d'exploitation.
2. Chercher le fichier zip télécharger et décompressé-le dans le dossier Program Files. Il devrait contenir plusieurs dossiers et fichiers, dont un dossier nommé drivers et un fichier nommé arduino.exe. Alors cliquer avec le bouton droit de la souris sur le fichier arduino.exe et placer un raccourci vers ce fichier sur le Bureau de l'ordinateur.
3. Relier le câble USB à la carte Arduino puis l'autre extrémité a l'ordinateur, Comme le montre la figure 18. On devrait voir des voyants s'allumer sur la carte Arduino Uno. Ceci indique simplement qu'elle est alimentée électriquement



Figure 17 : Le branchement de la carte Arduino avec l'ordinateur via un câble USB.

3.4.2.2. Interface du logiciel

Le logiciel Arduino a pour fonctions principales :

- Pouvoir écrire et compiler des programmes pour la carte Arduino.
- Se connecter avec la carte Arduino pour y transférer les programmes.
- De communiquer avec la carte Arduino.

L'interface du logiciel Arduino se présente de la façon suivante : (figure 19)

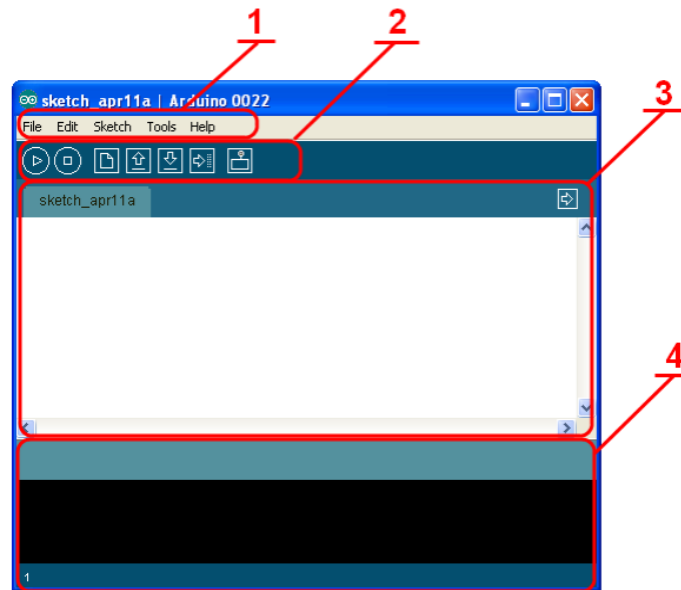


Figure 18 : Interface du logiciel Arduino

1. Options de configuration du logiciel.
2. Boutons pour la programmation des cartes.
3. Programme à créer.
4. Débogueur (affichage des erreurs de programmation). [19]

Le menu File dispose d'un certain nombre de choses qui vont être très utiles : (Figure 20).

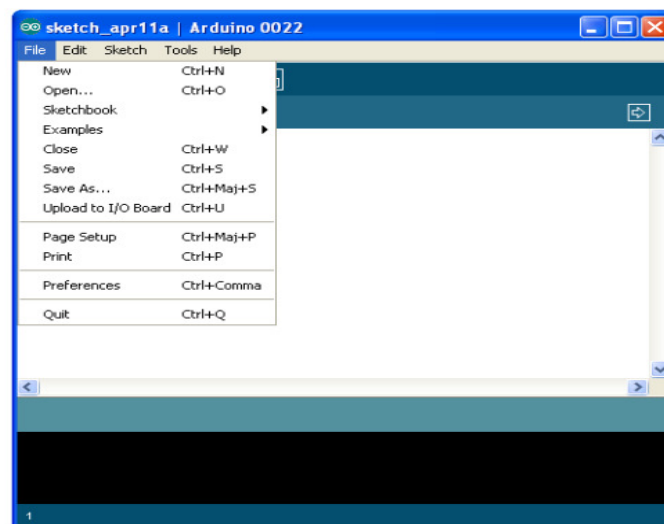


Figure 19 : Le menu du logiciel Arduino.

- New (nouveau) : va permettre de créer un nouveau programme. Quand on appuie sur ce bouton, une nouvelle fenêtre, identique à celle-ci, s'affiche à l'écran.
- Open (Ouvrir) : avec cette commande, on peut ouvrir un programme existant.

- Save / Save as (Enregistrer / enregistrer sous) : enregistre le document en cours / demande où enregistrer le document en cours.
- Example (exemples) : ceci est important, toute une liste se déroule pour afficher les noms d'exemples de programmes existant. [19]

Les boutons : (Figure 21)

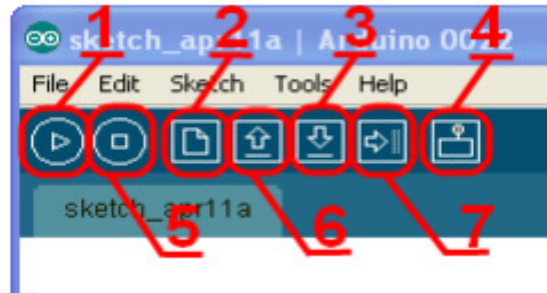


Figure 20 : Quelques boutons du logiciel.

1. Permet de vérifier le programme, il actionne un module qui cherche les erreurs dans le programme.
2. Créer un nouveau fichier.
3. Sauvegarder le programme en cours.
4. Liaison série 5 stop la vérification.
5. Charger un programme existant.
6. Compiler et envoyer le programme vers la carte. [19]

3.4.2.3. Le langage Arduino

Le logiciel de programmation des modules Arduino est une application **Java**, libre et multiplateforme, servant d'éditeur de code et de compilateur, et qui peut transférer le firmware et le programme au travers de la liaison USB. Le langage de programmation utilisé est un mélange de **C** et de **C++**, restreint et adapté aux possibilités de la carte. (Figure 22)

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

Structure d'un programme

Il y a trois phases consécutives:

1/La définition des constantes et des variables

2/La configuration des entrées et sorties
`void setup()`

3/La programmation des interactions et comportements
`void loop()`

Une fois la dernière ligne exécutée, la carte revient au début de la troisième phase et recommence sa lecture et son exécution des instructions successives. Et ainsi de suite.

Cette **boucle** se déroule des milliers de fois par seconde et anime la carte!

Figure 21 : le logiciel IDE.

→ Description d'un programme :

Un programme ou croquis (sketch) destiné à une carte Arduino est constitué de 3 parties : (figure 23)

Zone de définition des constantes ou des variables ou d'inclusion des bibliothèques

```
/* Ce programme fait clignoter une LED branchée sur la broche n°13
et fait également clignoter la diode de test de la carte */
int ledPin = 13; // LED connectée à la broche n°13
```

Fonction `setup()` qui contient les instructions d'initialisation

```
void setup()
{
  pinMode(ledPin, OUTPUT); // Definit la broche n°13 comme une sortie
}
```

Fonction `loop()` qui contient les instructions du programme

```
void loop()
{
  digitalWrite(ledPin,HIGH); // Met la sortie 13 au NL1 (diode allumée)
  delay(3000); // Attendre 3 s
  digitalWrite(ledPin,LOW); // Met la sortie 13 au NL0 (diode éteinte)
  delay(1000); // Attendre 1 s
}
```

Figure 22 : Exemple d'un programme (sous IDE).

La première partie permet de définir les constantes et les variables en déclarant leur type. Elle permet également l'inclusion des bibliothèques utilisées dans le programme au moyen de #include.

La fonction **setup ()** contient les instructions d'initialisation ou de configuration des ressources de la carte comme par exemple, la configuration en entrée ou sorties des broches d'E/S numériques, la définition de la vitesse de communication de l'interface série, etc... Cette fonction n'est exécutée qu'une seule fois juste après le lancement du programme.

La fonction **loop ()** contient les instructions du programme à proprement parlé. Cette fonction sera répétée indéfiniment tant que la carte Arduino restera sous tension.

- **Commentaire :**

Pour placer des commentaires sur une ligne unique ou en fin de ligne, il faut utiliser la syntaxe suivante :

// Cette ligne est un commentaire sur UNE SEULE ligne

Pour placer des commentaires sur plusieurs lignes :

/* Commentaire, sur PLUSIEURS lignes qui sera ignoré par le programme, mais pas par celui qui lit le code */

- **Les données, variables et constantes :**

Les différents types utilisés avec la programmation ARDUINO sont : [20]

Nom	Description
boolean	Donnée logique (true ou false) sur 8 bits
byte	Entier non signé sur 8 bits
int	Entier signé sur 16 bits
unsigned int	Entier non signé sur 16 bits
long	Entier signé sur 32 bits
unsigned long	Entier non signé sur 32 bits
float	Nombre à virgule flottant sur 32 bits
char	Caractère sur 8 bits. Seuls les caractères ayant pour code ASCII une valeur 0 à 127 sont définis

Figure 23 : Les différents types utilisés

- **Les bibliothèques de fonctions**

Une bibliothèque est un ensemble de fonctions utilitaires mises à disposition des utilisateurs de l'environnement Arduino. Les fonctions sont regroupées en fonction de leur appartenance à un même domaine conceptuel (mathématique, graphique, tris, etc). L'IDE ARDUINO comporte par défaut plusieurs bibliothèques externes. Pour les importer dans votre programme, vous devez utiliser le menu suivant :

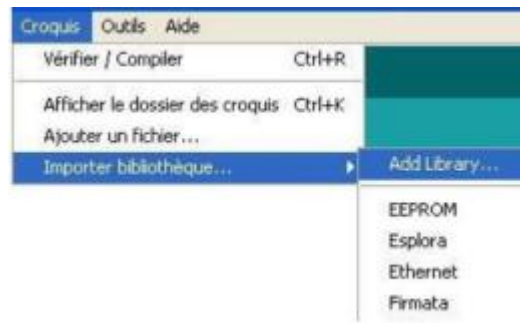


Figure 24 : menu pour importation des bibliothèques.

