

République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la recherche scientifique
Université 8Mai 1945 – Guelma
Faculté des sciences et de la Technologie
Département d'Electronique et Télécommunications

99



Mémoire de fin d'étude
Pour l'obtention du diplôme de Master Académique

Domaine : Sciences et Technologie
Filière : Electronique
Spécialité : Instrumentation

Commande et surveillance d'une maison intelligente à distance
par microcontrôleur ARDUINO



Présenté par :

Mechighel Ahmed Rami

Sous la direction de : Mr. Boukaache Abdennour

Jun 2018

Remerciement :

18/3808

Je remercie tout d'abord Dieu tout puissant pour la volonté, la santé et la patience, qu'il me a donnée durant toutes ces longues années.

J'exprime mon profondes gratitude à mes parents pour leurs encouragements, leur soutien et pour les sacrifices qu'ils ont enduré.

Je tiens également à exprimer mes vifs remerciements à mon encadreur Monsieur " boukaache abdennour " pour avoir d'abord proposé ce thème.

Je remercie les membres de jury d'examen pour l'honneur qu'ils me font en participant au jugement de ce travail.

Je tiens à remercier vivement tous ceux qui m'ont aidé de près ou de loin à accomplir ce travail.

Je remercie aussi à tous les enseignants du département d'électronique qui ont contribué à mon formation.

Enfin je tiens à exprimer mon reconnaissance à tous mes amis et collègues pour leur soutien moral et matériel...

Dédicace :

Je dédie ce modeste travail à :
Mes très chers parents qui ont tant sacrifié pour moi ;
Mes frères et mes sœurs ;
Mes oncles et mes tantes, cousins et cousines ;
Tous mes amis(es) ;
Toute notre promotion instrumentation 2018 ;
Tous ceux qui ont contribué de près ou de loin à la
réalisation de
ce travail.

SOMMAIRE

Introduction générale	1
CHAPITRE 1 : LA DOMOTIQUE	
1. Historique	2
2. Définition.....	2
3. Usage de la domotique	3
3.1. Le confort	3
3.1.1. Gestion du chauffage	4
3.1.2. L'éclairage intelligent.....	4
3.1.3. Arrosage extérieur.....	4
3.1.4. Ouverture et fermeture des portes.....	4
3.1.5. Stores intérieurs ou extérieurs.....	5
3.1.6. Répondre au téléphone.....	5
3.2. La sécurité	5
3.2.1. Détection de fumée, de gaz.....	5
3.2.2. Détection d'inondation.....	5
3.2.3. Alerte à distance.....	5
3.2.4. Système de surveillance.....	6
3.3. La gestion d'énergie	6
3.4. Le multimédia	7
4. La domotique pour la sante et la protection des personnes	7
5. La domotique et l'environnement.....	7
6. Les appareils domotiques	8
7. Principe de fonctionnement d'un système domotique.....	8
8. Le réseau domotique.....	9
8.1. La domotique par courant porteur.....	9
8.2. Le réseau local LAN (Ethernet) :	10
9. Le marché de la domotique :	12
CHAPITRE 2 : LES MICROCONTROLEURS ET L'ARDUINO	
1. Introduction :	14
2. Un peu d'histoire :	14

3.	Définition d'un microcontrôleur :.....	15
4.	L'architecture d'un microcontrôleur :	16
5.	Domaine d'application des microcontrôleurs :.....	18
5.1.	Systèmes de communication:.....	18
5.2.	Systèmes biomédicales:.....	18
5.3.	La domotique :	19
5.4.	L'automatique industrielle :	19
5.5.	Electronique de consommation:.....	19
6.	Les avantages des microcontrôleurs :.....	19
7.	Les inconvénients des microcontrôleurs :	19
8.	Les types de microcontrôleurs :.....	20
8.1.	Microcontrôleurs pour usage générale :	20
8.2.	Microcontrôleurs pour le traitement de signaux :	20
8.3.	Microcontrôleurs pour communication :.....	20
8.4.	Microcontrôleurs à faible consommation d'énergie :	21
9.	Le système ARDUINO :.....	21
9.1.	Définition :	21
9.2.	Les différents types de L'ARDUINO :	21
9.2.1.	La carte ARDUINO UNO :	21
9.2.2.	La carte Arduino Leonardo :.....	22
9.2.3.	La carte Arduino Mega :.....	22
9.2.4.	La carte Arduino Due :	23
9.2.5.	La carte Arduino Nano :	23
9.2.6.	La carte Arduino Mini-Pro :	24
9.2.7.	La carte Arduino Yun :	24
9.3.	Pour quoi L'ARDUINO UNO ?.....	24
10.	Description de la carte « Arduino Uno » :	26
10.1.	Le Microcontrôleur ATmega328 :	28
10.2.	L'alimentation de la carte :.....	28
10.3.	Les Entrées et les sorties :.....	29
10.4.	La mémoire :.....	30
10.5.	La Communication :	30
11.	Interface de programmation IDE	31

11.1.	Description de l'interface	32
11.2.	Structure du programme :	33
11.2.1.	La Définition des constantes et des variables :	33
11.2.2.	Configuration des entrées/sorties :	33
11.2.3.	Programmation des interactions et comportements :	34
11.2.4.	Les commentaires :	34
7.	Conclusion :	34

CHAPITRE 3: REALISATION DE L'APPLICATION

1.	Introduction	35
2.	Description de projet	35
3.	Les composants du système.....	36
3.1.	Le DIIT 11	36
3.2.	Le capteur de mouvement HC-SR501 (P.I.R)	39
3.3.	Le relais 5v et l'arduino	42
3.4.	Allumage automatique de la lampe à la détection de mouvement.....	44
3.5.	Le Shield Ethernet.....	45
3.5.1.	Branchement d'un Arduino sur un réseau local	46
3.5.2.	Lecture et écriture sur le shield via le réseau internet :	47
4.	Conception de la page WEB.....	48
5.	Schéma de réalisation de projet :	50
6.	L'interface de commande et d'affichage :	52
7.	Conclusion :	53
	Conclusion générale :	54

LISTE DE FIGURES :

Figure 1.1 : Usage de la domotique	4
Figure 1.2 : Principe de fonctionnement d'un système domotique	9
Figure 1.3 : Fonctionnement d'un système client/serveur	12
Figure 1.4 : Parts de marché des différents segments où sont installés les produits domotiques pour le résidentiel en Europe [source]	13
Figure 2.1: L'évolution du microcontrôleur	15
Figure 2.2 : Schéma fonctionnel d'un system a microcontrôleur	16
Figure 2.3 : Schéma simplifié du contenu type d'un microcontrôleur	17
Figure 2.4 : Carte ARDUINO UNO	22
Figure 2.5 . Carte arduino Leonardo	22
Figure 2.6 : Carte arduino Mega	23
Figure 2.7 : Carte arduino Due	23
Figure 2.8 : Carte arduino Nano	24
Figure 2.9 : Carte arduino Mini-Pro	24
Figure 2.10 : Carte arduino Yun	24
Figure 2.11 : Sschéma des ports	26
Figure 2.12 : Le microcontrôleur ATmega328	28
Figure 2.13 : Interface IDE	31
Figure 2.14 : Barre d'actions	32
Figure 2.15: Message d'Erreur	33
Figure 2.16 : Définition des constantes et des variables.	33
Figure 2.17 : Configuration des Entrées/Sorties	34
Figure 2.18 : Programmation des interactions et comportements	34
Figure 3.1 : Principe de fonctionnement de projet domotique	35
Figure 3.2 : Le capteur dht11	36
Figure 3.3 : Brochage d'un capteur DHT11	37
Figure 3.4 : Format des BIT de données	38
Figure 3.5 : Les étapes de communication entre le dht11 et le microcontrôleur	38
Figure 3.6 : Présentation des deux plaquettes à effet pyroélectrique de capteur P.I.R	39
Figure 3.7 : Principe de fonctionnement de capteur PIR	40
Figure 3.8 : Lentille de Fresnel multiple pour le PIR	41
Figure 3.9 : Champ couvert par lentille de Fresnel multiple	41

Figure 3.10 : Détecteur PIR avec et sans la lentille sur le capteur.	42
Figure 3.11 : Les bornes d'un relais 5v	43
Figure 3.12 : La borne normalement ouvert	44
Figure 3.13 : La borne normalement fermée.	44
Figure 3.14 : Activation du relais de l'éclairage par le capteur de mouvement PIR	45
Figure 3.15 : le shield Ethernet	45
Figure 3.16 : Liaison entre arduino et le shield Ethernet	46
Figure 3.17 : Branchement d'un arduino sur un réseau local	47
Figure 3.18 : Configuration du shield Ethernet sur arduino IDE	47
Figure 3.19 : Branchement d'un arduino sur un réseau internet	48
Figure 3.20 . Communication entre le client et l'arduino	49
Figure 3.21 : Présentation de la fonction « <i>client.println</i> »	50
Figure 3.22 : Schéma du projet	51
Figure 3.23 : Relais 5v à 4 chaines	51
Figure 3.24 : Saisie de l'adresse IP dans le navigateur internet	52
Figure 3.25 : Le panneau de contrôle	53

LISTE DES TABLEAUX :

Tableau 2.1 : Caractéristiques techniques de la carte UNO	27
Tableau 3.1 : Caractéristique d'un capteur DHT11	36
Tableau 3.2 : caractéristiques du capteur PIR	42

Introduction générale :

La domotique ne date pas d'aujourd'hui, d'ailleurs c'est un terme plutôt vague, n'importe quel système électronique permettant d'interagir avec le bâtiment pour en améliorer le confort est déjà un système domotique. Il y a quelque temps, la domotique c'était les lumières qui s'allumaient lorsque la nuit tombait mais avec l'évolution des technologies informatiques l'utilisateur a des possibilités de plus en plus de contrôle sur les fonctionnalités de son logement.

En effet, la domotique permet par exemple d'optimiser l'utilisation de l'éclairage, du chauffage afin de réduire notre consommation en énergie. C'est dans ce cadre que se situe notre projet de fin d'études, il a pour objectif de réaliser un système électronique à base d'un microcontrôleur de type ARDUINO pour l'aide à l'administration des équipements domestiques. Ce système permet le pilotage et la surveillance des différents dispositifs disponibles dans la maison ainsi que la description des services fournis et les actions qu'on peut les invoquer.

Ce projet sera composé d'un circuit permettant de connecter plusieurs relais à un réseau local ou internet. Un microcontrôleur «ARDUINO» a les capacités de faire cette liaison grâce au *Shield Ethernet* correctement configuré puis utilisé comme un serveur web. Par conséquent, depuis un ordinateur ou un Smartphone, la page web soumise par notre ARDUINO à travers le réseau affiche la température et l'humidité de la pièce et les boutons des différents relais à piloter par la pression sur un bouton de changement d'état. En gros, il s'agit d'un système de surveillance d'une maison pilotable à distance.

Pour bien expliquer ce travail ; nous avons choisi d'organiser notre mémoire en trois chapitres :

- Le premier chapitre présente les définitions et l'utilisation d'un système domotique.
- Le deuxième chapitre est consacré à la présentation des microcontrôleurs et la description de la carte ARDUINO Uno utilisée dans ce projet.
- Dans le troisième chapitre ; nous présentons une description détaillée de notre système domotique ainsi que les composants utilisés et l'interface WEB créée par notre circuit.

CHAPITRE 1 : LA DOMOTIQUE

1. Historique

Le mot domotique vient de la combinaison du mot latin « *domus* » qui signifie « domicile », et du suffixe « *-tique* », couramment employé pour évoquer le terme des technologies.

Les premiers travaux de domotique sont apparus dans les années 70 avec les problématiques énergétiques dues aux crises pétrolières. Ces crises marquent le début du développement de l'électronique pour les bâtiments. Au départ, la domotique contrôle seulement les prises, l'éclairage et les volets roulants grâce à une télécommande. Au fur et mesure, de nouveaux objets se mettent en réseau comme les thermostats et les alarmes. Mais, c'est véritablement à partir de la fin du 20^e siècle, que la domotique va se démocratiser. Ce développement peut être expliqué par deux raisons:

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

Depuis les années 2000, avec le développement des technologies sans fil comme le wifi ou le Bluetooth, la miniaturisation des composants électroniques, l'avènement des appareils mobiles, l'invasion des écrans tactiles et des télévisions connectées, les ingénieurs peuvent désormais proposer au public des produits – maisons intelligentes ou systèmes domotiques – bien plus puissants et simples d'utilisation [1].

2. Définition

La domotique est l'ensemble des techniques de l'électronique, d'automatisme, de l'informatique et des télécommunications utilisées dans les bâtiments, plus ou moins « interopérables » et permettant de centraliser le contrôle des différents systèmes et sous-systèmes de la maison et de l'entreprise (chauffage, volets roulants, porte de garage, portail

d'entrée, prises électriques,...etc.). La domotique vise à apporter des solutions techniques pour répondre aux besoins de confort (gestion d'énergie, optimisation de l'éclairage et du chauffage), de sécurité (alarme) et de communication (commandes à distance, signaux visuels ou sonores,...etc.) que l'on peut retrouver dans les maisons, les hôtels, les lieux publics,...etc.

Actuellement, la domotique est utilisée pour rendre une maison "*intelligente*". En effet celle-ci consiste en l'automatisation et la centralisation de différents appareils électriques de la maison. Ainsi la domotique est utilisée pour le contrôle de la maison.

