

Remerciements

Je tiens à exprimer mes plus vifs remerciements à Dr BENADDA Belkacem qui fut pour moi un enseignant durant ma carrière en Master et un bon encadreur attentif et disponible malgré ses nombreuses charges. Sa compétence, sa rigueur scientifique et sa clairvoyance m'ont beaucoup appris. Ils ont été et resteront des moteurs de mon travail ainsi qu'un bon modèle à suivre dans le domaine de la télécommunication.

Je souhaite aussi remercier mes parents, ainsi que ma sœur, et mon frère, pour l'intérêt qu'ils ont toujours porté à ce que je réalise. Merci d'avoir toujours été là et de m'avoir tant aidé et encouragé.

J'exprime tous mes respects et remerciements à l'ensemble des membres de mon jury Dr Messieurs Bahri S.A. et Dr Djennas S.A. J'adresse toute ma gratitude à tous mes ami(e)s et à toutes les personnes qui m'ont aidé dans la réalisation de ce travail.

Enfin, les mots les plus simples étant les plus forts, j'adresse toute mon affection à ma famille, et en particulier à ma maman et mon père. Merci pour avoir fait de moi ce que je suis aujourd'hui et de m'avoir fait croire à mes capacités.



Abstract:

In the late 90s, the term Smart Home was considered something that could only come out of science fiction novels and movies. From then until now, and particularly in the last 5 to 10 years, a lot of things has changed and new concepts have emerged, taking for example the Web 2.0 paradigm and the notion of the Internet of Things, along with dozens if not hundreds of technologies that use these same paradigms. By taking these concepts, adjusting and merging them with open source projects (mostly software API's and libraries), our work to design a close-to-complete Smart Home prototype was achieved.

Our system was designed to contain mainly a central control unit and four types of wireless nodes, the control board which is a Raspberry Pi accounts for the smart side of the system, the one responsible of decision making and information processing.

The wireless nodes are divided into two classes, the first class contains the control nodes and the second represents the sensing nodes. The sensing nodes are two types mainly: the first is the Sensor Node; it collects readings on temperature, humidity and gas levels inside the house, and the second is the Motion Detector which is used by the control board to realize the function of automatic lights. For the control nodes, there is the HVAC controller, which is a board employed specifically to control conventional HVAC systems by switching on and off its different components (compressor, fans and the heater). Another control node is the wireless power switch, that gives us the ability to control any electrical appliance in the house remotely (we used it mainly for automatic lights). All the wireless nodes, except the power switch, are built around the Arduino UNO board: for this, we had to make new shields to improve the workings of any Arduino UNO board so it can achieve the purpose of each node. The power switch was designed using an AVR microcontroller while making sure that this board will be Arduino compatible, this means that we can easily flash this power switch with Arduino code using its proper IDE.

The RF communication between the nodes and the central board is achieved by implementing the cheap RF module NRF24L01 on each board of this system, and to secure the RF links, we managed to create a secure and authenticated communication and control protocol that we named HomeRF. On the software side, and in addition to the Arduino sketches of each node, we realized two apps on the Raspberry Pi (main control board), the first is the main control daemon which is responsible for decision making, and a second app to serve as secure web server allowing the access to manually control the lights or the electrical devices inside the house.

Keywords: Smart Home, NRF24L01, Raspberry Pi, ATmega328, ARDUINO, Nodes, Sensor Node, RF Node, RF, 2.4Ghz, Home Automation, Open Source, Motion Detector, RF Power Switch, Wireless Power Switch, Automatic Lights, Arduino Shield.

Résumé

Ce projet aborde la conception et l'intégration d'un Smart Home. La solution proposée est réalisée par plusieurs modules. Le cœur de ce système est une carte Raspberry Pi, elle fonctionne comme un élément central responsable de l'intelligence et la prise de décision pour contrôler les périphériques de la maison connectée. L'un des nœuds est chargé d'acquérir des lectures sur la température, l'humidité et d'éventuelles fuites de gaz, un autre est implémenté pour la détection de présence ou de mouvement des occupants de la maison, et le troisième est un module de contrôle HVAC qui agit sur le système de climatisation/chauffage central. Ces périphériques, ce sont des nœuds communicants sans-fils conçus autour le matériel Arduino, plus spécifiquement la carte Arduino UNO, dans cette conception on a adapté la notion des Shields Arduino pour convertir n'importe qu'elle carte Arduino UNO en un nœud communicant dont la fonction est dépendante du type de Shield utilisé. Le dernier élément est un interrupteur sans-fil conçu comme une carte compatible Arduino créé spécialement pour ce projet. La communication entre les nœuds et l'élément centrale est assurée par des composants sans fils, en exploitant un protocole sécurisé qu'on a défini particulièrement pour cette maison intelligente. L'aspect logiciel de cette maison intelligente engendre les Sketch Arduino de chaque nœud communicant, le serveur web pour la commande manuel, et le processus de contrôle, ces deux derniers sont implémentés sur l'élément central qui est la carte Raspberry Pi.

Mots clés : Maison Communicante, Smart Home, Maison intelligente, Domotique, ARDUINO, Raspberry PI, Communication Sans-fils, Cryptage, ATmega328, Shields Arduino, Radiofréquence, NRF24L01, RF.

Tables des Matières

<i>Introduction</i>	7
Chapitre I : Etat de L'Art	9
I.1. Evolution Historique	9
I.2. Définitions	9
I.2.1. Principes de la domotique	9
I.2.2. Définition de la Maison Intelligente	10
I.2.3. Intelligence Ambiante	10
I.2.4. Objets Communicants	11
I.3. Etat de l'Art Technologique	11
I.3.1. Le Réseaux Domestique	11
I.3.2. L'infrastructure Electrique	12
I.3.3. L'Infrastructure de Communication	12
I.3.3.1 Le Réseaux Haute Débit	12
I.3.3.2 Le Réseaux Téléphonique	13
I.3.3.3 Le Réseaux Informatique	13
I.3.3.4 Convergence du Réseaux Multimédia	14
I.3.4 Les Technologies de la Domotique	14
I.3.4.1 Les Protocoles Sans-fil	14
I.3.4.2 Courants Porteurs	16
I.3.4.3 Protocoles Filaires	17
I.3.5 Logiciels de la Domotique	20
I.4. Exemples des Maisons Intelligentes	22
I.4.1 Aux Etats-Unis	22
I.4.2 En Asie	23
I.4.3 En Europe	24
I.5. Le Système de Domotique Proposé	26
I.6. Conclusion	28
Chapitre II : Conception des Shields Arduino pour Objets Communicants	29
II.1 Présentation d'Arduino	29
II.1.1 Le Module Arduino	29
II.1.2 Le l'environnement de développement Arduino	30
II.1.3 Le Module Radio NRF24L01	31
II.2 Shields Arduino proposés	32
II.2.1 Le Shield de Capteurs	32
II.2.1.1 Choix de Capteurs	33
II.2.1.2 Capteur de Température	33
II.2.1.3 Capteur de Gaz et de Fumée	34
II.2.1.4 Capteur d'Humidité	35
II.2.1.5 La Transmission-Réception Radio Fréquence	35
II.2.1.6 Réalisation de Shield de Capteurs	37
II.2.1.7 Algorithme Proposé	39
II.2.2 Le Shield de Contrôle HVAC	40
II.2.2.1 Compatibilité avec les systèmes HVAC	40
II.2.2.2 Thermostat à 4 fils	41

II.2.2.3 Conception de Shield HVAC	41
II.2.2.4 Algorithme Proposé	44
II.2.3 Shield de Détection de Mouvement	45
II.2.3.1 Les Détecteurs PIRs	45
II.2.3.2 Détecteurs a Micro-Onde	45
II.2.3.3 Conception de Shield	45
II.2.3.4 Algorithme Proposé	47
Chapitre III : Interrupteur Sans-fil	48
III.1 l'Eclairage Automatique	48
III.1.1 Définition	48
III.1.2 Son Fonctionnement	48
III.1.3 Le Rôle de l'Eclairage Automatique	48
III.1.4 Les Modes d'Utilisation de l'Eclairage Automatique	49
III.1.4.1 Prévention des intrusions	49
III.1.4.2 Prévention des risques d'accident	49
III.2 L'Interrupteur Sans-fil et La Détection de Mouvement	49
III.2.1 Composants Essentiels	50
III.2.2 Préparation de l'ATmega328	50
III.2.2.1 Une Carte Compatible Arduino	50
III.2.2.2 Gravure du Boot Loader Arduino	51
III.3 Conception d'un Interrupteur Sans-fil	52
III.3.1 Choix de Relais	52
III.3.2 L'Alimentation	53
III.3.3 Schéma Final	54
III.3.4 Algorithme de Fonctionnement	57
III.4 Conclusion	58
Chapitre IV : L'Organe Central et le Processus de Contrôle	59
IV.1 L'Organe Central	59
IV.1.1 Rappel	59
IV.1.2 Le Raspberry Pi	60
IV.1.3 Le Raspberry Pi Modèle B	60
IV.1.4 Les Ports GPIO	61
IV.1.5 Aspects Logiciel	61
IV.1.6 Mode de Travail	62
IV.2 Le Processus du Contrôle	62
IV.2.1 Le Protocol HomeRF	63
IV.2.1.1 Le Champ HMAC-SHA-1	64
IV.2.1.2 Le Champ Charge AES	65
IV.2.2 Déroulement du Processus	67
IV.2.2.1 Chargement du Fichier de Configuration	67
IV.2.2.2 Boucle de Contrôle	69
IV.3 Le Serveur Web avec Node.js	70
Conclusion	71
Table des Figures	72
ANNEXE : Codes Sources	73
Bibliographie	74

Introduction :

L'internet des Objets (IoT) s'annonce comme une évolution sans précédent. Les objets sont désormais capables de communiquer entre eux, d'échanger, de réagir et de s'adapter à leur environnement à un niveau beaucoup plus large. Souvent qualifié de 3^{ème} vague de la révolution des nouvelles technologies de l'information, succédant à l'avènement de 'Internet dans les années 1990, puis celui du Web 2.0 dans les années 2000, l'Internet des Objets marque une nouvelle étape dans l'évolution du cyberspace. Cette révolution facilite la création d'objets intelligents permettant des avancées dans de multiples domaines, l'un des domaines les plus affectés par l'émergence de l'IoT est la domotique.

En effet, la prolifération de nouveaux moyens de communication et les nouvelles solutions de traitement de l'information bouleversent les lieux de vie. Le logement, devenu un espace de vie intelligent, doit non seulement être adapté aux personnes qui y vivent, à leurs situations et besoins, mais aussi être prêt à accueillir de nouveaux systèmes conçus pour soulager le quotidien, décupler les possibilités et atteindre un niveau supérieur de services et confort (accès à Internet, télétravail, suivi des consommations, recherche de l'information etc.). Mais malgré l'implication de nombreuses entreprises dans le domaine force est de constater que peu d'applications sont aujourd'hui opérationnelles et diffusées à grande échelle. Les solutions commerciales dans le marché de la domotique sont dominées par des gadgets de control intelligentes comme la lumière automatique ou le thermostat intelligent, mais la solution complète de la domotique appelée la maison intelligente, reste inaccessible au consommateur commun à cause des couts et l'incompatibilité de la majorité de ces solutions avec les maisons déjà construits.

La disponibilité des microcontrôleurs (AVR, PIC ...) et les systèmes embarqué à faible cout comme Arduino, BeagleBone, Raspberry PI, avec leur propre plateformes Open Source, a capturé l'attention des inventeurs en électronique, des experts, et les petites firmes de domotique. Ces outils-là, sont appliqués pour créer des nouveaux concepts dans ce domaine en fournissant un large espace de développement avec des possibilités d'implémentation très vastes et une diversité exceptionnelle, et éventuellement l'apparition d'un nouveau type de ces solutions qui s'appelle le box domotique dans le marché. Avec un seul but, garantir le confort et améliorer la qualité de vie. Dans cette thèse de Master, on va proposer une architecture et un design, d'une maison intelligente sous la forme d'une box domotique à faible cout, qui prend une carte Raspberry Pi comme élément intelligent, et des cartes Arduino comme des cartes de communications et de contrôle, en détaillant le fonctionnement de couche radio fréquence du système, les algorithmes et les protocoles proposé, les technologies utilisé, et expliquer un peu les circuits proposé et le matériel utilisé.

Le mémoire est organisé en 4 chapitres, le premier fait objet d'un état de l'art sur la technologie actuelle des maisons intelligentes : solution commerciale, et notre solution open source proposée dans le cadre de ce projet. Les composantes de notre solution vont être détaillées, au niveau du second chapitre les applications et modules basée sur la technologie ARDUINO, le troisième aborde le développement et l'intégration d'une technologie ARDUINO propre au projet, le quatrième chapitre est consacré à la conception de l'organe centrale de notre projet Maison intelligente.

Chapitre I : Etat de L'Art

I.1. Evolution Historique :

Les premières applications de domotique sont apparues au début des années 1980. Elles sont nées de la miniaturisation des systèmes électroniques et informatiques. Le développement des composants électroniques dans les produits domestiques a amélioré les performances tout en réduisant les coûts de consommations en énergie des équipements :

- Une démarche visant à apporter plus de confort, de sécurité et de convivialité dans la gestion des habitations a ainsi guidé les débuts de la domotique.
- Cela fait maintenant plus de 20 ans que la domotique ne cesse de porter des innovations sur le marché. Mais c'est seulement depuis les années 2000 que la domotique semble être plus intéressante. Certaines institutions de recherche et d'industrie travaillent sur un concept de maison intelligente qui pourrait éventuellement faire naître de nouvelles technologies et attirer d'avantage de consommateurs.
- Actuellement l'avenir de la domotique est assuré. La domotique séduit de plus en plus de particuliers désireux de mieux gérer les nombreuses fonctionnalités de leur maison.
- L'un des espoirs sur lesquels se reposent les professionnels de la domotique consiste à faire de ce concept le meilleur soutien possible pour la réalisation des tâches au quotidien. Depuis 2008, les scientifiques et spécialistes réfléchissent par exemple sur des robots guidant les gens au quotidien.
- Encore une fois, les possibilités sont infinies, limitées uniquement par les technologies disponibles.

I.2. Définitions :

I.2.1. Principes de la domotique :

La domotique, tout le monde en parle sans connaître vraiment de quoi il retourne. Il suffit de consulter les catalogues des constructeurs pour s'en convaincre aisément. Les dictionnaires regorgent de définitions toutes plus ou moins semblables. Le dictionnaire encyclopédique Hachette, édition de 1995, nous dit que la domotique est « l'informatique appliquée à l'ensemble des systèmes de sécurité et de régulation des tâches domestiques destinés à faciliter la vie quotidienne "Vaste programme ! Où donc s'arrête l'électricité, où s'arrêtent les automatismes, où commence la domotique ?

S'il y a encore une quinzaine d'année, il suffisait d'un électricien pour réaliser toute l'installation électrique d'un bâtiment, il en est tout autrement aujourd'hui. Les compétences requises sont multiples – électronique, automatismes, sécurité, thermique, énergétique -, car tous les équipements domestiques sont étroitement couplés. Tous ces équipements sont reliés les uns aux autres par des liaisons spécialisées filaires ou hertziennes.

La centrale permet de gérer l'ensemble de ces matériels et peut-être reliée à un réseau informatique soit directement en IP sur Ethernet, soit par un modem téléphonique sur un réseau commuté résidentiel.

On peut résumer par dire que la domotique est le domaine technologique qui traite de l'automatisation du domicile, d'où l'étymologie du nom qui correspond à la contraction des termes "maison" (en latin "domus") et "automatique". Elle consiste à mettre en place des réseaux reliant différents type d'équipements (électroménager, hifi, équipement domotique, etc.) dans la maison. Ainsi, elle regroupe tout un ensemble de services permettant l'intégration des technologies modernes dans la maison.

1.2.2. Définition de la Maison Intelligente :

Les définitions de la maison intelligente provoquent des fois des ambiguïtés, principalement la confusion entre les termes « Domotique », et « Maison Intelligente ». La domotique (home automation) aujourd'hui ce terme est plutôt remplacé par celui de maison intelligente qui signifie un paradigme qui se positionne en successeur de la domotique, bénéficiant des avancées en informatique ubiquitaire que l'on dénomme aussi l'informatique ambiante, intégrant notamment l'internet des objets. Outre la dimension dominante de l'informatique, la maison intelligente telle que représentée dans les années 2010 se veut également plus centrée utilisateur, s'éloignant de l'approche technophile caractéristique de la domotique des années 1990. Le principe de fonctionnement d'une maison intelligente consiste à centraliser le contrôle-commande. À la différence d'une installation électrique classique, les circuits de commande et de puissance sont séparés. Il devient ainsi possible d'établir des liens entre les organes de commande et les récepteurs d'ordres, qui appartiennent habituellement à des sous-systèmes indépendants. En 2009, les premiers box de domotique sont apparus sur le marché. Contrairement aux précédentes solutions filaires souvent très coûteuses, les box de domotique utilisent la puissance d'Internet et du sans-fil. Avec ou sans abonnement, elles permettent une utilisation ouverte et sont pilotables depuis un ordinateur, un smartphone ou une tablette tactile. L'installation est très simple, et se fait en quelques minutes par un utilisateur non expérimenté.

1.2.3. Intelligence Ambiante

Du fait des évolutions technologiques et de la miniaturisation, l'ordinateur n'a plus une activité seulement centrée sur son utilisateur final, l'informatique prend aussi en compte l'environnement grâce à des informations issues de capteurs qui peuvent communiquer entre eux. L'environnement peut être très divers : la maison, la voiture, le bureau, la salle de réunions. Avant le développement

du concept intelligence ambiante, le terme informatique ubiquitaire a été proposé par Weiser en 1988^[1] dans le projet Xerox PARC pour désigner un modèle d'interaction homme-machine dans lequel le traitement de l'information relatif aux activités de la vie quotidienne a été intégré dans les objets.

La vision de l'informatique ubiquitaire a été permise par l'utilisation de très nombreux dispositifs miniaturisés, économiques, robustes et connectés par réseau que l'on a pu dès lors installer dans des endroits où les personnes réalisent leurs activités quotidiennes. Ensuite, Uwe Hansmann de la société IBM a proposé le terme informatique « pervasive » pour décrire cette technologie. Dans la littérature, il existe une grande confusion concernant l'utilisation de ces deux termes et celui d'intelligence ambiante. Ils sont souvent utilisés de façon indistincte. Pourtant, nous considérons plus pertinente la distinction proposée par Augusto J. C.^[2] qui affirme que le concept d'intelligence ambiante recouvre un plus large domaine que la simple disponibilité omniprésente de ressources et qu'elle nécessite l'application de l'intelligence artificielle pour atteindre son but, à savoir avoir un comportement judicieux et faire preuve d'initiative. Ainsi à la différence des deux autres concepts, l'intelligence ambiante fait appel à des contributions d'autres domaines comme l'apprentissage automatique, les systèmes multi-agents ou la robotique.

I.2.4. Objets Communicants :

Le M2M (lire Machine-to-Machine) regroupe les solutions permettant aux machines de communiquer entre elles ou avec un serveur central sans intervention humaine. Il s'agit d'un marché récent en plein essor, poussé par des dynamiques favorables au niveau technologique et économique et son développement variera en fonction du secteur d'activité. Cependant, on peut d'ores-et-déjà identifier celui de la sécurité et de la santé qui figurent parmi les plus prometteurs... Au niveau domotique on observe bien entendu le même essor de ces objets communicants. Les objets de la vie quotidienne s'équipent de solutions de communication, comme les chaudières (Thermibox d'ELM Leblanc qui s'appuie sur l'expertise M2M de Webdyn), les appareils électroménagers, etc. Si les fonctionnalités premières de nos équipements domestiques d'hier restent les mêmes, ceux-ci voient leur capacités décuplées du fait de cette « interconnexion ». Mais on observe aussi de nombreux objets spécialisés qui sont apparus il y a peu de temps.

I.3. Etat de l'Art Technologique :

I.3.1. Le Réseaux Domestique :

Au sein d'une habitation, le réseau peut être divisé en quatre « sous-réseaux » parmi lesquels on retrouve le réseau électrique, domotique, multimédia et de communication. Il est à noter que cette nomenclature est ici simplifiée et concentrée sur la partie « technologique électrique et électronique ». En effet, pour être réellement exhaustif, ce réseau domestique comprend aussi le réseau d'eau, le réseau de chauffage, réseau de circulation d'air, réseau de déchet etc. Les réseaux

électriques et de communication sont interconnectés aux infrastructures de distribution des opérateurs. La « libéralisation » des marchés de l'énergie et des télécommunications a conduit que les fournisseurs d'infrastructure ne sont plus les opérateurs de services.

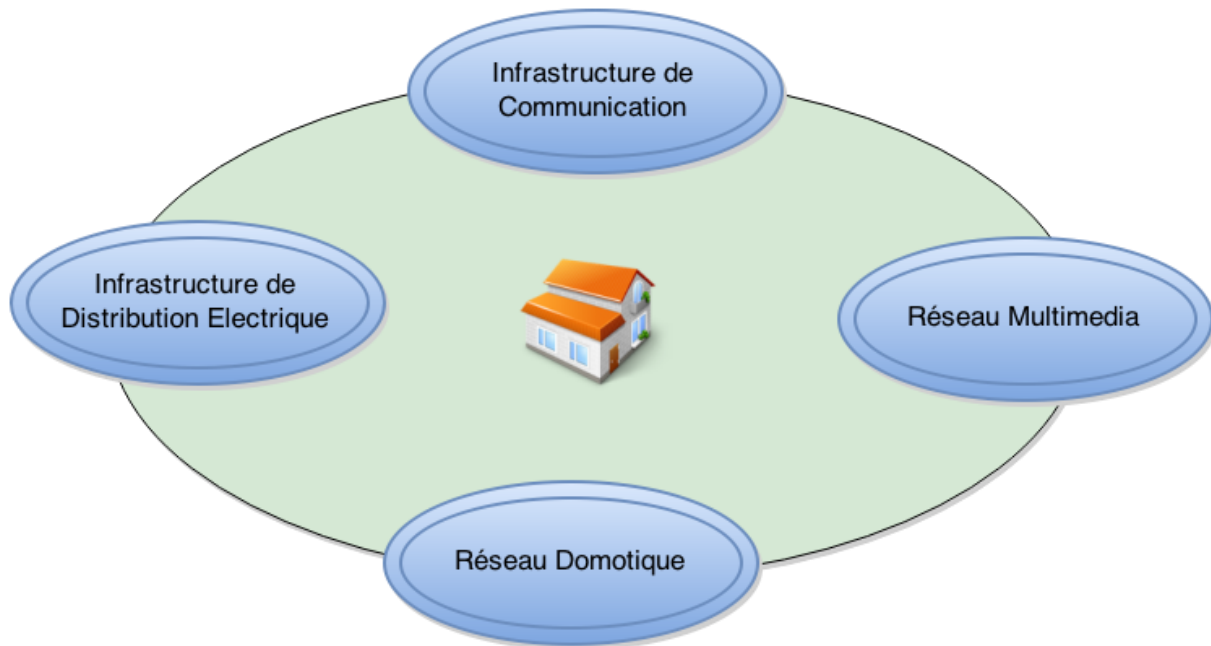


Figure 1. Le Réseaux Domestique.

I.3.2. L'infrastructure Electrique :

Installation indispensable au bon fonctionnement d'un logement, il s'agit d'une part de l'infrastructure externe d'alimentation électrique du logement mais aussi du câblage électrique interne de l'ensemble des pièces regroupé sur le tableau électrique.

Dans ce réseau électrique interne, on y retrouve à la fois le circuit de puissance (c'est-à-dire l'alimentation de l'installation) et le circuit de commande (interrupteurs, variateurs, ...). Pour résumer, le lien entre les interrupteurs et les circuits à commander est câblé une fois pour toute dans les murs, sans aucune possibilité d'évolution (sauf engagement de travaux).

Ce type d'installation est parfaitement maîtrisé par les électriciens.

I.3.3. L'infrastructure de Communication :

Il s'agit ici de l'infrastructure de communication qui va permettre de « connecter » l'habitat avec le monde extérieur, cela comprend donc à la fois l'accès à la téléphonie (fixe et mobile) et l'accès au haut débit Internet.

I.3.3.1 Le Réseaux Haute Débit :

Quand l'ADSL (2Mbit/s, 256Kbit/s) est disponible, c'est actuellement le meilleur compromis par rapport aux solutions bas débit de type RTC (56Kbit/s, 33,6Kbit/s) ou RNIS (128Kbit/s, 128Kbit/s) et celles par câble (2Mbit/s, 256Kbit/s) ou satellite (512Kbit/s, 128Kbit/s). L'ADSL se décline suivant trois

versions différentes suivant la localisation de l'habitation (ADSL, Re-ADSL et ADSL2+). Avec le dégroupage, il est possible de s'adresser au fournisseur d'accès de son choix et de bénéficier d'offres ADSL regroupant l'accès Internet, la téléphonie illimitée et la télévision sur IP. L'ADSL ne peut fournir des débits élevés qu'aux foyers proches d'un central téléphonique.

Les technologies de téléphonie mobile dites de 3ème génération (ou 3G et 3G+) permettent d'accéder à Internet depuis son téléphone portable ou son PC portable. A l'heure où beaucoup renoncent à leur ligne fixe au profit d'un abonnement à la téléphonie mobile, ces types de technologies 3G/3G+ sont des solutions intéressantes, alternatives à l'ADSL, pour accéder à Internet, à condition toutefois qu'elle soit couverte par ce type de réseau !