L'instruction suivante sera alors ajoutée au début de votre programme :

```
#include<la_bibliothèque.h>. [20]
```

Exemple :

```
#include <SPI.h>           // Importation de la bibliothèque SPI
```

3.4.2.4. Les étapes de développement d'un programme

- **Editer le programme à partir d'Arduino.**
- **Compilation du programme :**

Dans cette partie, on vérifie si le code contient des erreurs de syntaxes. En cas d'anomalie de compilation, le compilateur renseigne sur le type d'erreur et la ligne où elle se trouve. [21]

- **Sélection de la cible et du port série :**

Avant de transférer le programme vers la carte Arduino, il faut, si ce n'est déjà fait, sélectionner la bonne carte Arduino (la bonne cible) depuis le menu **Tools>Board** (**Outils>Carte**). La carte doit évidemment être connectée à l'ordinateur via un câble USB.

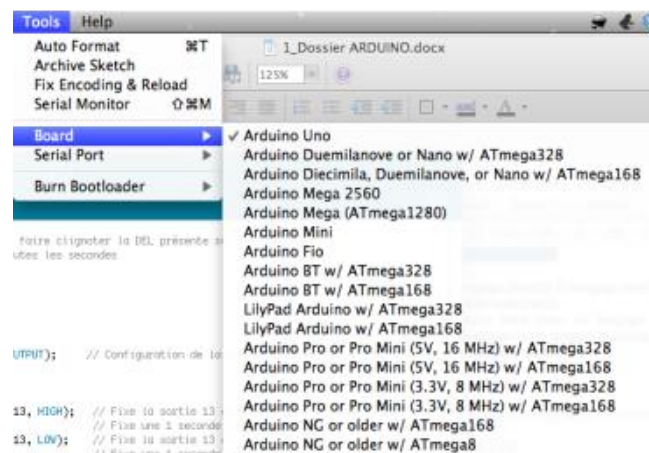


Figure 25 : menu pour la sélection de la bonne carte.

Vous devez également sélectionner le bon port série depuis le menu **Tools > Serial Port (Outils > Port Série)**. [21]

- **Transfert du programme vers la carte ARDUINO :**

Une fois que vous avez sélectionné le bon port série et la bonne carte Arduino, cliquez sur le bouton **UPLOAD** (Transfert vers la carte) dans la barre d'outils, ou bien sélectionner le menu **File>Upload to I/O board (Fichier > Transférer vers la carte)**. [21]

4. Conclusion

Dans ce chapitre, nous avons expliqué le système Arduino d'une manière général, ainsi que les domaines d'applications, puis nous avons cité des différents types de cette carte et une présentation dans un tableau comparatif.

Ensuite, nous avons projeté la lumière sur l'Arduino Uno, en expliquant les principes caractéristiques de la carte Arduino Uno, avec les deux parties essentielles qui sont la partie matérielle et la partie logiciels de la carte.

Enfin, ces outils matérielles et logiciels sont nécessaires pour concevoir notre système de verrouillage de porte et système d'alerte dans le chapitre suivant.

Sommaire

1. Introduction :	44
2. Définition du système :	44
3. Le schéma théorique du montage :	44
4. Principe de fonctionnement :	45
5. Description des composants utilisés :	46
5.1. Arduino UNO :	46
5.2. Les serrures connectées RFID :	46
5.3. Module GSM SIM900 :	49
5.4. L'écran LCD I2C :	52
5.5. Servomoteur :	54
5.6. Clavier 4X4 :	57
5.7. Buzzer :	58
5.8. Les LEDs :	59
6. Résultats pratiques	61
7. Conclusion	62

1. Introduction

Après avoir étudié la modélisation de chaque partie de notre projet nous arrivons à le réaliser pratique. Dans ce dernier chapitre, nous allons concrétiser notre projet par la réalisation d'une serrure électronique (servomoteur) codée à base de la carte Arduino Uno.

Ce chapitre va présenter notre système et le schéma de réalisation, son principe de fonctionnement, ainsi les composants utilisés et leur connectivité avec la carte Arduino. La fin du chapitre sera consacrée aux résultats pratiques de notre réalisation : système de verrouillage de porte et système d'alarme codée.

2. Définition du système

Le système d'accès intelligents à domotique est un dispositif électronique permettant de verrouiller ou déverrouiller tous les ouvrants de manière contrôlable et programmable au part avant dans le logiciel IDE, il permet d'apporter du confort et d'augmenter le niveau de sécurité au domicile, en s'appuyant sur le système Arduino et le RFID ainsi d'autre composant, il consiste à faire vérifier un badge et enter un mot de passe pour la confirmation d'identité ou le lancer une alarme en cas d'intrusion.

3. Le schéma théorique du montage

Cette application permet à l'utilisateur de contrôler l'accès dans son habitat par l'identification des tags et la saisie d'un code de sécurité à l'aide d'un clavier matricielle (figure 27).

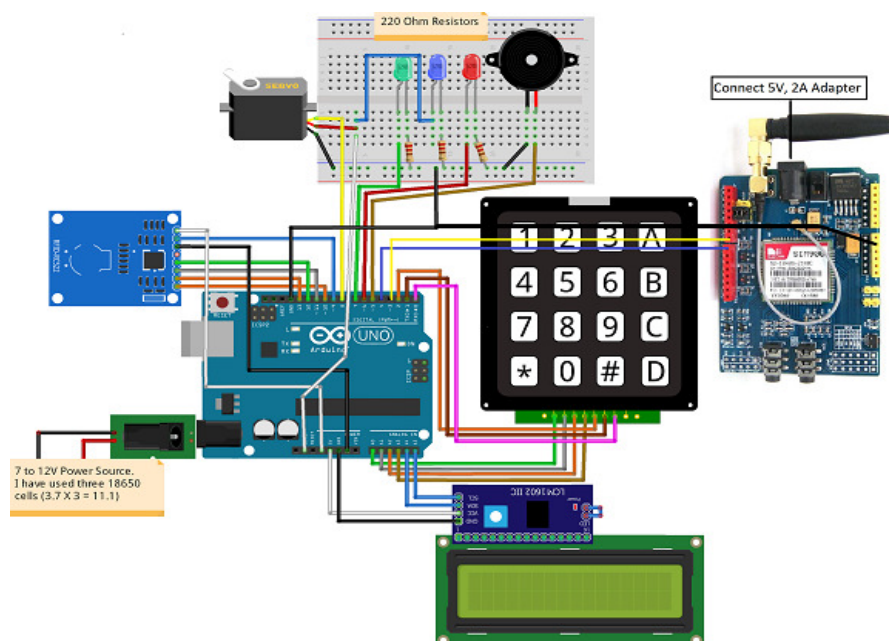


Figure 26 : Montage en plaque d'essai.

4. Principe de fonctionnement

Le principe de fonctionnement de notre système est très simple, il est constitué d'un système d'alarme pour lutter contre les intrusions et doté d'une sirène (buzzer), et le système de verrouillage de porte contrôlé par un servomoteur. Un affichage sur smartphone sera employé dans ce cas pour l'envoi des SMS, et enfin deux solutions d'accès pour sécuriser l'accès de notre local, par un code d'identification via le clavier 4*4 et une identification par carte RFID. La carte Arduino sera le cerveau et le centre de tous les équipements utilisés. Pour éviter tous ce qui est câblage, un module GSM SIM900 est associé à la carte Arduino pour communiquer avec le smartphone ce qui nous offre un contrôle à distance, et tous ces équipements nous permettent de créer un système de sécurité et qui fonctionnent comme suit :

- En scannant la bonne étiquette et après saisie du véritable code au clavier notre système va l'interroger : Si le mot de passe est correct l'accès sera autorisé : La LED vert s'allume, la serrure s'ouvre et une notification va être envoyée directement au smartphone, qui porte le message « door open » (qui veut dire « porte ouverte »).
- En scannant la mauvaise étiquette ou en entrant un mot de passe erroné l'accès sera refusé : La LED rouge s'allume, le buzzer sonne, la serrure reste fermée et une notification d'alerte va s'envoyer directement au smartphone porte le message « code erroné ».
- En scannant la bonne étiquette mais avec la saisie d'un mauvais mot de passe, l'accès sera refusé : La LED rouge s'allume, le buzzer sonne, la serrure reste fermée et une notification va être envoyée directement au smartphone porte le message « code erroné ».

On peut donner des commandes aussi à distance pour arrêter le système en envoyant un message «close» à la carte Arduino qui veut dire fermeture et il ne reviendra au mode normal que lorsqu'on enverra le message d'ouverture «open» à Arduino. Pendant le temps d'arrêt, il n'analysera aucune balise et ne recherchera que le message de réouverture.

L'ouverture de la porte est également possible en envoyant un message à Arduino et lui donnant un ordre d'ouverture.

5. Description des composants utilisés

La conception et la réalisation du système de verrouillage de porte et le système d'alarme ont besoin de plusieurs outils qui sont généralement réparties en deux parties , la partie Hardware qui présente les équipements utilisé dans le projet et la partie Software qui représente le logiciel IDE utilisé, et définit dans le chapitre précédent.

La partie Hardware du projet comporte :

5.1. Arduino UNO :

Le système Arduino qui est le cerveau du projet il permet d'interagir tous les composants entre eux et leur donner des actions et fonction fixe et précise.

Dans le chapitre qui précède le système est définit et expliquer avec tous ces aspect.

5.2. Les serrures connectées RFID

Pour la fonction d'accès et sécurisation de l'habitat le module choisis est le RFID, c'est pour éviter les intrusions et l'accès par des personnes indélicates ainsi l'utilisateur peut contrôler l'accès de son domicile.

5.2.1. Définition :

Le système RFID (Radio Frequency Identification) est une technologie très attractive pour les entreprises qui offrent la possibilité d'une gestion automatique du nombre conséquent d'informations qu'elle doit traiter. Les équipements adaptés à ce système permettent de synchroniser les flux physiques avec les flux d'informations.