Le développement de la domotique est, entre autre, la conséquence de la miniaturisation des systèmes électroniques et informatiques. Le développement des composants électroniques dans les produits domestiques a amélioré leurs performances, rationalisé et réduit les coûts de consommations en énergie de ces équipements. La combinaison de ce processus avec l'apparition sur le marché de services de communication performants est directement liée à l'émergence de systèmes innovants orientés vers la communication et les échanges internes et externes dans les lieux de vie ou de travail. Il s'agit donc d'une démarche visant à apporter plus de confort, de sécurité et de convivialité dans la gestion des bâtiments.

3. Usage de la domotique

Les domaines d'application sont au cœur de la vie quotidienne. Ils peuvent être regroupés selon 4 grands thèmes : la sécurité, le confort, la gestion de l'énergie et le multimédia. [2]

3.1. Le confort

L'accroissement du niveau de confort des habitations a été le premier objectif de la domotique. Les fonctions de commande à distance simples et qui agissent sur différents types d'appareils sont maintenant banalisées. En effet, il est possible d'activer à distance des fonctions qui ont pour but de recréer une ambiance ou un état prédéfinis dans la maison. Il est donc facile d'imaginer un nombre illimité des fonctions qui pourraient faciliter le confort quotidien dans la maison. Tout ceci n'est pas de la science-fiction : c'est tout à fait réalisable aujourd'hui, et cela assez simplement.



Figure 1.1 : Usage de la domotique

Voici quelques exemples d'usage de la domotique pour le confort :

3.1.1. Gestion du chauffage

Programmer en fonction des heures et de l'utilisation des pièces de la maison et à l'aide de thermostats et de capteurs utiles au contrôle intelligent du chauffage. Des thermostats intelligents et connectés assurent un contrôle à distance à partir d'un Smartphone ou un réseau local.

3.1.2. L'éclairage intelligent

Assure l'éclairage adéquat d'un endroit comme l'entrée de l'habitation, le couloir, le garage, le parking lors de son utilisation.

3.1.3. Arrosage extérieur

Croisé avec un détecteur d'humidité et un programmeur, l'arrosage sera optimisé et économique.

3.1.4. Ouverture et fermeture des portes

Tous les types de portes (battantes, basculantes, sectionnelles, coulissantes ...) peuvent être

motorisés. Une télécommande permet de déclencher l'action d'ouverture depuis un Smartphone ou un réseau bien connecté.

3.1.5. Stores intérieurs ou extérieurs

Ils peuvent être commandés automatiquement mais en plus être actionnés à l'aide de détecteurs de vent ou de pluie (fermeture), de soleil (ouverture).

3.1.6. Répondre au téléphone

Décrocher son téléphone à distance et depuis n'importe quelle pièce pour éviter de se déplacer.

3.2. La sécurité

Les capteurs domotiques installés dans un bâtiment peuvent avoir pour mission de détecter les mouvements, la présence d'un individu, la fumée...etc. Toutes ces données permettent d'optimiser la sécurité des maisons et de leurs habitants face aux vols et aux accidents domestiques.

On donne quelques exemples d'usage de la domotique pour La sécurité :

3.2.1. Détection de fumée, de gaz

Assure une protection contre l'incendie ou contre l'asphyxie en déclenchant une sirène ou en envoyant un message à une centrale qui signale l'anomalie par téléphone.

3.2.2. Détection d'inondation

Prévient les dégâts des eaux et peut enclencher la coupure automatique d'arrivée d'eau ce qui peut être utile pour des personnes ayant des pertes de vigilance.

3.2.3. Alerte à distance

En cas d'absence, permet d'être informé en temps réel de ce qui se passe au domicile via un Smartphone, un mail ou un SMS.

3.2.4. Système de surveillance

L'installation d'un système domotique nous permet de garder un œil sur notre maison, pendant nos absences. En effet, il nous est désormais possible de surveiller nos biens et nos animaux domestiques même à distance. Nous pouvons ainsi partir en week-end ou en vacances sereinement. Les caméras numériques, offrent une excellente qualité des images et de la vidéo sur les Smartphone ou sur PC. En plus, les appareils de prise de vues sont paramétrables à distance.

La vidéosurveillance permet alors de diffuser, stocker et relire des séquences vidéo d'une grande clarté issues de caméras, en local ou via Internet. Le stockage des images se fait directement sur disque dur. Avec le matériel domotique, nous pouvons dès lors surveiller notre maison ou notre résidence secondaire à partir de n'importe quel ordinateur équipé d'Internet ou d'un téléphone mobile adapté tout en étant relié à notre système d'alarme existant.

3.3. La gestion d'énergie

L'un des enjeux de la domotique est d'améliorer significativement l'efficacité énergétique des bâtiments. Les maisons dites « intelligentes » ou connectées sont équipées d'un ensemble de technologies innovantes permettant d'améliorer de manière globale leurs performances énergétiques sans perte de confort. Parmi ces technologies, de nombreux automatismes : gestion des volets, de la ventilation, gestion des équipements de chauffage rendent les maisons réactives aux conditions extérieures (climat) et intérieures (usage), l'objectif final étant de réduire les dépenses quotidiennes d'énergie tout en préservant le confort des habitants.

Un système domotique peut diminuer de 40% à 70% la facture d'énergie du domicile sans toucher au confort de vie. [3]

La domotique va influencer la dépense d'énergie sur deux éléments :

- ✓ La suppression de la consommation électrique inutile : L'économie d'énergie, c'est avant tout supprimer une énergie dépensée alors qu'elle n'est pas utilisée. En effet, le gaspillage d'énergie peut être limité avec des produits domotiques afin de ne pas avoir de lampe oubliée à la cave pendant plusieurs jours, un éclairage surdimensionné ou une lampe allumée en plein jour.

- ✓ La distribution de chaleur dans les pièces : Une régulation « intelligente » de l'énergie dans les pièces est essentielle, non seulement pour les factures d'énergie, mais aussi pour le confort de vie. Le réglage des températures est simple et visuel, chaque pièce peut bénéficier d'un réglage qui lui est propre (absence/présence, jour/nuit) et une commande à distance par téléphone est possible afin de mettre la maison sur «confort» ou bien « économie » lors de l'absence des habitants.

3.4. Le multimédia

Les solutions domotiques pour le multimédia permettent de profiter de différents médias simultanément dans différentes pièces de la maison. Les gens pourront ainsi écouter la chaîne Hi-fi dans le salon en surfant sur internet et imprimer une recette sur l'imprimante du bureau. Pendant ce temps leurs enfants profiteront dans leur chambre de leur radio préférée tout en jouant en réseau sur leur ordinateur respectif.

4. La domotique pour la santé et la protection des personnes

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

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

5. La domotique et l'environnement

En dehors des économies d'énergie évidentes que l'on peut réaliser en automatisant la programmation des systèmes de chauffage, la domotique est un formidable moyen pour éviter de dépenser de l'argent et surtout moins consommer pour protéger l'environnement. Les nouvelles installations peuvent même donner une estimation des gains énergétiques effectués et des rejets en CO2 évités dans l'environnement.

6. Les appareils domotiques

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

7. Principe de fonctionnement d'un système domotique

La domotique est utilisée pour le contrôle de la maison. Ce contrôle ne s'exerce pas n'importe comment. En effet le contrôle d'un appareil se fait suite à l'analyse de certaine donnée par exemple la force du vent extérieure. En effet un capteur est placé à l'extérieur et si le vent souffle trop fort les stores peuvent s'ouvrir. Ce principe est adapté à différents outils électriques de la maison. De plus, toutes ces informations sont centralisées dans un boîtier permettant à l'utilisateur de vérifier les différentes données (température de la maison, luminosité, force du vent...), d'entrer ses propres paramètres (par exemple, entrer une température idéale pour la maison, et si la température relevée est plus faible que celle voulue par l'utilisateur, les radiateurs se déclenchent).

Pour cela différents outils sont utilisés. Tout d'abord il faut utiliser différents types de capteurs (pour la température, la lumière, le vent ...) Il faut ensuite pouvoir traiter ces données. On utilise pour cela un ordinateur ou un système similaire (microcontrôleur) pour traiter les données et ainsi renvoyer les informations enregistrées par les capteurs afin d'agir en conséquence sur les appareils électriques (actionneurs).

La figure ci-dessous explique le principe de fonctionnement de la domotique

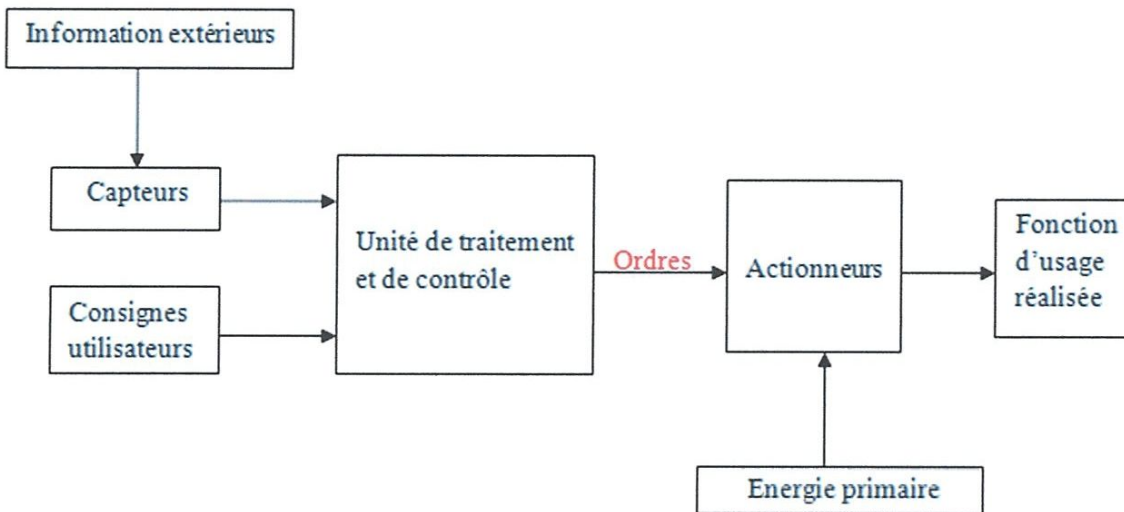


Figure 1.2 : Principe de fonctionnement d'un système domotique

8. Le réseau domotique

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

Il existe deux types de réseau domotique :

8.1. La domotique par courant porteur

Le réseau électrique CPL (Courant Porteur en Ligne) fait circuler les données à travers les câbles électriques existants. C'est une technologie permettant le transfert d'informations numériques en passant par les lignes électriques. De ce fait, il s'agit d'une alternative aux traditionnels câbles.

Le principe des CPL consiste à superposer au courant électrique de 50 Hz un signal à plus haute fréquence et de faible énergie. Ce deuxième signal se propage sur l'installation électrique et peut être reçu et décodé à distance. Ainsi le signal CPL est reçu par tout récepteur CPL qui se trouve sur le même réseau électrique.

On classe traditionnellement les CPL en deux catégories en fonction du débit offert:

- **Les CPL à haut débit :** utilisent des modulations multi porteuses de type OFDM dans la bande (bande1, 6 à 30MHz). La technique CPL haut-débit permet de faire passer des données informatiques sur le réseau électrique, et ainsi étendre un réseau local existant ou partager un accès Internet existant via les prises électriques grâce à la mise en place de boîtiers spécifiques. Dans l'état actuel de la technique, les débits atteints sont compris entre 14 Mbit/s et 200 Mbit/s.
- **Les CPL à bas débit :** utilisent des techniques de modulations assez simples, par exemple quelques porteuses (mais une seule à la fois) en modulation de fréquence. Les bandes des fréquences utilisées sont comprises entre 9 et 150 kHz. On utilise La technique CPL bas-débit pour mettre en réseau des appareils électriques dans des domaines aussi divers que les machines à laver, les volets roulants, le chauffage, ou aussi pour des applications de gestion du réseau de transport et de distribution électrique ou encore de télérelève des compteurs électriques. Les débits sont typiquement de 2,4 à 20 kbit/s.

En haut-débit comme en bas débit, la communication est soumise à des bruits et à de fortes atténuations. Il est donc nécessaire de mettre en œuvre de la redondance, par exemple sous la forme de codes correcteurs d'erreurs.

Un coupleur intégré en entrée des récepteurs CPL élimine les composantes basses fréquences avant le traitement du signal. Le modem transforme un flux de bits en signal analogique pour l'émission et inversement en réception, celui-ci inclut les fonctions d'ajout de la redondance et de reconstitution du flux de bits original ou correction d'erreur. [4]

8.2. Le réseau local LAN (Ethernet) :

C'est le réseau le plus utilisé dans les systèmes domotiques. Le mot LAN signifie Local Area Network (en français Réseau Local). Il s'agit d'un ensemble des machines appartenant à une même organisation et reliés entre eux dans une petite aire géographique par un réseau, souvent à l'aide d'une même technologie nommée « Ethernet »

L'Ethernet (aussi connu sous le nom de norme IEEE 802.3) est un standard de transmission de données pour un réseau local basé sur le principe suivant : tous les machines de réseau Ethernet sont connectées à une même ligne de communication constituée de câble cylindrique.

Toutes les machines d'un réseau Ethernet sont reliées à une même ligne de transmission, et la communication se fait à l'aide d'un protocole appelé CSMA/CD (*Carrier Sense Multiple Access with Collision Detection*) ce qui signifie qu'il s'agit d'un protocole d'accès multiple avec surveillance de porteuse (*Carrier Sense*) et détection de collision. [5]

Avec ce protocole toutes les appareils sont autorisés à émettre sur la ligne à n'importe quel moment et sans notion de priorité entre les appareils. Cette communication se fait de façon simple :

- ✓ Chaque machine vérifie qu'il n'y a aucune communication sur la ligne avant d'émettre
- ✓ Si deux machines émettent simultanément, alors il y a collision (c'est-à-dire que plusieurs trames de données se trouvent sur la ligne au même moment)
- ✓ Les deux machines interrompent leur communication et attendent un délai aléatoire, puis la première ayant passé ce délai peut alors réémettre.