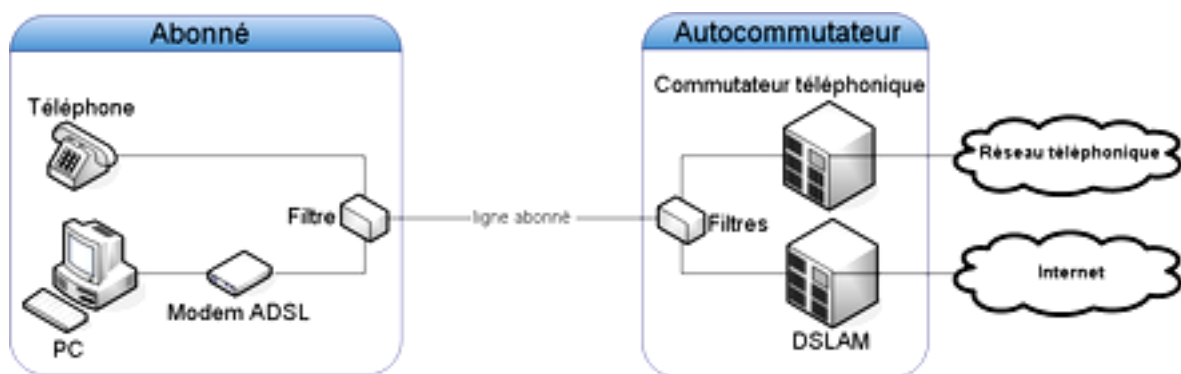


Figure 2. Schéma d'une liaison ADSL

1.3.3.2 Le Réseaux Téléphonique :

L'installation téléphonique classique repose donc sur le raccordement filaire du logement au réseau de l'opérateur. On parle souvent de la ligne RTC. La ligne téléphonique arrive donc jusqu'au logement et est distribuée dans l'ensemble des pièces par l'intermédiaire de prises en T. Ce type d'installation n'autorise que des échanges à bas débit et n'offre aucune évolutivité ni possibilité de réorganisation. C'est pourquoi ce type d'infrastructure téléphonique tend à disparaître, tant à l'échelle inter bâtiment qu'intra bâtiment (émergence de la téléphonie sur IP, rapidement appelée voix sur IP).

L'existence de la téléphonie en technologie réseau commuté des années 80 (téléphonie fixe réseau RTC) est en débat. Sa disparition totale est sans doute plus problématique, mais de plus en plus d'abonnés choisissent les solutions des fournisseurs d'accès internet, ce qui n'est pas sans poser des problèmes pour certains services demandant une forte qualité de services téléphonique. Le doublement des voies de communication par la téléphonie mobile est alors la solution incontournable (GSM ou GPRS), notamment pour les systèmes de téléalarmes indispensable pour les services de maintien à domicile.

1.3.3.3 Le Réseaux Informatique :

Il s'agit ici du réseau informatique interne au logement, c'est-à-dire le réseau qui connecte l'ensemble des éléments informatiques présent au sein de l'habitat, parmi ceux-ci, on peut lister :

Ordinateur(s) fixe(s) ; Ordinateur(s) portable(s) ; Tablette PC ; PDA ou Pocket pc ; Téléphone mobile ; Imprimante ; Serveur bureautique ; Serveur domotique ; Serveur multimédia. Ces différents appareils sont interconnectés par des réseaux sans fil (wifi) ou filaire (réseau VDI ou courant porteurs).

I.3.3.4 Convergence du Réseaux Multimédia :

Pour pouvoir écouter de la musique chez soi (ou écouter la radio), il est toujours possible et encore d'actualité de transporter son poste radio ou son lecteur CD ou MP3 dans chaque pièce de la maison ou d'en installer un peu partout dans ces pièces. Cependant, dans cette ère technologique, il est envisageable de mutualiser tous les appareils et de déployer alors un réseau de diffusion sonore.

L'image prend de plus en plus d'importance dans notre quotidien et dans nos maisons et devient un élément de confort à part entière. Plusieurs télévisions sont maintenant disponibles au sein d'un même foyer (entre deux ou trois en moyenne).

La maison ne doit plus comporter aucune prise TV ni téléphonique, elles sont remplacées par des prises de communication universelle RJ 45 reliées à une boîtier de communication sur lequel on peut connecter un téléviseur, un téléphone ou un ordinateur. Dans cette configuration, chaque prise RJ-45 20 est raccordée directement au boîtier de communication et peut donc évoluer indépendamment des autres (transformer un poste téléphonique en poste informatique devient donc très facile). L'infrastructure VDI (pour Voix, Données, Images) permet la distribution, comme son nom l'indique, de trois types de données : le signal téléphonique, les données informatiques et la télévision, et permet d'apporter un maximum de souplesse dans la distribution. La distribution peut aussi se faire par wifi ou courant porteur. Ce réseau unifie donc tous les besoins en termes de téléphonie, interphonie, diffusion audio et vidéo, informatique et de télévision, en fonction des sources (hertzien, satellite, ADSL, internet, ou à partir de source locale (ordinateur, enregistreur, DVD etc.) et à condition que les adaptateurs de convergence le permette.

I.3.4 Les Technologies de la Domotique :

L'essence même d'une installation domotique est la communication entre ses différents éléments. Pour cela, beaucoup de protocoles sont nés, car chaque constructeur a réalisé son protocole de communication, ce qui a conduit à une situation très complexe. Les protocoles présentés ci-dessous ne sont pas des protocoles propriétaires : la plupart sont standardisés et/ou ouverts.

I.3.4.1 Les Protocoles Sans-fil :

Les protocoles sans fil sont très appréciés actuellement, la grande liberté de placement des capteurs et interrupteurs qu'ils apportent leurs permettent d'être placés dans des endroits parfois improbables, bien souvent dans ce que l'on appelle les « derniers mètres », ces endroits depuis lesquels on a besoin d'informations, mais où il est relativement coûteux de câbler un bus de terrain dédié. Ils permettent en plus de ne pas avoir à câbler certaines parties, afin de pouvoir les rénover/ré-agencer plus facilement dans le futur. Ces protocoles imposent parfois l'utilisation de

piles, le principal défaut est donc la durée de vie de ces dernières, dans certains cas, elle chute à quelques mois, ce qui est évidemment très contraignant. La faible portée (espace libre : 300 m, une habitation : environ 30 m) de ces équipements les réservent à des usages bien définis, mais dans le cas d'une maison individuelle, les limitations sont pour la plupart acceptables.

Les protocoles présentés ci-après utilisent les fréquences 868 MHz pour l'Europe et 315 MHz pour l'Amérique du Nord :

EnOcean: est une technologie radio (868 MHz) normalisée IEC (ISO/IEC 14543-3-10) promue par l'*EnOcean Alliance* et par la société EnOcean.

Le but de ce protocole est de faire dialoguer divers appareils en utilisant la récolte d'énergie environnante. Les équipements *EnOcean* sont donc sans fil et sans pile ! L'énergie récoltée dans l'environnement peut provenir de divers principes physiques :

- effet **piézoélectrique inverse** ;
- effet **photoélectrique** ;
- effet **Seebeck**.

Des recherches sont en cours, pour récupérer l'énergie provenant de vibrations, ou l'énergie du champ électromagnétique environnant. Il est bien sûr évident que l'optimisation énergétique qui a dû être menée est très poussée, afin de pouvoir soutenir des transmissions radio avec si peu d'énergie. Une super-capacité est souvent ajoutée au sein de ces équipements, afin qu'ils puissent émettre même en cas de pénurie de leur énergie primaire ; et certains affichent plusieurs mois d'autonomie dans ces conditions.

La communication entre dispositifs se fait via un appairage manuel préalable ; ensuite, chaque dispositif peut adresser jusqu'à 20 autres dispositifs. Cette norme est libre en ce qui concerne l'implémentation, néanmoins, beaucoup d'acteurs intègrent l'*EnOcean Alliance* afin de pouvoir bénéficier de licences des brevets concernant la récolte d'énergie détenus par l'Alliance.

802.15.4 : La norme 802.15.4 est une norme IEEE pour les réseaux sans fil de la famille des LR-WPAN (*Low Rate Wireless Personal Area Network*). Sur le modèle OSI, ce protocole correspond aux couches physiques et liaison de données, et permet de créer des réseaux sans fil de type maillé ou en étoile. Il est relativement aisé de trouver chez les revendeurs spécialisés des émetteurs-récepteurs 802.15.4 incluant des micro-processeurs et 128 ko de mémoire vive embarquée pour mettre en œuvre toute sorte d'applications au-dessus de 802.15.4.

6LoWPAN: est une abréviation de *IPv6 Low power Wireless Personal Area Networks*. Ce projet de l'IETF vise à définir les mécanismes d'encapsulation et de compression d'en-tête des protocoles IPv4 et surtout IPv6 pour la norme 802.15.4.

Ce projet, bien que disposant déjà de produits commercialisés, n'est pas encore aussi mature que les autres solutions présentées plus haut. Il devrait arriver à maturité à moyen terme, et reçoit pour le moment un très bon accueil par les acteurs du milieu, ce qui devrait lui donner un avenir radieux. L'intégration de la pile 6LowPAN a été réalisée dans le noyau Linux depuis la version 3.3 et le travail continue sur ce sujet.

Z-Wave : Z-Wave a été développé par la société danoise Zen-Sys qui a été rachetée par la société américaine Sigma Designs en 2008. Il communique en utilisant une technologie radio de faible puissance dans la bande de fréquence de 868,42 MHz. Le protocole radio Z-Wave est optimisé pour des échanges à faible bande passante (entre 9 et 40 kbps) et des appareils sur pile ou alimentés électriquement, par opposition au Wi-Fi par exemple, qui est prévu pour des échanges à haut débit et sur des appareils alimentés électriquement uniquement.

Z-Wave fonctionne dans la gamme de fréquences sous-gigahertz, qui dépend des régions (868 MHz en Europe, 908 MHz aux US, et d'autres fréquences suivant les bandes ISM des régions). La portée est d'environ 50 m (davantage en extérieur, moins en intérieur). La technologie utilise la technologie du maillage (Mesh Network) pour augmenter la portée et la fiabilité.

ZigBee : (dont le nom provient de l'analogie avec une abeille qui zigzague pour trouver son point d'arrivée) est un protocole libre régi par la *ZigBee Alliance*.

Le protocole ZigBee fonctionne généralement au-dessus de 802.15.4, il implémente les couches réseaux et applicatives du modèle OSI. Cette mise en œuvre permet de profiter des avantages de la norme 802.15.4 en termes de communication. Les principaux ajouts sont l'ajout des couches réseau et applicatives qui permettent, entre autres, de chacune réaliser du routage de messages ; l'ajout des ZDO (*ZigBee Device Object*) régis par la spécification ; et l'ajout des objets personnalisés par les constructeurs.

Ce protocole souffre tout de même de certains problèmes, le plus important étant un problème d'interopérabilité. Comme vu précédemment, le protocole laisse aux constructeurs la possibilité de définir leurs propres objets applicatifs. Les constructeurs ne s'en sont bien sûr pas privés, ce qui cause des incompatibilités totales, certains constructeurs ayant ré-implémenté leurs propres protocoles non documentés au-dessus de ZigBee. L'intégration de la pile ZigBee/802.15.4 est réalisée dans le noyau Linux depuis la version 2.6.31. ZigBee amorce sa mutation vers un réseau IP via la spécification *Smart Energy Profile version 2.0*.

1.3.4.2 Courants Porteurs :

Les protocoles utilisant les courants porteurs sont très prisés actuellement, car ils permettent de réduire le câblage et n'ont supposément pas recours aux radiofréquences. Ils présentent néanmoins des désavantages, ils sont très rapidement perturbés par l'environnement électrique (radiateurs, gradateurs...), ils ne franchissent pas, ou très mal, les transformateurs électriques, et le

rayonnement électromagnétique des câbles dans lesquels ils passent en font de très bons émetteurs radio.

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

I.3.4.3 Protocoles Filaires :

Les protocoles sans fil sont souvent appuyés par un bus de terrain qui permet d'étendre les capacités globales de l'installation. On distingue parmi les protocoles filaires deux grandes familles, les centralisés, ceux qui utilisent un automate ou un serveur pour régir l'ensemble de l'installation ; et l'autre catégorie, les protocoles décentralisés où les capteurs et actionneurs dialoguent directement les uns avec les autres, sans point central. Chaque approche a ses avantages et inconvénients.

Modbus : est un ancien protocole placé dans le domaine public fonctionnant sur le mode de couche applicative maître-esclave. Il fonctionne sur différents médias : RS-232, RS-485 ou Ethernet. Ce protocole impose une centralisation du fait de son utilisation d'un maître. Il supporte jusqu'à 240 esclaves. Son utilisation pour la domotique est maintenant anecdotique ou réservée aux projets économiques à la construction.

DALI : (*Digital Addressable Lighting Interface*) est un protocole standardisé IEC 60929 et IEC 62386 qui permet de gérer une installation d'éclairage par l'intermédiaire d'un bus de communication à deux fils. Il est le successeur du 0 - 10 V pour la variation de l'intensité lumineuse. Ses capacités sont limitées (64 luminaires répartis en 16 groupes au maximum par bus), mais on peut utiliser des gestionnaires pouvant interconnecter plusieurs bus (des automates, par exemple). Le DALI est souvent utilisé dans le tertiaire (bureau) ou, dans une moindre mesure, dans l'habitat résidentiel.

DMX512 : Parmi les méthodes de contrôle d'éclairage, on retrouve plusieurs normes bien définies et utilisées depuis longtemps. C'est le cas du DMX512, plus communément appelé DMX (*Digital Multiplexing*). Il est principalement utilisé dans le monde de la scène (concerts, plateaux télé, spectacles son & lumière), pour le contrôle de l'éclairage dynamique. Le DMX 512 est, à ce jour, le protocole le plus répandu et le plus universel, il est utilisé partout et par tous les fabricants de matériel d'éclairage scénique, ce qui permet de trouver des blocs de puissance gradateurs capables de varier plusieurs équipements, à des prix très abordables. Ces blocs peuvent

aussi supporter des puissances supérieures à ce qu'on pourrait faire en DALI. Le DMX 512 utilise une liaison série RS-485 pour contrôler 512 canaux en leur affectant une valeur comprise entre 0 et 255.

1-Wire : est un bus de communication, très proche dans le fonctionnement du bus I²C. Ce bus n'est actuellement plus beaucoup utilisé pour la domotique, bien que certaines installations subsistent.

KNX : est un standard ouvert (ISO/IEC 14543-3) né de la fusion de trois spécifications de protocoles : EIB, EHS et Bâtibus. Il est surtout utilisé en Europe. KNX est décrit par une spécification écrite par les membres de l'association KNX, qui se charge également de la promotion et du logiciel de configuration de référence (logiciel propriétaire ETS). Différentes couches physiques peuvent être mises en place pour KNX, la paire torsadée et Ethernet sont les plus répandues, mais d'autres peuvent également être rencontrées, bien que très marginales : infrarouge, courant porteur et transmission radio. Ces couches physiques sont en réalité très lentes (excepté Ethernet) et pénalisent le protocole pour les réseaux conséquents.

À l'utilisation, ce protocole se révèle décentralisé, les capteurs communiquent directement aux actionneurs qu'ils doivent régir, sans passer par un point central. La configuration d'un réseau se fait avec le logiciel propriétaire dédié ETS (conçu par l'association KNX), d'autres logiciels existent, mais ont une visibilité très faible comparativement au mastodonte ETS. Lors d'un changement dans le comportement du réseau, le fonctionnement du protocole impose un rechargement total du micro-logiciel (*firmware*) du ou des équipements concernés (capteur ou actionneur).

La mise en œuvre est relativement complexe et le protocole révèle des possibilités au final assez faibles et qui dépendent très fortement des micro-logiciels des équipements. Là aussi, une installation n'ayant qu'une marque d'équipements est préférable, afin de profiter au maximum des possibilités de ceux-ci.

L'évolutivité d'une installation de ce type est très faible, à moins d'avoir conservé toute la configuration mise en place (micro-logiciels compris, ce qui peut rapidement être lourd), et la logique de fonctionnement est assez complexe à appréhender pour un non-habitué.

LonWorks : est un réseau de bâtiment au niveau terrain historiquement créé par l'entreprise californienne Echelon, qui fournit maintenant le matériel de base (puces avec le protocole LonTalk embarqué). Ce réseau utilise comme protocole LonTalk, standardisé ANSI/CEA-709.1-B, et de libre utilisation. Il est très utilisé comme bus de terrain pour commander des équipements CVC (chauffage, ventilation, climatisation), ainsi que pour la commande de l'éclairage. Géographiquement, il est principalement utilisé aux États-Unis, en Angleterre, et dans plusieurs pays d'Asie ; ce qui en fait probablement le réseau de ce type avec le plus d'installations à travers le monde.

De la même manière que KNX, LonWorks est un réseau du type décentralisé. Cela lui permet de communiquer à très longue distance avec une vitesse de 78 kb/s. La vitesse dépend de la couche

physique utilisée, parmi lesquelles se trouvent : paire torsadée, courant porteur, fibre optique et ondes radio.

LonWorks a plusieurs avantages, mais l'un des plus importants est l'interopérabilité. L'utilisation de SNVT (*Standard Network Variable Type*), variables de réseau standardisées, pour la communication entre nœuds, oblige les intégrateurs à réaliser des configurations propres. De plus, les fabricants sont fortement incités à créer leurs nouveaux produits en respectant l'utilisation des SNVT, ce qui assure une compatibilité maximale entre les marques.

LonMark International est l'association dédiée à la promotion et à l'évolution des équipements compatibles avec LonWorks. Elle gère et maintient les normes liées au développement entre les fabricants qui font partie de cette association ; elle gère également la publicité de la norme et des produits, les certifications, l'annulation/création de SNVT, etc.

Il existe plusieurs logiciels pour mettre en œuvre un réseau LonWorks : NL220, LonMaker, entre autres. La flexibilité logicielle qu'offre LonWorks aux fabricants est telle, que n'importe qui peut développer un logiciel capable de mettre en marche un réseau de ce type, en se conformant aux documentations. Outre ces avantages, se trouve la possibilité de créer des « *LNS Plugins* », logiciels qui permettent la configuration d'un produit ou d'un réseau au travers d'une interface graphique indépendamment du logiciel utilisé pour créer le réseau. La configuration du réseau est sauvegardée dans la « *LNS Database* », base de données de taille très modeste qui définit tout le réseau et qui est commune à tous les logiciels de configuration. Des projets sur LonWorks et Linux sont mis en œuvre, tels que *lonworks4linux*, mais ils ne sont pas encore bien définis.

Protocol xPL : Le projet *xPL* vise à fournir un protocole unifié pour le contrôle et la supervision de tous les équipements de la maison. Ce protocole se veut simple dans sa structure, tout en mettant à disposition un panel de fonctionnalités important. Il dispose, par exemple, de fonctions d'auto-découverte et d'auto-configuration qui lui permettent d'être « *Plug and Play* », contrairement à beaucoup d'autres protocoles de domotique. De part ses origines complètement libres, il est implémenté dans beaucoup de logiciels libres de domotique, mais il est très dur de trouver du matériel compatible pour équiper un habitat. Il est simple à mettre en œuvre et fait partie d'appareils comportant le principe « branchez et utilisez ». Sa devise: « Léger sur le câble par conception ». Dans un réseau local, il utilise le protocole UDP.

BACnet : (*Building Automation and Control networks*) est un protocole de communication spécifié par l'ASHRAE et est un standard ISO et ANSI. Il s'agit d'un protocole de couche réseau pouvant être utilisé au-dessus de plusieurs technologies de couches liaison et physique, telles que LonTalk, UDP/IP...

BACnet intègre également la couche application grâce à un ensemble d'objets dédiés. Ces objets représentent les informations gérées et échangées par les appareils. Son approche objet au plus proche des couches applicatives en fait un bon candidat comme protocole de haut niveau dans une

installation GTB ou domotique. BACnet est souvent perçu comme le protocole qui permettra d'unifier tous les autres, grâce à ses fonctionnalités avancées. Il est donc actuellement très apprécié pour la supervision dans la Gestion Technique de Bâtiment.

I.3.5 Logiciels de la Domotique :

OpenHAB : (*Open Home Automation Bus*) est un logiciel de domotique écrit en Java avec OSGi et publié sous la licence GPL v3. *OpenHAB* embarque un serveur Web pour son interface, il dispose également d'applications pour iOS et Android. Le logiciel peut être commandé d'une façon particulière : en lui envoyant des ordres via XMPP (Protocol De Communication). Le développement est actif, de nombreux modules de communication avec d'autres protocoles devraient arriver dans les versions ultérieures.

FHEM : est un serveur de domotique écrit en Perl sous licence GPL v2. Ce logiciel d'origine allemande permet de gérer les protocoles FS20, 1-Wire, X10, KNX et EnOcean. Sa documentation et ses forums majoritairement en allemand sont un point repoussant pour beaucoup d'utilisateurs.

HEYU : est un programme de domotique utilisable en ligne de commande. Ce programme est écrit en C et est sous licence GPL v3 (les anciennes versions disposent d'une licence particulière). HEYU est spécifiquement destiné au protocole X10, pour communiquer avec ce réseau, l'interface privilégiée est le CM11A de XA0 Inc. Ce projet n'est depuis peu plus très actif, son ouverture tardive et son utilisation exclusive de X10 ont sans doute provoqué son abandon.

DomotiGa : est un logiciel de domotique pour GNU/Linux écrit en Gambas et sous licence GPL v3, ses origines sont néerlandaises. Ce logiciel est compatible avec les réseaux 1-Wire, KNX, X10, xPL, Z-Wave et bien d'autres encore.

MisterHouse : est un logiciel multi-plate-forme écrit en Perl sous licence GPL. Ce logiciel est vieillissant et ne semble plus maintenu, il est revient tout de même régulièrement en avant lors des recherches sur la domotique libre. De part ses racines américaines, ce logiciel permet de gérer les réseaux X10, Z-Wave, EIB, 1-Wire.

Domogik : est un logiciel écrit en Python sous licence GPL v3+. C'est né sur le forum *ubuntu-fr.org* entre plusieurs personnes qui souhaitaient un logiciel de domotique. Il est en développement actif et permet pour le moment une gestion basique de l'habitat. Son architecture repose sur le protocole *xPL* en interne. Il étend petit à petit ses fonctionnalités vers les protocoles les plus utilisés en domotique. Pour le moment, les protocoles suivants sont gérés : x10, 1-Wire,

LPX800, Teleinfo, RFID, Wake on LAN / Ping. Le logiciel dispose d'une interface Web et d'une application Android.

Calaos : est une solution commerciale de domotique basée sur un automate *Modbus* et un serveur domotique GNU/Linux. Le code des applications est en majorité libre sous licence GPL v3. La solution est destinée en priorité à des maisons neuves avec une intégration forte au moment de l'étude de la construction. Une installation Calaos utilise des automates qui permettent le pilotage de tous les éléments électriques de la maison, ainsi que de faire l'acquisition des interrupteurs, sondes de température, capteurs de présence, etc. L'automate est capable d'interagir avec des bus de terrain tels que le DALI ou le KNX. Viens ensuite le serveur domotique, qui va piloter l'automate et ainsi gérer l'ensemble des règles de la maison (comme l'appui sur un interrupteur qui engendre le lancement d'un scénario). Il donne également accès à la maison sous différentes formes d'interfaces : Web, tactile (basée sur les EFL), applications mobiles, etc. Un système Calaos est aussi capable de gérer des caméras IP, ainsi que le multi-pièce (*multi-room*) audio.

OpenRemote : Le but de OpenRemote est de créer une pile logicielle libre que les constructeurs pourront intégrer à très bas coût dans leurs produits, afin de créer des surfaces de contrôle pour le bâtiment. OpenRemote supporte un grand nombre de protocoles parmi lesquels : X10, KNX, Z-Wave, ZigBee, 6LoWpan, etc. L'idée étant de réutiliser les écrans déjà présents dans les lieux de vie tels que les *smartphones*, les tablettes et les ordinateurs de bureau. Sont donc actuellement supportés : Android, iOS, ainsi que GWT pour les applications Web. L'ensemble du code est sous licence AGPL.

LinuxMCE : est une distribution GNU/Linux qui vise à fournir une solution complète et intégrée de gestion multimédia, de domotique, de téléphonie et de sécurité pour une maison. Elle repose sur beaucoup de logiciels libres, tels que VLC, Asterisk, Xine, etc. Tous ces logiciels sont mis en œuvre conjointement afin de créer un ensemble cohérent. Beaucoup de code additionnel permet de réaliser les diverses interfaces de configuration et de contrôle. Cette distribution en développement permet de gérer les protocoles de domotique suivants : KNX, 1-Wire, DMX, X10, EnOcean...Elle prend pour base d'anciennes versions (10.04, pour la version en développement) de Kubuntu. Elle représente probablement la solution libre la plus aboutie actuellement, ses développeurs la comparent à des solutions propriétaires à plus de 100 000 \$ USD. Malheureusement, elle impose l'utilisation d'une distribution dédiée, les plus débrouillards pourront la décortiquer pour en extraire les composants qui les intéressent et recréer un serveur de domotique sur un de leur serveur domestique.

Beaucoup d'autres solutions libres existent, on peut par exemple citer **openAMI**, **Minerva**, **Freedomotic**, etc. Cette liste est loin d'être exhaustive — des recherches sur **Gitorious** ou des plates-

formes équivalentes peuvent vous retourner des dizaines de résultats —, les logiciels présentés ci-dessus sont principalement ceux ayant une bonne visibilité sur le Web.