Le terme RFID englobe toutes les technologies qui utilisent les ondes radio pour identifier automatiquement des objets ou des personnes.

Le système RFID autrement dit l'identification par radiofréquence est une technologie qui permet de mémoriser et de récupérer des informations à distance grâce à une étiquette qui émet des ondes radio (figure 28). [05]

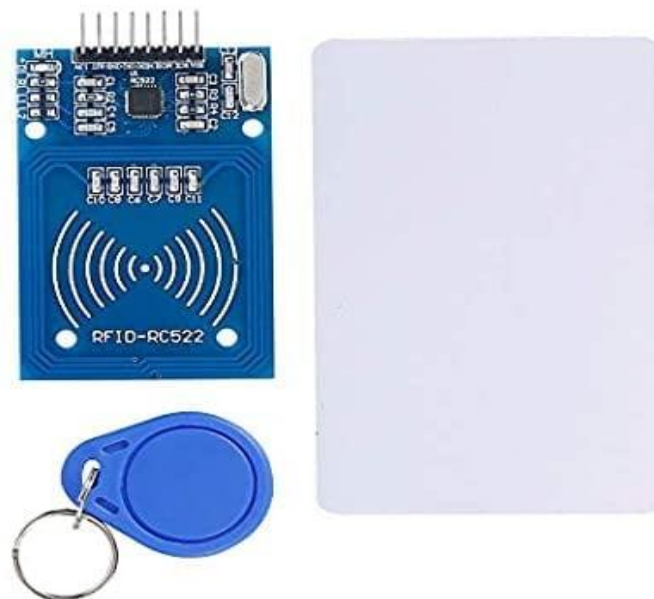


Figure 27 : modèle RFID RC522.

5.2.2. Les composants d'un système RFID

- **Système RFID** : Un système RFID (Radio Fréquence Identification) se compose de transpondeurs (aussi nommés étiquettes, marqueurs, tags, identifiants...) et d'un ou plusieurs interrogateurs (aussi nommés lecteurs, bases stations...). (figure 29)
- **Interrogateurs RFID** : Ce sont des dispositifs actifs, émetteurs de radiofréquences qui vont activer les tags qui passent devant eux en leur fournissant l'énergie dont ils ont besoin pour fonctionner.
- **Tag RFID** : C'est un dispositif récepteur, Il est muni d'une puce contenant les informations et d'une antenne pour permettre les échanges d'informations.
- **Interface** : L'interface est le support de transmission de l'énergie et des données. Dans le cadre des systèmes RFID, il s'agit de l'air. [05]

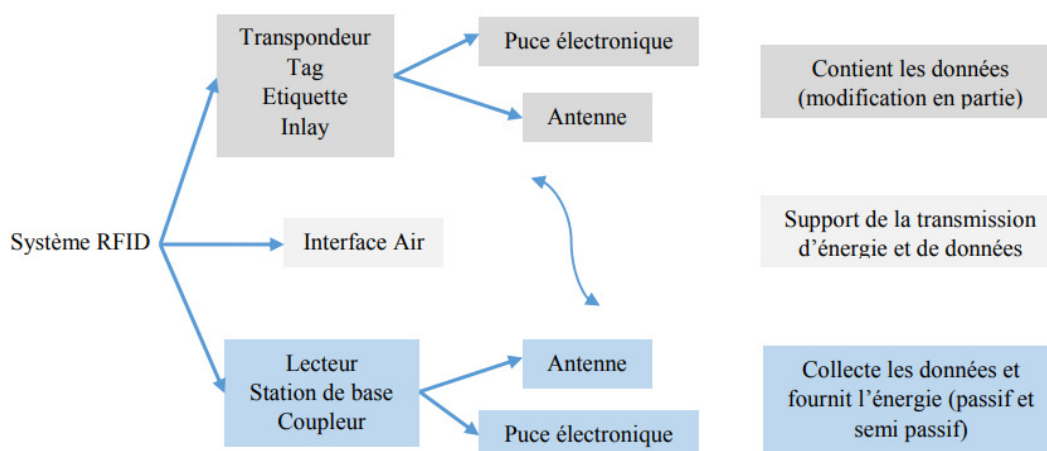


Figure 28 : Les composants d'un système RFID.

5.2.3. Les gammes de fréquences RFID

La RFID a 4 types des gammes dans le spectre radio qui sont :

- LF : 125 kHz - 134,2 kHz : basses fréquences.
- HF : 13,56 MHz : hautes fréquences.
- UHF : 860 MHz - 960 MHz : ultra hautes fréquences.
- SHF : 2,45 GHz : super hautes fréquences. [05]

5.2.4. Module RFID RC522

Ce module est conçu pour communiquer dans la bande 13.56 MHz permettant ainsi d'interagir avec des tags RFID à une distance comprise entre 5cm et 1.5m, à faible coût, de petite taille, sans contact carte à puce pour lire et écrire, instruments intelligents et des dispositifs portatifs développés mieux. Or la fréquence de fonctionnement de ce module est 13.56 MHz donc on peut le classer comme un système RFID hautes fréquences dont les tags et le lecteur sont couplés magnétiquement. Le module RFID Mifare RC522 possède un seul lecteur sous forme d'une petite puce électronique et deux tags, le premier sous forme d'une carte et le deuxième sous forme d'une clé. [5]

5.2.5. Connectique du module RFID-RC522

Le module RFID-RC522 13.56 Mhz compatible Arduino, il dispose 8 broches dont 7 uniquement sont utilisées couramment. Dans l'ordre :

- SDA : Slave Select, actif à l'état bas, (généré par le maître) --> pin 10.
- SCK : Horloge (généré par le maître) --> pin 13.
- MOSI : Master Output, Slave Input (généré par le maître) --> pin 11.
- MISO : Master Input, Slave Output (généré par l'esclave) --> pin 12.
- GND : à mettre à la masse --> pin GND.
- RESET : Pin de réinitialisation --> pin 9.
- 3.3V : pour l'alimentation --> pin 3V3.

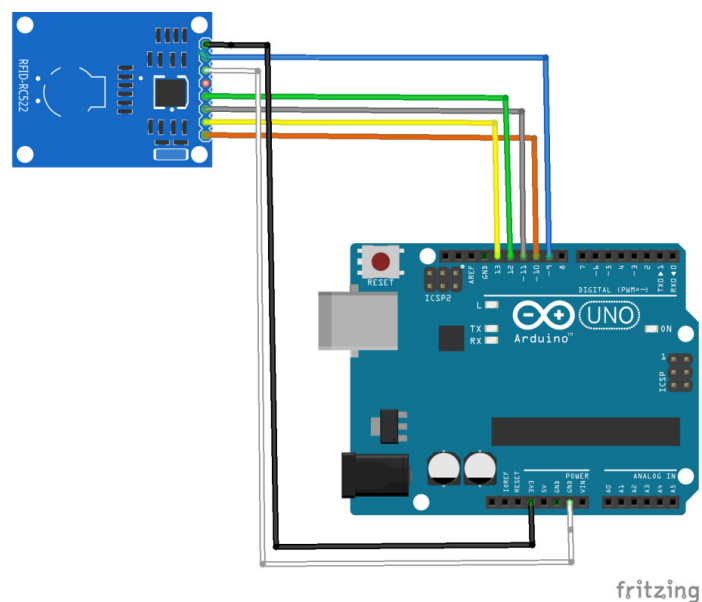


Figure 29 : câblage RFID RC522 avec Arduino UNO.

5.3. Module GSM SIM900 :

Le module GMS/GPRS de chez SeedStudio est une carte d'interface compatible Arduino. Elle permet d'envoyer et recevoir des SMS (Figure 31), des données ou des communications vocales depuis le réseau mobile. Le module est basé sur le circuit SIM900 de la société SIMCOM. Il est contrôlé via les commandes AT depuis une carte Arduino.

Le module est livré avec une antenne patch déportée. Un connecteur au dos de la platine est prévu pour recevoir une carte SIM ainsi qu'une pile Lithium CR1220. La communication entre le module et une carte Arduino est réalisée par la liaison série asynchrone : UART ou une liaison série logicielle. [22]



Figure 30 : les photos de module GSM

5.3.1. Les caractéristiques principales :

Transmission	Voix, données et SMS
Alimentation	3.4 V à 4.5 V, typiquement 3.8 V
Bandes de fréquences	GSM 850 MHz, EGSM 900 MHz, DCS 1800 MHz, PCS 1900 MHz
Courant absorbé	GSM900 : 147 mA (900 mA max) GSM1800 : 127 mA (700 mA max) GSM1900 : 113 mA (650 mA max)
Puissance d'émission	Class 4 (2 W) pour bandes GSM/EGSM Class 1 (1 W) pour bandes DCS/PCS
Lecteur de carte SIM	Intégré au module, supporte les cartes SIM 3v et 1,8v
Température d'utilisation	- 40°C à + 85 °C
Antenne	Externe par connecteur MMCX
Masse	3.4 g
Taille	24 mm x 24 mm x 3mm
Normes respectées	GSM07.07 et GSM07.05
GPRS Data Services	GPRS multi-slot class (MSC) 10 (4+1, 3+2), GPRS PBCCH/PCCCH support, GPRS Class B and CC
Messages SMS (Short Message Service)	Il est ainsi possible d'envoyer de petites quantités de données via le réseau mobile (ASCII ou données hexadécimales)
Embarque une stack TCP/UDP	Permettant d'envoyer des données sur un serveur web
Contrôle à l'aide de commandes AT	Commandes standard : GSM 07.07 et 07.05 Commandes étendues : SIMCOM AT Commands

Tableau 4 : principales caractéristique du SIM900.

5.3.2. Vue sur l'ensemble du module GSM :

Les différentes parties d'un module GSM sont présentés sur la figure 32.