Ce principe est basé sur plusieurs contraintes :

- ✓ Les paquets de données doivent avoir une taille maximale.
- ✓ Il doit y avoir un temps d'attente entre deux transmissions.

Le temps d'attente varie selon la fréquence des collisions :

- ✓ Après la première collision une machine attend une unité de temps
- ✓ Après la seconde collision la machine attend deux unités de temps
- ✓ Après la troisième collision la machine attend quatre unités de temps
- ✓ ... avec bien entendu un petit temps supplémentaire aléatoire

Le mode de fonctionnement le plus utilisé pour le réseau local dans les systèmes domotique, c'est l'environnement « client/serveur ». Ce mode signifie que des machines clientes (des machines faisant partie du réseau) contactent un serveur, une machine généralement très puissante en termes de capacités d'entrée-sortie (l'unité de contrôle), qui leur fournit des services. Ces services sont des programmes fournissant des données telles que l'heure, des fichiers, une connexion, etc.

Un système client/serveur fonctionne selon le schéma suivant :

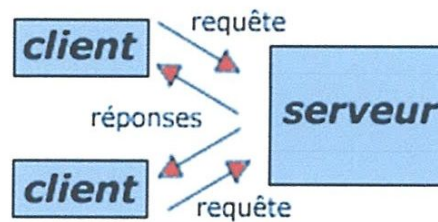


Figure 1.3 : Fonctionnement d'un système client/serveur

- Le client émet une requête vers le serveur grâce à son adresse IP et le port, qui désigne un service particulier du serveur.
- Le serveur reçoit la demande et répond à l'aide de l'adresse de la machine cliente et son port. [6]

9. Le marché de la domotique :

Le marché de la domotique rassemble des acteurs provenant d'univers différents en raison d'une grande diversité d'équipements domotiques. Les acteurs historiques du secteur, à savoir les fabricants de matériels électriques (Schneider, Hager, Legrand, Delta Dore, Theben, ABB) côtoient désormais les opérateurs télécoms et les sociétés informatiques qui proposent des offres permettant d'automatiser une partie des équipements du logement. Ces solutions dites « partielles » sont confrontées aujourd'hui à des solutions globales, totalement intégrées au bâti, et proposées par les fabricants de matériels électriques.

En effet, comme représenté par le schéma ci-dessous, le haut de gamme et le moyen de gamme représentaient 60 % des ventes en équipement domotique.

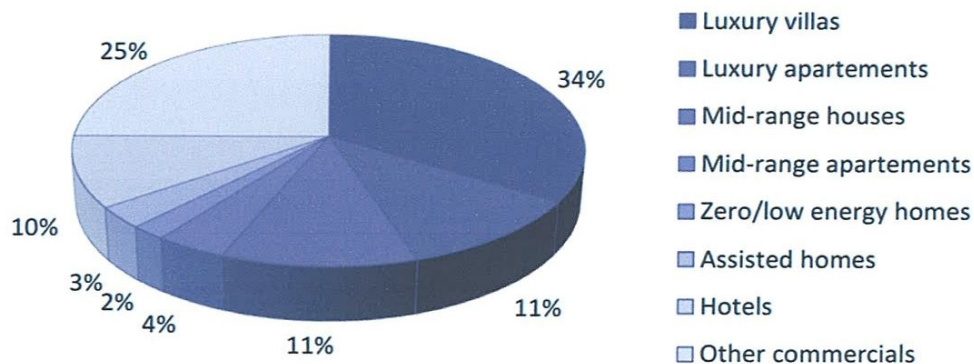


Figure 1.4 : Parts de marché des différents segments où sont installés les produits domotiques pour le résidentiel en Europe [source]

La croissance attendue pour les prochaines années est en forte augmentation grâce à la conjugaison de plusieurs facteurs : baisse des prix permettant de toucher significativement le moyen de gamme, développement d'outils performants de configuration facilitant la mise en œuvre par les installateurs non spécialisés. Egalement, la généralisation du haut débit ouvre la voie à une offre plus attrayante avec la transmission d'images et l'utilisation d'outils ludiques sur Smartphones et Tablettes. Toutes les conditions sont donc réunies pour faire du marché de la domotique un marché de masse. De plus, avec le vieillissement de la population, le maintien des personnes âgées à domicile génère un besoin d'automatisme et de communication, et joue en faveur d'une domotique généralisée. La tendance des prix devrait continuer à la baisse et, en conséquence accélérer la démocratisation de la domotique. [7]

CHAPITRE 2 : LES MICROCONTROLEURS ET L'ARDUINO

1. Introduction :

Le microcontrôleur est au microprocesseur ce qu'est le singe à l'homme : un cousin. Il est un peu plus bête mais en réalité beaucoup plus utilisé ; il existe tout autour de nous et nous simplifie bien la tâche lorsqu'il est dans notre téléphone portable, notre voiture (airbag, tableau de bord...etc.) ou notre chaîne HI-FI.

Le microcontrôleur est taillé pour l'embarqué. Il est beaucoup plus facile à mettre en œuvre qu'un processeur et les dernières versions atteignent des performances remarquables. Il est conçu pour se suffire à lui-même ; un pentium a besoin de périphériques pour pouvoir fonctionner contrairement au microcontrôleur qui lui possède sa propre RAM, sa ROM et le programme pour exécuter les tâches, tel un système d'exploitation.

Nous voyons donc que ses possibilités sont quasi infinies et ses domaines d'application sont très larges ; c'est pourquoi il est tant apprécié des électroniciens amateurs dans plusieurs applications.

2. Un peu d'histoire :

Un petit retour en arrière, dans les années 40. Le premier ordinateur fut l'ENIAC, apparu en 1945. Il coûtait 10 millions de dollars, contenait 20 000 tubes, pesait 30 tonnes, consommait 0,2 Méga Watts et occupait une pièce de 150m². Chacune des unités fonctionnelles tenait sur plusieurs cartes électroniques.

L'évolution des technologies a permis la miniaturisation et l'intégration de plus en plus de fonctions sur une surface de plus en plus petite. Tout d'abord, l'intégration est faite au niveau des cartes électroniques puis au niveau des circuits intégrés.

L'avènement des transistors (1958), des circuits intégrés (1968) et des circuits intégrés à haute densité (1978) a permis de construire des ordinateurs de plus en plus petits et de plus en plus puissants. Chacune des intégrations laisse leurs traces dans les différents noms que l'on trouve aujourd'hui : mini-ordinateur, microprocesseur, micro-ordinateur, microcontrôleur...etc.

Le diagramme ci-dessous montre les grandes étapes de l'évolution. Chaque étape a été marquée par un processeur différent.

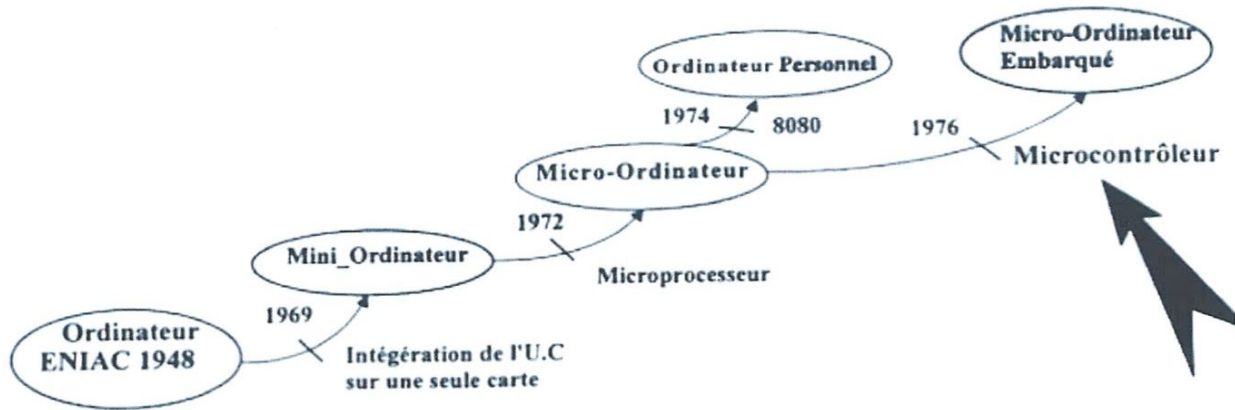


Figure 2.1: L'évolution du microcontrôleur

Voici les grandes étapes d'évolution :

- **Mini-ordinateur** : première intégration, cet ordinateur contient maintenant une unité Centrale de traitement sur une seule carte.
- **Microprocesseur** : deuxième intégration, l'unité centrale est contenue dans un seul circuit, c'est le CPU. Ce fut le 4004 en 1971, le 8008 en 1972 puis le 8080 en 1974. Ce dernier eut un tel succès qu'il est considéré comme le père de nos microprocesseurs et microcontrôleurs d'aujourd'hui.
- **Micro-ordinateur** : c'est le nom qui désignera désormais un ordinateur conçu autour d'un microprocesseur.
- **Microcontrôleur** : troisième intégration, l'ensemble des 3 unités (centrale, mémoires et périphériques) sont maintenant contenues dans un seul circuit. Ce fut le 8048 en 1976 puis le 8051 en 1980.
- **Micro-ordinateur embarqué** : C'est un ordinateur spécialisé et conçu autour d'un microcontrôleur. Il est capable de réagir à des événements plus rapides. [8]

3. Définition d'un microcontrôleur :

Un microcontrôleur, est un composant électronique qui rassemble tous les éléments d'un "mini-ordinateur" et qui se présente sous la forme d'un circuit intégré. Il permet de réaliser des systèmes et montages électroniques programmés. Cela veut dire que l'on pourra, avec le même montage, réaliser des fonctions très différentes qui dépendront du programme qui aura été programmé dans le microprocesseur.

Les microcontrôleurs améliorent donc l'intégration et le coût (lié à la conception et à la réalisation) d'un système à base de microprocesseur en rassemblant les éléments essentiels d'un tel système dans un seul circuit intégré. On parle alors de "système sur une puce" (en anglais : "System On chip").

Le microcontrôleur possède des entrées /sorties pour pouvoir interagir avec l'extérieur, par exemple avec des capteurs ou des actionneurs.

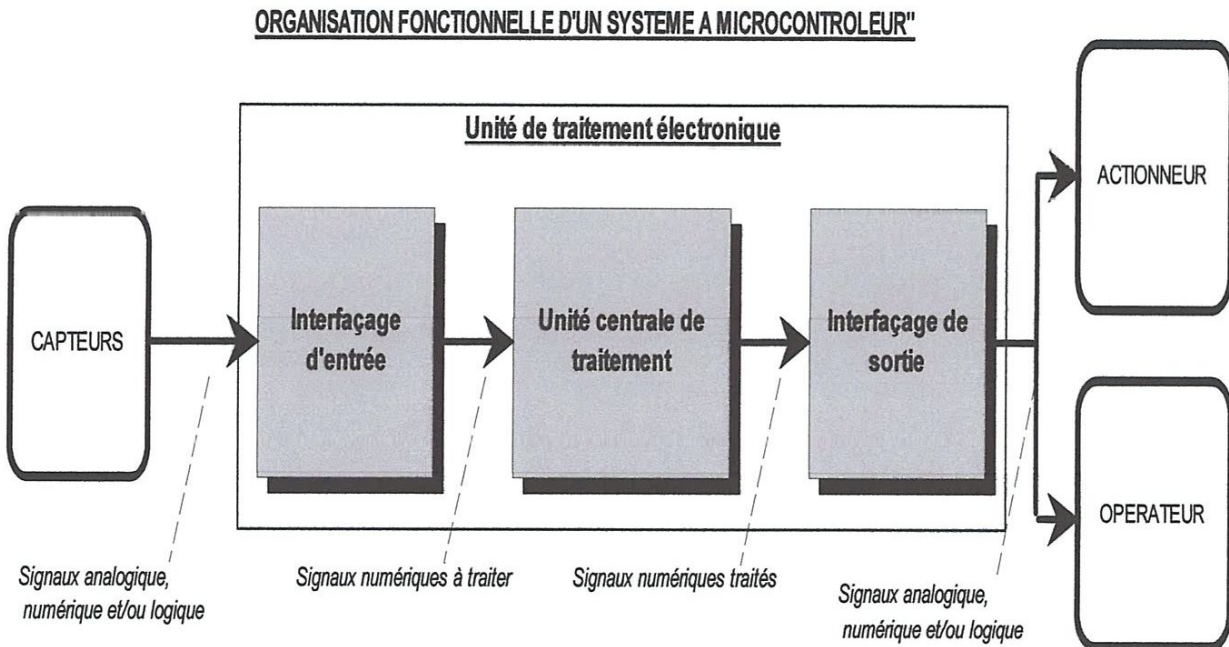


Figure 2.2 : Schéma fonctionnel d'un system a microcontrôleur

4. L'architecture d'un microcontrôleur :

Il existe plusieurs familles de microcontrôleurs, se différenciant par la vitesse de leur processeur et par le nombre de périphériques qui les composent. Toutes ces familles ont un point commun c'est de réunir tous les éléments essentiels d'une structure à base de microprocesseur sur une même puce.