Voici un aperçu des différentes technologies utilisées en domotique, à ces technologies, on peut ajouter celles qui auront peut-être une grande présence dans le futur, telle **GreenNet** qui a pour ambition de créer des réseaux de minuscules micro-ordinateurs équipés de processeurs à 72 MHz et de liaisons à 2 Mb/s, basés sur la récolte d'énergie ; ou encore **CAN**, avec **CANopen** qui arrive depuis le monde des bus embarqués. On assiste globalement à une grande convergence des technologies de domotique vers l'Internet, à travers l'utilisation de plus en plus massive de technologies réseau telles qu'IP. Globalement, les grands silos de données qui ont été constitués au fil des ans par les différents constructeurs s'ouvrent progressivement au monde extérieur.

I.4 Exemples des Maisons Intelligentes :

Plusieurs habitats ayant le sens de l'exploration du monde et l'amélioration on imaginer certains concepts techniques visant deux objectifs majeurs. Un premier objectif est d'optimiser le confort, le bien-être et la qualité de vie de ses habitants et dans ce cas, la maison et ses différents appareils électroménagers sont équipés de capteurs et de fin de course. Au-delà des objectifs de confort, des loisirs et de surveillance technique, un second objectif est de veiller sur les personnes ayant des handicaps moteurs, visuels auditifs ou cognitifs ainsi que sur les personnes âgées, dans le cadre du maintien à domicile. Dans ce cas, les dispositifs intégrés à la maison ont parfois des fonctionnalités de surveillance biomédicales. Tous ces dispositifs sont reliés en réseau à un serveur central qui collecte et traite les données afin de fournir d'une part des diagnostics de situations et coordonner les procédures d'assistance, d'intervention, et de commande, d'autre part, les différents dispositifs d'assistance, appareils domestiques et appareils de régulation du confort. Nous proposons ici une liste des principales expérimentations qui se sont déroulées à travers le monde :

I.4.1 Aux Etats-Unis :

- L'université de Floride propose la « Maison de Matilda » qui comporte un système de localisation des habitants basé sur les ultrasons. L'objectif du projet est d'optimiser le confort en contrôlant l'environnement en fonction des habitudes de mobilité des usagers. Expérimenté en laboratoire, le système de localisation est composé de pilotes ultrasons (récepteurs), installés aux quatre coins de la pièce surveillée, et de deux balises légères (émetteurs), placées sur les épaules de la personne. Un calcul précis de l'emplacement de la personne dans la pièce est alors effectué par triangulation ^[3].
- La même équipe propose un projet de 2ème génération de maison intelligente nommé «Gator Tech Smart House » qui comporte un grand nombre d'équipements « intelligents » : boîte aux lettres, porte d'entrée, lit, salle de bain, miroirs (utilisés comme pense bête),

plancher sensitif, etc. Le système de localisation est identique et tous les dispositifs sont couplés à des capteurs et des actionneurs reliés à une plateforme opérationnelle optimisant le confort et la sécurité des personnes âgées à domicile ^[4].

- Le laboratoire « House_n » du MIT propose le projet de « la maison du futur » dont le but est d'équiper une maison intelligente afin d'entreprendre des études qualitatives et quantitatives sur les relations entre l'environnement, les comportements des individus et les technologies de l'information. L'objectif est d'identifier et de proposer de nouveaux services. Les capteurs sont installés sur des meubles et sur des appareils domestiques (lave-vaisselle, machine à laver, dispositifs dans la salle de bains, etc.). Le principe est basé sur trois types de traitements : la collecte des changements d'états des capteurs, l'enregistrement par l'utilisateur de ses propres activités en utilisant l'interface MSE et la construction de modèles, basés sur les réseaux bayésiens, permettant de prédire les activités de l'utilisateur. Au démarrage de l'étude, les sujets disposent d'un PDA doté de l'interface MSE pour saisir leurs activités. Les principaux problèmes rencontrés dans cette étude ont concerné : la qualité et le nombre d'échantillons d'activités car plus d'un mois de collecte sont nécessaires, l'interprétation des images vidéo et la sensibilité des capteurs ^[5].

1.4.2 En Asie :

En Asie, une quinzaine de maisons intelligentes sont en cours de développement. L'objectif majeur, pour la plupart, est de maintenir des personnes âgées à domicile en leur créant un environnement adapté et confortable tout en exploitant des technologies d'assistance.

- La maison intelligente d'Osaka, issue du projet de Matsuoka, détecte automatiquement les événements inhabituels causés par certaines pathologies ou accidents. Elle est équipée de 167 capteurs et de 17 appareils électroménagers instrumentés (réfrigérateur, télévision, cuisinier, climatiseur, etc.). Les différentes activités de la personne sont déterminées en fonction des spécificités des capteurs et des informations telles les activités de lever, coucher, préparation des repas, toilette, douche et de bureau, etc. Les résultats de l'étude montrent que pour une famille de quatre personnes observées durant une année, le système a détecté 73 situations anormales dont 19 étaient réelles ^[6].
- En Corée, le projet « Intelligent Sweet Home » propose une maison intelligente dédiée aux personnes âgées et handicapées incluant notamment un lit robotisé intelligent équipé d'une main intelligente, un fauteuil roulant motorisé ainsi qu'un élévateur intelligent permettant de déplacer la personne entre le lit et le fauteuil roulant. Le but est d'assister la personne dans sa mobilité : aller au lit, s'asseoir dans le fauteuil, tendre un livre ou un journal, le

remettre dans l'étagère, etc. Des interfaces homme machine spécifiques ont été conçues pour contrôler automatiquement l'ensemble des dispositifs électromécaniques [7].

- Deux universités Coréennes ont développé une maison intelligente capable de détecter les différentes modes de vie du résidant dans le but d'anticiper les besoins du résidant et offrir un service domotique approprié. Le système exploite des détecteurs pyroélectrique pour localiser la personne et une expérimentation a été menée dans une chambre mesurant 4 x 4 x 2,5 mètres instrumentée par 12 capteurs placés dans les plafonds. Une évaluation des performances à travers une expérimentation a été faite et a conclu au développement d'un algorithme de suivi-multiples capable de reconnaître les trajectoires des résidents, lorsqu'ils se trouvent dans la même pièce [8].

I.4.3 En Europe :

En Europe, on trouve également un grand nombre de projets de maisons intelligentes dédiées d'une part à la gestion du confort et d'autre part, au maintien à domicile des personnes âgées :

- La maison intelligente de « British Telecom » et « Anchor Trust » en Angleterre, surveille à distance l'activité de la personne. Elle comporte des capteurs infrarouges et des capteurs de contacts magnétiques sur les portes d'entrée ainsi que sur la porte du réfrigérateur. Un capteur de température est également placé dans l'environnement de vie pour surveiller la température ambiante. Basé sur l'étude du style de vie, le système déclenche une alarme lorsqu'une situation anormale est détectée [9].
- Le projet européen « Handicom », issu de l'initiative AAL-119 « Ambient Assistive Living » du traité européen, élabore une infrastructure ouverte de services dont l'objectif est d'améliorer la qualité de vie des personnes âgées et handicapées dans le monde. Considérant la clé de la réussite par l'intégration des besoins spécifiques de chaque personne, le principe est de concevoir l'ensemble des services appropriés, quel que soit l'environnement et le contexte dans lequel se trouve la personne : ce qui inclus non seulement l'environnement de vie « Indoor » et l'environnement extérieur « Outdoor ».
- Le projet « Global Village Initiative » s'appuie ainsi sur les capacités des technologies intelligentes au travers d'un programme de recherche international faisant collaborer, la recherche, les industries et les acteurs sociaux. Ce projet de grande envergure a pour but d'aboutir, sur la base des innovations de la recherche, à un déploiement et une offre commerciale soutenue par une action internationale [10].
- Le projet « PROSAFE » à Toulouse a été développé pour la surveillance continue des personnes âgées à domicile. L'objectif est d'assurer le maintien des personnes âgées

autonomes dans leur logement et d'envoyer automatiquement, sans intervention humaine, un signal d'alarme en cas d'urgence ^[11]. Le système est composé de capteurs de mouvements (infrarouge passif), interfacés à une unité de calcul. Plusieurs chambres en Institution et à domicile ont été instrumentées. Une unité de calcul placée à l'extérieur de la chambre ou de l'appartement permet d'évaluer la mobilité et plusieurs critères d'activités de la personne. Une première étude consistait à établir des modèles d'habitudes de présence et d'absence des personnes en utilisant des techniques d'apprentissage basées sur les réseaux de neurones et les statistiques. Des travaux de validation ont été menés sur plusieurs mois sur sites habités pour corréliser les diagnostics automatisés et les comportements du patient. La surveillance s'effectue essentiellement de nuit et mettait en évidence les habitudes de : lever, coucher, sortie, douche, etc. Les résultats statistiques traitaient la mobilité, l'immobilité et l'agitation au lit. Une étude dédiée à la classification des trajectoires de position a été menée pour pouvoir identifier les déambulations habituelles et inhabituelles ^[12].

- Le système « ERGDOM » est un gestionnaire auto-adaptatif pour la régulation du confort d'hiver dans l'habitat et la réduction de la consommation énergétique. Un réseau de capteurs de présence infrarouges distribués permet au système de connaître les habitudes de présence de l'occupant dans chaque pièce. Des bornes de commandes du système sont placées dans des endroits stratégiques de l'appartement permettant notamment à l'occupant d'augmenter ou diminuer la puissance des convecteurs. Le concept ERGDOM se base sur l'exploitation des techniques d'apprentissage pour prendre en considération les habitudes de présence des occupants, leurs préférences en matière de confort et la réaction thermique de l'habitat pour prédire l'ambiance et anticiper la présence des personnes à leur domicile ^[13].

I.5 Le Système de Domotique Proposé :

Comme on a vu dans les sections précédentes, il y'a plusieurs normes d'interconnexion, avec une multitude logicielle pour contrôler sa maison, cependant il y a des éléments qui persistent dans toute application domotique, notamment dans la conception d'une maison intelligente. Ces éléments sont des objets communicants : des capteurs de température, des interrupteurs, détecteurs de présence PIR...etc tous combinés avec un élément central, qui est le cœur du système : organe responsable de la décision et la commande des autres objets. On propose un système dont la communication entre les différents nœuds est une liaison sans-fil sécurisée.

L'élément central est conçu autour d'une carte Raspberry Pi équipée avec un module RF et un module GSM pour permettre les alertes à distances en temps réelles. La carte Raspberry Pi doit être reliée à un switch Ethernet ou un hub Wi-Fi, pour garantir l'accès au serveur web de control que nous avons développé et embarqué dans cette dernière. Cette approche permet l'accès à l'organe central en utilisant des smartphones ou des terminaux via des browsers internet. En plus, on a intégré des fonctionnalités qui nécessitent la connectivité Internet. Notre proposition consiste à concevoir quatre types d'objets communicants : le nœud des capteurs (température, humidité, détection de Gaz et fumé), le contrôleur HVAC, les détecteurs de mouvement PIR et les interrupteurs (interrupteur générique, pour allumer ou éteindre n'importe quelle appareils électriques). Ces objets, sont construits au-dessus de la plateforme Arduino. Les trois premiers sont des cartes Arduino UNO dotées des Shields qu'on a proposés et qui seront détaillés dans le chapitre suivant. Le dernier type d'objet communiquant créé, est une solution pour un interrupteur sans-fil compatible avec les outils de développement software Arduino, les détails de son fonctionnement seront présentés dans le chapitre 3.

La solution qu'on va présenter est schématisée dans la page suivante par la Fig-3.

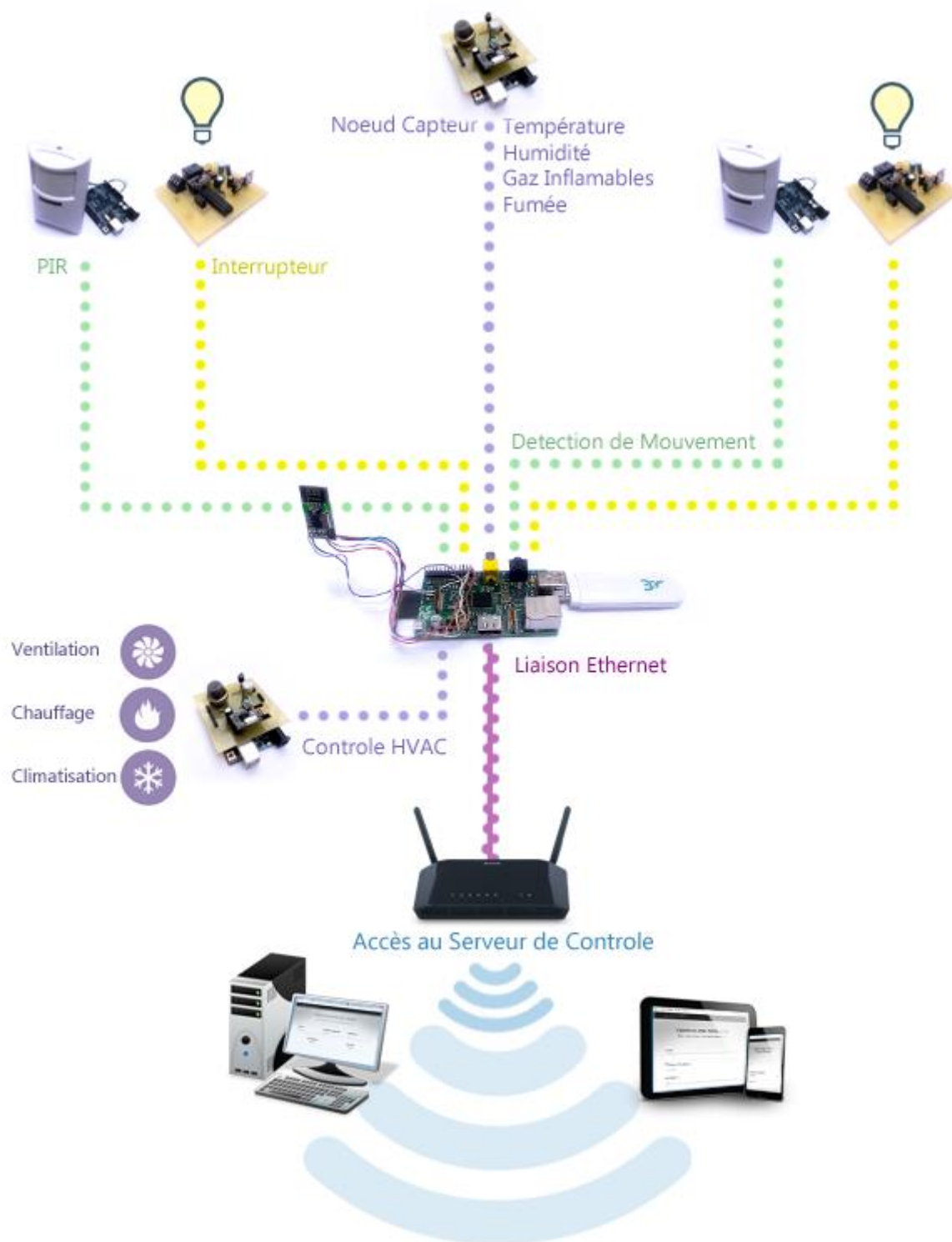


Figure 3. Schéma Synoptique Globale de la Maison Communicante Proposée.

I.6 Conclusion

Dans cette partie on a présenté une vue globale de la technologie de la domotique automatisée et intelligente. On a terminé l'étude par une proposition globale d'une maison communicante à réaliser. Dans les chapitres qui vont suivre on va détailler petite à petit la conception, réalisation et intégration des différentes composantes de notre maison intelligente.

Chapitre II : Conception des Shields Arduino pour Objets Communicants

II.1 Présentation d'Arduino :

Arduino est une technologie qui fait associer un environnement de développement avec un circuit électronique à base d'un microcontrôleur AVR, distribué sous la licence du matériel libre (les schémas électrique sont disponible gratuitement). Arduino était l'origine destiné principalement à la programmation multimédia interactive

en vue de spectacle ou d'animations artistiques. Mais cela n'était pas exclusif. La carte électronique peut être programmée pour analyser et produire des signaux électriques, de manière à effectuer des tâches très diverses comme la domotique (le contrôle des appareils domestiques, éclairage, chauffage...), le pilotage d'un robot, etc. L'interface de programmation est inspirée du traitement environnementale Processing. Ce dernier est à son tour inspiré de l'environnement de programmation Wiring^[14].

Arduino peut être utilisé pour construire des objets interactifs indépendants (prototypage rapide), ou bien pour la connexion à un ordinateur pour communiquer avec des logiciels (Macromedia Flash, traitement de données...).



II.1.1 Le Module Arduino:

Un module Arduino est généralement construit autour d'un microcontrôleur Atmel AVR (ATmega328 ou ATmega2560 pour les versions récentes, ATmega168 ou ATmega8 pour les plus anciennes), et de composants complémentaires qui facilitent la programmation et l'interfaçage avec d'autres circuits.

Chaque module possède au moins un régulateur linéaire 5 V et un oscillateur à quartz 16 MHz (ou un résonateur céramique dans certains modèles). Le microcontrôleur est préprogrammé avec un boot loader de façon à ce qu'un programmeur dédié ne soit pas nécessaire.

Les modules sont programmés au travers d'une connexion série RS-232, mais les connexions permettant cette programmation diffèrent selon les modèles. Les premiers Arduino possédaient un port série, puis l'USB est apparu sur les modèles récents, tandis que certains modules destinés à une utilisation portable se sont affranchis de l'interface de programmation, relocalisée sur un module USB-série dédié (sous forme de carte ou de câble).

L'Arduino utilise la plupart des entrées/sorties du microcontrôleur pour l'interfaçage avec les autres circuits. Le modèle Diecimila par exemple, possède 14 entrées/sorties numériques, dont 6 peuvent produire des signaux PWM, et 6 entrées analogiques. Les connexions sont établies au travers de

connecteurs femelle HE14 situés sur le dessus de la carte, les modules d'extension venant s'empiler sur l'Arduino (on les appelle des Shields Arduino). Plusieurs sortes d'extensions ou Shields sont disponibles dans le commerce. Les modules d'origine des différentes versions de l'Arduino sont fabriqués par la société italienne Smart Projects. Quelques-unes des cartes de marque Arduino ont été conçues par la société américaine SparkFun Electronics. Dix-sept versions des cartes de type Arduino ont été produites et vendues dans le commerce à ce jour, et les plus utilisés sont la carte Arduino UNO R3 et la carte Arduino Méga.

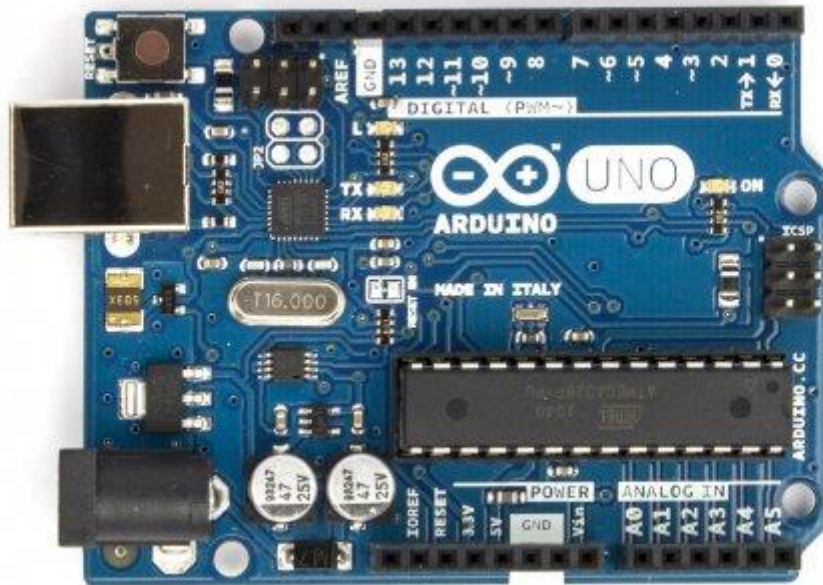
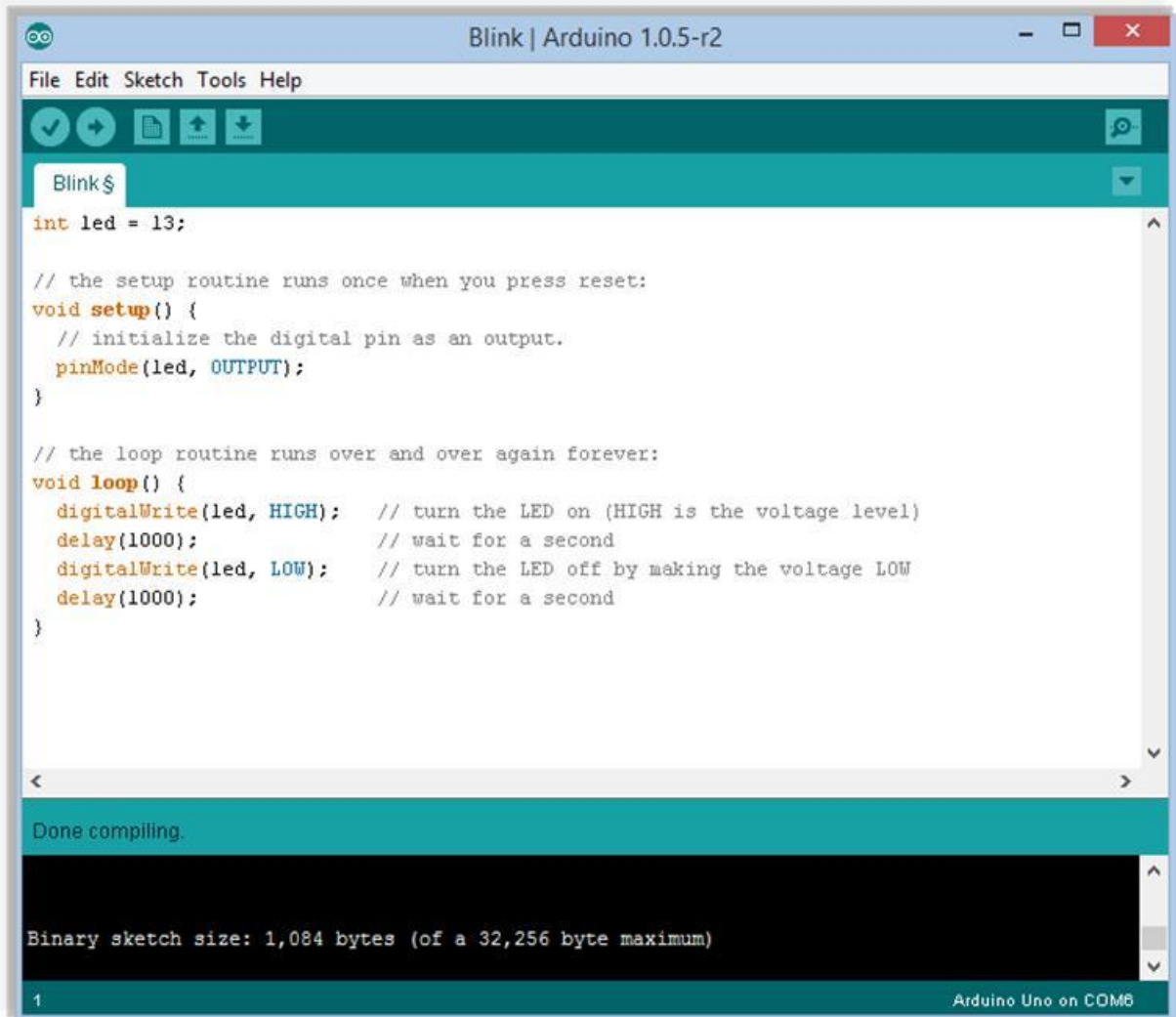


Figure 4. Une Carte Arduino UNO R3

II.1.2 Le l'environnement de développement Arduino:

Le logiciel de programmation des modules Arduino est une application Java, libre et multiplateforme, servant d'éditeur de code et de compilateur, qui peut transférer le programme dit sketch à travers une liaison série (RS-232, Bluetooth ou USB selon le module). Il est également possible de se passer de l'interface Arduino, et de compiler et uploader les programmes via l'interface en ligne de commande. Le langage de programmation utilisé est le C++, compilé avec avr-g++ 3, et lié à la bibliothèque de développement Arduino, permettant l'utilisation de la carte et de ses entrées/sorties. La mise en place de ce langage standard enrichie le développement de programmes sur les plateformes Arduino, et le rend plus intéressant.



```
Blink | Arduino 1.0.5-r2
File Edit Sketch Tools Help
Blink $
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led, LOW);  // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}

Done compiling.

Binary sketch size: 1,084 bytes (of a 32,256 byte maximum)

1 Arduino Uno on COM6
```

Figure 5. Le logiciel Arduino avec l'exemple Blink

II.1.3 Le Module Radio NRF24L01 :

Les moyens de faire communiquer sans fil deux microcontrôleurs ne manquent pas. Si on prend le cas d'Arduino, les plus populaires sont peut-être les modules XBee. L'inconvénient majeur de ces modules est leur coût (environ 4000DA). Pour des projets Wireless, surtout s'ils demandent plusieurs liaisons, ce coût peut-être trop élevé. Une alternative intéressante est fournie par les puces nRF24L01+ de Nordic Semiconductors. On les trouve montées sur des circuits, prêtes à l'emploi pour environ 300DA.