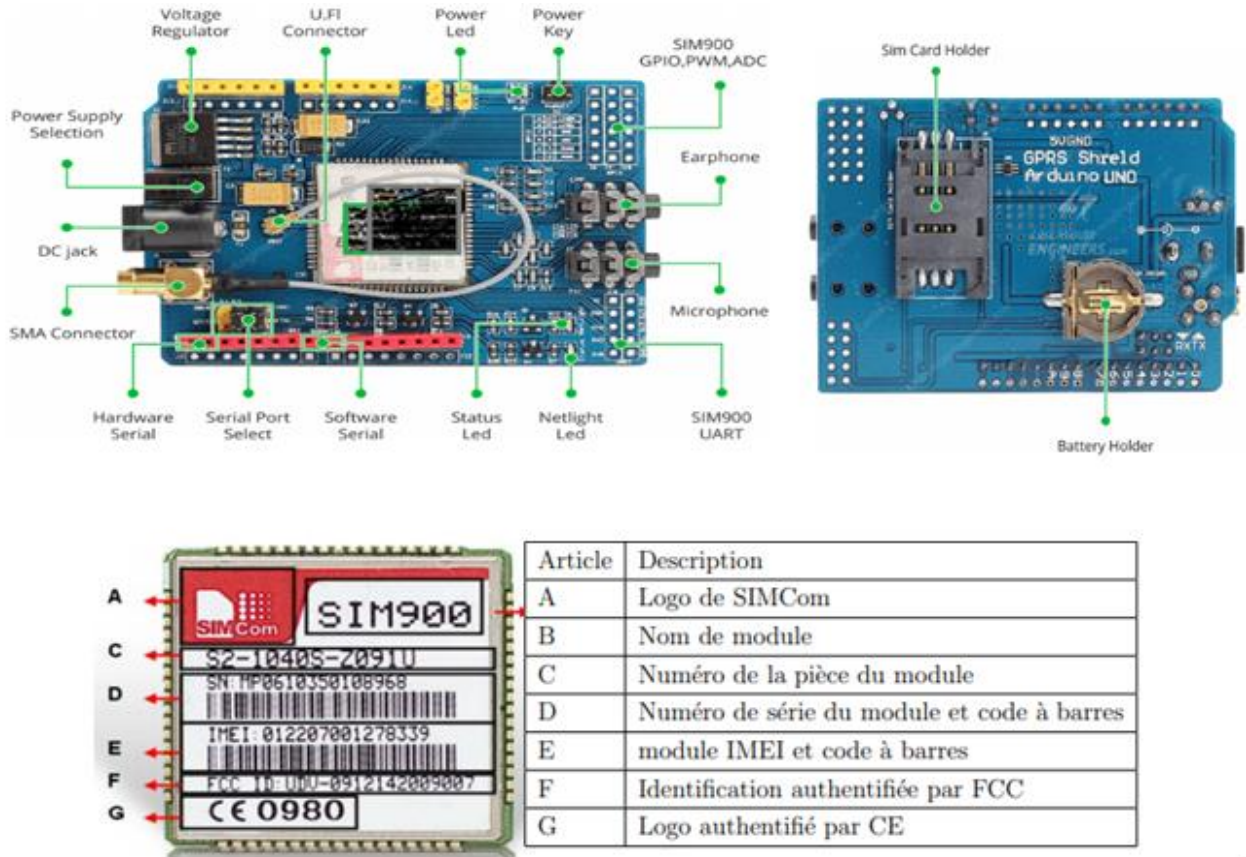


Figure 31 : Vue d'ensemble du module GSM.

5.3.3. Branchement avec l'Arduino

Pour le branchement entre l'Arduino Uno et le Sim900 on doit utiliser 3 fils, le 1er GND avec GND, le PIN7 de la carte sim900 avec PIN3 de l'Arduino et PIN8 du sim900 avec le PIN4 de l'Arduino.

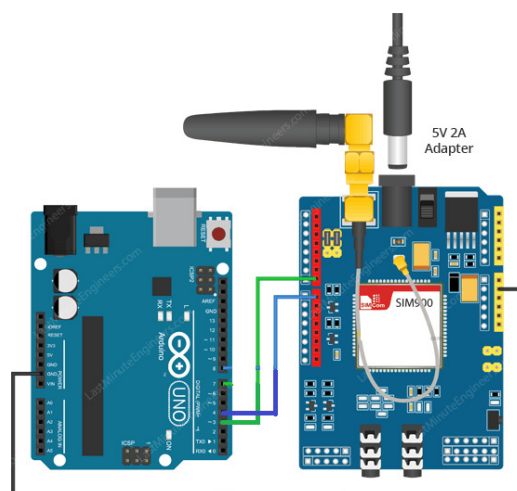


Figure 32 : Branchement du sim900 avec l'Arduino.

5.4. L'écran LCD I2C

Les afficheurs LCD (Liquid Crystal Display) sont devenues incontournables dans toutes applications qui demandent la visualisation de paramètres, il s'agit donc d'une interface Homme/Machine. Ils sont très utilisés dans les montages à microcontrôleur, et permettent une grande convivialité. Ils peuvent aussi être utilisés lors de la phase de développement d'un programme, car on peut facilement y afficher les valeurs de différentes variables.

Vu de l'extérieur, les écrans LCD alphanumériques sont essentiellement caractérisés par leur taille. Deux modèles se rencontrent très fréquemment et sont les meilleurs marché, celui ayant 2 lignes et 16 colonnes d'affichage et celui ayant 4 lignes et 20 colonnes d'affichage. [5]

5.4.1. Connecteur de l'afficheur LCD

Ces deux écrans (afficheur LCD 16x2 et 20x4) ont exactement la même connectique, c'est à dire un connecteur 16 broches. Ce connecteur véhicule plusieurs signaux dont une partie forme un bus de communication parallèle 4 ou 8 bits selon la configuration choisie ainsi que les signaux permettant de contrôler la communication entre l'Arduino et l'écran. [05]

La figure 34 ci-dessous donne la nomenclature des broches de ce connecteur.

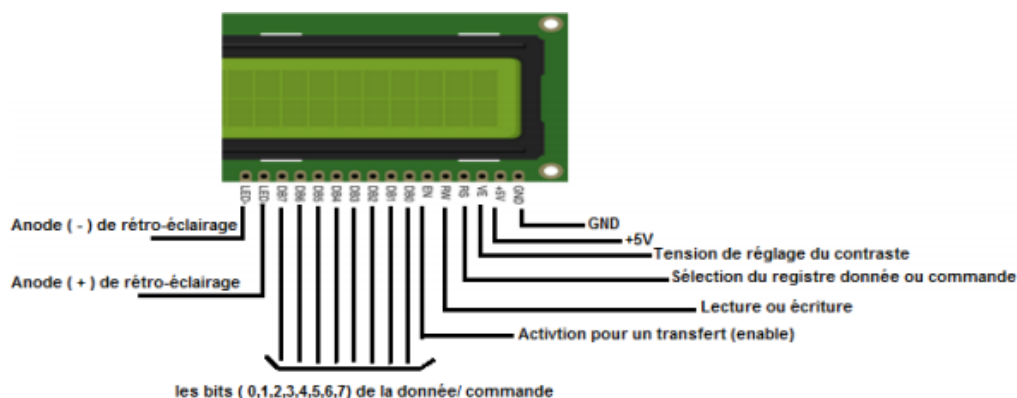


Figure 33: Nomenclature du connecteur de l'afficheur LCD.

5.4.2. Communication avec le LCD

L'afficheur LCD peut fonctionner en mode 4 bits ou en mode 8 bits. En mode 8 bits, les octets sont transférés sur les lignes DB0 à DB7. En mode 4 bits les octets sont transférés en deux fois sur les lignes DB4 à DB7. Piloter directement un LCD est un processus relativement compliqué. Toutefois, il y'a des bibliothèques qui permet de les utiliser aisément sans avoir à plonger dans la datasheet. [23]

5.4.3. Pilotage via le bus I2C

Il existe également de petits modules permettant d'interfacier un écran LCD avec un bus I2C. Cette solution peut être intéressante si on manque désespérément de broches sur la carte Arduino comme dans notre projet puisqu'au lieu de monopoliser 6 à 7 broches, l'écran n'en utilisera plus que 2. Toutefois, le module que l'on rencontre le plus souvent est construit autour du PCF8574P de NXP, un circuit permettant d'augmenter le nombre d'entrées sorties numériques via l'I2C. [23]



Figure 34 : Afficheur LCD 16x2 avec I2C.

5.4.4. Caractéristiques :

Deux afficheurs LCD 16x2 I2C seront utilisés dans notre projet, une pour afficher l'action de l'ouverture ou fermeture de la porte.

L'afficheur LCD 16x2 I2C dispose des caractéristiques techniques suivantes :

- Affichage : 16 colonnes, 2 lignes.
- Tension : 5V uniquement nécessaires.
- Courant : 135 mA typique. avec rétro-éclairage allumé.
- Modes : I2C / 9 600 baud de communication série. [05]

5.4.5. Connectivité avec Arduino :

- SCL brancher avec A5 de l'Arduino.
- SDA brancher avec A4 de l'Arduino.
- GND avec le GND (la masse).
- VCC pour l'alimentation avec 5V de la carte Arduino.