Alors, Le microcontrôleur est un dérivé du microprocesseur. Sa structure est celle des systèmes à base de microprocesseurs. Il est donc composé d'une unité centrale de traitement et de commande (équivalente au microprocesseur), de mémoires et de ports d'entrées/sorties.

En plus de cette configuration minimale, les microcontrôleurs sont dotés d'un ou plusieurs systèmes de comptage (TIMER) et quelques-uns sont dotés d'un convertisseur analogique/numérique (CAN) intégré.

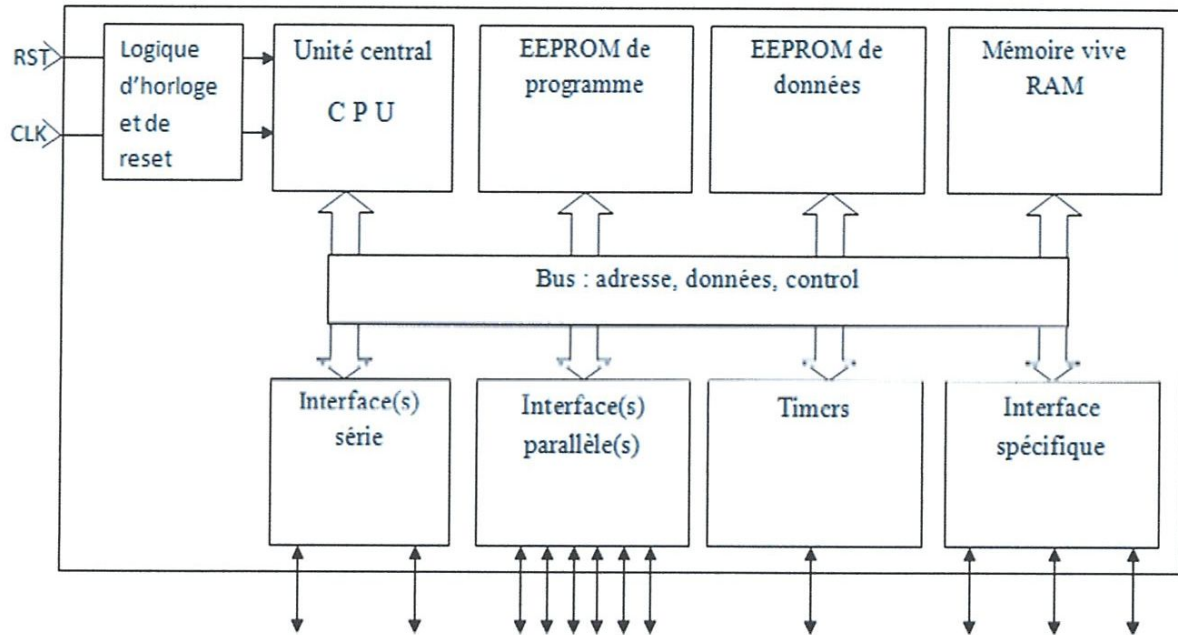


Figure 2.3 : Schéma simplifié du contenu type d'un microcontrôleur

Voici généralement ce que l'on trouve à l'intérieur d'un microcontrôleur:

- **Un processeur (CPU)** : c'est une unité centrale de traitement et de commande.
- **La mémoire vive (RAM)** : *Random Access Memory* ou mémoire à accès aléatoire. Cette mémoire perd l'information lorsqu'elle n'est plus alimentée. Pour pouvoir travailler normalement le microcontrôleur doit pouvoir souvent stocker des données temporaires quelque part dans la RAM.
- **Les bus** : pour écrire une donnée en mémoire, l'UC doit d'abord spécifier l'adresse de la mémoire, puis envoyer la donnée, et en dernier lieu, envoyer un signal qui validera la mémorisation de la donnée. Tous ces signaux seront véhiculés par des « bus », ensembles de « conducteurs », sur lesquels viennent se brancher les mémoires et les interfaces des périphériques.

On distingue 3 types de bus :

- bus d'adresse.
- bus de données.
- bus de contrôle (pour les signaux de service).

- **La mémoire de donnée (EEPROM)** : pour stocker les données.
- **La mémoire programme (ROM, OTPROM, UVPROM ou EEPROM)** : pour stocker le programme.
- **Les interfaces parallèles** pour la connexion des entrées / sorties, ces circuits d'interfaces peuvent piloter des matériels très différents, (moteur pas à pas, afficheur LCD, thermistance, communication avec PC ou d'autres microcontrôleurs, etc.).
- **Les interfaces séries** (synchrone ou asynchrone) pour dialoguer avec d'autres unités.
- **Les Timers** pour générer ou mesurer des signaux avec une grande précision temporelle. (Compteurs d'impulsions d'horloge interne ou d'événements externes). [9]

5. Domaine d'application des microcontrôleurs :

Le microcontrôleur apparaît comme un système extrêmement complet et performant, capable d'accomplir une ou plusieurs tâches très spécifiques, pour lesquelles il a été programmé. Ces tâches peuvent être très diverses, si bien qu'on trouve aujourd'hui des microcontrôleurs presque partout: dans les appareils électroménagers (réfrigérateurs, fours à micro-ondes...), les téléviseurs et magnétoscopes, les téléphones sans fil, les périphériques informatiques (imprimantes, scanners...), les voitures (airbags, climatisation, ordinateur de bord, alarme...), les avions et vaisseaux spatiaux, les appareils de mesure ou de contrôle des processus industriels,...etc.

Voici quelques domaines d'utilisation des microcontrôleurs :

5.1. Systèmes de communication:

Les microcontrôleurs sont souvent utilisés dans les téléphones portables et les téléphones fixes, mais ils se retrouvent plutôt dans les Smartphones et les PDA (*Personal Digital Assistant*).

5.2. Systèmes biomédicales:

Les instruments de mesure (par exemple mesure de la glycémie) ou les organes artificiels,...etc.

5.3. La domotique :

Les systèmes pour gérer la sécurité dans les moyens de transport (par exemple : les passages à niveau « Ascenseur »), dans les bâtiments (par exemple: alarme incendie, effractions) etc.

5.4. L'automatique industrielle :

Dans ce domaine l'utilisation des microcontrôleurs devient par les automates programmables industriels : API (un modèle de microcontrôleur spécialisé pour l'industrie).

5.5. Electronique de consommation:

Appareil HI-FI, TV, vidéo, magnétoscope, vidéoprojecteurs, télécommande,...etc.

6. Les avantages des microcontrôleurs :

- Diminution de l'encombrement du matériel et du circuit imprimé.
- Simplification du tracé du circuit imprimé.
- Augmentation de la fiabilité du système (Nombre de composants diminué).
- Intégration en technologie MOS, CMOS, ou HCMOS et diminution de la consommation.
- Le microcontrôleur contribue à réduire les coûts à plusieurs niveaux:
 - Moins cher que les composants qu'il remplace.
 - Diminution des coûts de main d'œuvre (conception et montage)
- Environnement de programmation et de simulation évolués.

7. Les inconvénients des microcontrôleurs :

- le microcontrôleur est souvent surdimensionné devant les besoins de l'application.
- Investissement dans les outils de développement.
- Écrire les programmes, les tester et tester leur mise en place sur le matériel qui entoure le microcontrôleur.
- Incompatibilité possible des outils de développement pour des microcontrôleurs de même marque.

8. Les types de microcontrôleurs :

8.1. Microcontrôleurs pour usage générale :

Les fabricants de microcontrôleurs, comme Atmel et Microchip, offrent des familles de microcontrôleurs à usage général. Dans ce types d'appareils, il existe souvent différentes configurations disponibles, tels que 8 bits, 16 bits et 32 bits. La taille de mot se réfère à la taille des nombres binaires qui peuvent être traitées par le microcontrôleur. En outre, les dispositifs à usage général sont en mémoire et configurations périphériques différents. Les microcontrôleurs d'usage général ont normalement un ensemble de fonctionnalités qui seraient utiles dans une variété d'applications et peuvent être conçus en produits tels que les appareils ménagers et les produits de consommation et la domotique.

8.2. Microcontrôleurs pour le traitement de signaux :

Comme la vitesse et la puissance de traitement des microcontrôleurs ont augmenté, les fabricants ont combiné les caractéristiques d'un microcontrôleur avec des fonctionnalités d'un processeur de signal numérique (ou DSP). Par exemple, MICROCHIP offre la gamme de produits dsPIC qu'ils appellent les contrôleurs de signaux numériques (ou DSC), qui présentent des caractéristiques de microcontrôleurs et des fonctionnalités de DSP dans un seul noyau. Les microcontrôleurs de traitement du signal combinent généralement la mémoire intégrée et les jeux d'instructions simples de microcontrôleurs avec le circuit de traitement arithmétiques de signaux efficaces trouvées sur DSP. Les microcontrôleurs de traitement du signal sont utilisés dans des applications de traitement du son, des images et de la vidéo.

8.3. Microcontrôleurs pour communication :

Parmi les nombreux types de circuits périphériques qui peuvent être construits pour les microcontrôleurs sont des interfaces de communication, et en particulier des circuits d'interface sans fil. Les microcontrôleurs conçus pour les applications de communication comprennent des sections de traitement des protocoles de communication tels que le Wi-Fi, Bluetooth, ZigBee, bus CAN, infrarouge, USB et Ethernet. Ils peuvent être trouvés dans les dispositifs sans fil et des dispositifs de réseau câblé tels que ceux dans les applications automobiles.

8.4. Microcontrôleurs à faible consommation d'énergie :

La nécessité d'un processeur dans les appareils alimentés par batterie a stimulé le développement de microcontrôleurs qui attirent peu de puissance, mais offrent la vitesse de traitement nécessaire dans les petits appareils de consommation. Dans certains cas, les microcontrôleurs servent de dispositifs de gestion de batterie pour surveiller la charge et la décharge des batteries, telles que des cellules au lithium-ion, dans des dispositifs électroniques portables. D'autres microcontrôleurs à faible puissance sont conçus pour être toujours sous tension et comprennent typiquement un mode actif pour le traitement et un mode de veille pour surveiller un signal tout en tirant une infime quantité de courant.

9. Le système ARDUINO :

9.1. Définition :

ARDUINO est une technologie qui fait associer un environnement de développement avec un circuit électronique à base d'un microcontrôleur AVR de la société Atmel (constructeur de microcontrôleurs), distribué sous la licence du matériel libre (les schémas électrique sont disponible gratuitement). La carte électronique ARDUINO 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.

9.2. Les différents types de L'ARDUINO :

Il existe plusieurs types de circuits ARDUINO. La différence entre ces types sont en termes de nombre d'entrées et sorties, ainsi que le microcontrôleur la vitesse du processeur trouvé à l'intérieur.

Voici quelques types des cartes arduino :

9.2.1. La carte ARDUINO UNO :

La carte arduino UNO (*utilisé pour notre projet*) est la carte idéale pour découvrir l'environnement ARDUINO. Elle permet à tout débutant de se lancer dans tous ses premiers petits projets. Comme c'est la carte la plus utilisée, il est très facile de se référer aux tutoriels très nombreux sur le net et ainsi de ne pas rester seul dans son exploration.

Sa simplicité devient par contre un handicap lorsqu'il s'agit de multiplier les périphériques, de manipuler des algorithmes lourds ou d'interagir avec les systèmes Android pour lesquels d'autres cartes arduino sont plus adaptées.

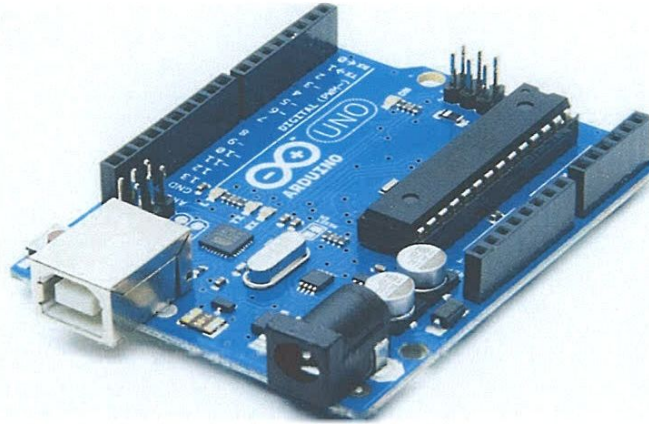


Figure 2.4 : Carte ARDUINO UNO

9.2.2. La carte Arduino Leonardo :

C'est la carte qui est prévue pour succéder à la carte Arduino Uno en présentant des caractéristiques équivalentes mais une ergonomie revue et une stabilité plus éprouvée. Sa diffusion moins importante limite le support utilisateur disponible sur le net.

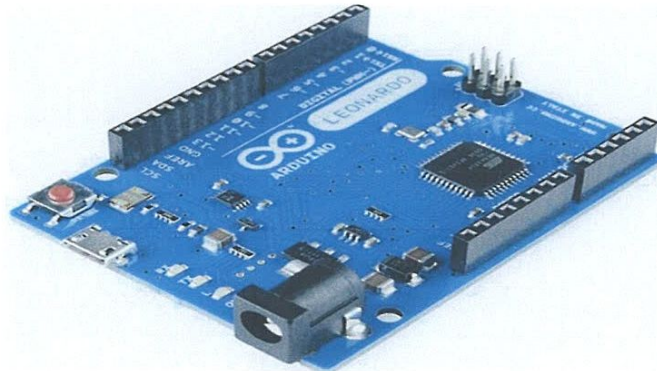


Figure 2.5 : Carte arduino Leonardo

9.2.3. La carte Arduino Mega :

La carte Arduino Mega est la carte la plus diffusée après la carte Arduino Uno. Elle offre un nombre d'entrées/sorties beaucoup plus important (54 contre 14), un processeur plus

puissant doté d'une mémoire plus large qui permet d'exploiter des algorithmes plus complexes.