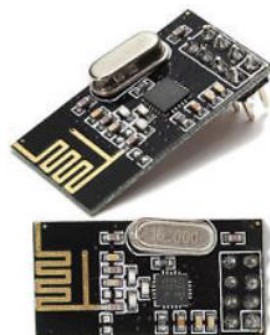


Figure 6. Le module NRF24L01

Le NRF24L01 est un émetteur-récepteur half-duplex, c'est à dire que la puce peut soit émettre soit recevoir, qui opère dans la bande des 2.4GHz (comme le Wi-Fi et le Bluetooth). La communication entre la puce et un microcontrôleur est assurée par un bus SPI (supporté par les cartes Arduino). La vitesse de transmission des données est configurable et peut atteindre 2Mbps. La portée maximale est d'environ 100m pour un débit de 250kbps. Cette puce utilise un protocole propriétaire appelé Enhanced ShockBurst. La puce peut travailler sur 126 canaux de communication. En mode récepteur elle peut écouter sur 6 canaux simultanément. Dans ce cas, chaque canal est associé à une adresse. En cas de réception d'un paquet pour cette adresse, la puce gère le protocole en envoyant un ACK en retour puis en désassemblant le paquet pour ne garder que son contenu utile (les données transmises) dans une file d'attente (FIFO) spécifique pour chaque canal d'écoute. Cette dernière peut être lue par le microcontrôleur associé à l'aide de la liaison série SPI. La taille des données transmises ne peut pas dépasser les 32 octets par paquet.

Préambule 1 Octets	Adresse 2-5 Octets	La charge 1-32 Octets	CRC 1-2 Octets
-----------------------	-----------------------	-----------------------	-------------------

Figure 7. Format d'un paquet ShockBurst

II.2 Shields Arduino proposés :

Les Shields sont des "extensions" pour rajouter des fonctions à une carte Arduino, sous forme des cartes conçues pour se connecter directement sur ce dernier. Le plus connu de toute est le Shield Ethernet qui permet d'ajouter un port Ethernet à une carte Arduino. L'avantage du Shield est de minimiser les soudures et les fils, par une simple installation. Pour notre solution, on propose trois cartes Shields Arduino. Ces Shields ont pour but de transformer n'importe quelle carte Arduino en un objet communicant par le biais des ondes électromagnétiques, en utilisant le module NRF24L01. Des fonctionnalités spécifiques vont être implémentées pour chaque type de nœud. Les trois Shields proposés sont : le Shield des capteurs (température, humidité...), le Shield de contrôle HVAC, et le Shield de détection de mouvements.

II.2.1 Le Shield de Capteurs :

Le Sensor Node ou le nœud capteur est un terme qui désigne un objet communicant ou plus spécifiquement une carte électronique qui collecte les informations sur l'état physique de l'environnement et les transmet. Il contient quatre unités de base : l'unité de capture, l'unité de traitement, l'unité de transmission, et l'unité de contrôle d'énergie. Selon le domaine d'application, il peut aussi contenir des modules supplémentaires tels qu'un système de localisation (GPS), ou bien un système générateur d'énergie (cellule solaire).

Le capteur est le dispositif transformant l'état d'une grandeur physique observée en une grandeur utilisable, telle qu'une tension électrique, une hauteur de mercure, une intensité ou la déviation d'une aiguille. Il est généralement composé de deux sous-unités : le récepteur (reconnaissant l'analyte) et le transducteur (convertissant le signal du récepteur en signal électrique). Le capteur fournit des signaux analogiques, basés sur le phénomène observé, au convertisseur Analogique/Numérique. Ce dernier transforme ces signaux en un signal numérique compréhensible par l'unité de traitement.

Pour notre cas, la carte Arduino comprend tous ces unités de base sauf l'unité de capture et l'unité de transmission, elles sont représentées par les capteurs et le module NRF24L01 respectivement, ces deux derniers forment l'extension qu'on appelle le « Shield de Capteurs », et qui s'empile sur les cartes Arduino UNO.

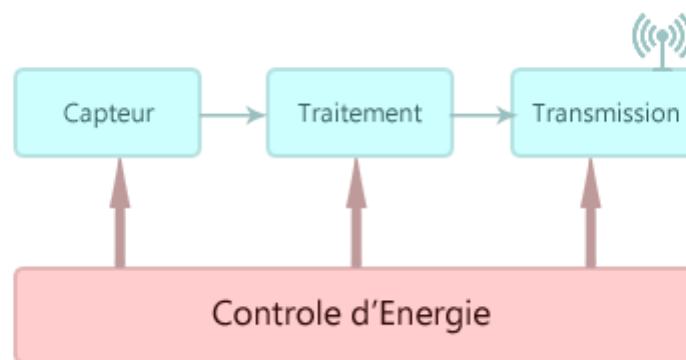


Figure 8. Schéma Block du Nœud Capteur

II.2.1.1 Choix de Capteurs :

Un système domotique vise à assurer des fonctions de sécurité et de confort dans la maison, pour cela il doit être capable au minimum, d'acquérir des lectures sur la température et de détecter s'il y a un danger d'incendie par la détection de fumée ou la présence d'un Gaz inflammable. Donc on a besoin de deux types de capteurs pour notre Shield, un capteur de température et un autre pour la détection des gaz inflammables et de fumées. On peut rajouter un capteur d'humidité pour que le système devienne capable d'exercer un contrôle climatique à l'intérieur de la maison.

II.2.1.2 Capteur de Température :

Il est possible de mesurer la température de plusieurs façons différentes qui se distinguent par le coût des équipements et la précision ainsi que le temps de réponse. Pour des raisons de coût et principalement la disponibilité, on a choisit la sonde LM335. Il s'agit d'un capteur de température précis, facilement calibré, et très simple d'usage, qui délivre une tension évoluant linéairement avec la température. La tension de sortie est la température en degrés Kelvin divisé par 10 (la température en degrés Kelvin est égale à la température en degrés Celsius à laquelle on ajoute la valeur 273). Il fonctionne comme une diode Zener dont la tension de claquage est directement

La concentration de détection gamme de 300 ppm à 10000 ppm est appropriée pour la détection des fuites de gaz. Le capteur peut fonctionner à des températures allant de -10 à 50 °C et consomme moins de 150 mA à 5 V. Il est utilisé pour les équipements des marchés de grandes consommations et industriel. Ce capteur est conçu pour détecter le LPG, i-butane, propane, méthane, alcool, hydrogène et la fumée. Il a une grande sensibilité et un temps de réponse rapide. Sa sensibilité peut d'ailleurs être ajustée par potentiomètre.

II.2.1.4 Capteur d'Humidité :

Il existe actuellement un grand nombre de capteurs d'humidité, reposant sur des principes très complexe (réservé à des applications en laboratoire) comme l'hygromètre à isotopes radioactifs, au plus simple comme l'hygromètre résistif. Ce dernier, fonctionne comme une résistance variable qui change sa valeur non-linéairement selon la valeur d'humidité relative de l'environnement, mais ce type de capteur est un peu moins précis que le capteur capacitif, qui presque pour le même prix, offre une précision excellente. C'est pour cette raison qu'on a choisis le capteur d'humidité capacitif SH1101. Il fonctionne comme un condensateur à plaques, l'électrode inférieure est déposée sur un substrat porteur, souvent un matériau en céramique, une fine couche hygroscopique en polymère agit comme le diélectrique, au-dessus duquel se trouve la plaque supérieure, qui agit comme la seconde électrode mais qui permet aussi le passage de la vapeur d'eau dans le polymère. Les molécules de vapeur d'eau entrent ou sortent dans le polymère hygroscopique jusqu'à ce que la vapeur d'eau soit en équilibre avec l'air ou le gaz ambiant. La rigidité diélectrique du polymère est proportionnelle à la teneur en vapeur d'eau. De même, la rigidité diélectrique affecte la capacité, qui est mesurée et traitée par la carte Arduino pour donner une mesure d'humidité relative ^[16].



Figure 13. Capteur SH1101

II.2.1.5 La Transmission-Réception Radio Fréquence :

L'élément commun à tous les Shields de ce système domotique est le module radio NRF24L01. L'implémentation de ce module dans un circuit avec un microcontrôleur qui supporte la communication SPI est simple. Il faut juste identifier les pins du bus SPI de la carte Arduino utilisé, et les attacher aux pins SPI respectives du module RF. On peut consulter le tableau suivant pour les différentes cartes Arduino :

MODULE PIN	DESCRIPTION	UNO	MEGA
1	GND	GND	GND
2	VCC	3.3V (See note)	3.3V (See note)
3	CE	DIO 8	DIO 8
4	CSN	DIO 7	DIO 7
5	SCK	DIO 13	DIO 52
6	MOSI	DIO 11	DIO 51
7	MISO	DIO 12	DIO 50
8	IRQ		

On remarque que le NRF24L01 marche avec une alimentation VCC de 3.3V, donc il faut le brancher avec le pin 3.3V de la carte Arduino. Le reste des pins tolèrent et marchent avec des tensions de 5V. Avec cette considération bien entendue, on n'aura pas une possibilité de griller le module RF. Mais dans nos essais pour tester une ligne de communication entre deux Arduino équipés des modules radios de même type avec une simple installation, c'est-à-dire que on a utilisé juste les modules directement avec les cartes Arduino sans ajouter d'autre composants électroniques, le résultat est aucun paquet RF n'a été transmis. Pour cela on a ajouté un condensateur de découplage (1uF-100uF) entre les bornes d'alimentation du module RF, ce condensateur sert à filtrer les harmoniques et le bruit du pin à 3.3V, et fournis une puissance électrique quand il y'a un pique de courant exercer par le module RF. Donc la meilleure implémentation obtenue pour le module NRF24L01 est comme celle-ci :

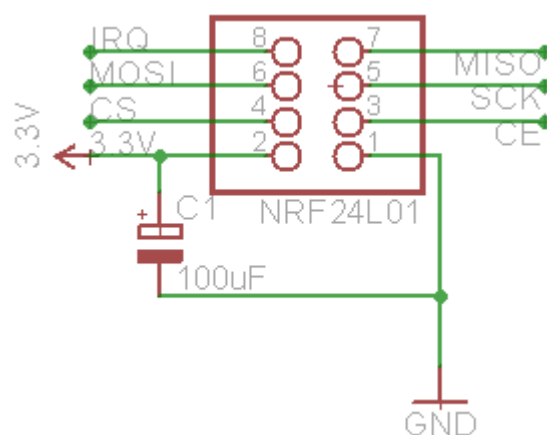


Figure 14.Schéma NRF24L01 dans les Shields

Après avoir validé le schéma électrique, on va basculer vers l'interface du logiciel CAD utilisée pour la conception du PCB. L'image générée pour le circuit imprimé de ce Shield est illustrée dans les figures suivantes :

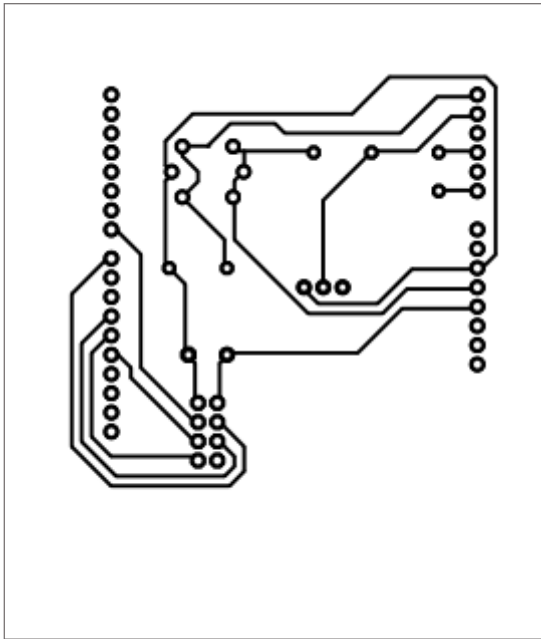


Figure 17. PCB Shield de Capteurs (échelle 1/1)

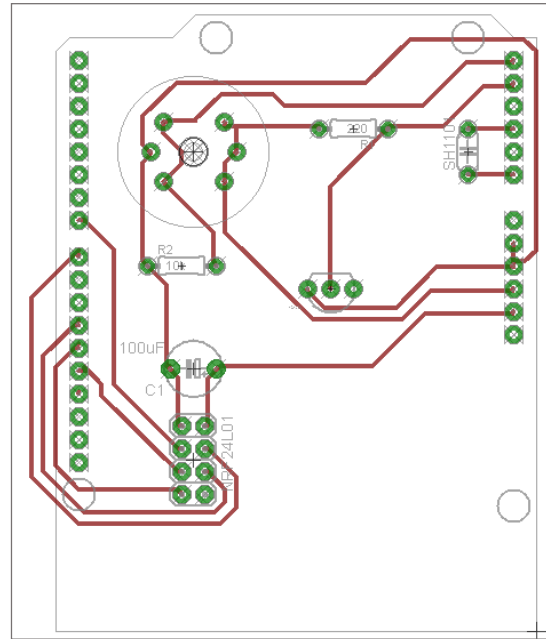


Figure 16. Aide Schéma

Les images suivantes montrent le circuit final après la réalisation de circuit imprimé et la soudure des composants :

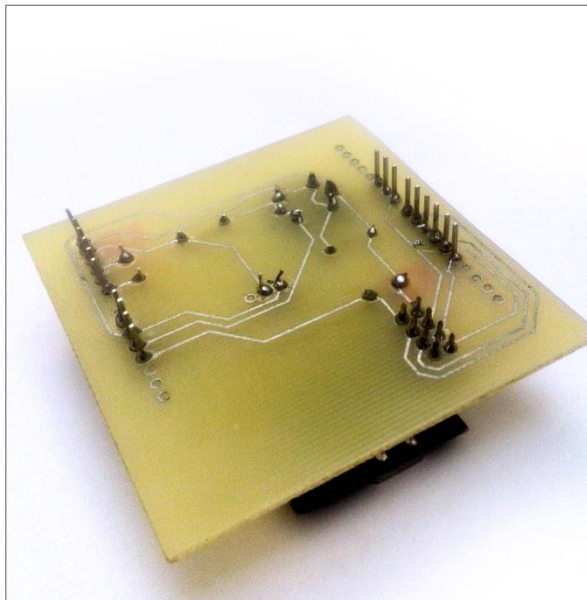


Figure 19. Le Shield de Capteurs

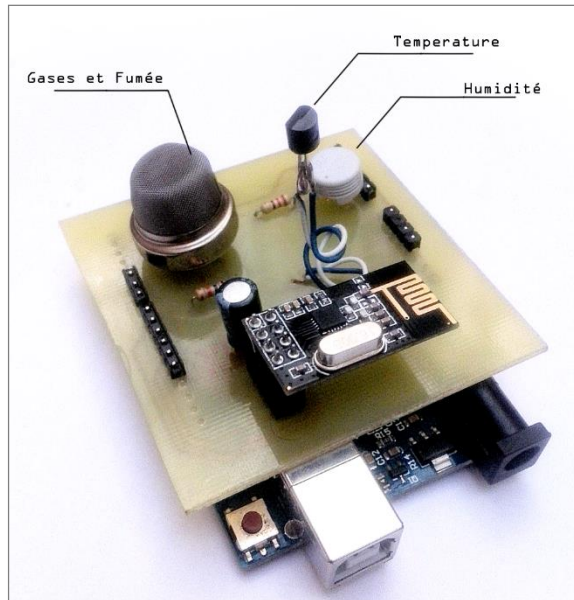


Figure 18. Le Shield empilé sur l'Arduino UNO

II.2.1.7 Algorithme Proposé :

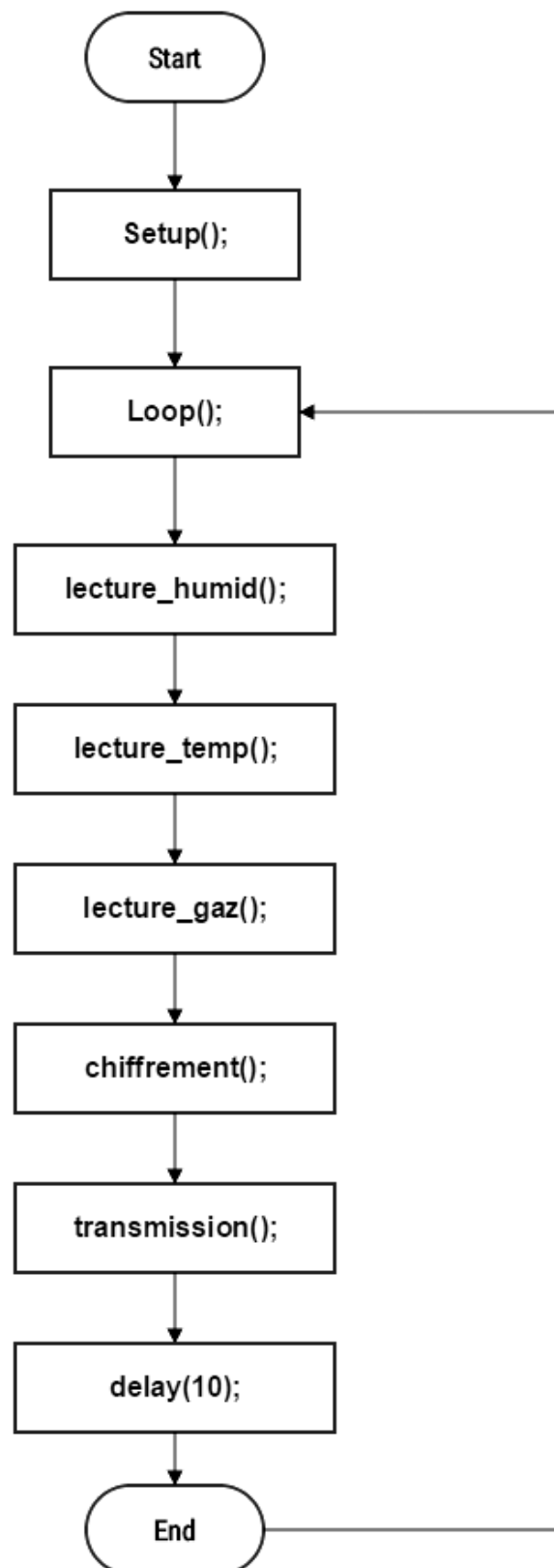


Figure 20. Algorithme Nœud Capteur

II.2.2 Le Shield de Contrôle HVAC :

Le HVAC (en anglais : Heating, Ventilation and Air Conditioning) est l'ensemble des techniques regroupant le contrôle aéraulique pour assurer un confort maximum. Ce qualificatif s'applique à tous types de bâtiments habitat, tertiaire, industriel. Pour notre maison intelligente le nœud de capteurs collecte et transmet les informations sur la température et l'humidité vers l'élément centrale, cet organe lui-même va calculer la décision pour allumer ou éteindre les chauffages et les climatiseurs, selon le processus de contrôle programmé et les préférences des occupants de la maison. Une fois la commande programmée, elle sera transmise par la liaison radio (en utilisant le NRF24L01), vers une carte Arduino UNO dotée d'un Shield opérant à l'image d'un thermostat d'ambiance, cet ensemble gère l'alimentation des climatiseurs et chauffages. On va introduire dans les paragraphes qui suivent la conception du Shield de contrôle HVAC qui permet aux cartes Arduino UNO de recevoir les commandes et de contrôler les éléments HVAC de la maison. Ce contrôleur HVAC va jouer le rôle d'un thermostat connecté, souvent appelé dans le marché de domotique le thermostat intelligent. Avec une bonne implémentation d'algorithmes de contrôle de température et d'humidité, ce Shield permet de faire une gestion climatique à l'intérieur de la maison, en assurant un confort thermique.

II.2.2.1 Compatibilité avec les systèmes HVAC :

Il y'a plusieurs types de systèmes HVAC dans la maison moderne, en Europe on trouve principalement la chaudière à gaz ou la chaudière mixte (elle passe automatiquement à l'énergie de secours lorsque la première est consommée) avec des climatiseurs individuels (Split System) pour chaque pièce ou chaque zone, ces installations sont notamment plus implémentées dans la Royaume Unie, la France, et l'Irlande. Dans le reste du monde principalement en Asie et en Amérique, les plus populaires sont basés sur des systèmes qu'on appelle « Conventional HVAC System », qui fait les deux fonctions de climatisation et chauffage avec un seul système, leur fonctionnement est montré dans la figure ci-dessous:

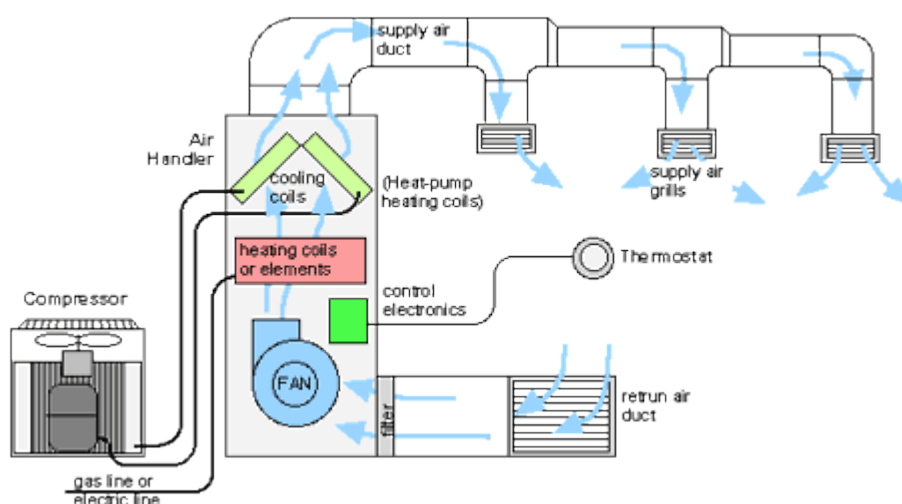


Figure 21. Système HVAC Typique (Amérique du Nord et en Asie)

On note l'objet dénommé Thermostat au centre de la figure précédente, c'est le moyen qui contrôle le mode d'opération de système HVAC. Il gère l'activation de ventilation (pour la distribution d'air chaud ou froid), le compresseur (pour refroidissement d'air), et le chauffage. Il y'a des standards industriels pour la fabrication et la production de ces systèmes, mais jusqu'aujourd'hui les régulations qui normalisent le câblage et le raccordement de système avec le thermostat n'existent pas^[17].

II.2.2.2 Thermostat a 4 fils :

Il existe des pays européens comme la France qui exige leurs propres standards sur les marchés locaux, mais le câblage le plus répandu dans le monde (sauf une minorité de pays européens) est le branchement à 4 fils. Dans ce type de raccordement, le thermostat est lié à 4 fils, ou plus pour une pompe de chaleur, ces fils sortent d'une boîte de contrôle HVAC ou s'installe un thermostat. Un fil pour chaque élément HVAC (ventilateur, climatisation, chauffage) et un fil pour une alimentation de 24VAC (220VAC/120VAC dans quelque systèmes). Pour faire déclencher un mode d'opération, le thermostat fait juste court-circuiter le fil d'alimentation avec les autres, dont la combinaison varie selon le mode d'opération. Si on souhaite l'air froid, le thermostat déclenche la ventilation avec le compresseur en même temps, la même chose pour l'air chaud sauf qu'il déclenche le chauffage à la place du compresseur^[18].

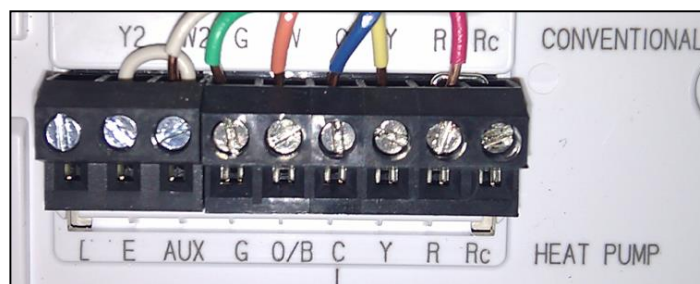


Figure 22. Un exemple de Sortie des câbles HVAC

Chaque fil est identifié par une lettre ou une couleur. Le rouge est la phase avec 24VAC, le blanc W est pour le chauffage ou la chaudière, le jaune Y pour le compresseur, et le vert G pour la ventilation. Le reste c'est pour refroidissement/chauffages auxiliaires ou secondaire (des fois utilisé dans les systèmes HVAC à deux étapes, c'est-à-dire le préchauffage ou pré-climatiseur suivis par la climatisation ou chauffage principale), ils peuvent être aussi utilisés pour une pompe à chaleurs. Mais la majorité des maisons ont des systèmes HVAC conventionnel qui presque toujours fournissent les quatre câbles mentionnés ci-dessus^[19].

II.2.2.3 Conception de Shield HVAC :

Pour un système HVAC avec quatre fils on a besoin d'un Shield qui peut remplacer le thermostat et ces interrupteurs dans le diagramme de raccordement suivant :

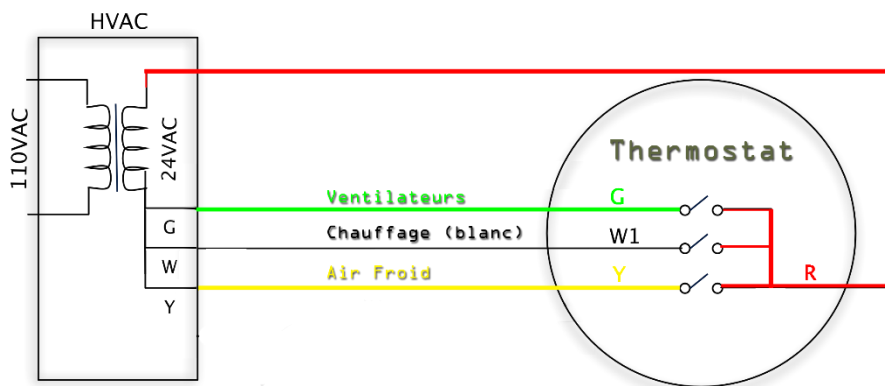


Figure 23. Diagramme de Raccordement

Donc on va implémenter trois relais contrôlé par une carte Arduino UNO. Pour une compatibilité avec les systèmes haute tensions comme le 220VAC ou le 120VAC, ces relais doivent accepter un ampérage jusqu'à 10A et une tension de 220V, en maximisant la compatibilité avec les systèmes HVAC courants. Comme le Shield des Capteurs, ce contrôleur nécessite l'implémentation d'un module NRF24L01, pour recevoir les commandes en provenance du nœud central. Notre circuit de commande relais contient un transistor bipolaire opérant en commutation, et un relais avec une diode de protection flyback. Pour commuter le transistor il faut juste appliquer une tension sur sa base, c'est le rôle de sortie numérique de la carte Arduino.