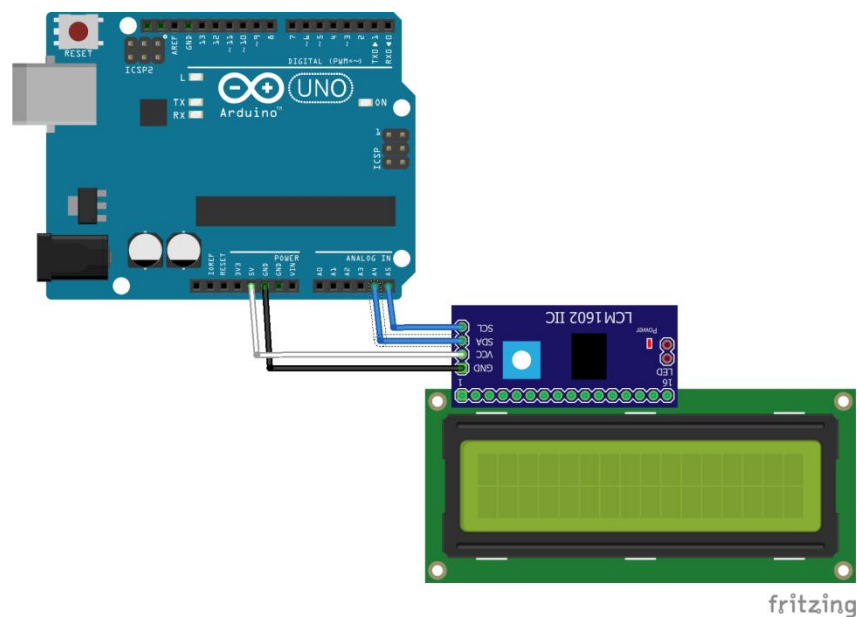


Figure 35 : Branchement écran LCD I2C avec Arduino.

5.5. Servomoteur

Le servomoteur est un moteur ayant la capacité de maintenir une position à un effort statique. Si c'est en continu, la position du servomoteur est vérifiée, et en fonction de la mesure, elle est corrigée. Ainsi, le système du servomoteur est dit immobilisé ou motorisé du fait qu'il atteigne des positions prédéterminées. [24]

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

5.5.1. Composition d'un servomoteur

Il est composé de plusieurs éléments visibles :

- Les fils, qui sont au nombre de trois (nous y reviendrons)
- L'axe de rotation sur lequel est monté un accessoire en plastique ou en métal
- Le boîtier qui le protège

Mais aussi de plusieurs éléments que l'on ne voit pas :

- un moteur à courant continu
- des engrenages pour former un réducteur (en plastique ou en métal)
- un capteur de position de l'angle d'orientation de l'axe (un potentiomètre bien souvent)

- une carte électronique pour le contrôle de la position de l'axe et le pilotage du moteur à courant continu. [24]

Voilà une image 3D (figure 37) de vue de l'extérieur et de l'intérieur d'un servomoteur :

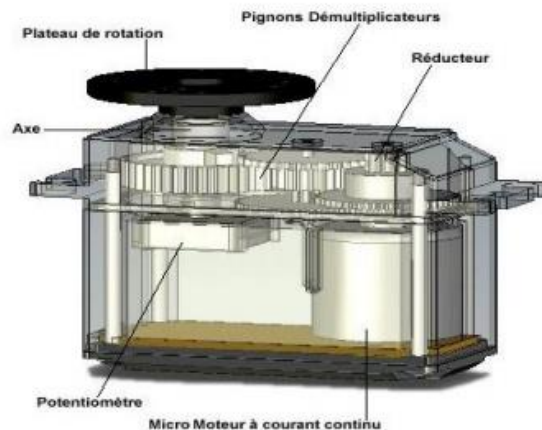


Figure 36 : schéma représentatif des composants de servomoteur à l'intérieur et extérieur. [24]

Le servomoteur a besoin de trois fils de connexion pour fonctionner. Deux fils servent à son alimentation, le dernier étant celui qui reçoit le signal de commande :

- **Rouge** : pour l'alimentation positive (4.5V à 6V en général).
- **Noir ou marron** : pour la masse (0V).
- **Orange, jaune, blanc, ...** : entrée du signal de command.



Figure 37 : les fils de servomoteur (alimentation, masse et l'entrée du signal)

5.5.2. Type des servomoteurs

Pour faire tourner la porte, une rotation de 90° ou 180° sera amplement suffisante pour l'ouverture, par contre, pour faire tourner le volet de fenêtre, une rotation de 360° est nécessaire. Nous allons utiliser les servomoteurs à rotation angulaire pour l'ouverture/fermeture des portes, ce qui permettra un gain au niveau du coût. Concernant notre projet le Micro servomoteur SG90 va bien répondre à nos besoins pour notre prototype d'un système domotique. [05]



Servomoteur à rotation continue S4303R	Servomoteur à rotation angulaire (Micro-Servo) SG90
	
<p><u>Caractéristiques :</u></p> <ul style="list-style-type: none"> • Alimentation : 4.8 à 6 V • Angle de rotation : 360° • Couple : 3.3 à 4.8 Kg.cm • Vitesse : 60 à 70 tr/min • Dimensions : 41x20x40 mm • Poids : 44g 	<p><u>Caractéristiques :</u></p> <ul style="list-style-type: none"> • Alimentation : 4.8 à 6 V • Angle de rotation : 180° • Couple : 1.3 Kg.cm • Vitesse : 0.12 sec/60° • Dimensions : 23.2 x 12.5 x 22 mm • Poids : 9g

Figure 38 : Caractéristiques des servomoteurs.

5.5.3. Fonctionnement du système

Un servomoteur se pilote par l'intermédiaire d'un câble à trois fils. Ce câble permet à la fois de l'alimenter et de lui transmettre des consignes de position par le fil de signal. Contrairement à un moteur CC simple, qui peut être piloté par des variations de tension, le servomoteur réagit en fonction d'une impulsion de durée variable. C'est la durée de ce signal qui détermine la rotation de l'axe donc la position de l'objet fixé dessus. L'impulsion détermine la position en absolu, pas en relatif : une durée précise correspond à une position précise de l'axe, toujours la même. Le signal doit être répété régulièrement (toutes les 20 ms). Ces caractéristiques expliquent qu'il faille une électronique complexe pour piloter un servomoteur, et que cette électronique lui soit dédiée. [24]

5.5.4. La connectivité avec l'Arduino

Nous l'avons vu plus haut, la connectique d'un servomoteur se résume à trois fils, deux pour l'alimentation positive et la masse qui sont brancher dans la plaque d'essai et le dernier pour le signal de commande dans le Pin 8 de l'Arduino Uno, voici ci-dessous le schéma du câblage :

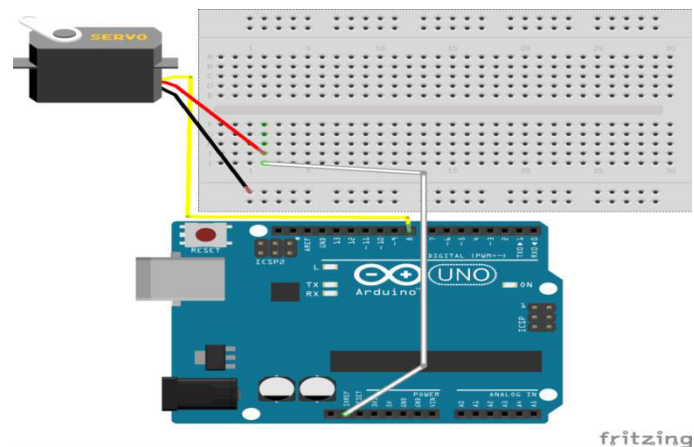


Figure 39 : Le câblage du servomoteur SG90 avec l'Arduino UNO.

5.6. Clavier 4X4

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

Ce clavier comprend 16 touches disposées en 4 lignes et 4 colonnes. L'appui sur une touche fait communiquer une ligne avec une colonne.

Le code étant numérique, donc il nous suffit de choisir un clavier seize touches dont il y a deux boutons spéciaux, un capable de changer facilement le mot de passe et l'autre efface l'écran en cas d'erreur de frappe.

En introduisant le code d'accès qui est par défaut 1234, suivant « * », la LED jaune s'allume signifiant l'activation du relais.

Si on introduit un code erroné plus de trois fois, le PIC génère un signal pour l'activation de l'alarme, et un autre pour les deux LED « clignotantes ».

Par sécurité, le système est totalement bloqué. Il est débloqué en introduisant le bon code.

Si nous voudrions changer le code d'accès, nous aurons obligé à suivre les étapes suivantes :

- Appuyer sur la touche « # » : l'utilisateur entre en mode de programmation.
- Entrer l'ancien code
- Appuyer sur « # ».
- Entrer le nouveau code d'accès désiré.
- Appuyer sur « * ».



Figure 40 : modèle d'un clavier matricielle 4x4.

5.6.1. Branchement avec la carte Arduino

Le branchement est assez simple, connectant les broches A0, A1, A2, A3 de la carte Arduino vers les lignes et les broches PIN0 , PIN1, PIN2 vers les colonnes, comme le montre la figure suivante :

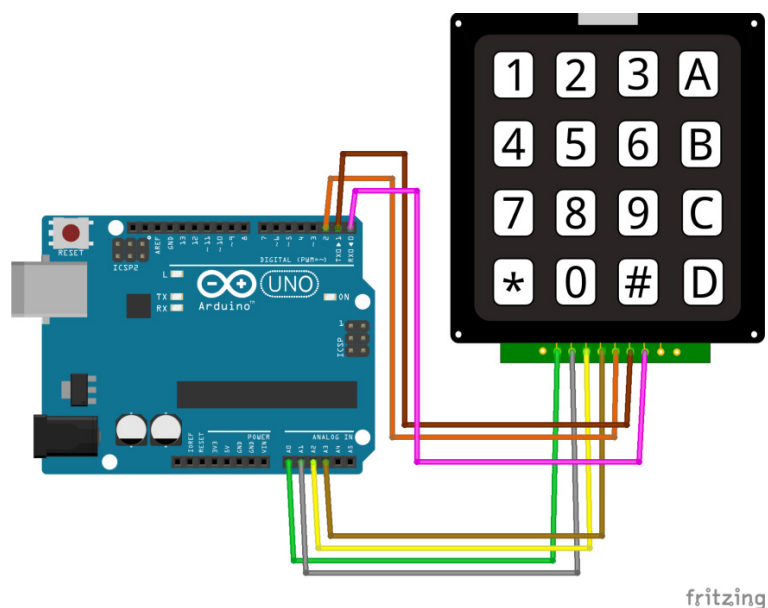


Figure 41 : Câblage clavier 4x4 avec Arduino UNO.