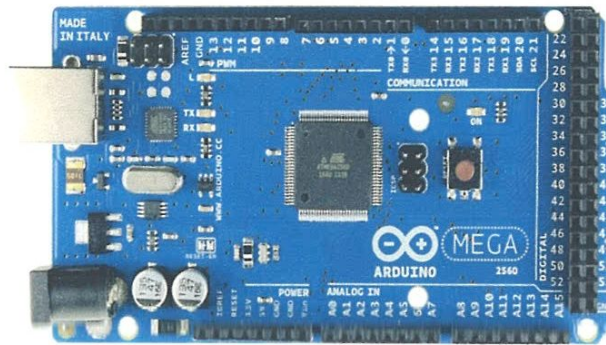


Figure 2.6 : Carte arduino Mega

9.2.4. La carte Arduino Due :

La carte Arduino Due est une évolution de la carte Arduino Mega et offre des performances réputées 3 fois supérieures. Elle permet de manipuler rapidement des algorithmes lourds particulièrement utiles dans le monde de la robotique par exemple.

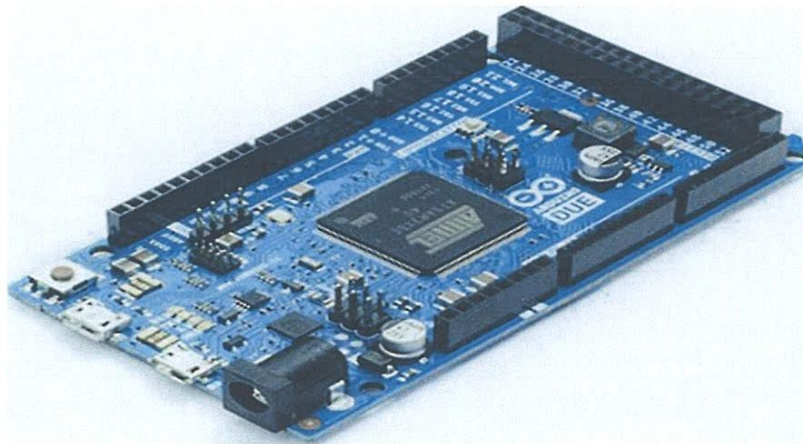


Figure 2.7 : Carte arduino Due

9.2.5. La carte Arduino Nano :

La carte Arduino nano n'est ni plus ni moins qu'une carte Arduino uno miniaturisée. Sa taille et son poids réduits la destinent à une utilisation dans des espaces réduits (en textile par

exemple) ou dans des applications de robotique ou de modélisme pour lesquels le poids et la taille sont des facteurs déterminant (hélicoptères, drones...).

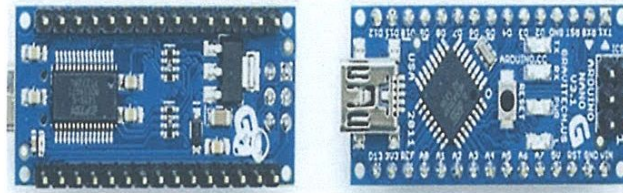


Figure 2.8 : Carte arduino Nano

9.2.6. La carte Arduino Mini-Pro :

La carte arduino Mini Pro est une carte Arduino Uno simplifiée à l'extrême permettant néanmoins de piloter de petits projets ou certains éléments d'un projet. Cette carte n'intègre pas de port USB ce qui rends sa connectivité délicate.

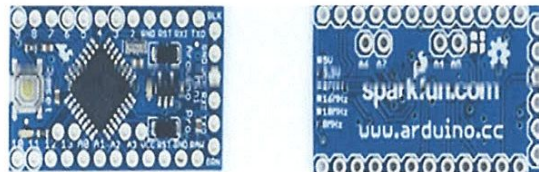


Figure 2.9 : Carte arduino Mini-Pro

9.2.7. La carte Arduino Yun :

La carte Arduino Yun, récemment proposée par Arduino, est conçue pour contrer les avantages de la carte Raspberry. Elle est un dérivé de la carte Leonardo et a pour objectif de combiner la puissance de Linux avec la facilité d'utilisation d'une carte Arduino. Elle est également la première carte Arduino à être dotée nativement d'un wifi intégré.



Figure 2.10 : Carte arduino Yun

Parmi cette riche collection des cartes, nous avons choisi une carte Arduino UNO pour notre projet. L'intérêt principal de cette carte est de faciliter la mise en œuvre d'une telle commande.

9.3. Pour quoi L'ARDUINO UNO ?

Par sa simplicité d'utilisation, Arduino est utilisé dans beaucoup d'applications comme l'électronique industrielle et embarquée ou la domotique. Il y a de nombreuses cartes électroniques qui possèdent des plateformes basées sur des microcontrôleurs disponibles pour l'électronique programmée. Tous ces outils prennent en charge les détails compliqués de la programmation et les intègrent dans une présentation facile à utiliser. De la même façon, le système Arduino simplifie la façon de travailler avec les microcontrôleurs tout en offrant aux personnes intéressées plusieurs avantages cités comme suit:

- **Le prix (réduits) :** les cartes Arduino sont relativement peu coûteuses comparativement aux autres plates-formes. La moins chère des versions du module Arduino peut être assemblée à la main, (les cartes Arduino pré-assemblées coûtent moins de 2500 Dinars).
- **Multi plateforme :** le logiciel Arduino, écrit en JAVA, tourne sous les systèmes d'exploitation Windows, Macintosh et Linux. La plupart des systèmes à microcontrôleurs sont limités à Windows.
- **Un environnement de programmation clair et simple :** l'environnement de programmation Arduino (le logiciel Arduino IDE) est facile à utiliser pour les débutants, tout en étant assez flexible pour que les utilisateurs avancés puissent en tirer profit également.
- **Matériel Open source et extensible :** les cartes Arduino sont basées sur les Microcontrôleurs Atmel ATMEGA8, ATMEGA168, ATMEGA328, les schémas des modules sont publiés sous licence, et les concepteurs des circuits expérimentés peuvent réaliser leur propre version des cartes Arduino, en les complétant et en les améliorant. Même les utilisateurs relativement inexpérimentés peuvent fabriquer la version sur plaque d'essai de la carte Arduino, dont le but est de comprendre comment elle fonctionne pour économiser le coût.
- **Logiciel Open Source et extensible :** le logiciel Arduino et le langage Arduino sont publiés sous licence «*open source*», disponible pour être complété par des programmeurs expérimentés. Le logiciel de programmation des modules Arduino est

une application JAVA multi plateformes (fonctionnant sur tout système d'exploitation), servant d'éditeur de code et de compilateur, et qui peut transférer le programme au travers de la liaison série (RS232, Bluetooth ou USB selon le module).

- **Une bibliothèque riche :** Nombreuses librairies disponibles avec diverses fonctions implémentées. Ils existent aussi des nombreux conseils, tutoriaux et exemples en ligne.
- **Existence de « shield » (boucliers en français) :** ce sont des cartes supplémentaires qui se connectent sur le module Arduino pour augmenter les possibilités comme par exemple : afficheur graphique couleur, interface Ethernet, GPS,...etc.

10. Description de la carte « Arduino Uno » :

La carte Arduino Uno est basée sur un microcontrôleur ATmega328 cadencé à 16 MHz. C'est la plus récente et la plus économique carte à microcontrôleur d'Arduino. Des connecteurs situés sur les bords extérieurs du circuit imprimé permettent d'enficher une série de modules complémentaires. Elle peut se programmer avec le logiciel Arduino « IDE ». Le contrôleur ATmega328 contient un *bootloader* qui permet de modifier le programme sans passer par un programmeur. [10]. La carte Uno est dotée de :

- de 14 entrées/sorties (dont 6 fournissent la sortie PWM).
- 6 entrées analogiques.
- un cristal à 16 MHz.
- une connexion USB.
- une prise jack d'alimentation.
- un en-tête ICSP.
- Une fonction reset.

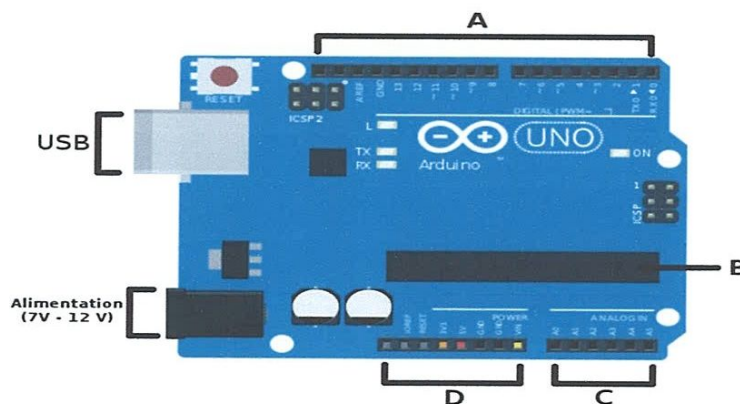


Figure 2.11 : Schéma des ports

- A: ce sont les pattes (pin) dites digitales (0,1) ou "tout ou rien" ; elles offrent en sortie du 5 V et acceptent en entrée du 5 V sur le même principe. La patte 13 est équipée d'une résistance.
- B: le microcontrôleur.
- C: ce sont les pattes dites analogiques, valeur entre 0 V et 5 V.
- D: les différentes pattes d'alimentation :
 - Rouge* : sortie 5 V(+)
 - Orange* : sortie 3.3 V(+)
 - Noire* : les masses (-)
 - Jaune* : entrée reliée à l'alimentation (7 V-12 V)
- L'USB sert pour l'alimentation de la carte et le transfert des programmes qu'on souhaite charger dans le microcontrôleur.
- L'alimentation 7 V-12 V servira à alimenter la carte lorsqu'elle est en production (non reliée à l'ordinateur).

Tableau 2.1 : Caractéristiques techniques de la carte UNO

Microcontrôleur	ATmega328P
Tension de fonctionnement	5V
Tension d'entrée (recommandé)	7-12V
Tension d'entrée (limite)	6-20V
E / S numériques Pins	14 (dont 6 fournissent la sortie PWM)
PWM numérique E / S Pins	6
Pins d'entrée analogique	6
DC Courant par I O Pin /	20 mA
Courant DC pour 3.3V Pin	50 mA
Mémoire Flash	32 KB dont 0.5 KB utilisée par le <i>bootloader</i>
Mémoire SRAM	2 KB
Mémoire EEPROM	1 KB
Vitesse de l'horloge	16 MHz
Longueur :	68,6 mm
Largeur	53,4 mm
Poids	25 g

10.1. Le Microcontrôleur ATmega328 :

Un microcontrôleur ATmega328 est un circuit intégré qui rassemble sur une puce plusieurs éléments complexes (les transistors; les résistances et les condensateurs) dans un espace réduit (boîtier en plastique noir muni d'un certain nombre de broches). Le microcontrôleur ATmega328 est constitué par un ensemble d'éléments qui ont chacun une fonction bien déterminée. Il est en fait constitué des mêmes éléments que sur la carte mère d'un ordinateur. [11]

La figure I.2 montre un microcontrôleur ATmega 328, qu'on trouve sur la carte Arduino.

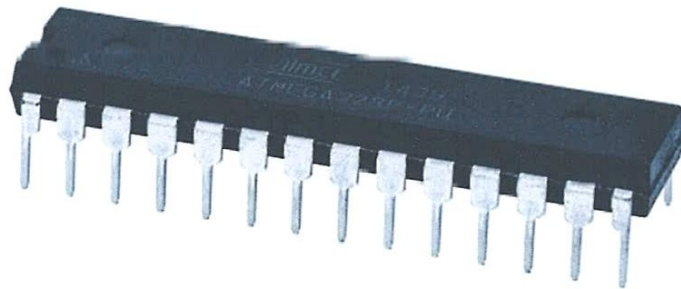


Figure 2.12 : Le microcontrôleur ATmega328

10.2. L'alimentation de la carte :

La carte Arduino Uno peut être alimentée via la connexion USB ou avec une alimentation externe. La source d'alimentation est automatiquement sélectionnée. Une alimentation externe peut provenir soit d'un adaptateur AC-DC ou d'une batterie. L'adaptateur peut être connecté en branchant une prise 2.1mm dans la prise d'alimentation de la carte ou à partir d'une batterie connectée dans le pin (ou broche) GND et V-in (alimentation externe). Le processeur peut fonctionner sur une alimentation externe de 6 à 20 volts. Cependant, si la tension est inférieure à 7V, le pin 5V peut fournir moins de cinq volts et le processeur peut devenir instable. Si la tension est supérieure à 12V, le régulateur de tension peut surchauffer et endommager la carte. La plage recommandée est de 7 à 12 volts.

Les pins (ou broches) d'alimentation sont les suivantes:

- **V-in** : Tension d'entrée à la carte Arduino à l'aide d'une source d'alimentation externe (par opposition à 5 volts de la connexion USB ou une autre source d'alimentation régulée). Si l'alimentation en tension est faite par l'intermédiaire de la prise d'alimentation, on pourra y accéder via ce pin.
- **5 V** : Ce pin délivre un 5V régulé par la carte. Le processeur peut être alimenté soit à partir de la prise d'alimentation DC (7-12 V), le connecteur USB (5 V), ou le pin V-in de la carte.