Le code Arduino fait allumer ou éteindre un relais simplement en basculant l'état (HIGH ou LOW) de la pin numérique lié au transistor. Le circuit final est démontré dans la figure suivante :

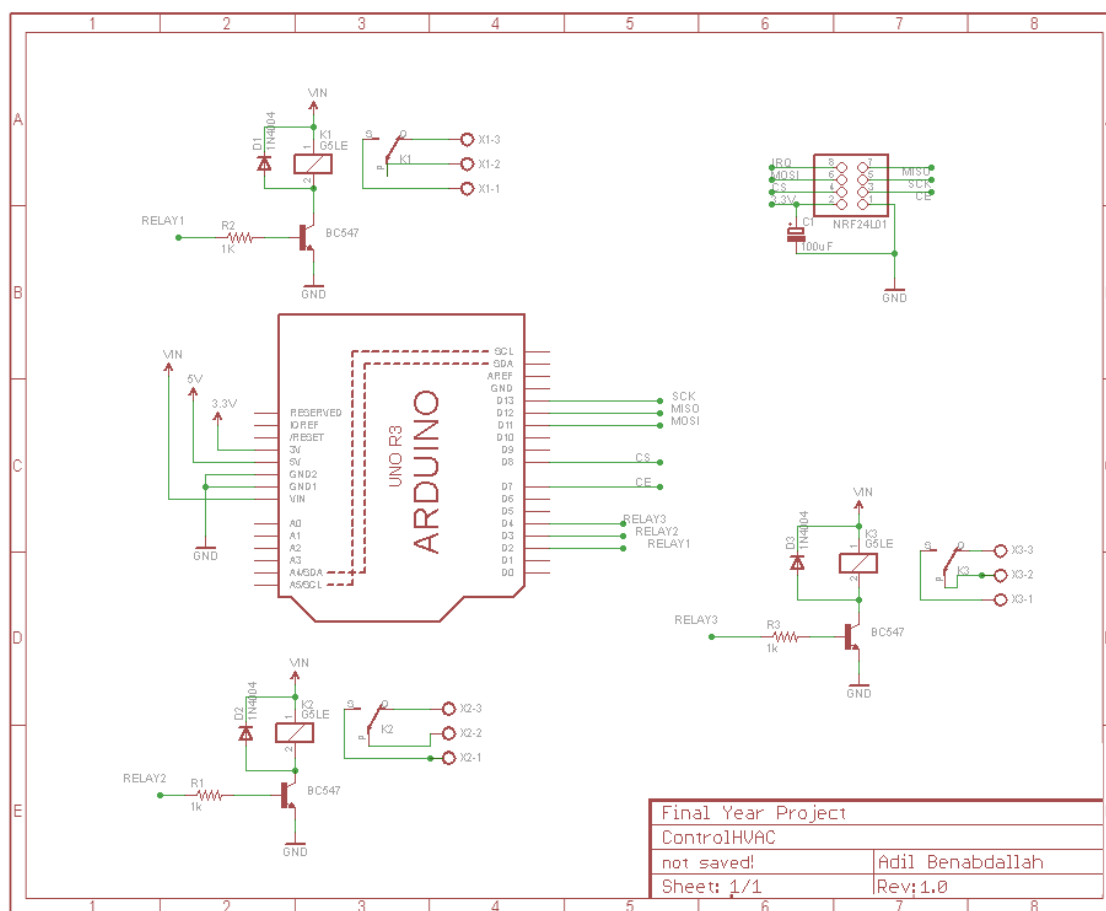


Figure 24. Schéma Shield de Controle HVAC

Après la conception et le schéma on a produit le circuit imprimé final du Shield de Contrôle HVAC :

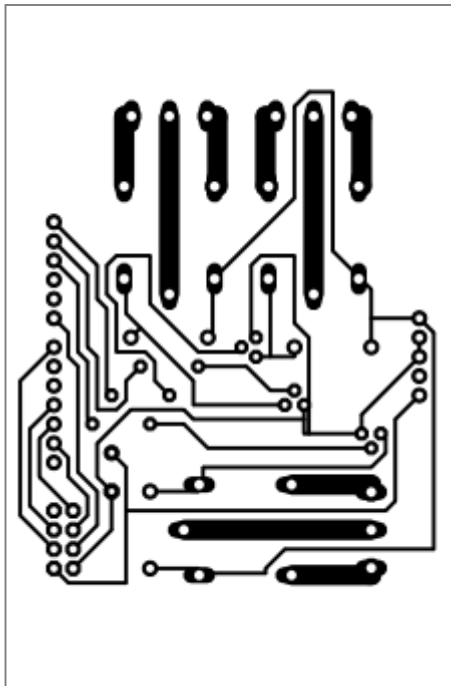


Figure 25. PCB de Shield (échelle 1/1)

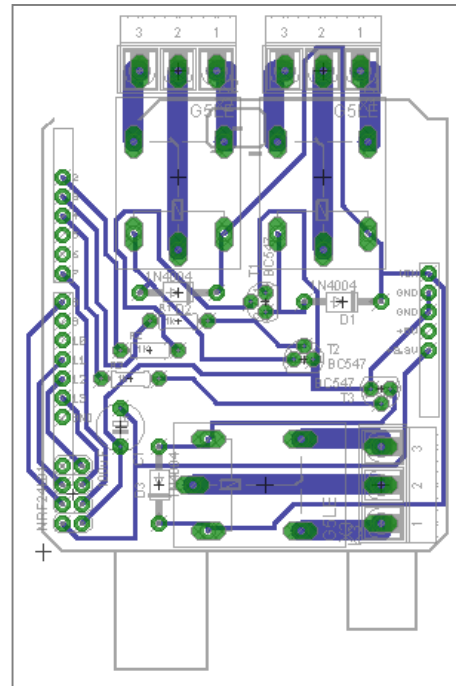


Figure 26. Aide Schéma

II.2.2.6 Algorithme Proposé :

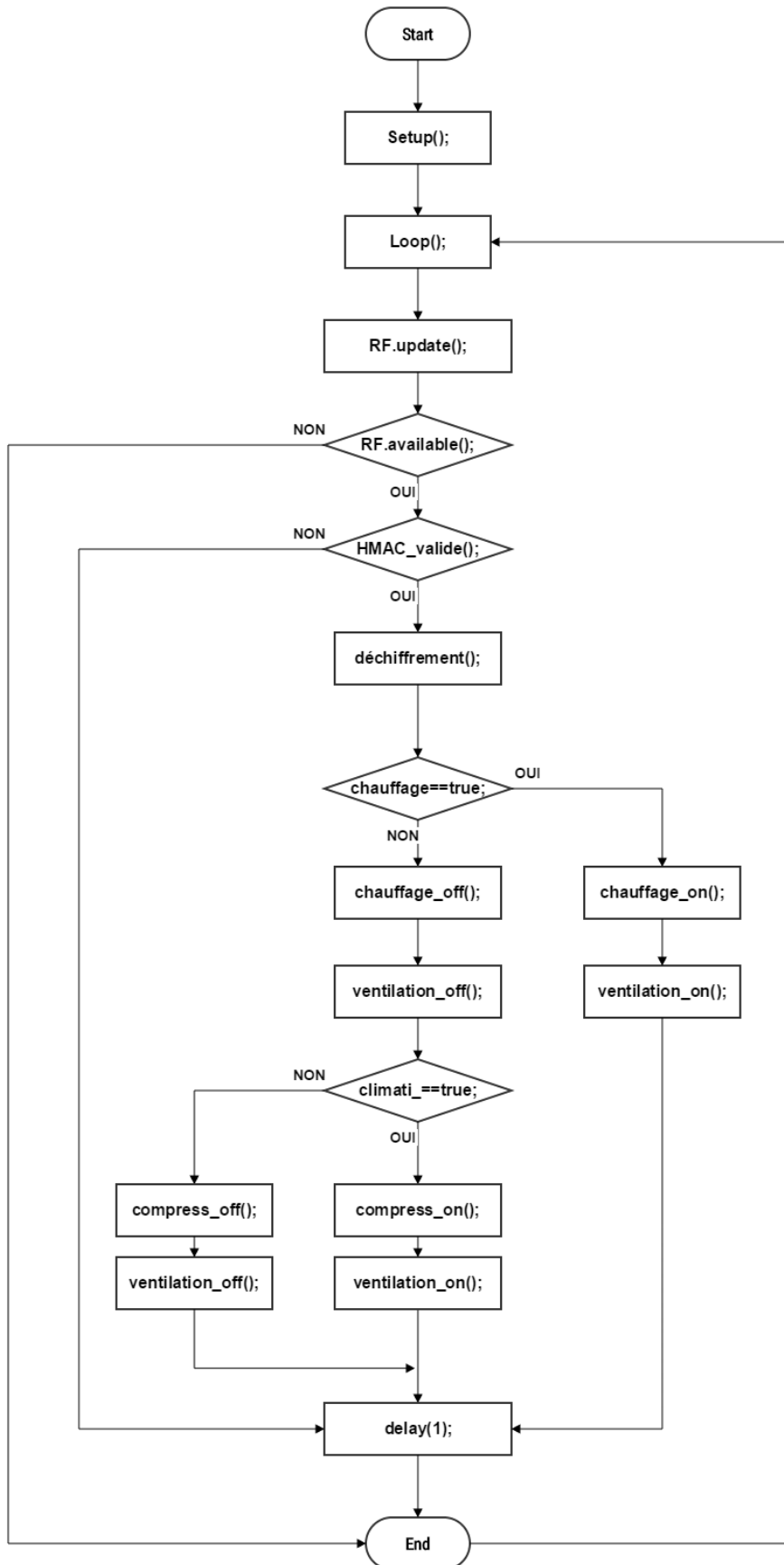


Figure 27. Algorithme du Controlleur HVAC

II.5.3 Shield de Détection de Mouvement :

Pour notre maison intelligente on va faire une implémentation de lumière automatique, pour cela le système doit détecter la présence des personnes dans les chambres, cette fonctionnalité est atteinte par l'installation de capteurs de mouvement dans chaque pièce, ces capteurs sont classés en deux types majeurs :

II.5.3.1 Les Détecteurs PIRs :

L'infrarouge ou rayonnement infrarouge est également désigné par rayonnement thermique et fait partie des radiations électromagnétiques. Tout objet et donc également le corps humain émet un rayonnement thermique dont la longueur d'onde est en fonction de sa température. Ce rayonnement infrarouge n'est néanmoins pas perceptible par l'œil humain. Le rayonnement thermique de l'homme se situe dans la bande infrarouge et il est détecté par des détecteurs passifs d'infrarouge. Ces détecteurs n'émettent eux-mêmes aucune radiation. Ils sont seulement récepteurs et c'est pourquoi on les qualifie de « détecteurs passifs d'infrarouge ». Ils possèdent des capteurs qui réagissent au rayonnement thermique. Si le rayonnement thermique est constant, ils ne délivrent pas de signal. Pour pouvoir réagir aux mouvements, les détecteurs utilisent un système optique composé de lentilles et de miroirs qui divisent la zone à surveiller en zones passives et actives à la manière d'un jeu d'échec, donc ils ne réagissent qu'aux variations du rayonnement thermique. Un mouvement de la main suffit pour détecter la présence d'une personne.

II.5.3.2 Détecteurs à Micro-Onde :

Ce sont des capteurs qui exploitent l'effet Doppler. Comme un radar, ce type de capteur émet des ondes électromagnétiques dans le spectre des micro-ondes vers un angle solide précis et attend la réponse pour avoir s'il y'a un retard ou décalage entre la période de transmission et la période de réception des ondes, s'il y'a un retard donc on a quelque chose qui bouge. Il y'a des systèmes notamment dans le domaine de la sécurité, qui utilisent des technologies hybrides, c'est-à-dire inclure ces deux types de capteurs dans une seule boîte ou carte électronique pour une détection plus efficace.

II.5.3.3 Conception de Shield :

Un capteur de mouvement typique contient deux types de fils, les fils d'alimentation (deux), et les fils de sortie ; généralement un ou deux fils pour la sortie selon le constructeur. S'il y'a deux sorties, ils représentent un interrupteur qui est ouvert s'il y'a aucun mouvement, et fermé s'il y'a une détection. Pour les capteurs avec un seul fil de sortie, lorsque la détection, ils font juste basculer l'état de ce pin vers un état HIGH de 5V.

Ce Shield dédié au PIR ne nécessite aucun composant électronique, il a besoin de l'alimentation et le branchement de sa sortie vers une entrée numérique ou analogique d'une carte du type Arduino UNO. Donc on peut éviter la réalisation du circuit imprimé de ce Shield et raccorder le NRF24L01 et le PIR directement aux pins Arduino. C'est cette approche qui a été adoptée. Toutefois, on a conçu le schéma du circuit imprimé pour des éventuelles implémentations.

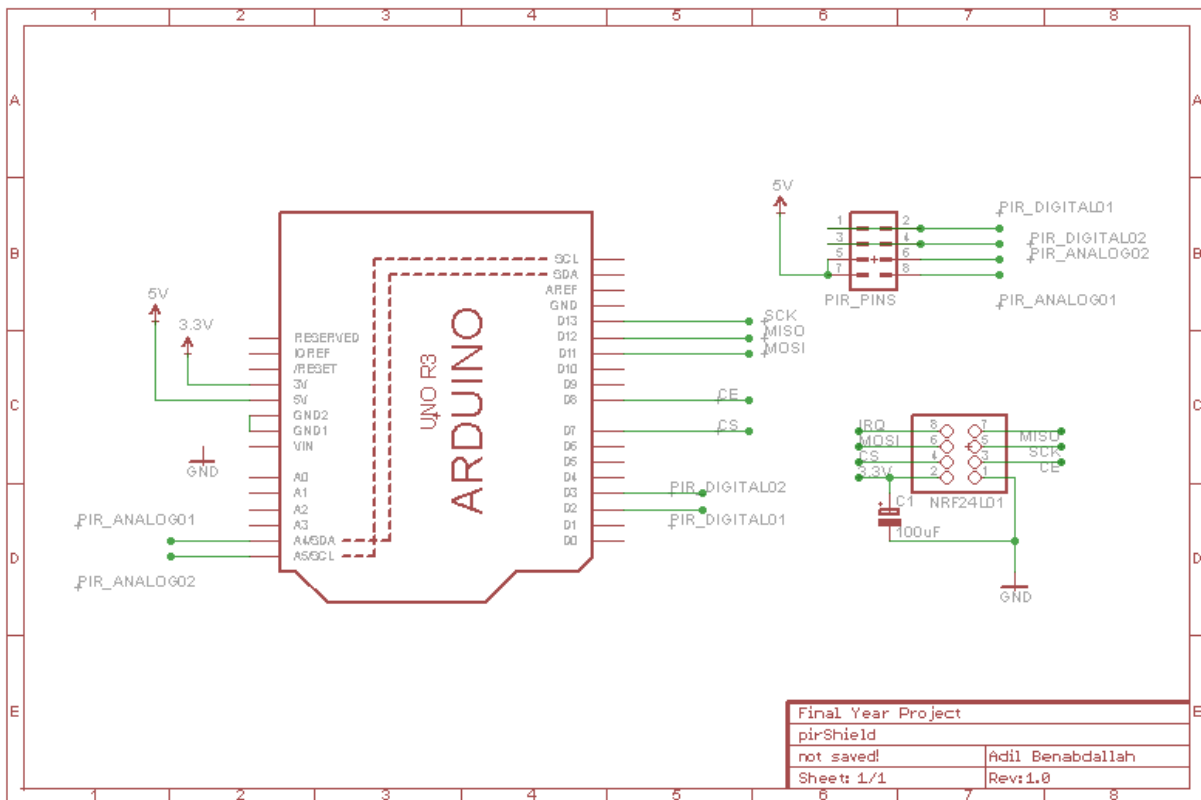


Figure 30. Schéma Shield Detecteur de Mouvement

Le Shield présenté ci-dessus peut accepter jusqu'aux quatre capteurs de mouvement, l'interface est à 8 broches (dans le schéma, le block dénommé PIR_PINS), les quatre entrées supérieures sont pour les capteurs a 1 fil (lié directement au entrés numérique d'Arduino), et les quatre qui restent, ils sont pour les capteurs a deux fils, puisque ils jouent le rôle d'un interrupteur, il faut juste raccorder un des fils d'un seul capteur avec le 5V (pin 7 ou 5 de PIR_PINS) et l'autre avec l'entrée analogique ou numérique Arduino (pin 5 ou 3 de PIR_PINS). Les figures suivantes montrent le circuit imprimé de Shield de Détection de Mouvement :

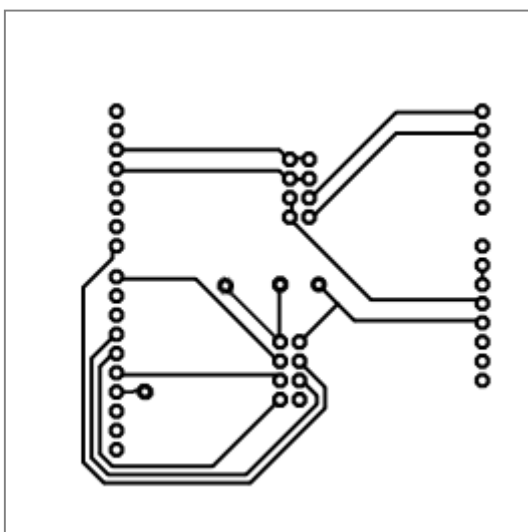


Figure 28. PCB Shield (échelle 1/1)

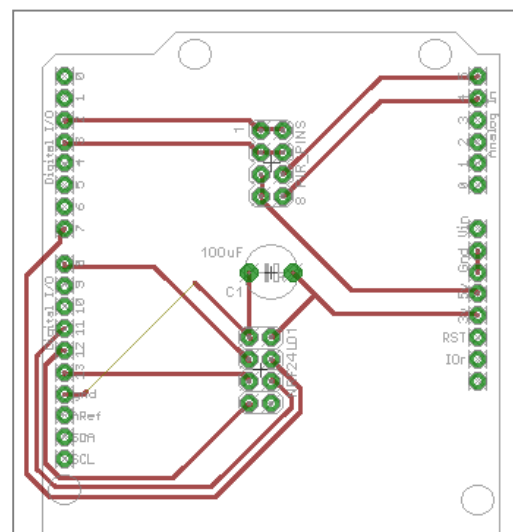


Figure 29. Aide Schéma

Note : Il faut connecter le strap en jaune dans le schéma.

II.5.3.4 Algorithme Proposé :

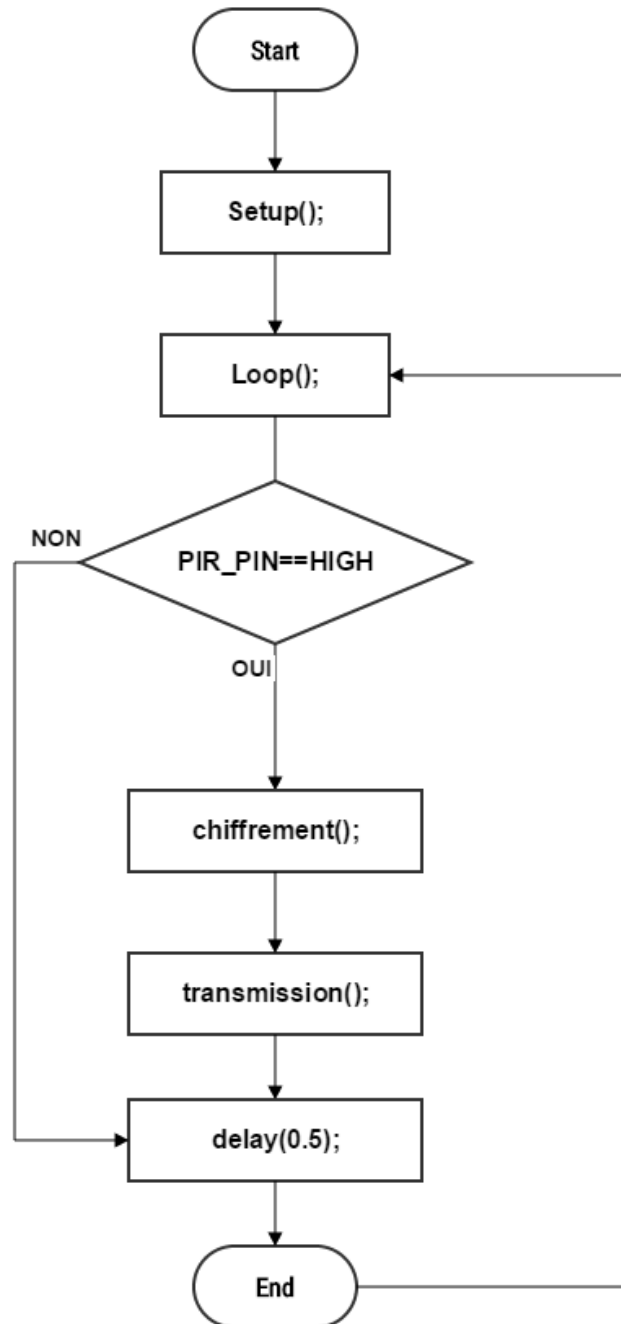


Figure 31. Algorithme de Detection de Mouvement

Chapitre III : Interrupteur Sans-fil

III.1 l'Éclairage Automatique :

III.1.1 Définition :

L'éclairage automatique est un système permettant d'éclairer une pièce ou une zone avec ou sans l'aide d'un interrupteur. Il se déclenche à l'aide d'un détecteur de mouvement (pour notre cas c'est l'Arduino avec les capteurs PIRs ou Micro-ondes). Dans une habitation, l'éclairage automatique présente plusieurs utilités, notamment celles de réduire la facture d'électricité et de bénéficier de confort supplémentaire ^[20].

III.1.2 Son Fonctionnement :

Grâce à un détecteur de mouvement, l'éclairage d'une pièce ou d'une zone extérieure peut se faire de manière automatique. En effet, le détecteur de mouvement, équipé de capteurs infrarouges, va repérer la chaleur d'un corps dans un champ déterminé selon les modèles. Lorsqu'il repère une variation de température, il déclenche l'éclairage. Ainsi, placé à des endroits stratégiques d'une maison, l'éclairage automatique peut s'avérer très utile, voire indispensable à la prévention des accidents domestiques.

III.1.3 Le Rôle de l'Éclairage Automatique :

Un éclairage automatique peut avoir plusieurs rôles, notamment celui de prévenir les intrusions, de limiter le risque d'accidents ou de chutes et surtout apporte un confort supplémentaire à l'habitation et ses occupants. Il apporte donc un plus grand confort de vie dans la maison, ce qui constitue une des principales avancées dans le mariage forcé de la domotique et du luminaire. Pour l'extérieur, l'éclairage automatique présente également une grande utilité. Il est d'ailleurs essentiellement sollicité pour ce type d'utilisations. En effet, lorsqu'on rentre chez soi le soir, les risques de trébucher sont importants parce que l'on se déplace dans l'obscurité jusqu'à trouver le premier interrupteur à portée de main. Avec un dispositif d'éclairage automatique, on ne court plus le risque de se prendre les pieds dans un tuyau ou un râteau laissé par terre par mégarde. L'on peut également décharger les courses du coffre de sa voiture, sans avoir besoin de laisser les phares allumés, ou rentrer à la maison pour ensuite allumer les lumières d'extérieures.

III.1.4 Les Modes d'Utilisation de l'Eclairage Automatique :

III.1.4.1 Prévention des intrusions :

Pour une utilisation fiable et sécurisée face aux cambriolages, l'éclairage automatique devra être puissant et couvrir les zones extérieures sensibles, telles que les portes ou fenêtres. Ainsi, dès lors qu'un intrus tente de s'approcher des lieux, un puissant éclairage l'incitera à fuir. Il faudra pour cela choisir un éclairage automatique avec différents niveaux de sensibilité équipés de détecteurs à infrarouge à grande portée. Ainsi, les intrus sont repérés de loin et n'ont pas l'occasion de dérober des objets laissés à l'extérieur.

III.1.4.2 Prévention des risques d'accident :

Pour une utilisation entrant dans le cadre d'une prévention contre les risques d'accident domestique, notamment à l'intérieur du logis, un éclairage automatique équipé d'un système détectant l'obscurité demeure plus adapté. Ainsi, la lumière ne s'allumera pas en plein jour et permettra à son utilisateur de réaliser des économies d'énergie. La détection d'obscurité peut aussi être utile pour le jardin parce que les lumières n'ont pas lieux de s'allumer pendant la journée, même s'il y a du mouvement. Installé dans un escalier, un détecteur de présence ou détecteur de mouvement couplé avec l'éclairage automatique diminuera de 90% le risque de chutes. Le risque de chute des personnes âgées dans un escalier la nuit est une source d'accidents domestiques qui peut être très facilement baissé avec un détecteur de présence ou un détecteur de mouvement couplé à l'éclairage automatique dans un escalier.