5.7. Buzzer

Ce composant a pour fonction d'associer un bip sonore à chaque appui sur chacune des touches du clavier matriciel, ainsi pour générer une alarme (sous forme d'un long bip ou sous forme de trois ou quatre bips successifs) dans le cas où le code inséré par l'utilisateur pour entrer au bureau est incorrect.

Un autre buzzer sera utilisé pour associer un bip sonneur à chaque identification des tags d'utilisateurs par RFID et pour générer une alarme sous forme des bips successive lors d'accès non autorisé (tags non autorisé pour accéder à la maison). [05]

5.7.1. Caractéristique du Buzzer

- Tension : 5V DC
- Sortie Min sonore à 10cm : 85dB
- Taille totale (Pin non inclus) : 12 x 9mm / 0,47 "x 0,35" (D * H)
- Matériaux : plastique, métal
- Poids net : 8g. [05]



5.7.2. Branchement avec l'Arduino :

Le Buzzer est placé dans la plaque d'essai en reliant son pin(+) avec la sortie numérique PIN5 de la carte Arduino et le le pin(-) avec la masse de la plaque d'essai.

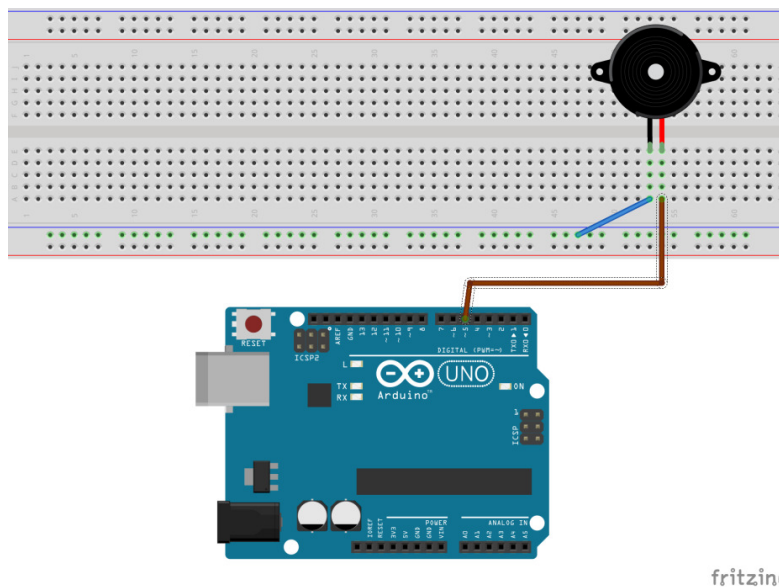


Figure 42 : Le câblage du buzzer dans la plaque d'essai et Arduino Uno.

5.8. Les LEDs

Une diode électroluminescente (**DEL** en français, ou **LED** en anglais : Light-Emitting Diode), est un dispositif optoélectronique capable d'émettre de la lumière lorsqu'il est parcouru par un courant électrique. Une diode électroluminescente ne laisse passer le courant électrique que dans un seul sens (le sens passant, comme une diode classique, l'inverse étant le sens bloquant) et produit un rayonnement monochromatique ou poly-chromatique non cohérent à partir de conversion d'énergie électrique lorsqu'un courant la travers. [05]

La diode électroluminescente (LED) émet de la lumière. Elle est polarisée : la patte '+' est la plus longue, l'autre patte est la patte '-'. Les broches numériques de l'Arduino, lorsqu'elles sont haut (HIGH), fournissent une tension de 5v.

L'utilisation des diodes électroluminescentes dans notre projet a pour but d'ajouter des signalisations sur l'état d'un actionneur. A titre d'exemple, si l'accès à l'habitat est autorisé à l'utilisateur (identification de son tag RFID), une LED verte sera allumée sinon une LED rouge sera allumée en cas d'erreur du mot de passe ou présenter la mauvaise étiquette.



Figure 43 : LEDs de différentes couleurs et le symbole de la diode électroluminescente

- **Résistances**

Une résistance est utilisée pour limiter le courant électrique qui traversera la LED. En effet la tension aux bornes d'une LED est généralement d'environ 1,5 V alors que la tension en sortie de l'Arduino est de 5 V. Bien que brancher directement la LED sur la carte soit possible il est recommandé d'ajouter une résistance pour éviter d'abîmer la LED. Le schéma ci-contre montre une résistance montée en série avec la LED : la résistance à la suite de la LED.

5.8.1. Branchement avec l'Arduino :

La patte négative des trois LEDs est liée avec des résistances placées au GND de la plaque d'essai.

- **LED verte** : La patte positive vers PIN 7 de l'Arduino Uno.
- **LED bleu** : La patte positive est branchée avec le fils rouge du servomoteur pour prévoir son alimentation et qu'il est en marche.
- **LED rouge** : La patte positive vers PIN 6 de l'Arduino Uno.

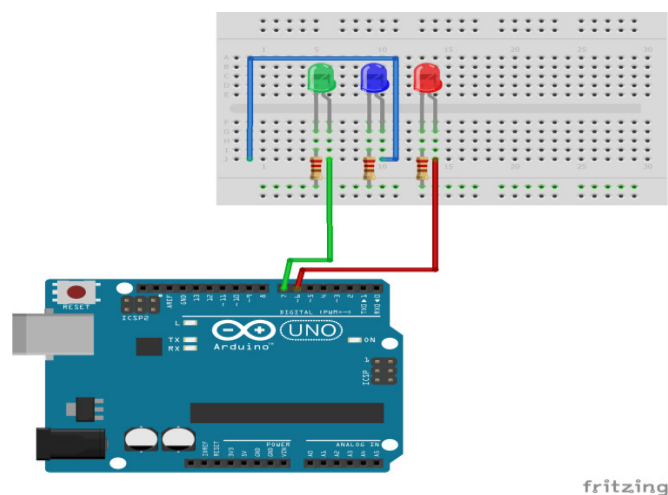


Figure 44 : branchement des LEDs avec l'Arduino.

6. Résultats pratiques

La sécurité dans ce projet a une importance capitale, nous avons utilisé un RFID RC522, afficheur LCD, clavier matricielle, un servomoteur, et le Buzzer ce dispositif fait partie du système d'alarme qui consiste à l'activation des effets sonore en cas d'intrusions tout ceci est contrôlé avec un microcontrôleur de l'Arduino qui est coordonné par notre programme écrit dans le logiciel IDE, la photos ci-dessous illustrent le branchement réel (figure 46) de tous les composants électronique utilisé pour le fonctionnement de ce système.

Ce branchement permet de sécuriser l'accès à l'habitat est assuré par la reconnaissance des clés ou tags RFID avec le module RFID plus un code d'entrée saisie au clavier. Une fois l'accès est autorisé, la LED verte s'allumera et le servomoteur tournera dans le sens où il ouvrira la porte de la maison suivis par la réception d'un SMS de confirmation associé au SIM900. Dans le cas où l'accès est non autorisé, la LED rouge et l'alarme vont être déclenchée (Buzzer) et également la réception d'un SMS d'alerte.

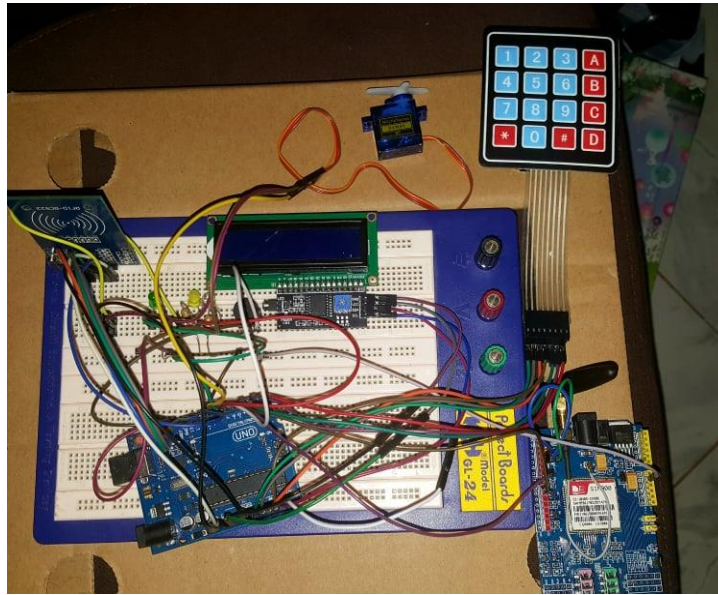
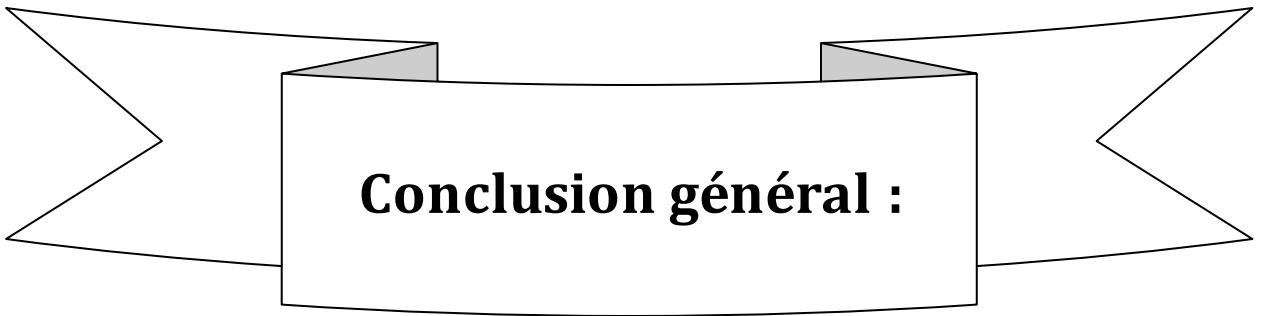


Figure 45 : la réalisation de la fonction d'accès sécurisé à l'habitat.