La fourniture d'une tension via les 5 V ou 3,3 V est déconseillée, elle contourne le régulateur, et peut endommager le processeur.

- **3,3 V** : Une alimentation de 3,3 volts générée par le régulateur. La consommation de courant maximale est de 50 mA.
- **GND** : Masse.

10.3. Les Entrées et les sorties :

Chacune des 14 broches numériques sur la carte Uno peut être utilisée comme une entrée ou une sortie, en utilisant les fonctions *pinMode()*, *digitalWrite()*, et *digitalRead()* du langage Arduino. Chaque broche peut fournir ou recevoir 20 mA en état de fonctionnement recommandée et a une résistance de pull-up interne (déconnecté par défaut) de 20-50 kOhms. Un maximum de 40 mA est la valeur qui ne doit pas être dépassée sur toutes les broches d'Entrée/Sorties pour éviter des dommages permanents au microcontrôleur.

En plus, certaines broches ont des fonctions spécialisées :

- **Interruptions Externes**: Broches 2 et 3, ces broches peuvent être configurées pour déclencher une interruption sur une valeur basse, sur un front montant ou descendant, ou sur un changement de valeur. Impulsion PWM (largeur d'impulsion modulée): broches 3, 5, 6, 9, 10 et 11. elles fournissent une impulsion PWM 8-bits à l'aide de l'instruction *analogWrite()*.
- **Série: 0 (RX) et 1 (TX)** : permet de recevoir (RX) et transmettre (TX) TTL données série. Ces pins sont connectés aux pins correspondants de l'USB-TTL puce Serial ATmega8U2.

- **LED:** broche 13, il y a une LED incluse dans la carte connectée à la broche 13. Lorsque la broche est au niveau HAUT, la LED est allumée, lorsque la broche est au niveau BAS, la LED est éteinte.
- **SPI** (Interface Série Périphérique) : broches 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ces broches supportent la communication SPI (Interface Série Périphérique) disponible avec la librairie pour communication SPI. Les broches SPI sont également connectées sur le connecteur ICSP qui est mécaniquement compatible avec les cartes Mega.
- **I2C:** broches 4 (SDA) et 5 (SCL) qui supportent les communications de protocole I2C (ou interface TWI (Two Wire Interface - Interface "2 fils"), disponible en utilisant la librairie `Wire/I2C` (ou TWI - Two-Wire interface - interface "2 fils").
- **les entrées analogiques :** L'Uno dispose de 6 entrées analogiques, A0 à A5, dont chacune fournit 10 bits de résolution (ou 1024 valeurs différentes). Par défaut, la tension est de 5 volts, mais il est possible de changer la limite supérieure de la gamme en utilisant la broche AREF et la fonction `analogReference()`.

10.4. La mémoire :

La mémoire de l'atmega328 se compose essentiellement sur :

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

10.5. La Communication :

L'Arduino uno à un certain nombre de moyens pour communiquer avec un ordinateur, une autre carte Arduino, ou autres microcontrôleurs. L'ATmega328 fournit UART TTL (5V) en communication série, disponible sur les broches numériques 0 (RX) et 1 (TX). La connexion série de l'Arduino est très pratique pour communiquer avec un PC, mais son

inconvenient est le câble USB, pour éviter cela, il existe différentes méthodes pour utiliser ce dernier sans fil.

11. Interface de programmation IDE

Le logiciel qui permet de programmer notre carte Arduino porte le nom d'IDE, ce qui signifie « Integrated Development Environment » ou encore « Environnement de Développement Intégré ». En effet, cette application intègre l'édition des programmes, la téléversement dans la carte Arduino et plusieurs bibliothèques.

L'IDE est une application Java, libre et multiplateformes (fonctionnant sur plusieurs système d'exploitation), servant d'éditeur de code et de compilateur, et qui peut transférer le firmware et le programme au travers de la liaison série (RS-232, Bluetooth ou USB selon le module). Il est également possible de se passer de l'interface Arduino, et de compiler les programmes via l'interface en ligne de commande.

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

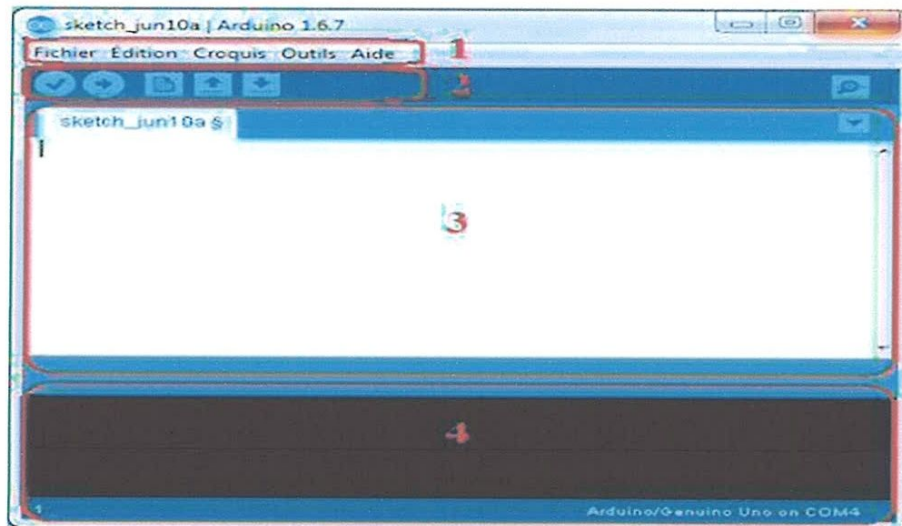


Figure 2.13 : Interface IDE

11.1. Description de l'interface

L'espace de programmation (IDE) des cartes Arduino comporte :

a. Menu :

Les différents éléments du menu (Fichier, Edition, Croquis et Outils) permettent de créer de nouveaux sketches (programmes), de les sauvegarder, et de gérer les préférences du logiciel et les paramètres de communication avec votre carte Arduino.

b. Barre d'action :

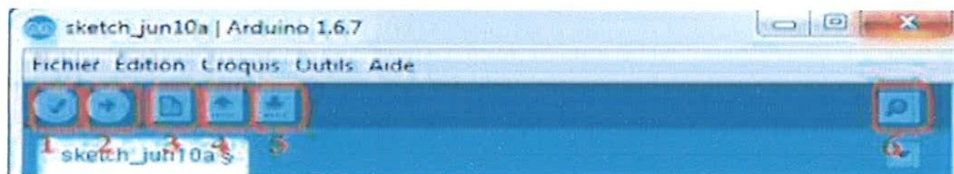


Figure 2.14 : Barre d'actions

1 : Bouton Verify (Compiler) : il permet de vérifier le programme pour trouver d'éventuelles erreurs. Cette procédure prend un certain temps d'exécution et lorsque elle est terminée, elle affiche un message de type « Binary sketch size : ... » Indiquant la taille du programme téléversé.

2 : Bouton Upload (Téléverser) : ce bouton permet de compiler et téléverser le programme sur la carte Arduino.

3 : Bouton Nouveau : Crée un nouveau code (ouvre une fenêtre d'édition vide)

4 : Bouton Ouvrir : Ouvre un programme existant dans la fenêtre courante.

5 : Bouton Sauver : Enregistre le programme.

6 : Moniteur Série : Ouvre la fenêtre du moniteur (ou terminal) série.

c. Fenêtre de Programmation :

L'éditeur où s'écrit le programme, chaque logiciel obéit à quelques notions pour pouvoir bien structurer le programme à fin de le compiler et éviter les erreurs de syntaxe et autres.

d. Barre des erreurs :

La barre des erreurs affiche les erreurs faites au cours du programme, comme l'oubli d'un point-virgule, le manque d'une accolade ou toute autre erreur dans les instructions (Figure 15).

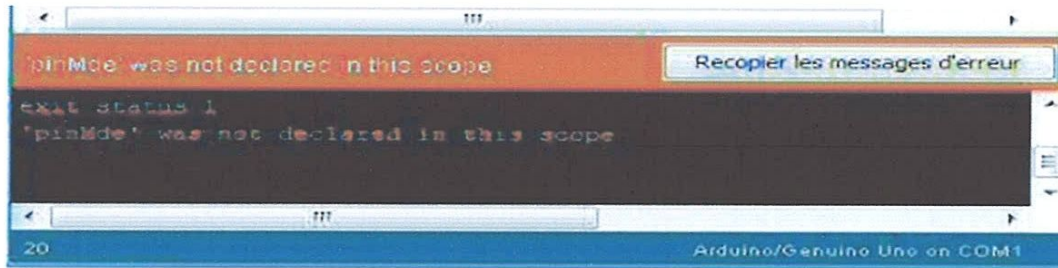


Figure 2.15: Message d'Erreur

11.2. Structure du programme :

La structure du programme pour la carte arduino comporte trois phases consécutives :

11.2.1. La Définition des constantes et des variables :

Cette partie est optionnelle, chaque entrée et sortie est définie et déclarée, en lui donnant un nom arbitraire et en lui affectant le numéro de l'entrée ou celui de la sortie voulue, sans oublier de préciser le type de la variable.

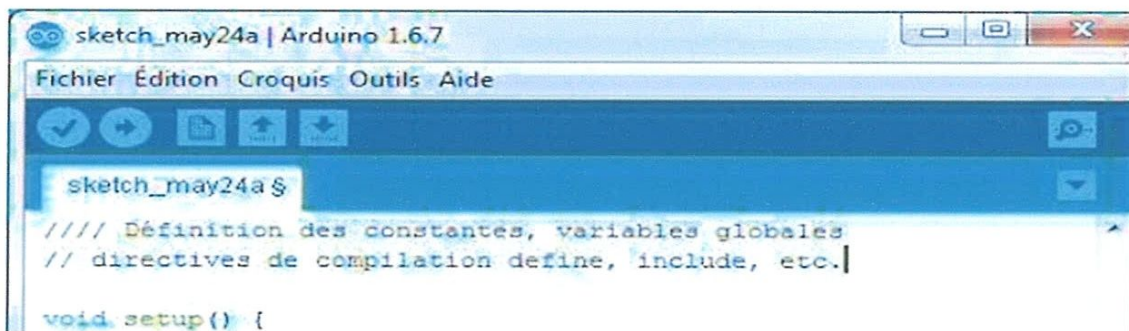


Figure 2.16 : Définition des constantes et des variables.

11.2.2. Configuration des entrées/sorties :

Les instructions viennent après « **void setup ()** » après avoir ouvert une accolade, on peut manipuler les broches de la carte en les configurant comme étant des entrées ou des sorties, selon les besoins. Les entrées analogiques pour les capteurs par exemple, ne sont soumises à aucune configuration, car la carte possède 6 entrées analogiques qui ne font que cela.

```
void setup() {  
  // initialisation des ressources de la carte,  
  // configuration des entrées/sorties,  
  // définition de la vitesse de fonctionnement du port série, etc.  
  // setup() n'est exécuté qu'une seule fois.  
}
```

Figure 2.17 : Configuration des Entrées/Sorties

11.2.3. Programmation des interactions et comportements :

Celles-ci viennent après « **void loop ()** » c'est la partie principalement, ou on rédige les instructions et les opérations comme la lecture des données, les boucles, les affectations,...etc. Chacune d'elle doit obligatoirement finir par un point-virgule.

```
void loop() {  
  // les instructions contenues ici sont exécutées indéfiniment en boucle  
  // Seule une coupure de l'alimentation de la carte ou un appui sur le bouton Reset  
  // permet de quitter le programme.  
}
```

Figure 2.18 : Programmation des interactions et comportements

11.2.4. Les commentaires :

Les commentaires doivent être écrits après un slash ou une étoile ou les deux, tandis que sur une ligne de code, on les écrit après deux slashes. [12]

12.Conclusion :

Dans ce chapitre nous avons présenté les microcontrôleurs et leur architecture interne, et on a pris une vue détaillé sur la carte Arduino y compris l'architecture interne aussi son environnement (partie logiciel).

CHAPITRE 3: REALISATION DE L'APPLICATION

1. Introduction

Depuis quelques années, les solutions de domotique se démocratisent de plus en plus. En effet, à une époque où il devient important de maîtriser sa consommation énergétique et où presque tout le monde dispose d'un Smartphone, il est tentant de pouvoir commander ses équipements domestiques (éclairage, chauffage, stores, etc.) et de consulter l'état de l'habitation (température, humidité, etc.) via un ordinateur ou un appareil mobile. L'objectif de notre projet est de créer un système de ce type.

2. Description de projet

En principe, un système domotique est composé d'actionneurs et de capteurs reliés à un réseau, afin de permettre leur gestion via une ou plusieurs interfaces homme/machines. En ce qui concerne la partie matérielle, nous avons utilisé une carte Arduino UNO pour leur grande polyvalence et leur faible coût. L'utilisation de cette carte comme un microcontrôleur rend le système extrêmement modulable. Ainsi dans ce projet on utilise un capteur de température et humidité « le DHT11 » et un détecteur de mouvement « P.I.R sensor » pour la détection des personnes dans la maison, on utilise aussi des relais 5v pour la liaison entre le circuit de commande (l'arduino) et les actionneurs (climatiseurs, chauffages, éclairages, etc.), et nous n'oublions pas le *Shield Ethernet* qui fait la connexion entre l'arduino et le réseau local.