III.2 L'Interrupteur Sans-fil et La Détection de Mouvement :

Pour un système d'éclairage automatique, on a choisit la carte Arduino UNO avec son Shield comme un détecteur de mouvement sans fil, mais pour que notre système domotique puisse réagir aux signaux transmis par ce capteur il faut qu'on implémente un interrupteur qui a la capacité de recevoir ces signaux et faire allumer ou éteindre la lumière selon les commandes reçues. La partie éclairage automatique de notre solution est montrée dans la figure suivante.

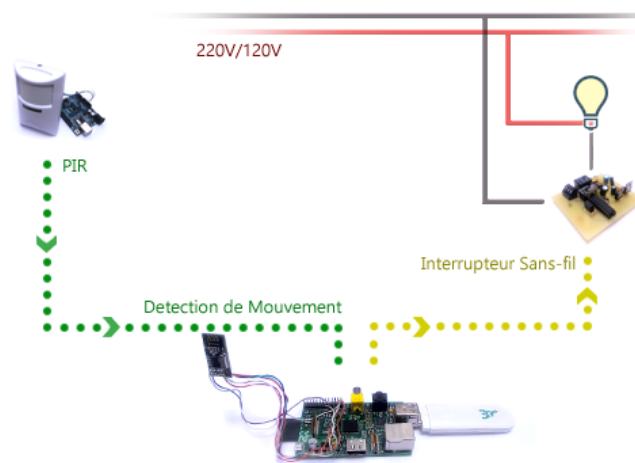


Figure 32. Lumière Automatique

Lorsqu'un mouvement est détecté par le PIR, l'Arduino va transmettre un paquet RF vers le Raspberry Pi. Selon l'algorithme du contrôle principal, le Raspberry Pi va prendre la fonction de calcul de décision d'allumer ou pas la lampe de la chambre (par exemple dans le jour la fonction de lumière automatique est désactivée). Si c'est le cas d'un allumage, donc il va envoyer un paquet RF vers l'interrupteur sans-fil. Pour raison d'économie d'énergie ce dernier contient un temporisateur ajustable (défini par l'élément centrale) ; après qu'une certaine durée est écoulée, la lampe sera éteinte par l'interrupteur lui-même.

On a déjà introduit le Shield PIR, qui est l'élément responsable de la détection du mouvement dans la figure précédente. Mais on n'a pas discuté notre proposition d'interrupteur sans-fil, dans les sections suivantes de ce chapitre on va détailler le fonctionnement et la réalisation de cet élément.

III.2.1 Composants Essentiels :

Un interrupteur à commande par câble est constitué d'un relais et un circuit d'adaptation, pour notre cas un module RF est responsable des communications. Le circuit d'adaptation ou de contrôle est une implémentation d'un circuit électronique compatible avec les outils de programmation Arduino. On a réalisé cette carte autour d'un microcontrôleur ATmega328. Elle contient les composants principaux suivants :

- . Des Relais
- . Un Microcontrôleur : ATmega328 flashé avec le boot loader Arduino
- . Un Module RF : NRF24L01
- . Des Régulateurs de tension

III.2.2 Préparation de l'ATmega328 :

Le model open source hardware choisi par les concepteurs d'Arduino donne accès aux fichiers de conception électronique et de fabrication électronique (circuit, typon). Cela permet à d'autres (entreprises ou inventeurs) de reproduire la carte, soit à l'identique (clone) soit en faisant évoluer le modèle. Bien entendu, c'est la seconde possibilité que nous préférons : la carte d'origine n'est pas parfaite, d'ailleurs la version UNO a changé plusieurs points essentiels, y compris les composants. Surtout, des modèles spécialisés qui sont apparus : pour la robotique, le contrôle de LEDs, la miniaturisation,... etc.

III.2.2.1 Une Carte Compatible Arduino :

Donc on va concevoir un modèle spécialisé qui joue le rôle d'un interrupteur sans-fil. Il est basé sur le circuit d'implémentation minimal d'un microcontrôleur AVR, dans ce cas c'est l'ATmega328p. La figure suivante montre le schéma d'un microcontrôleur AVR avec le nombre minimal des composants électronique.

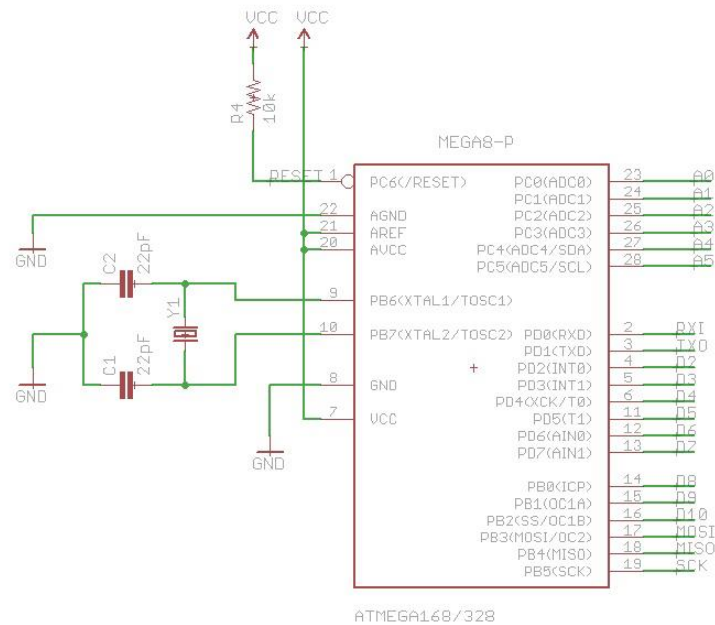


Figure 33. Circuit Minimal d'un AVR

On remarque que le minimum des composants constitue les éléments de l'horloge, et l'alimentation. Toutefois, avant d'implémenter ce circuit, il faut programmer l'ATmega328 à l'aide d'un programmeur AVR, mais pas n'importe quel programme ou code va être téléchargé sur le microcontrôleur. Pour que notre interrupteur sans-fil soit compatible avec l'environnement de développement Arduino, la première chose à faire est de graver le boot loader Arduino sur l'ATmega328.

III.2.2.2 Gravure du Boot Loader Arduino:

On peut acheter des ATmega328 déjà gravés avec ce boot loader, mais des fois ils coûtent le double d'un seul ATmega328 sans boot loader, donc pour le long terme il est préférable d'acheter un programmeur ISP (In-System Programming). Cependant, il existe une autre méthode pour télécharger le boot loader sur un AVR, en utilisant la carte Arduino UNO elle-même.

Puisque notre système domotique prend les cartes Arduino UNO comme des nœuds communicants, on va prendre une d'elles, et la programmer avec le sketch dénommé Arduino ISP, qui se trouve dans la liste d'exemples dans le menu **Fichier** de l'interface Arduino. Ce code, permet de graver un boot loader ou un code Arduino sur un microcontrôleur AVR supporté (de préférence ATmega328) à l'aide d'une carte Arduino UNO. Quand le logiciel termine le téléchargement de ce sketch, on va débrancher la carte UNO de l'ordinateur et faire le câblage nécessaire pour raccorder l'ATmega328 avec la carte Arduino UNO en consultant la figure dans la page suivante ^[21].

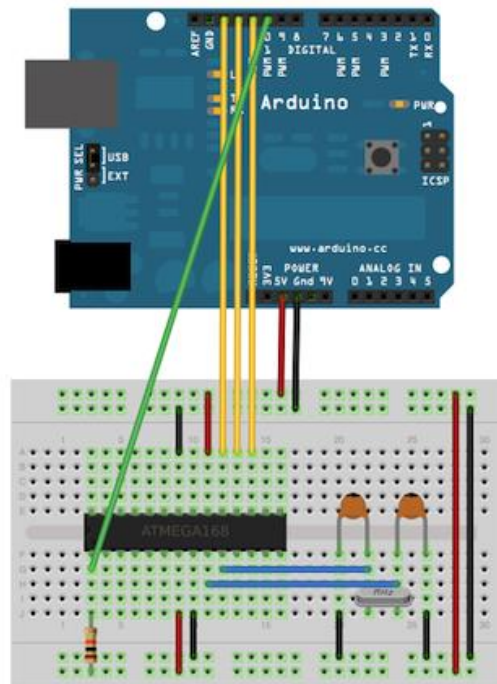


Figure 34. Raccordement pour Programmation ISP

Le raccordement fait juste relier l'interface SPI de l'ATmega328 de la carte UNO avec celle de l'ATmega328 qu'on souhaite programmer (les fils en jaune et vert), en fournissant une alimentation de 5V et une horloge de 16 Mhz. Maintenant on va brancher la carte UNO avec l'ordinateur, et sur l'interface graphique du logiciel Arduino, vers le menu « **Outils** », on va lancer « **Graver la séquence d'initialisation** », et le boot loader sera chargé sur le microcontrôleur. Maintenant on a un ATmega328 qu'on peut adapter dans un circuit pour obtenir une carte électronique compatible avec Arduino.

III.3 Conception d'un Interrupteur Sans-fil :

III.3.1 Choix de Relais :

Tous les pays européens et ainsi que la plupart des pays africains et asiatiques utilisent une tension nominale comprise entre 220 et 240 V. Quant au Japon, tous les pays d'Amérique du Nord, la plupart des pays d'Amérique centrale, et quelques pays d'Amérique du Sud emploient une tension entre 100 et 127 V. Au cours des années 1980, l'harmonisation décidée par l'Europe a été mise en œuvre par le CENELEC et a abouti au choix d'une tension de 220 V avec une fréquence de 50 Hz^[22]. Donc pour que notre interrupteur soit compatible avec la majorité des tensions utilisées, il faut qu'il supporte une tension jusqu'au 220V. Cependant, il y'a le problème de puissance, car il existe plusieurs types de lampes dans le marché avec des ampérages ou puissance différentes. Mais on peut dire que la limite en puissance d'une lampe dédiée à l'utilisation domestique est autour 200W. Avec une tension de 220V, ça va nous donner un courant de 0.9A, donc notre relais doit tolérer un ampérage minimal de 0.9A. Puisque on n'a pas trouvé un relais qui utilise une tension de commande 5V (même tension d'alimentation de l'ATmega328), et en-même temps répond aux conditions minimales citées antérieurement, on a choisi un relais 12V qui peut gérer une puissance de 2200VA à une

tension de 220V (10A), et de 1440VA à 120V (12A). Mais lorsque le filament d'une lampe à incandescence est froid, sa résistance ohmique est plus faible que celle dans l'état chaud, par un facteur de 10 [23]. Avec cette faible résistance, le courant d'appel va être 10 fois plus grand que le courant du fonctionnement normal, donc le relais choisis (220V-10A) fonctionnera comme un relais de 220V-1A. Avec une lampe de 60W, le courant est de $60W/220V = 0.27A$, en conséquence ce relais a la capacité de gérer trois lampes de 60W en-même temps, ou une seule lampe de 200W.



Figure 36. Relais BS-115C

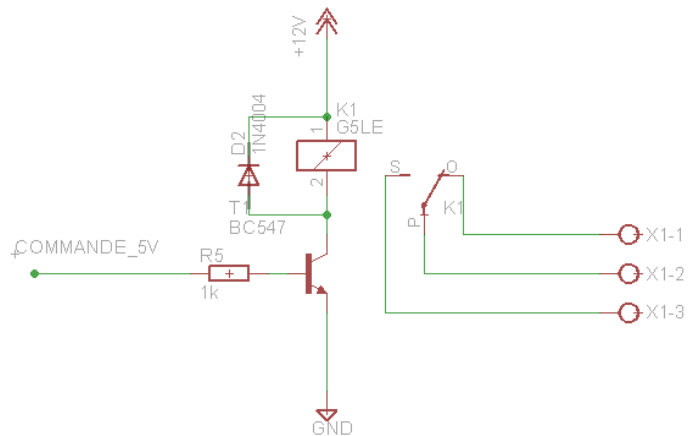


Figure 35. Circuit Implementé

III.3.2 L'Alimentation :

Le microcontrôleur ATmega328 fonctionne sur une tension d'alimentation de 5V, le relais sur 12V, et le module RF sur 3.3V. La nécessité de fournir ces différentes tensions enlève l'option d'alimentation avec batteries, car les régulateurs de tensions qu'on va utiliser, ont un rendement de puissance médiocre (environ 45%), c-à-dire ils consomment ou convertissent la majorité de puissance d'une batterie en énergie thermique, donc elle va être totalement consommé dans quelque jours ce qui rend cette implémentation inconmode. C'est pour ça cet interrupteur, et comme la majorité des interrupteurs sans fil, nécessite un module d'alimentation qui peut fournir la tension maximale utilisé par l'interrupteur, pour notre cas c'est la tension 12V du relais. Pour alimenter le ATmega328 avec 5V et le module NRF24L01 avec 3.3V, on va consacrer respectivement deux régulateurs de tensions le LM7805 et le LM317. Le LM7805 est un régulateur de tensions linéaire qui accepte une tension minimal de 7.3V et la convertie en 5V. On peut l'implémenter tell

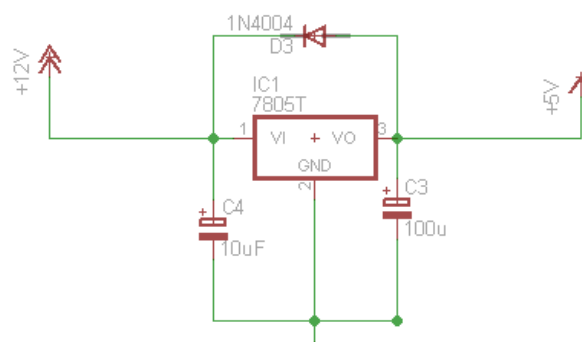


Figure 37. Schéma d'Implementation LM7805

qu'il est mais selon sa fiche technique il est préférable d'ajouter des condensateurs aux bornes d'entrée et de sortie du LM7805.

Le LM317 est aussi un régulateur linéaire mais un qui est ajustable, dont la tension de sortie peut être ajusté entre 1,25V et 37V. Pour une application de tension fixe, la broche de contrôle est alors généralement commandée avec deux résistances fixes. Les fiches techniques du fabricant fournissent des configurations standard pour la réalisation. Pour notre cas la combinaison de résistances utilisé pour une tension de sortie 3.3V est R3 :220Ω et R4 :330Ω, la figure suivante montre le circuit LM317 proposé :

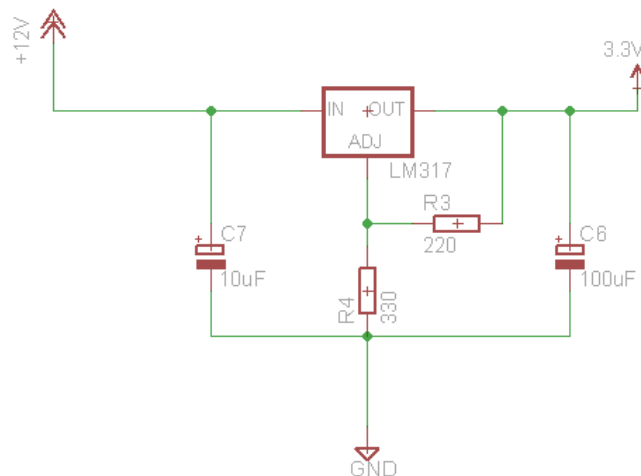


Figure 38. Circuit LM317

III.3.3 Schéma Final :

Notre solution pour un interrupteur sans fil est une carte ou plus spécifiquement un circuit électronique conçue autour d'un microcontrôleur ATmega328 qui contrôle un circuit de relais à commande. Pour avoir une compatibilité entre notre carte et l'environnement de développement Arduino, on a chargé cet AVR par le boot loader Arduino. Comme tous les nœuds mentionnés dans le chapitre précédent, l'implémentation du module RF NRF24L01 est nécessaire pour assurer une communication sans-fil avec l'élément centrale.

Cet interrupteur fonctionne sur trois tensions, on a le 5V pour le microcontrôleur fournit par le régulateur LM7805, le 3.3V fournit par le régulateur LM317, et finalement le 12V qui est la source d'alimentation principale arrivant d'un module ou adaptateur d'alimentation 12V.

Pour qu'on puisse programmer cet interrupteur avec un code Arduino sans retirer le microcontrôleur et refaire le câblage décrit dans la section de gravure du boot loader, on a ajouté un support 2x4 broches (comme celui du NRF24L01) relié aux pins SPI de l'AVR pour permettre la programmation SPI sans retirer l'ATmega328, il suffit juste de connecter les broches avec leurs câbles SPI respectives d'un Arduino ou programmeur ISP.

Le circuit final de l'interrupteur sans fil qu'on a conçu est schématisé dans la figure suivante :

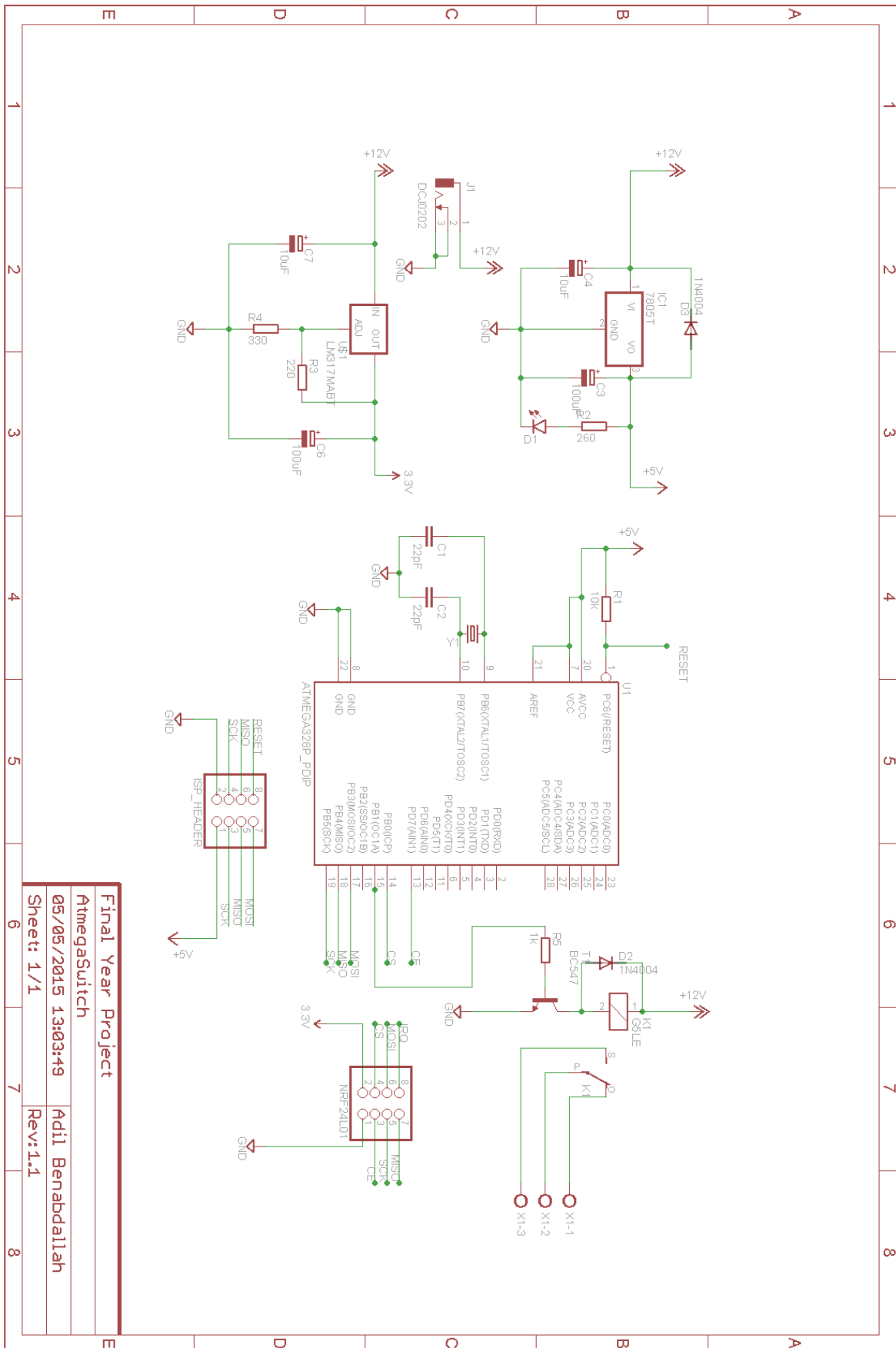


Figure 39. Schéma Final d'Interrupteur Sans-fil

Final Year Project	
AtmegaSwitch	
05/06/2015 13:03:49	Adil Benabdallah
Sheet: 1/1	Rev:1.1

Après la schématisation, on a produit le plan du circuit imprimé de l'interrupteur sans-fil suivant :

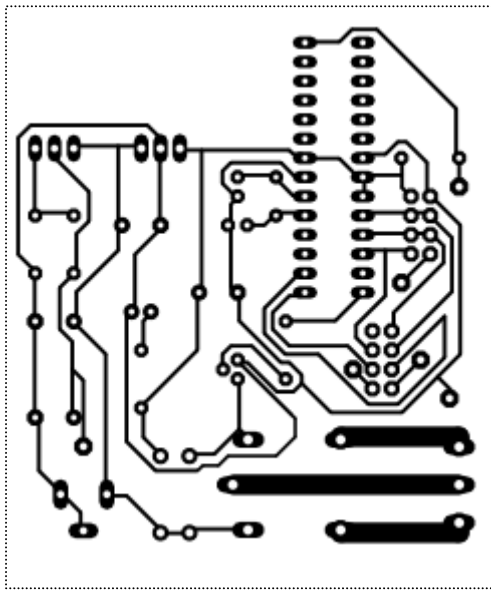


Figure 41. PCB Interrupteur Sans-fil

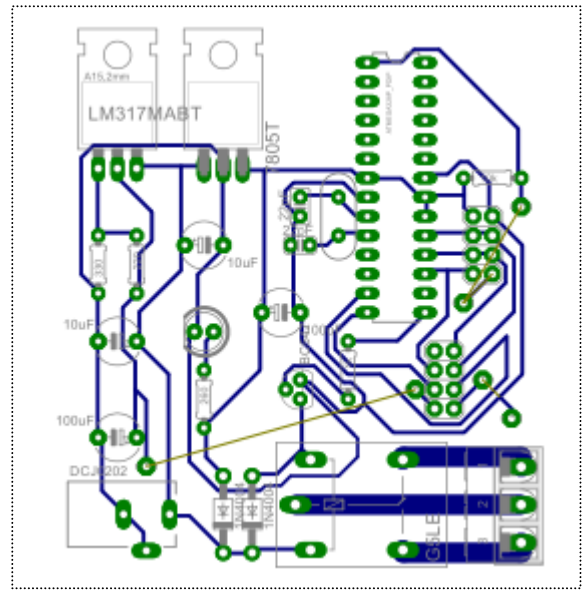


Figure 40. Aide Schéma

Remarque : Après la réalisation Il faut connecter les trois straps identifiés par les lignes jaunes à l'aide-schéma.