7. Conclusion

Dans ce dernier chapitre, nous avons réalisé un système d'automatisation d'ouverture et fermeture des portes en se servant d'un servomoteur comme serrure de porte et aussi pour augmenter la sécurité et avertir en cas d'intrusion en installant un système d'alarme liée à la porte et suivis par l'envoi des messages a un smartphone à chaque mouvement, et une commande à distance. Ce système est très remarquable et prometteur pour une haute sécurité pour les habitations et peut être plus développé.



Conclusion général :

Conclusion générale

Avec le grand progrès des technologies de la communication, de l'électronique et de l'informatique qui sont réunies pour former un seul domaine : la domotique, l'utilisateur est devenu capable de communiquer avec les différents équipements domestiques.

Le travail est constitué d'un aperçu général sur la domotique (définition, ses domaines d'application), puis nous avons détaillés dans le deuxième chapitre le système qui permet de gérer tout autre composants qui représente le système Arduino. La troisième partie du travail a été consacrée à notre réalisation pratique, qui consiste à la conception puis la réalisation d'un système de verrouillage de porte et système d'alarme et de déterminer tous les composant nécessaire pour la conception.

Au terme de notre travail, il convient de retenir que la réalisation de ce projet a permis d'avoir une notion approfondie dans la conception matérielle et logicielle des systèmes. Ainsi ; L'intérêt majeur est de permettre de gérer et contrôler d'une façon sécurisé l'accès à la domotique.

À travers ce modeste travail, nous avons eu l'occasion d'utiliser plusieurs outils informatiques et électroniques qui sont nécessaires pour la réalisation de notre projet de fin d'étude. Dans ce travail nous avons réalisé une maquette d'une porte dite intelligente à base d'Arduino UNO qui est le cerveau et le centre de tous les équipements utilisés représenter par un clavier pour saisir le mot de passe et un système d'identification dit RFID, de plus d'une communication sans fil (par un Smartphone), et le buzzer qui est considéré comme le système d'alarme et de plus d'une communication en SMS à travers le module sim900 .

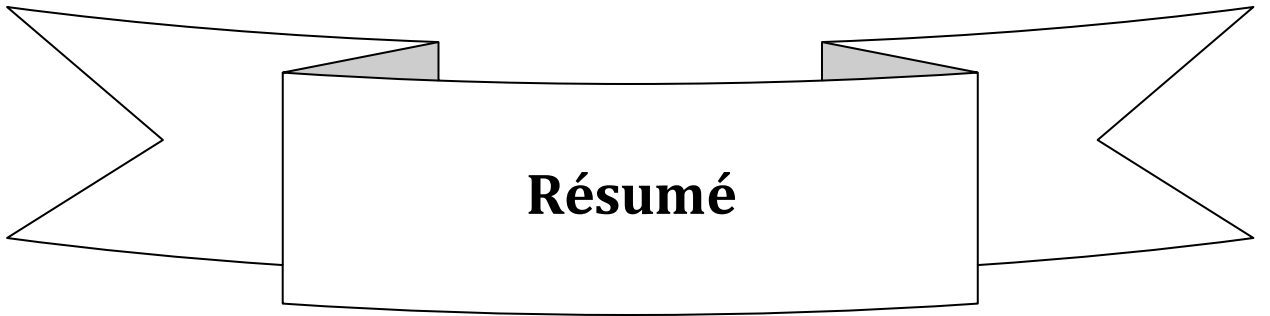
La porte reste fermé jusqu'après l'identification de la bonne carte et l'introduction d'un code à travers le clavier numérique qui équipe la serrure. Après validation du code et la carte, le système compare les données inséré au donnés stocké dans sa mémoire.

S'ils sont égaux, le voyant (LED) verte reste allumer, un message de bienvenue s'affiche et la porte s'ouvre puis se referme automatiquement. Une fois la porte fermée,

elle se verrouille à nouveau. Le mot de passe doit donc être connu à l' avance par l'utilisateur.

Si le code inséré n'est pas correcte, il affiche un message d'avertissement et finira par déclencher une alarme et une LED rouge s'allume puis l'envoi d'un SMS.

Enfin, cette progression et développement électronique et informatique dans les habitats et l'insertion dans la vie quotidienne est un grand avantage ajouté, qui permet de faciliter plusieurs tache habituelle et de fortifier le niveau sécurité, c'est un gain énorme en temps et économies d'énergie et qui devient indispensable dans les nouvelles constructions.



Résumé :

Ce projet de recherche s'intéresse à l'évolution de nos habitats, il aborde principalement la sécurité d'accès des domiciles en mettant en place un système de verrouillage de porte et système d'alerte.

Notre système doit garantir avant tout la sécurité en disposant d'un système ARDUINO RFID. La solution proposée est réalisée par plusieurs modules, dans un premier temps on utilise la carte RFID comme système de serrure de porte à clavier dans lequel vous deviez d'abord numériser la bonne étiquette, puis saisir le mot de passe correct pour ouvrir la serrure de porte.

Par la suite, ajouter le module Sim900 dans ce système pour en faire un système de serrure de porte et d'alerte basé sur un clavier RFID utilisant Arduino

Abstract:

This research project is interested in the evolution of our habitats; it mainly addresses the security of access to homes by setting up a door locking system and warning system.

Our system must above all guarantee security by having an ARDUINO RFID system. The proposed solution is realized by several modules, first of all we use the RFID card as a keypad door lock system in which you first had to scan the correct tag, and then enter the correct password to open the door lock.

Subsequently, add the Sim900 module in this system to make it an RFID keypad based door lock and alert system using Arduino

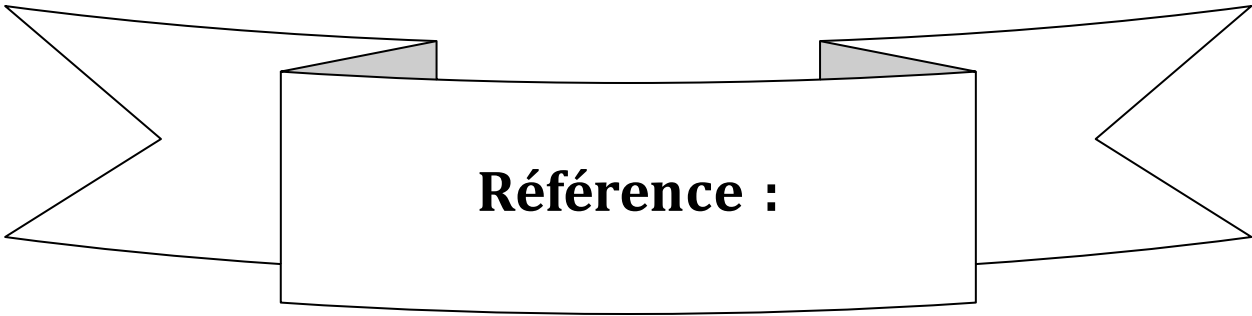
ملخص

يهتم هذا المشروع البحثي بالتطور السكني ، فهو يعالج بشكل أساسي أمن الوصول إلى المساكن من خلال إنشاء نظام قفل الباب ونظام الإنذار.

يجب أن يضمن نظامنا قبل كل شيء الأمان من خلال وجود نظام ARDUINO RFID. يتم تحقيق الحل المقترح من خلال عدة مكونات، أولاً وقبل كل شيء نستخدم بطاقة RFID كنظام قفل باب لوحة المفاتيح حيث كان علينا أولاً تحديد البطاقة الصحيحة ثم إدخال كلمة المرور الصحيحة لفتح القفل باب.

ثم إضافة وحدة Sim900 إلى هذا النظام لجعله قفل باب قائم على لوحة مفاتيح RFID ونظام تنبيه باستخدام

Arduino.

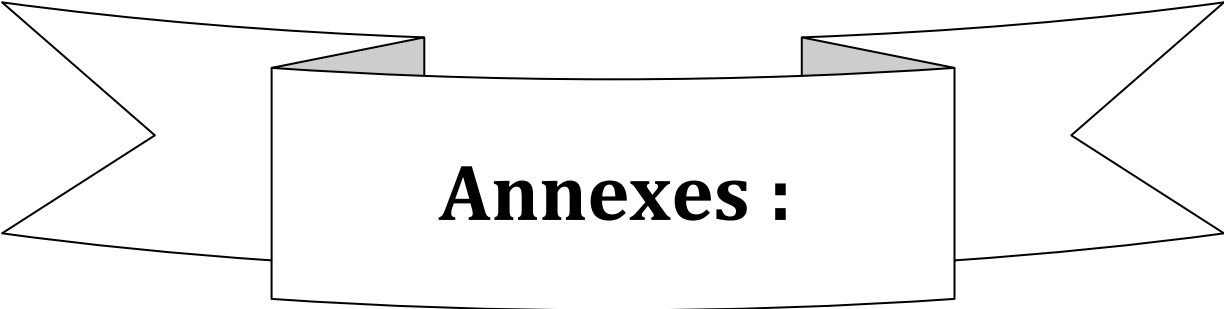


Référence :

Référence :