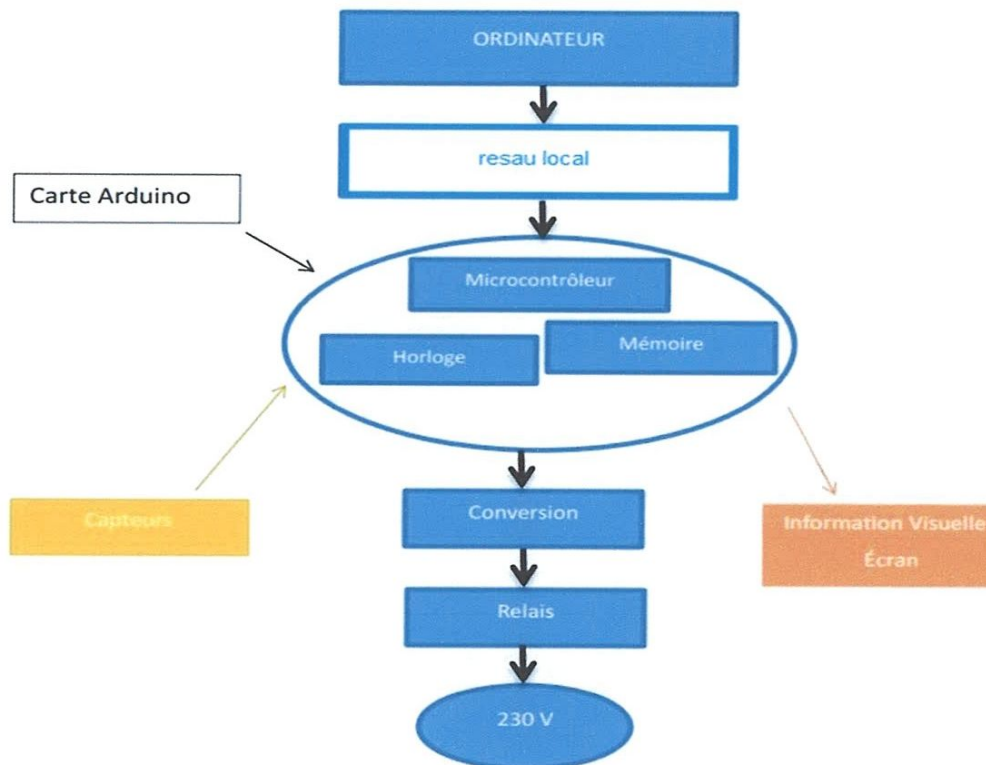


Figure 3.1 : Principe de fonctionnement de projet domotique

3. Les composants du système

3.1. Le DHT 11

Le DHT11 est un capteur de température et d'humidité « deux en un ». Capable de mesurer des températures de 0° à +50°C avec une précision de +/- 2° et des taux d'humidité relative de 20 à 80% avec une précision de +/- 5%. Une mesure peut être réalisée toutes les secondes. Le DHT11 est compatible 3.3 volts et 5 volts (le fabricant recommande cependant de toujours alimenter le capteur en 5 volts pour avoir des mesures précises). Son courant d'alimentation est de : 100 μ A maximum en veille, 2.5 mA maximum en dialogue (lors de la collecte de données). [13]

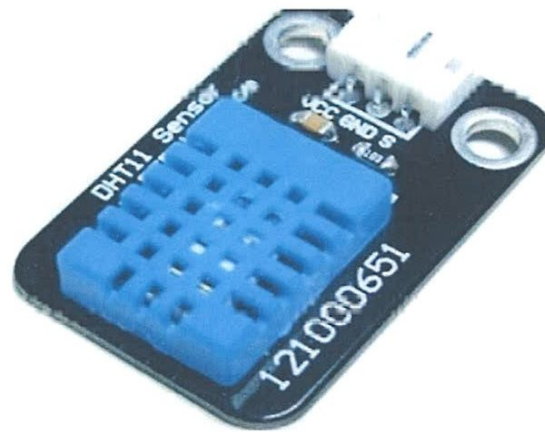


Figure 3.2 : Le capteur dht11

Pour résumer, les caractéristiques de ce capteur sont regroupées dans le tableau suivant :

Tableau 3.1 : Caractéristique d'un capteur DHT11

Humidité (relative %)	20 ~ 80%
Précision (humidité)	+/- 5%
Température	0 ~ +50°C
Précision (température)	+/- 2°C
Fréquence mesure max	1Hz (1 mesure par seconde)
Tension d'alimentation	3 ~ 5 volts
Stabilité à long terme	+/- 1% par an
Consommation maximale	2.5mA (lors de la collecte de données)
Dimensions	15.5mm x 12mm x 5.5mm

Le brochage du capteur est comme le suivant :

- La broche n°1 est la broche d'alimentation (5 volts ou 3.3 volts).
- La broche n°2 est la broche de communication. Elle sera connectée à l'entrée utilisée pour lire les données sur l'Arduino. Celle-ci doit impérativement être reliée à l'alimentation via une résistance de tirage de 4.7K ohms (il s'agit d'une sortie à collecteur ouvert).
- La broche n°3 n'est pas utilisée et ne doit pas être câblée.
- La broche n°4 est la masse du capteur (GND).

La ligne filaire peut faire jusqu'à 20 m de longueur entre le capteur et le microcontrôleur [13].

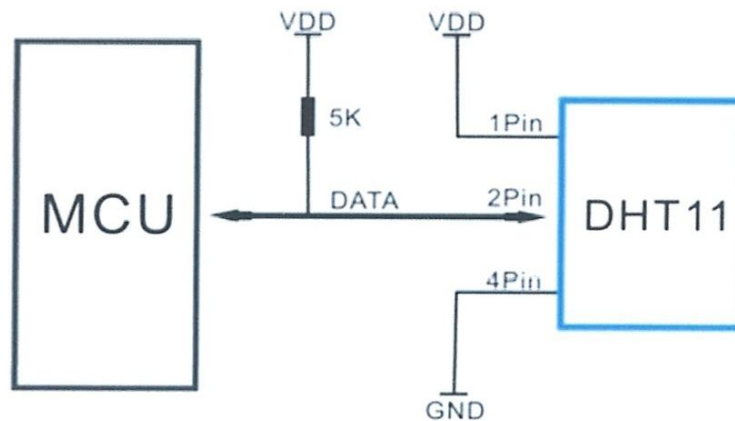


Figure 3.3 : Brochage d'un capteur DHT11

Le dht11 a la particularité de communiquer avec le microcontrôleur au moyen d'une seule ligne série bidirectionnelle (une unique broche d'entrée / sortie) de type OneWire (Un fil).

Le protocole de communication est le suivant :

Lorsque le microcontrôleur envoie un signal de démarrage, le DHT11 passe du mode faible consommation électrique au fonctionnement en mode dialogue et envoie l'accusé de réception suivi des données (Si le signal de départ est valide). Il repasse ensuite en mode veille jusqu'à la prochaine sollicitation.

La communication avec un capteur DHT11 se fait en 3 étapes :

- Demander une mesure : Forcer l'état "0" durant 18 ms puis état "1" durant 40 μ s.
- Accusé de réception du DHT11 : Forcer l'état "0" durant 54 μ s puis état "1" durant 80 μ s.

- Lecture des données : Les données suivent l'accusé de réception et sont codées dans une série de 40 bits (sous forme de cinq valeurs de 8 bits chacune) dont le format est comme suit :

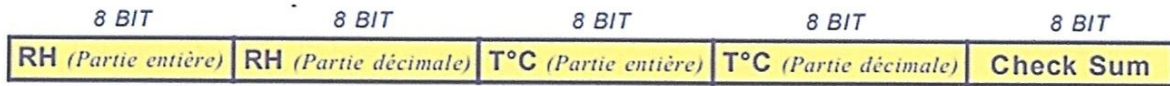


Figure 3.4 : Format des BIT de données

La « *Check Sum* » est calculée par la somme des quatre premières données, donc avec les 32 premiers BIT.

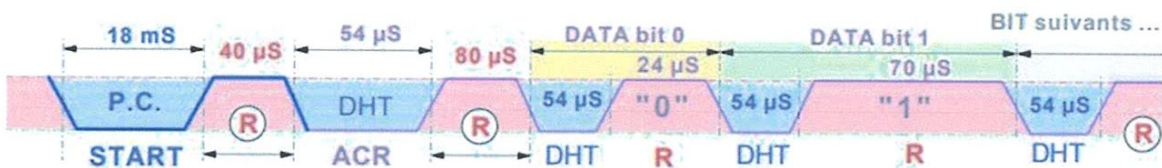


Figure 3.5 : Les étapes de communication entre le dht11 et le microcontrôleur

- En bleu clair l'état zéro forcé par le P.C. ou en réponse par le DHT11.
- En rose pâle l'état "1" maintenu par la résistance de forçage R (Pull Up).
- En bleu épais la requête du P.C. et en violet fin la réponse du capteur. [14]

La broche de communication des capteurs DHT11 est de type "collecteur ouvert". La sortie du capteur ne génère pas de tension. Elle ne fait que commuter (via un transistor) la tension au niveau de la résistance de tirage R sur la ligne de données. Dans ce contexte, la valeur *HIGH* est la tension de la résistance de tirage et *LOW* est la tension à la masse (0 volt).

La mise en œuvre de dht11 sur ARDUINO est d'autant plus facile que la bibliothèque DHT11.h est disponible en ligne et se charge des protocoles de dialogue de type série. Celle-ci contient notamment la fonction *read11* (car notre composant s'appelle DHT 11), qui prend en paramètre le numéro du pin à laquelle le capteur est connecté et qui effectue la mesure et permet de renvoyer l'état du capteur et de signaler si celui a rencontré un problème ou non. Elle contient aussi des variables membres permettant simplement d'accéder aux valeurs de la température ou de l'humidité.

3.2. Le capteur de mouvement HC-SR501 (P.I.R)

Un capteur PIR (*Passive Infra Red sensor* : Capteur infrarouge passif) utilise l'effet Pyroélectrique. La pyroélectricité (du pyrus grec, du feu et de l'électricité) est la propriété de certains cristaux qui sont naturellement polarisés électriquement, et ont la capacité de générer une tension lorsqu'ils sont chauffés ou refroidis. Le changement de température modifie légèrement la position des atomes à l'intérieur de la structure cristalline, de sorte que la polarisation du matériau change. Ce changement de polarisation provoque une tension à travers le cristal. Si la température reste constante à sa nouvelle valeur, la tension pyroélectrique disparaît progressivement à cause du courant de fuite. Le capteur est dit passif car il n'y a pas de diode infrarouge pour éclairer la zone de détection, le capteur fonctionne de façon autonome.

Le principe de fonctionnement et le suivant :

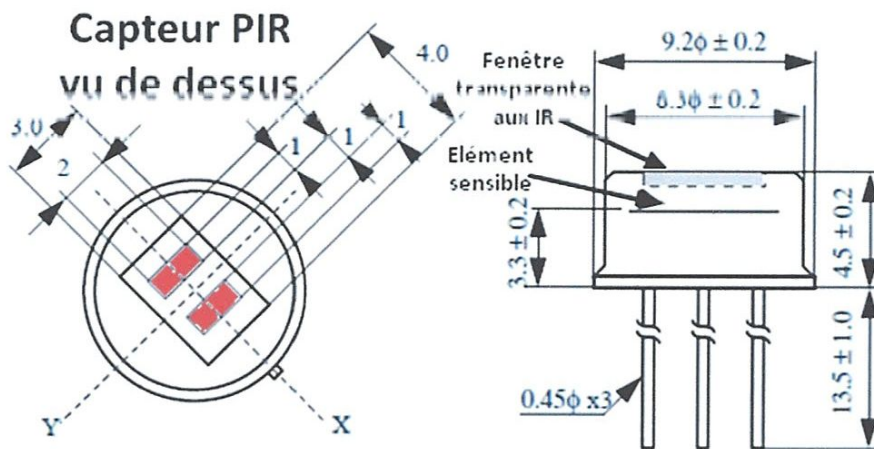


Figure 3.6 : Présentation des deux plaquettes à effet pyroélectrique de capteur P.I.R

Vu de dessus, le capteur monté dans son boîtier à une forme circulaire. Une fenêtre bloque la lumière visible et laisse passer le rayonnement infrarouge vers les deux plaquettes à effet pyroélectrique (en rouge sur la vue de dessus). Sur le schéma situé à droite, on voit le boîtier cylindrique et les broches de connexion. La fenêtre infrarouge est représentée en gris. Elle se trouve au-dessus de l'élément sensible, comportant les deux plaquettes pyroélectriques.

Le rayonnement infrarouge arrive sur la fenêtre, la traverse et provoque une variation de tension au niveau des plaquettes. La fenêtre ne fait pas grand-chose à part arrêter la lumière visible et laisser passer le rayonnement infrarouge.