Voilà une image de l'interrupteur sans fil après la réalisation :

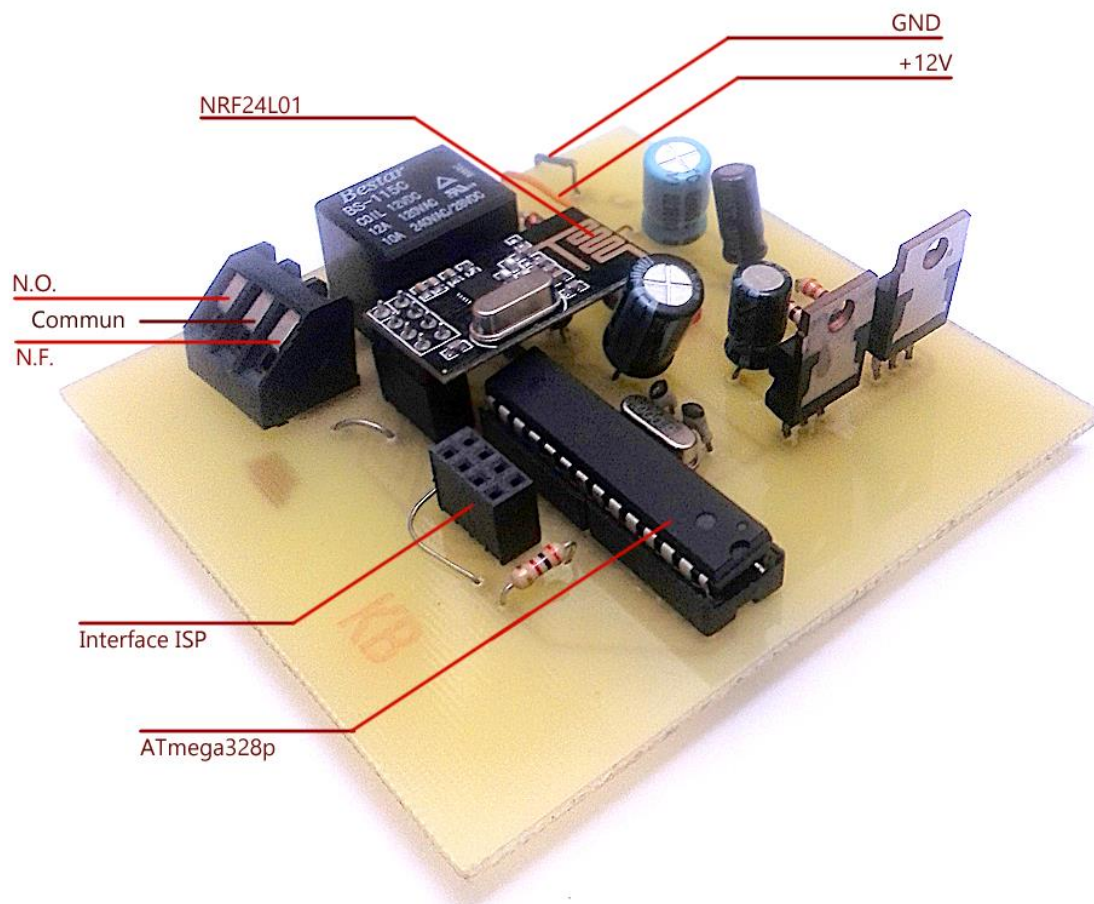


Figure 42. Photo de L'interrupteur Sans-fil Réalisé

III.3.4 Algorithme de Fonctionnement :

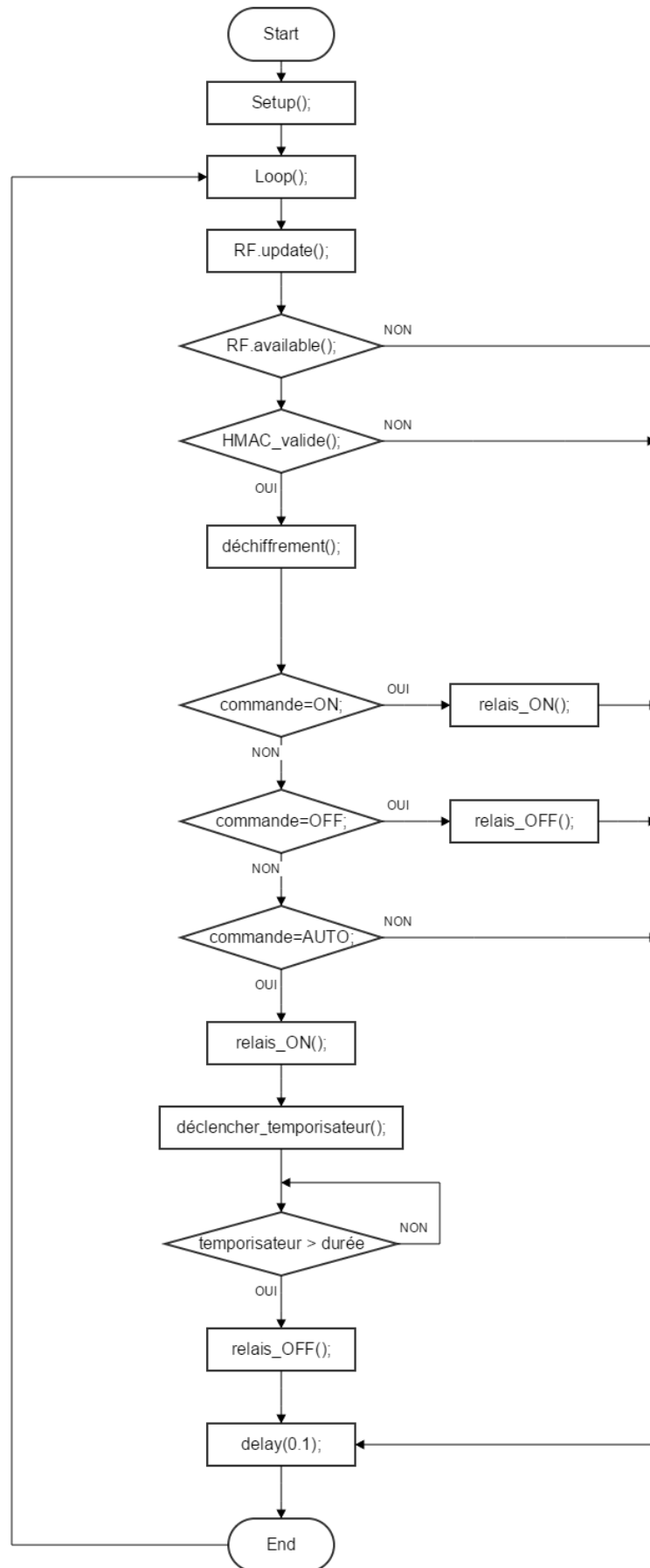


Figure 43. Algorithme Interrupteur Sans-fil

III.4 Conclusion

Dans ce chapitre on a exposé la conception d'un interrupteur d'allumage. Ce dernier est conçu autour d'un clone Arduino amélioré. L'ARDUINO est utilisé uniquement en raison des outils open source largement disponible. On a expliqué que cet interrupteur est utilisé pour le contrôle d'allumage, cependant, dans ce chapitre on a donné en bref d'autres fonctionnalités de ce dernier. Par le biais d'un terminal mobile, cet interrupteur peut également être utilisé pour activer à distance des équipements électronique, tel que la cafetière, le four, le bain, l'arrosage automatique...etc. Il a également la possibilité de simuler la présence des habitants si ces derniers sont absents. Beaucoup de fonctionnalités peuvent être ajoutées et adapté en exploitant les caractéristiques d'adaptation de l'élément central de commande et de contrôle qui vont être détaillés sur un plan de conception dans le chapitre suivant.

Chapitre IV : L'Organe Central et le Processus de Contrôle

IV.1 L'Organe Central :

IV.1.1 Rappel :

Les nœuds communicants constituent la grande partie matérielle dans ce système domotique. On peut distinguer deux groupes de nœuds, il y a des nœuds qui ont comme fonction d'acquies des informations sur l'état physique à l'intérieur de la maison comme les nœuds capteurs (température, humidité et gaz) et les détecteurs de mouvement, et il y a le deuxième groupe qui fait la commande ou le contrôle des appareils électriques de la maison, comme le contrôleur HVAC et l'interrupteur sans-fil pour l'éclairage automatique. Mais ces éléments ont besoin d'un élément central qui va prendre la tâche de gérer et traiter les informations assemblées, et en même temps faire calculer la décision d'allumer ou d'éteindre les différents instruments électriques de la maison comme le système HVAC de la maison (via le contrôleur HVAC), et l'éclairage de chaque pièce (via les interrupteurs sans-fils). L'organe central de notre système domotique est une carte électronique qui s'appelle le Raspberry Pi, mais avant d'introduire cet élément final, on va faire un rappel du notre système domotique par la visualisation du schéma synoptique suivant :

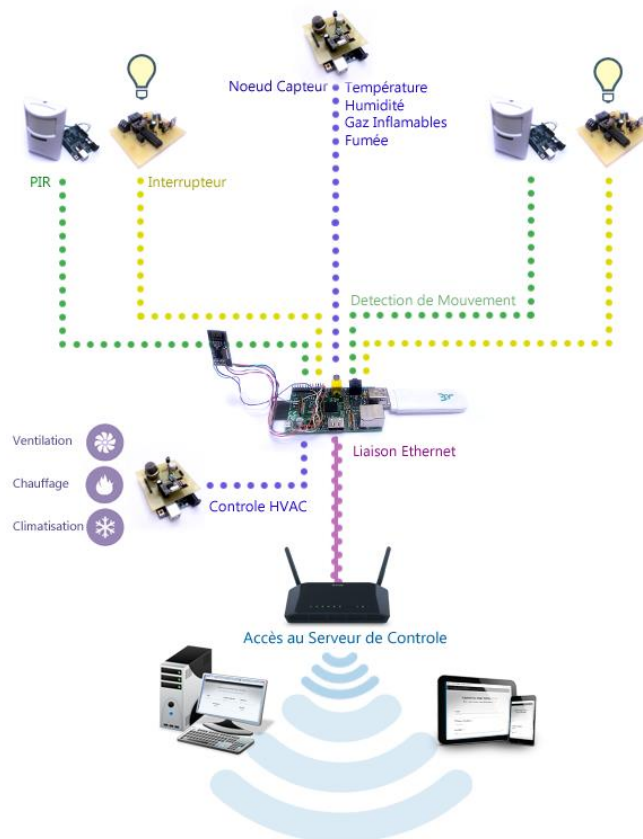


Figure 44. Schéma Synoptique du Système Domotique

IV.1.2 Le Raspberry Pi :

Le Raspberry Pi est un ordinateur à mono-carte avec un processeur ARM conçu par le créateur de jeux vidéo David Braben, dans le cadre de sa fondation Raspberry Pi. Cet ordinateur, qui a la taille d'une carte de crédit, est destiné à encourager l'apprentissage de la programmation informatique ; il permet l'exécution de plusieurs variantes du système d'exploitation libre GNU/Linux et des logiciels compatibles. Il est fourni nu (carte mère seule, sans boîtier, alimentation, clavier, souris ni écran) dans l'objectif de diminuer les coûts et de permettre l'utilisation de matériel de récupération. Son prix de vente était estimé à 25 \$, soit 19,09 €, début mai 2011. Les premiers exemplaires ont été mis en vente le 29 février 2012 pour environ 25€. Début 2015, plus de cinq millions de Raspberry Pi ont été vendus ^[24].

IV.1.3 Le Raspberry Pi Modèle B:

Le Modèle B est la version matérielle de la carte Raspberry Pi qu'on a utilisé, il se compose d'une "carte mère miniature" sur laquelle on retrouve un processeur ARM intégré et cadencé à 700 MHz. Il embarque également un contrôleur graphique Broadcom Videocore IV permettant d'utiliser les sorties vidéo présentes et l'affichage sur moniteur ou TV par exemple, avec une taille du mémoire RAM de 512Mo. Intéressons-nous aux caractéristiques techniques de ce mini PC Raspberry Pi Type B. Ce modèle B est équipé des entrées/sorties suivantes :

- Deux ports USB 2.0
- Un port RJ45 (pour liaison Ethernet)
- Un lecteur de cartes SD
- Un port HDMI
- Une sortie audio (jack 3,5 mm)
- Une sortie RCA

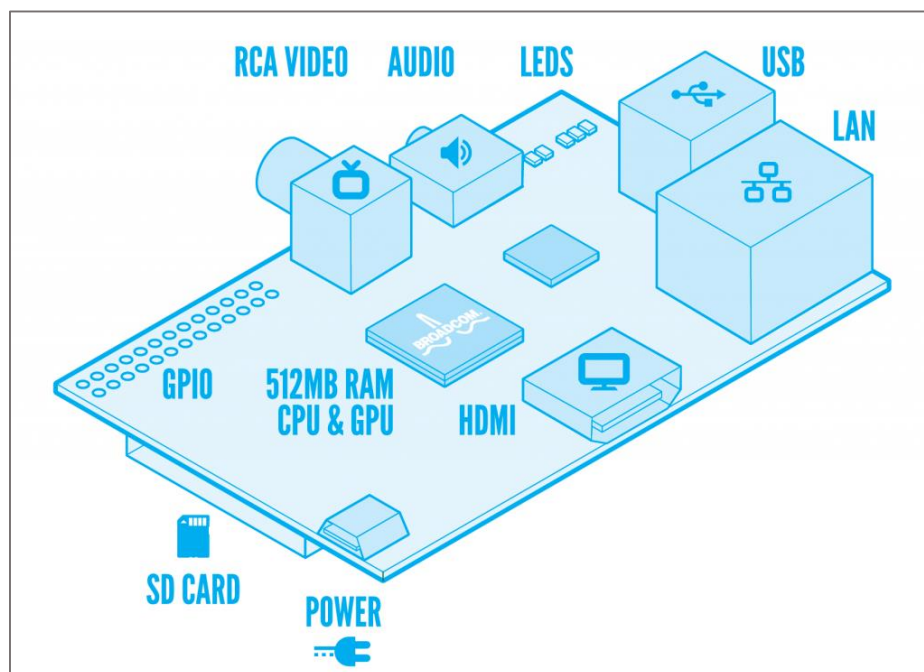


Figure 45. Raspberry Pi Modèle B

IV.1.4 Les Ports GPIO :

Le Raspberry Pi possède, en plus des connectiques classiques USB, HDMI, etc... un connecteur GPIO. GPIO signifie en anglais « General Purpose Input Output » et pourrait être traduit en français par entrées/sorties numériques. Ces entrées/sorties permettent d'étendre les fonctionnalités du Raspberry Pi en lui donnant la possibilité d'agir sur des LEDs ou des afficheurs LCD par exemple, lire l'état d'un interrupteur, d'un capteur, etc...

Ce connecteur GPIO dispose de différents types de connexion :

- Des broches utilisables en entrée ou sortie numérique tout ou rien.
- Des broches pour une interface I2C.
- Une interface SPI (le module NRF24L01 est connecté à cette interface)
- Les broches Rx et Tx pour la communication UART avec les périphériques séries.
- Des broches pouvant être utilisé en PWM ("Pulse Width Modulation") permettant le contrôle de puissance ou PPM.















Raspberry Pi Model A & B (P1 Header)					
PIN #	NAME			NAME	PIN #
	3.3 VDC Power	1		5.0 VDC Power	2
8	SDA0 (I2C)	3		DNC	4
9	SCL0 (I2C)	5		0V (Ground)	6
7	GPIO 7	7		TxD (UART)	15
	DNC	9		RxD (UART)	16
0	GPIO 0	11		GPIO1	1
2	GPIO2	13		DNC	14
3	GPIO3	15		GPIO4	4
	DNC	17		GPIO5	5
12	MOSI	19		DNC	20
13	MISO	21		GPIO6	6
14	SCLK	23		CE0	10
	DNC	25		CE1	11
		26			

Figure 46. Pins GPIO et Leur Fonctions

IV.1.5 Aspects Logiciel :

Toutes les applications compatibles avec le système d'exploitation et le processeur ARM, ou utilisant un environnement d'exécution virtuel (Java, émulateurs...) sont susceptibles de fonctionner : KOffice, Python... Les principales contraintes portent sur les performances du processeur et la mémoire vive disponible (256 Mo). Ce dernier point a toutefois été corrigé avec l'arrivée des versions embarquant 512 Mo de mémoire vive ce qui est suffisant pour lancer le processus du contrôle et le serveur web de la maison intelligente sur le système d'exploitation Raspbian (une distribution Linux Debian pour le Raspberry Pi).

IV.1.6 Mode de Travail :

Le Raspberry Pi peut fonctionner en mode Headless, avec ni clavier ni écran, il faut juste le brancher avec un routeur ou switch (Routeur Wi-Fi est meilleur) pour l'accès à distance par SSH ou Telnet, et lui ajouter un Modem 3G USB pour la connectivité SMS. On peut trouver l'adresse du Raspberry Pi sur le réseau Ethernet par un simple scan IP à travers la plage IP locale. L'Access se fait par le biais d'un client SSH comme Putty, après qu'on gagne l'accès (utilisateur: pi, mot de passe: raspberry) on va lui attribuer une adresse IP statique en remplaçant tous les lignes du fichier `/etc/network/interfaces` par les lignes suivantes :

```

ifacé eth0 inet static
address 192.168.1.100
netmask 255.255.255.0
gateway 192.168.1.1

```

Maintenant on peut l'accéder sans refaire le scan IP, en établissant une connexion SSH vers l'adresse IP 192.168.1.100 .

IV.2 Le Processus du Contrôle :

C'est le programme principal qui va gérer les informations et les commandes du contrôle qui circulent dans le système domotique, il prend comme fonction le calcul de décisions et la collection d'informations sur la maison et ces occupants, tout en assurant un confort et une qualité de vie optimale. Ce processus va interagir avec le serveur web local pour exécuter les requêtes des utilisateurs, par exemple le contrôle manuel de lumière. Pour la communication RF avec les autres nœuds on va proposer notre propre protocole qu'on dénomme HomeRF conçu spécifiquement pour les modules NRF24L01 mais on peut l'adapter pour d'autres modules RF. La figure suivante montre les deux piles protocolaires implémentées :

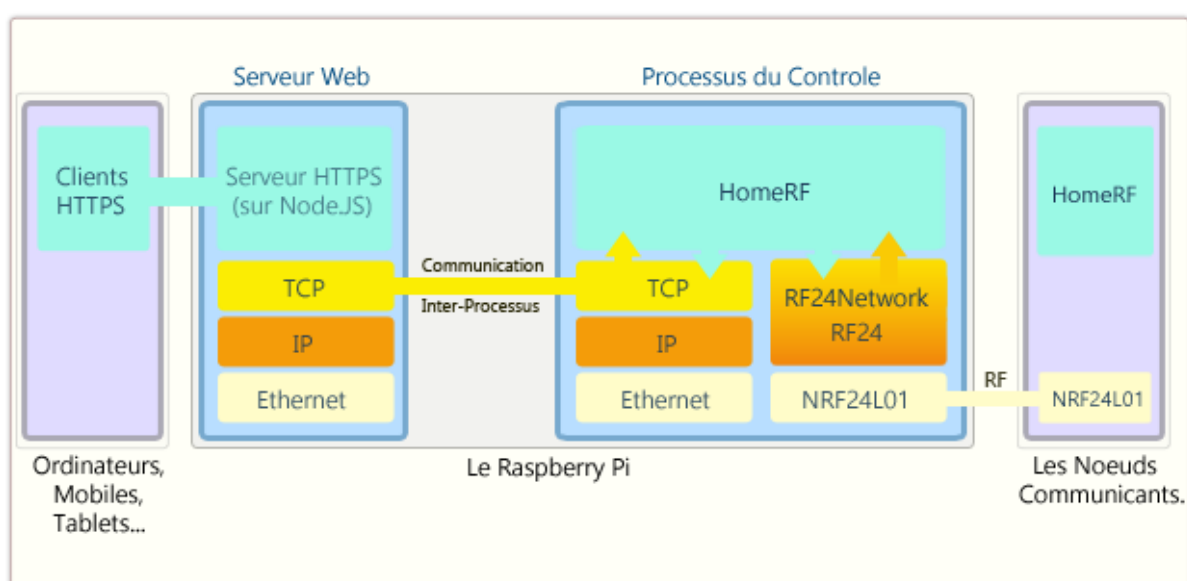


Figure 47. Piles Protocolaires du Système Domotique

Pour assurer la meilleure performance possible en exploitant les ressources matérielles du Raspberry Pi, le processus principal est programmé avec le langage C++, mais on note qu'il est dur (s'il n'est pas incommode) de programmer ou implémenter un serveur http ou bien HTTPS avec ce langage, c'est pour ça qu'on a implémenté un serveur HTTPS avec le langage NodeJS. Ce dernier fournit des outils et des bibliothèques faciles à utiliser comme la Framework ExpressJS qu'on a implémenté pour la réalisation d'un service HTTPS. Ces deux programmes doivent se communiquer entre eux, puisque ils s'exécutent dans la même machine il est nécessaire d'utiliser un protocole de communication interprocessus, et les sockets TCP/IP sont les plus répandus pour ce travail. Pour avoir interagir avec le matériel RF qui est le module NRF24L01, la bibliothèque RF24 fait l'abstraction de toute capacité de ce module RF et les exporte en fonctions ou procédures prêtes à l'emploi. La bibliothèque RF24Network exploite ces fonctions pour fournir des nouvelles fonctionnalités comme :

- Adresse d'Hôte : adressage logique.
- Routage des paquets RF d'un nœud à un autre dans une topologie maillée.
- Fragmentation et Réassemblage des paquets RF (version de développement).

IV.2.1 Le Protocol HomeRF :

Le protocole HomeRF est proposé autour de la bibliothèque RF24Network, il est responsable de l'encapsulation, le codage et l'authentification des données et les commandes RF échangées entre les nœuds communicants et le processus de contrôle. Pour cela on a créé une bibliothèque spécifique pour notre protocole HomeRF, et pour qu'il y ait une compatibilité entre le Raspberry Pi et les nœuds on a créé deux versions de cette bibliothèque, une pour le Raspberry Pi (C++) et l'autre pour le compilateur Arduino.

La figure suivante montre la structure d'un paquet HomeRF :

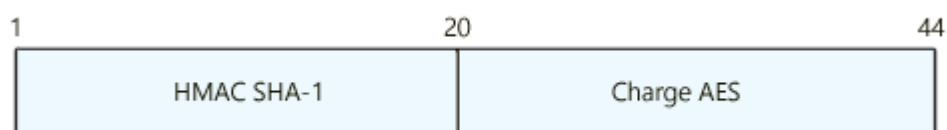


Figure 48. Paquet HomeRF

La taille du paquet est de 44 octets, avec un en-tête de 20 octets qui sert à l'authentification et la vérification de l'intégrité des données envoyées dans le champ Charge AES. Ce dernier contient les données cryptées et les champs nécessaires pour les déchiffrer en connaissant la clé AES.

IV.2.1.1 Le Champ HMAC-SHA-1 :

Un HMAC, de l'anglais Keyed Hash Message Authentication Code (code d'authentification d'une empreinte cryptographique de message avec clé), est un type de code d'authentification de message (CAM), ou MAC en anglais (Message Authentication Code), calculé en utilisant une fonction de hachage cryptographique en combinaison avec une clé secrète. Comme avec n'importe quel CAM, il peut être utilisé pour vérifier simultanément l'intégrité de données et l'authenticité d'un message. Cette clé est partagée entre le Raspberry Pi et tous les nœuds communicants mais jamais envoyé sur le réseau domestique (RF ou filaire). N'importe quelle fonction itérative de hachage, comme MD5 ou SHA-1, peut être utilisée dans le calcul d'un HMAC ; le nom de l'algorithme résultant est HMAC-MD5 ou HMAC-SHA-1. La qualité cryptographique du HMAC dépend de la qualité cryptographique de la fonction de hachage et de la taille et la qualité de la clé. Une fonction itérative de hachage découpe un message en blocs de taille fixe et itère dessus avec une fonction de compression ^[25]. La taille de la sortie HMAC est la même que celle de la fonction de hachage, et dans le cas du SHA-1 c'est 160 bits c'est-à-dire 20 octets pour le champ HMAC-SHA-1. On remarque que ce champ prend presque la moitié de la taille du paquet HomeRF, le résultat d'un compromis entre la performance et la sécurité : le calcul d'un HMAC-SHA-256 (32 octets) est lent sur les cartes Arduino (ATmega328), ce qui reste est le HMAC-SHA-1 (20 octets) avec presque le même niveau de sécurité et une performance optimale.

La figure suivante montre l'utilité du code HMAC-SHA-1 :

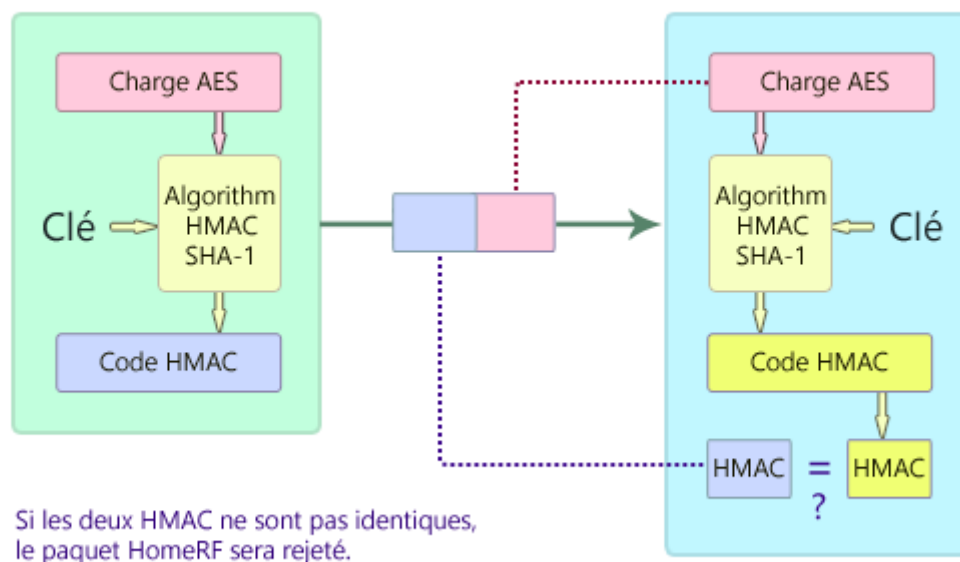


Figure 49. Processus de Vérification HMAC

IV.2.1.2 Le Champ Charge AES :

Pour la confidentialité des données, on a utilisé le standard AES-128. Comme premièrement accessible au public de la NSA pour le chiffrement avec le classement "top secret", l'Advanced Encryption Standard (AES) est l'un des algorithmes de cryptage aujourd'hui le plus fréquemment utilisé, et le plus sécuritaire. Son histoire de succès a débuté en 1997, lorsque le NIST (National Institute of Standards and Technology) a annoncé la recherche pour un successeur au standard de cryptage vieillissant DES. Un algorithme, appelé "Rijndael", développé par les cryptographes belges Daemen et Rijmen, a excellé dans la sécurité aussi bien dans la performance et la flexibilité. Il est arrivé en tête de plusieurs compétiteurs, et a été annoncé officiellement comme le nouveau standard de cryptage AES en 2001. Les algorithmes sont basés sur plusieurs substitutions, permutations et transformations linéaires, chacune réalisée sur des blocs de données de 16 octets - de là venant le terme chiffrement par bloc. Ces opérations sont répétées plusieurs fois, appelées "rondes". Durant chaque ronde, une clé de ronde unique est calculée de la clé de cryptage, et incorporée dans les calculs. Basé sur cette structure de bloc de l'AES, le changement d'un seul bit soit dans la clé, ou soit dans le bloc de texte brut donne un bloc de texte de chiffrement complètement différent ; un avantage clair sur les chiffrements de flux traditionnels. Finalement, la différence entre AES-128, AES-192 et AES-256, c'est la longueur de la clé : 128, 192 ou 256 bits – tous des améliorations drastiques comparées à la clé de 56 bits de DES. Voici un exemple : Le craquage d'une clé AES à 128 bits avec un super-ordinateur à la fine pointe de la technologie prendrait plus de temps que l'âge présumé de l'univers. En date d'aujourd'hui, il n'existe pas d'attaque possible et pratique contre le AES. Ainsi, le AES demeure le standard de cryptage préféré pour les gouvernements, les banques et les systèmes de haute sécurité autour du monde.