- [1] T.Mekhalfia et T.Ghadbane,"Etude et réalisation d'un système de commande à distance des installations électriques pour la domotique", mémoire master, université Mohammed V de Rabat, 2018.
- [2] H. Hamouchi, "Conception et réalisation d'une centrale embarquée de la domotique « Smart Home »", mémoire master, université Mohammed V de Rabat, 2015.
- [3] <https://www.maison-et-domotique.com/47938-comment-domotiser-sa-maison/>
- [4] <https://heatzy.com/blog/25/8/2017/domotique-les-protocoles-pour-connecter-vos-objets-connects>.
- [5] A.Elhammoumi et M. Slimani, " Conception et réalisation d'un prototype d'une maison domotique intelligente « My Smart Home » ", Université Hassan 1er Faculté des Sciences et Technique – Settat 2016.
- [6] K. El yahiaoui ET A.Boukoutaya, "Réalisation d'une maison intelligente à base d'Arduino", Projet fin de cycle, université Mohamed V faculté des sciences rabat, 2016.
- [7] <http://www.cea.fr/comprendre/Pages/nouvelles-technologies/essentiel-sur-domotiquemaison-connectee>.
- [8] A. Belhamel," étude et réalisation d'une serrure électronique codée (a base du pic 16f877/arduinouno) ", Université larbi ben m'hidi de oum el bouaghi 2017.
- [9] <https://www.latribune.fr/supplement/serrurier-serrure-connectee-tout-ce-qu-il-faut-connaître-775545.html>
- [10] Livre,« support de cours carte Arduino uno » .
- [11] Carte de développement ARDUINO Lycée Gustave Eiffel.
- [12] Arduino pour les Nuls poche, 2e édition (Poche pour les Nuls) (French Edition).
- [13] Arduino - Apprendre à développer pour créer des objets intelligents.
- [14] Tutoriel carte Arduino pour débutant.
- [15] Arduino à l'école Cours pour l'apprentissage des bases de l'électronique et de la programmation sur Arduino.
- [16] http://www.mon-club-elec.fr/pmwiki_reference_arduino/pmwiki.php?n=Main.MaterielUno
- [17] PRESENTATION DE LA CARTE ARDUINO UNO.
- [18] MehalaineNourelhouda, " Étude et réalisation d'un système intelligent pour la commande d'éclairage publique et surveillance de quelques paramètres atmosphériques ", UNIVERSITE LARBI BEN M'HIDIOUM EL BOUAGHI 2018.
- [19] Arduino pour bien commencer en électronique et en programmation.
- [20] Développement informatique Algorithmique, Didacticiel Arduino.
- [21] La carte ARDUINO UNO, LP CONDORCET.
- [22] Module GSM/GPRS SEEDSTUDIO, LycéeGustaveEiffel .

- [23] S.M'hid, "conception et réalisation d'une Maison intelligente" ; Mémoire licence ; Université Sidi Mohammed Ben Abdellah, 2018.
- [24] M.Mehtati et S.Abdeli, "Smart House ", Mémoire Projet de Fin d'Etude, Université ABOU BEKR BELKAID 2017.



Annexes

Programme de la carte ARDUINO UNO :

```
// Include required libraries
#include <MFRC522.h>
#include <LiquidCrystal_I2C.h>
#include <Keypad.h>
#include <SoftwareSerial.h>
#include <Servo.h>
#include <SPI.h>
// Create instances
SoftwareSerialSIM900(3, 4); // SoftwareSerial SIM900(Rx, Tx)
MFRC522 mfrc522(10, 9); // MFRC522 mfrc522(SS_PIN, RST_PIN)
LiquidCrystal_I2C lcd(0x27, 16, 2);
Servo sg90;

// Initialize Pins for led's, servo and buzzer
// Blue LED is connected to 5V
constexpr uint8_t greenLed = 7;
constexpr uint8_t redLed = 6;
constexpr uint8_t servoPin = 8;
constexpr uint8_t buzzerPin = 5;
char initial_password[4] = {'1', '2', '3', '4'}; // Variable to store initial password
String tagUID = "29 B9 ED 23"; // String to store UID of tag. Change it with your tag's
UID
char password[4]; // Variable to store users password
booleanRFIDMode = true; // boolean to change modes
booleanNormalMode = true; // boolean to change modes
charkey_pressed = 0; // Variable to store incoming keys
uint8_t i = 0; // Variable used for counter
// defining how many rows and columns our keypad have
const byte rows = 4;
const byte columns = 4;
```

```
// Keypad pin map
charhexaKeys[rows][columns] = {
  {'1', '2', '3', 'A'},
  {'4', '5', '6', 'B'},
  {'7', '8', '9', 'C'},
  {'*', '0', '#', 'D'}
};
byterow_pins[rows] = {A0, A1, A2, A3};
bytecolumn_pins[columns] = {2, 1, 0};
// Create instance for keypad
Keypad keypad_key = Keypad(makeKeymap(hexaKeys), row_pins, column_pins, rows,
columns);
void setup() {
  pinMode(buzzerPin, OUTPUT);
  pinMode(redLed, OUTPUT);
  pinMode(greenLed, OUTPUT);
  sg90.attach(servoPin); //Declare pin 8 for servo
  sg90.write(0); // Set initial position at 0 degrees
  lcd.begin(); // LCD screen
  lcd.backlight();
  SPI.begin(); // Init SPI bus
  mfrc522.PCD_Init(); // Init MFRC522
  // Arduino communicates with SIM900 GSM shield at a baud rate of 19200
  SIM900.begin(19200);
  // AT command to set SIM900 to SMS mode
  SIM900.print("AT+CMGF=1\r");
  delay(100);
  // Set module to send SMS data to serial out upon receipt
  SIM900.print("AT+CNMI=2,2,0,0,0\r");
  delay(100);
  lcd.clear(); // Clear LCD screen
}
void loop() {
```

```
if (NormalMode == false) {
    // Function to receive message
    receive_message();
}
else if (NormalMode == true) {
    // System will first look for mode
    if (RFIDMode == true) {
        // Function to receive message
        receive_message();
        lcd.setCursor(0, 0);
        lcd.print(" Door Lock");
        lcd.setCursor(0, 1);
        lcd.print(" Scan Your Tag ");
        if (! mfr522.PICC_IsNewCardPresent()) {
            return;
        }
        // Select one of the cards
        if (! mfr522.PICC_ReadCardSerial()) {
            return;
        }
        //Reading from the card
        String tag = "";
        for (byte j = 0; j < mfr522.uid.size; j++)
        {
            tag.concat(String(mfr522.uid.uidByte[j] < 0x10 ? " 0" : " "));
            tag.concat(String(mfr522.uid.uidByte[j], HEX));
        }
        tag.toUpperCase();
        //Checking the card
        if (tag.substring(1) == tagUID)
        {
            // If UID of tag is matched.
            lcd.clear();
        }
    }
}
```

```
lcd.print("Tag Matched");
digitalWrite(greenLed, HIGH);
delay(3000);
digitalWrite(greenLed, LOW);
lcd.clear();
lcd.print("Enter Password:");
lcd.setCursor(0, 1);
RFIDMode = false; // Make RFID mode false
    }
else
    {
        // If UID of tag is not matched.
        lcd.clear();
        lcd.setCursor(0, 0);
        lcd.print("Wrong Tag Shown");
        lcd.setCursor(0, 1);
        lcd.print("Access Denied");
        digitalWrite(buzzerPin, HIGH);
        digitalWrite(redLed, HIGH);
        send_message("Someone Tried with the wrong tag \nType 'close' to halt the system.");
        delay(3000);
        digitalWrite(buzzerPin, LOW);
        digitalWrite(redLed, LOW);
        lcd.clear();
        }
    }
    // If RFID mode is false, it will look for keys from keypad
if (RFIDMode == false) {
key_pressed = keypad_key.getKey(); // Storing keys
if (key_pressed)
    {
password[i++] = key_pressed; // Storing in password variable
lcd.print("*");
```

```
    }
    if (i == 4) // If 4 keys are completed
    {
        delay(200);
        if (!(strcmp(password, initial_password, 4))) // If password is matched
        {
            lcd.clear();
            lcd.print("Pass Accepted");
            sg90.write(90); // Door Opened
            digitalWrite(greenLed, HIGH);
            send_message("Door Opened \nIf it was't you, type 'close' to halt the system.");
            delay(3000);
            digitalWrite(greenLed, LOW);
            sg90.write(0); // Door Closed
            lcd.clear();
            i = 0;
            RFIDMode = true; // Make RFID mode true
        }
        else // If password is not matched
        {
            lcd.clear();
            lcd.print("Wrong Password");
            digitalWrite(buzzerPin, HIGH);
            digitalWrite(redLed, HIGH);
            send_message("Someone Tried with the wrong Password \nType 'close' to halt the
            system.");
            delay(3000);
            digitalWrite(buzzerPin, LOW);
            digitalWrite(redLed, LOW);
            lcd.clear();
            i = 0;
            RFIDMode = true; // Make RFID mode true
        }
    }
}
```

```
    }
  }
}
// Receiving the message
void receive_message()
{
  char incoming_char = 0; // Variable to save incoming SMS characters
  String incomingData; // for storing incoming serial data
  if (SIM900.available() > 0)
  {
    incomingData = SIM900.readString(); // Get the incoming data.
    delay(10);
  }
  // if received command is to open the door
  if (incomingData.indexOf("open") >= 0)
  {
    sg90.write(90);
    NormalMode = true;
    send_message("Opened");
    delay(10000);
    sg90.write(0);
  }
  // if received command is to halt the system
  if (incomingData.indexOf("close") >= 0)
  {
    NormalMode = false;
    send_message("Closed");
  }
  incomingData = "";
}
// Function to send the message
void send_message(String message)
```

```
{  
SIM900.println("AT+CMGF=1"); //Set the GSM Module in Text Mode  
delay(100);  
SIM900.println("AT+CMGS=\"+XXXXXXXXXXXXX\"");  
delay(100);  
SIM900.println(message); // The SMS text you want to send  
delay(100);  
SIM900.println((char)26); // ASCII code of CTRL+Z  
delay(100);  
SIM900.println();  
delay(1000);  
}
```