Le mode d'opération ECB (Electronic Codebook), il s'agit du mode le plus simple à implémenter. Le message à chiffrer est subdivisé en plusieurs blocs qui sont chiffrés séparément les uns après les autres. Le gros défaut de cette méthode est que deux blocs avec le même contenu seront chiffrés de la même manière, on peut donc tirer des informations à partir du texte chiffré en cherchant les séquences identiques. On obtient dès lors un « dictionnaire de codes » avec les correspondances entre le clair et le chiffré d'où le terme Codebook. C'est pour cette raison qu'on a utilisé le mode d'opération CBC (Code Block Chaining ou Enchaînement en Blocs) pour chiffrer les données RF. Pour éviter la vulnérabilité du mode ECB ; le CBC applique sur chaque bloc un 'OU exclusif' avec le chiffrement du bloc précédent avant qu'il soit lui-même chiffré. De plus, afin de rendre chaque message unique, un vecteur d'initialisation (IV) est utilisé, ce vecteur est généré aléatoirement par des fonctions TRNG (True Random Number Generator). Et pour qu'un récepteur portant la clé AES, puisse déchiffrer les données, il faut que la source envoie le vecteur d'initialisation aussi.

Donc les champs utiles dans la Charge AES sont : le vecteur d'initialisation et le block des données chiffrées, mais on ajoute un troisième champ qui désigne le type des données envoyées.

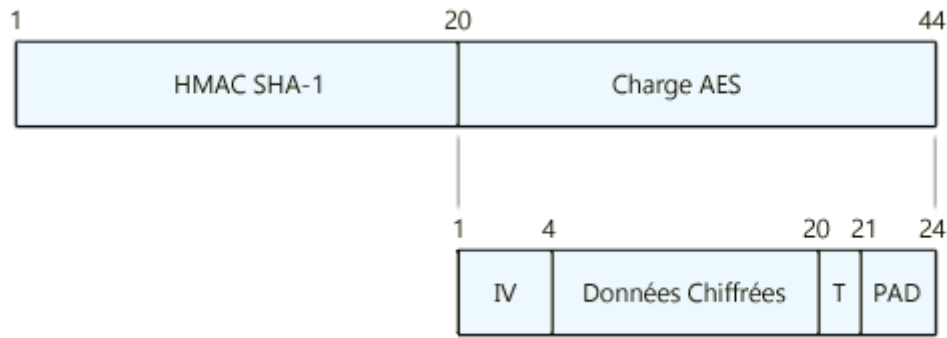


Figure 50. L'Encapsulation des Données Chiffrées

Les champs de la Charge AES sont :

* IV : Vecteur d'Initialisation sur 4 octets.

* Données Chiffrées sur 16 octet.

* T : Type des données envoyées, un caractère ASCII (1 octet) :

- 'H' : pour les commandes HVAC.

- 'S' : pour les données du Sensor Node (gaz, humidité, et température).

- 'L' : pour les commandes de lumières.

- 'P' : pour les données de la détection de mouvement.

* PAD : Bourrage de 3 octets pour que la taille du paquet HomeRF soit un multiple de 4.

La taille des données en état clair est inférieure ou égale à 16 octets, et puisque on utilise le AES-128, donc le block chiffré sera toujours de 16 octets. Dans la figure suivante on visualise la structure des données envoyées avant le chiffrement ou après le déchiffrement :

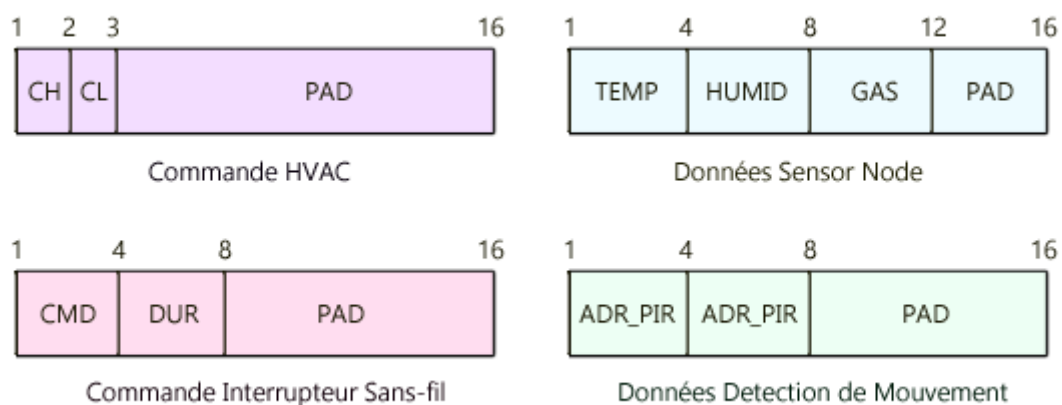


Figure 51. Structure des Données et Commandes HomeRF

Les données sont principalement envoyées par le Nœud de Capteurs et le Nœud Détecteur de Mouvement :

1. Données de Type 'S' :

- TEMP : Température en virgule flottante sur 4 octet.
- HUMID : Humidité en virgule flottante sur 4 octet.
- GAS : Niveaux de gaz en virgule flottante sur 4 octets.

2. Données de Type 'P' :

- ADR_PIR : Adresse du Nœud Détecteur de Mouvement source sur 4 octet (entier).
- ADR_PIR : Champ redondant.

Les paquets RF de commande sont envoyés par le Raspberry Pi pour contrôler les interrupteurs et le contrôleur HVAC, on distingue deux types :

1. Commande de Type 'T' :

- CL : Pour activer ou désactiver la climatisation (un booléen sur 1 octet).
- CH : Pour activer ou désactiver le chauffage (un booléen sur 1 octet).

2. Commande de Type 'L' :

- CMD : Chaîne de caractère ASCII sur 3 octets, avec trois valeurs possible :
 - 'ONN' : Pour fermer le circuit de l'interrupteur sans fil (allumer).
 - 'OFF' : Pour ouvrir le circuit (éteindre).
 - 'AUT' : Pour la lumière automatique.
- DUR : Si le champ CMD a comme valeur 'AUT', le champ DUR va indiquer la durée du temporisateur en milliseconde (4 octet).

On note que le PAD (bourrage) est rempli par l'algorithme AES-128 jusqu'au on atteint 16 octets.

IV.2.2 Déroulement du Processus :

Le processus du contrôle est divisé en deux portions principales, la première partie charge les variables et les données à partir du fichier de configuration qui est l'intermédiaire principal de personnalisation dans ce système domotique. La deuxième partie c'est le côté Daemon, c'est-à-dire c'est une boucle infinie qui écoute sur les évènements et réagit par le calcul de décision.

IV.2.2.1 Chargement du Fichier de Configuration :

Le fichier de configuration contient des balises qu'on peut modifier leurs valeurs pour changer ou définir le fonctionnement du système domotique, il y'a environ 17 balises qu'on peut agir sur. On peut mentionner quelques variables configurables :

Hystérésis en Température : (par exemple 1°)

Ici on suppose que le dispositif de chauffage fonctionne en tout ou rien : soit il fonctionne à pleine puissance, soit il est éteint, la même chose pour le compresseur. C'est le cas de la

plupart des systèmes HVAC. Dans la pratique on fait en sorte que le thermostat fonctionne avec une hystérésis : la température à laquelle il « bascule » n'est pas exactement la même selon que la température est en train de monter ou de descendre. On obtient au cours du temps le fonctionnement suivant : la température est inférieure à 18 °C, le chauffage fonctionne, et la température augmente progressivement ; quand elle atteint 18,5 °C, le contrôleur HVAC arrête le chauffage ; la température redescend progressivement, quand elle atteint 18 °C le chauffage recommence à fonctionner, et le cycle recommence. L'hystérésis permet d'éviter qu'au voisinage de la consigne le chauffage soit très rapidement allumé et éteint. Le résultat est que la température oscille autour de la consigne. Cela convient à une utilisation domestique.

La Plus Faible Température : (par exemple 5°)

Quand la température de la maison atteint cette température le chauffage sera activé automatiquement jusqu'à ce qu'on obtienne une température supérieure à la somme de cette variable et l'hystérésis (6°).

La Plus Haute Température : (par exemple 40°)

Quand la température de la maison atteint cette température le compresseur sera activé automatiquement jusqu'à ce qu'on obtienne une température inférieure à la différence entre cette variable et l'hystérésis (39°).

Niveau Critique du Gaz :

La valeur de cette balise indique à quel niveau du gaz dans l'air on va déclencher une alerte et envoyer des notifications SMS vers la liste de personnes concernées. Après des tests, la valeur préférée est de 400ppm.

Durée du Temporisateur :

Il indique la valeur en millisecondes du temporisateur pour tous les interrupteurs sans-fils.

Géolocalisation : (true ou false)

Cette balise contrôle la fonction de Géolocalisation par Internet. Si elle est active, le système va chercher les heures de lever et coucher du soleil selon les coordonnées géographiques détectées en utilisant l'adresse IP WAN. Ces heures vont être configurés automatiquement comme le temps où le système va éteindre tous les lampes (lever du soleil), et le temps où il va activer la fonction de lumière automatique (coucher de soleil).

Heures d'Activation et Désactivation de Lumière Automatique :

Si l'option de Géolocalisation n'est pas fonctionnelle, l'utilisateur doit spécifier avec ces deux balises, les temps où la fonction de lumière automatique sera désactivée et activée.

Les Adresses RF des Elément Unique :

Il existe un Raspberry Pi (Elément Central), un Contrôleur HVAC et un Nœud Capteur, on peut configurer les adresses RF de ces éléments par la modification ces trois balises respectives dans le fichier de configuration. Il est préférable de choisir l'adresse RF 00 pour l'élément central.

Les Détecteur de Mouvement et leurs Interrupteurs :

On peut ajouter jusqu'au 124 détecteurs de mouvement avec 124 interrupteurs sans-fil, le processus du contrôle gère ces adresses en employant un tableau ou vecteur des adresses RF. Les adresses RF sont regroupées en pairs, la première adresse est pour le détecteur, et la deuxième pour l'interrupteur.

IV.2.2.2 Boucle de Contrôle :

Quand le processus a terminé de charger presque tous les variables du système domotique et d'établir la liaison IPC (un socket TCP) avec le serveur web local, il se résume par basculer vers une boucle du contrôle fermée. Le déroulement de cette boucle est récapitulé dans la figure suivante :

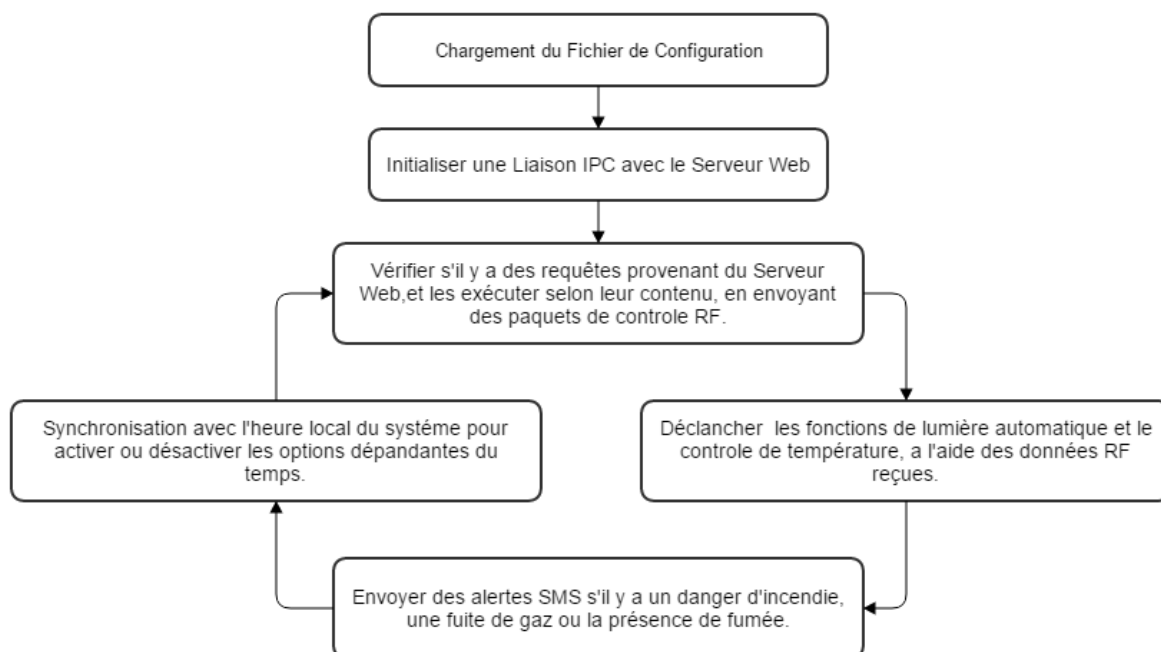


Figure 52. Boucle de Contrôle

IV.3 Le Serveur Web avec Node.js :

Node.js est une plateforme logicielle libre basé sur JavaScript, orientée vers les applications web. Elle utilise la machine virtuelle V8 développé par Google et implémenté sous la licence Open Source du MIT. Node.js contient une bibliothèque de serveur HTTP intégrée, ce qui rend possible de faire tourner un serveur web sans avoir besoin d'un logiciel externe comme Apache, NGINX ou Lighthttpd, et permettant de mieux contrôler la façon dont le serveur web fonctionne. Notez qu'il existe des Frameworks Web comme Express.js qui sont basés sur Node.js. Ils nous permettent d'éviter les tâches répétitives qui sont imposées par la nature bas niveau de Node.js, mais ils restent quand même plus complexes à utiliser que des langages comme PHP.

Express.js fournis aussi des fonctionnalités comme :

- Le support de plusieurs Template Engines HTML ; ce sont des moteurs qui convertissent un fichier texte (un pseudocode) vers un fichier HTML en temps réel.
- Gestion de session et cookies.
- Traitement et l'encapsulation des méthodes http comme GET et POST.
- Contrôle d'erreur.

Les Frameworks ou les modules Node.js principaux qu'on a implémenté sont Express.js v4 et Hogan.js pour le Template Engine. Le serveur conçu est un serveur sécurisé (HTTPS) qui emploie un certificat et une clé générés localement sur le Raspberry Pi. Et pour limiter l'accès, le serveur demande une combinaison de nom d'utilisateur et mot de passe lorsqu'on essaye de l'accéder à partir d'un navigateur web pour avoir la possibilité de contrôler les différents lumières ou appareils de la maison.

La figure suivante visualise l'un des scénarios possibles lorsqu'on implémente ce serveur :

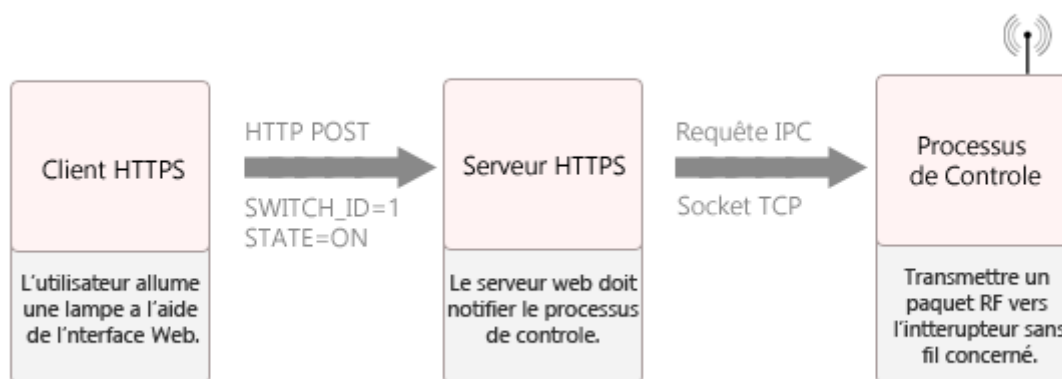


Figure 53. Rôle du Serveur Web

Conclusion

Ce travail illustre le fonctionnement d'un prototype du système domotique, par la conception d'un box de domotique basé sur l'ordinateur mono-carte Raspberry Pi qui interroge et contrôle les différents nœuds Arduino en exploitant la connectivité en radiofréquence des modules NRF24L01. Le marché de la domotique en Algérie est presque inexistant, pour pénétrer, ou précisément, créer ce marché il faut qu'on fournisse des appareils ou des box qui supportent l'infrastructure déjà existante dans la maison algérienne (niveaux de tensions, climatiseurs électriques, installation électrique, installation gaz de ville ...), tout ça va être employé ou vendu avec un coût accessible au consommateur algérien. Notre prototype est le premier pas vers ce but, le commercialiser à grande échelle nécessite quelque optimisation, par exemple ; on peut utiliser la soudure par technique de montage en surface pour réduire la taille des cartes électroniques conçues, le code du processus de contrôle peut aussi être optimisé en implémentant des algorithmes plus robustes comme la gestion de consommation d'énergie, et le contrôle de la température ambiante en exploitant l'intelligence artificielle. Le point fort de ce prototype c'est qu'il est extensible, on peut par exemple contrôler des rideaux électriques ou n'importe quel instrument électrique en rajoutant d'autres interrupteurs sans-fils et modifier le code pour incorporer cette nouvelle fonctionnalité. On peut dire que le point faible de cette conception est la basse autonomie des nœuds communicants, car ils soient toujours branchés avec une alimentation du secteur, mais on peut lutter contre ce problème si on modifie un peu les schémas et les optimise pour une consommation ultra-basse pour avoir la possibilité d'utiliser des piles qui durent plus que six mois.

Avec l'aide des outils, les plateformes, et les bibliothèques Open Source, et les communautés qu'elles supportent, le développement de nouveaux produits libres n'a pas été aussi facile, c'est pour cette raison qu'on va publier ce prototype comme un projet open source sous licence GPLv3, et comme Khalil Gibran a dit :

“Vous ne donnez que peu lorsque vous donnez vos biens. C'est lorsque vous donnez de vous-mêmes que vous donnez réellement.”

Table des Figures :

Figure 1. Le Réseaux Domestique.	12
Figure 2. Schéma d'une liaison ADSL.....	13
Figure 3. Schéma Synoptique Globale de la Maison Communicante Proposée.	27
Figure 4. Une Carte Arduino UNO R3.....	30
Figure 5. Le logiciel Arduino avec l'exemple Blink.....	31
Figure 6. Le module NRF24L01.....	31
Figure 7. Format d'un paquet ShockBurst.....	32
Figure 8. Schéma Block du Nœud Capteur.....	33
Figure 10. Circuit LM335 pour le Shield	34
Figure 9. Circuit LM335 (fiche techn.).....	34
Figure 11. Le MQ2 adapté pour Arduino.....	34
Figure 12. Le MQ2 etSchéma d'implémentation.....	34
Figure 13. Capteur SH1101	35
Figure 14.Schéma NRF24L01 dans les Shields.....	36
Figure 15. Schéma Final de Shield des Capteurs	37
Figure 17. Aide Schéma.....	38
Figure 16. PCB Shield de Capteurs (échelle 1/1).....	38
Figure 18.Le Shield empilé sur l'Arduino UNO.....	38
Figure 19.Le Shield de Capteurs	38
Figure 20. Algorithme Nœud Capteur.....	39
Figure 21. Système HVAC Typique (Amérique du Nord et en Asie)	40
Figure 22. Un exemple de Sortie des câbles HVAC.....	41
Figure 23. Diagramme de Raccordement	42
Figure 24.Schéma Shield de Controle HVAC.....	42
Figure 25. PCB de Shield (echelle 1/1).....	43
Figure 26. Aide Schéma.....	43
Figure 27. Algorithme du Controlleur HVAC	44
Figure 29. PCB Shield (échelle 1/1).....	46
Figure 28. Aide Schéma.....	46
Figure 30. Schéma Shield Detecteur de Mouvement	46
Figure 31. Algorithme de Detection de Mouvement.....	47
Figure 32. Lumière Automatique	49
Figure 33. Circuit Minimal d'un AVR.....	51
Figure 34. Raccordement pour Programmation ISP.....	52
Figure 36. Circuit Implementé.....	53
Figure 35. Relais BS-115C.....	53
Figure 37. Schéma d'Implementation LM7805	53
Figure 38. Circuit LM317.....	54
Figure 40. Schéma Final d'Interrupteur Sans-fil.....	55
Figure 41. Aide Schéma.....	56
Figure 42. PCB Interrupteur Sans-fil.....	56
Figure 43. Photo de L'interrupteur Sans-fil Réalisé	56
Figure 44. Algorithme Interrupteur Sans-fil	57
Figure 45. Schéma Synoptique du Système Domotique	59
Figure 46. Raspberry Pi Modèle B	60
Figure 47. Pins GPIO et Leur Fonctions	61
Figure 48. Piles Protocolaires du Système Domotique	62
Figure 49. Paquet HomeRF	63
Figure 50. Processus de Vérification HMAC	64
Figure 51. L'Encapsulation des Données Chiffrées.....	64
Figure 52. Structure des Données et Commandes HomeRF.....	64
Figure 53. Boucle de Contrôle	64
Figure 54. Rôle du Serveur Web.....	64

ANNEXE : Codes Sources

Les programmes réalisés pour la conception de cette maison intelligente sont publiés comme des logiciels libres sous la licence GPLv3, tous les fichiers requis pour lancer ce system sont uploadé sur le site d'hébergement et de gestion de logiciels GitHub ; le lien permanant :

<https://github.com/SINOVA/Nova-Smart-Home>

Code source du processus de contrôle (C++) :

<https://github.com/SINOVA/Nova-Smart-Home/blob/master/ControlDaemon/NovaHomeDaemon.cpp>

Code source de la bibliothèque HomeRF :

<https://github.com/SINOVA/Nova-Smart-Home/blob/master/ControlDaemon/HomeRF.cpp>

<https://github.com/SINOVA/Nova-Smart-Home/blob/master/ControlDaemon/HomeRF.h>

Code Source Serveur Web (fichier app.js pour l'initialisation du serveur) :

<https://github.com/SINOVA/Nova-Smart-Home/blob/master/HomeControlServer/app.js>

Code Source Serveur Web (fichier routes pour contrôler les requêtes) :

<https://github.com/SINOVA/Nova-Smart-Home/blob/master/HomeControlServer/routes/index.js>

Bibliographie:

- [1] (en) John Krumm (2009). « Ubiquitous Computing Fundamentals »
- [2] (en) AUGUSTO, J. C. (2009). « Ambient Intelligence: Opportunities and consequences of its use in smart classrooms »
- [3] (en) S. Helal, B. Winkler, C. Lee, Y. Kaddoura, L. Ran, C. Giraldo, S. Kuchibhotla, W. Mann (2003). « Enabling location-aware pervasive computing applications for the elderly ».
- [4] (en) S. Helal, W. Mann, H. El-Zabadani, J. King, Y. Kaddoura, E. Jansen (2005). « The Gator Tech Smart House: a programmable pervasive space ».
- [5] (en) E. M. Tapia, S. S. Intille, K. Larson (2004) « Voice controlled smart house ».
- [6] (en) K. Matsuoka (2004) « Aware home understanding life activities »
- [7] (en) Park and all (2007) « Robotic smart house to assist people with movement disabilities »
- [8] (en) K. N. Ha, K. C. Lee, S. Lee (2006) « Development of PIR sensor based indoor detection system for smart home »
- [9] (en) N. M. Barnes, N. H. Edwards, D. A. D. Rose, P. Garner, (1998) « Lifestyle monitoring technology for supported independence »
- [10] (en) M. Mokhtari, « Global Village Initiative »
- [11] (en) M. Chan, E. Campo, D. Estève, « PROSAFE, a multisensory remote monitoring system for the elderly or the handicapped »
- [12] (fr) M. Chan, E. Campo D. Esteve (2004) « L'apprentissage des pratiques et des habitudes de vie, une clef de l'automatisation domotique » – Revue « Signaux »
- [13] (fr) E. Campo, J.P. Scotto Di Rinaldi, D. Estève, N. Bailly, F. Benard (2003) « Développement d'une nouvelle génération de gestionnaire d'énergie auto-configurable pour l'habitat: le concept ERGDOM »
- [14] <http://fr.wikipedia.org/wiki/Arduino>
- [15] « LMx35 Precision Temperature Sensors » <http://www.ti.com/lit/ds/symlink/lm235.pdf>
- [16] « RELATIVE HUMIDITY SENSOR HS 1100 / HS 1101 »
<https://www.parallax.com/sites/default/files/downloads/27920-Humidity-Sensor-Datasheet.pdf>
- [17] [18] [19] https://wiki.xtronics.com/index.php/Thermostat_signals_and_wiring
- [20] <http://www.detecteur-de-presence.info/eclairage-automatique.html>
- [21] <http://www.arduino.cc/en/Tutorial/ArduinoToBreadboard>
- [22] http://fr.wikipedia.org/wiki/Électricité_domestique
- [23] <http://cp.literature.agilent.com/litweb/pdf/5988-6917EN.pdf>
- [24] http://fr.wikipedia.org/wiki/Raspberry_Pi
- [25] http://fr.wikipedia.org/wiki/Keyed-Hash_Message_Authentication_Code