Les deux plaquettes sensibles peuvent “voir” jusqu’à une certaine distance (essentiellement liée à la sensibilité du capteur). Lorsque les deux plaquettes détectent la même quantité d’IR, le capteur est inactif. Il s’agit de la quantité d’IR ambiante rayonnée dans la pièce ou par des murs à l’extérieur. Lorsqu’un corps chaud comme un humain ou un animal passe, il modifie d’abord la quantité d’IR sur la moitié du capteur PIR, ce qui crée une différence entre les deux plaquettes du capteur et une impulsion positive. Lorsque le corps chaud quitte la zone de détection, l’inverse se produit et le capteur génère une variation négative. Ce sont ces impulsions générées lors des changements qui sont détectées. [15]

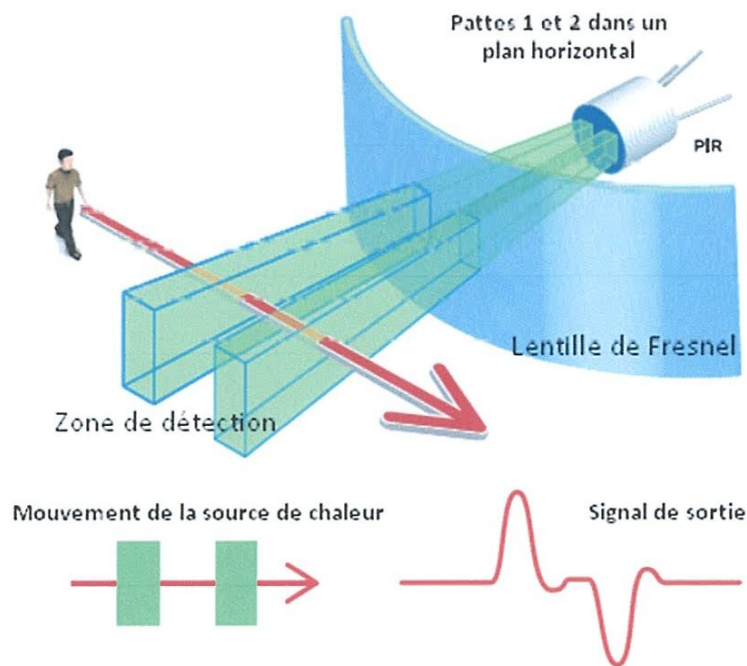


Figure 3.7 : Principe de fonctionnement de capteur PIR

Pour améliorer la détection du capteur de base, on utilise une lentille pour projeter le rayonnement IR sur le capteur. C’est pour ça, les fabricants de capteur ont développé des lentilles spéciales réalisés en matière plastique nommés « lentilles de Fresnel multiples ».



Figure 3.8 : Lentille de Fresnel multiple pour le PIR

Sur cette lentille prélevée sur un détecteur PIR. On voit les différentes zones correspondantes à autant de lentilles individuelles. Chaque zone envoie l'image de sa "région" sur le capteur. Donc l'emplacement d'une lentille de Fresnel multiple améliore la distance de détection de capteur.

Chacune des lentilles correspond à une zone de détection. La forme sphérique de l'ensemble permet de réaliser un capteur capable de détecter un mouvement dans un angle de 100°, voire plus. L'image ci-dessus indique les zones de détection en vue de dessus. [15]

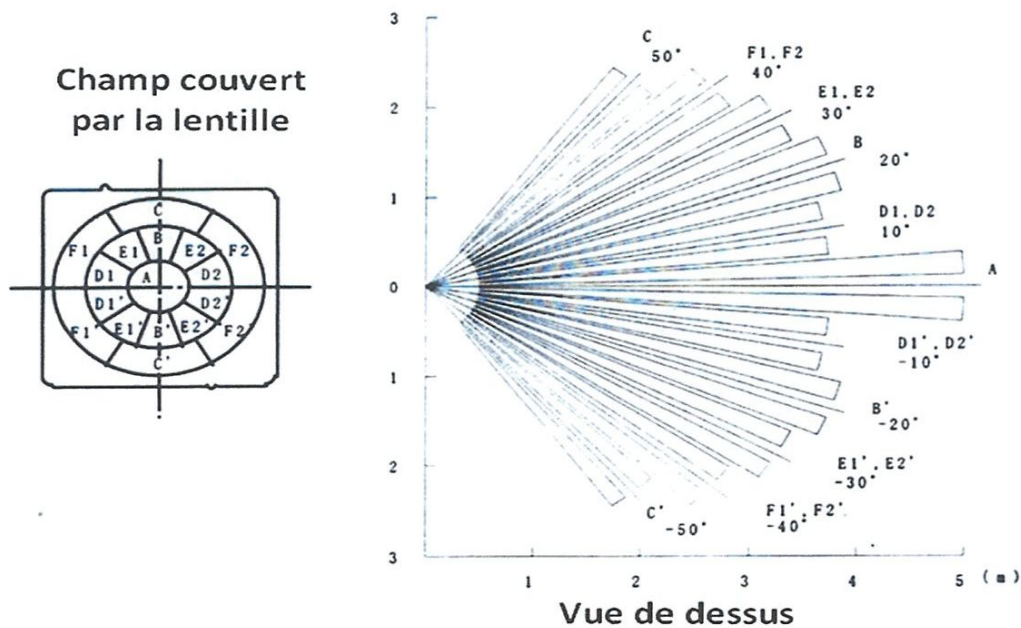


Figure 3.9 : Champ couvert par lentille de Fresnel multiple

C'est cette multitude de zones qui fait qu'un capteur situé à 2 ou 3 mètres de hauteur (sous un projecteur extérieur par exemple) est à même de détecter une personne qui s'aventure dessous.

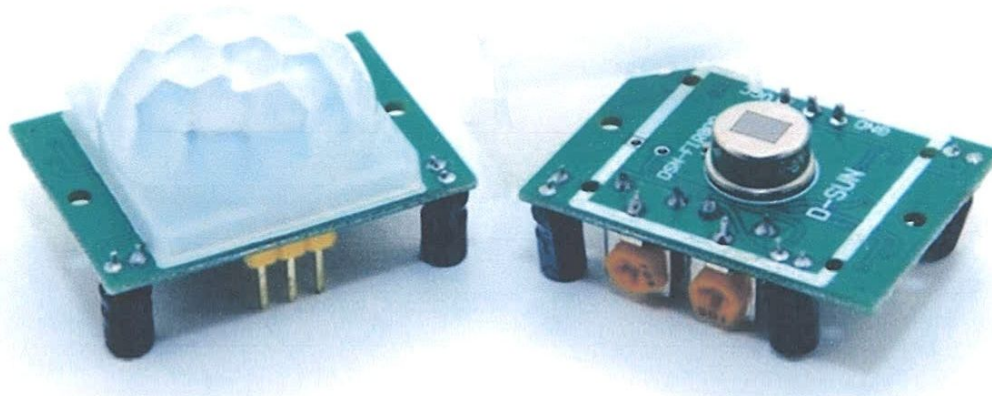


Figure 3.10 : Détecteur PIR avec et sans la lentille sur le capteur.

Tableau 3.2 : caractéristiques du capteur PIR

Tension d'alimentation	5V – 20V
Consommation	65mA
Sortie TTL	3.3V, 0V
Temps pendant lequel la sortie est à 1	Ajustable (3s à 5min)
Temps de verrouillage Ti	~0.2 sec à 3s selon fabricant
Sensibilité	moins de 120°, jusque 7 mètres
Température	- 15 à +70°C
Angle	<100 °
Dimension	32*24 mm, distance entre trous de montage 28mm, vis M2Diamètre de la lentille : 23mm

3.3. Le relais 5v et l'arduino

Une des choses les plus utiles que nous pouvons faire avec un Arduino est de contrôler les appareils à plus haute tension (120-240V) comme les ventilateurs, les lumières, les appareils de chauffage et autres appareils électroménagers. Comme l'Arduino fonctionne à 5V, il ne peut pas contrôler directement ces appareils à plus haute tension, mais nous pouvons utiliser un relais 5V pour commuter le courant 120-240V et utiliser l'Arduino pour contrôler le relais.

L'Arduino peut être programmé pour activer le relais lorsqu'un certain événement se produit, par exemple lorsque la détection d'un mouvement par le capteur P.I.R. N'importe quel capteur peut être utilisé pour déclencher ou éteindre le relais. Le déclencheur n'a même pas besoin d'être un capteur. Il peut se produire à des intervalles de temps définis, il peut être déclenché par simple pression d'un bouton.

Ce relais dispose de trois bornes haute tension (NC, C et NO) qui se connectent à l'appareil que vous souhaitez contrôler. L'autre côté dispose de trois broches basse tension (GND, VCC et Signal) qui se connectent à l'Arduino.



Figure 3.11 : Les bornes d'un relais 5v

- ✓ NC: normalement fermé 120-240V
- ✓ NO: normalement ouvert 120-240V
- ✓ C: commun
- ✓ Ground (Terre): se connecte à la broche de masse sur l'Arduino
- ✓ 5V VCC: Connecte la broche 5V de l'Arduino
- ✓ Signal: Transmet le signal de déclenchement de l'Arduino qui active le relais

À l'intérieur du relais est un interrupteur 120-240V qui est connecté à un électro-aimant. Lorsque le relais reçoit un signal HAUT sur la broche de signal, l'électroaimant se charge et déplace les contacts de l'interrupteur ouvert ou fermé.

Le relais a deux types de contacts électriques à l'intérieur - normalement ouvert (NO) et normalement fermé (NC). Celui que nous utiliserons dépendra de si nous voulons que le signal 5V allume ou éteigne l'interrupteur. Le courant d'alimentation 120-240V entre dans le relais à la borne commune (C) dans les deux configurations. Pour utiliser les contacts normalement ouverts, utilisez le terminal NO. Pour utiliser les contacts normalement fermés, utilisez la borne NC. [16]

Dans la configuration normalement ouverte, lorsque le relais reçoit un signal HIGH, le commutateur 120-240V se ferme et permet au courant de circuler de la borne C vers la borne NO. Un signal LOW désactive le relais et arrête le courant. Donc, si nous voulons que le signal HIGH allume le relais, nous utilisons la borne normalement ouvert.



Figure 3.12 : La borne normalement ouvert

Dans la configuration normalement fermée, un signal HIGH ouvre le commutateur et interrompt le courant 120-240V. Un signal LOW ferme l'interrupteur et permet au courant de circuler de la borne C vers la borne NC. Par conséquent, si nous voulons que le signal HIGH éteigne le courant 120-240V, nous utilisons la borne normalement fermée.

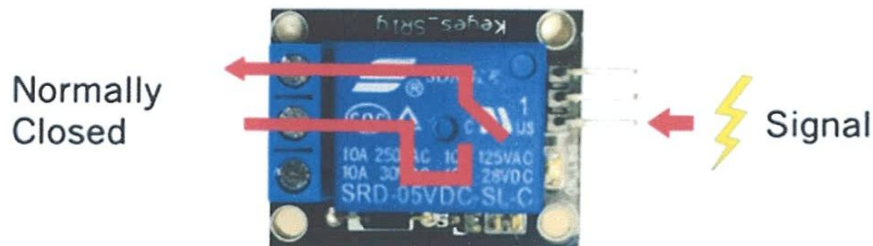


Figure 3.13 : La borne normalement fermée.

3.4. Allumage automatique de la lampe à la détection de mouvement

Cette partie permet d'activer l'éclairage d'une pièce de la maison d'une manière automatique lors de la détection de passage d'une personne (par le capteur PIR). Alors, l'éclairage de la zone de marche est activé pendant un intervalle de temps.

l'utilisation du capteur PIR avec l'arduino est très simple. Le programme IDE doit établir une connexion, lire les données reçues du capteur du mouvement (la valeur « val » dans le programme), puis allumer ou éteignez la lampe suivant cette valeur.

```
val = digitalRead(inputPin);
if(val == HIGH){
  digitalWrite(relayPin4, LOW); // relaypin4 est le pin de relai connecté avec la lampe
  delay(reldelay);
} else {
  digitalWrite(relayPin4, HIGH);
}
```

Figure 3.14 : Activation du relais de l'éclairage par le capteur de mouvement PIR

3.5. Le Shield Ethernet

L'un des shield les plus intéressants que nous pouvons monter sur la plate-forme Arduino est certainement le shield Ethernet, car il permet de nombreuses applications réseau telles que le contrôle à distance des systèmes et des utilisateurs, l'accès Web et la publication des données. Il joue le rôle d'une carte réseau. L'intégration de bibliothèques open-source sur Arduino IDE fait le reste.

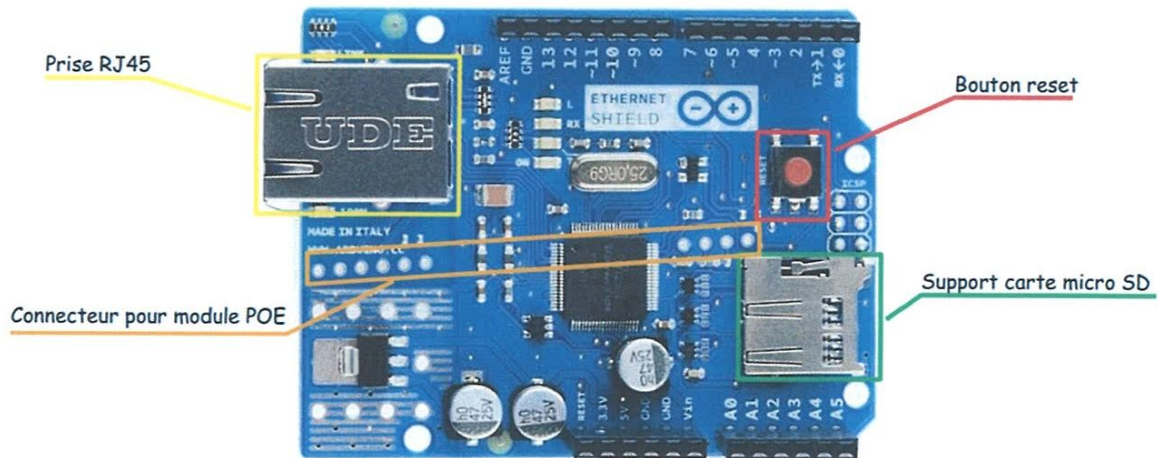


Figure 3.15 : Le Shield Ethernet

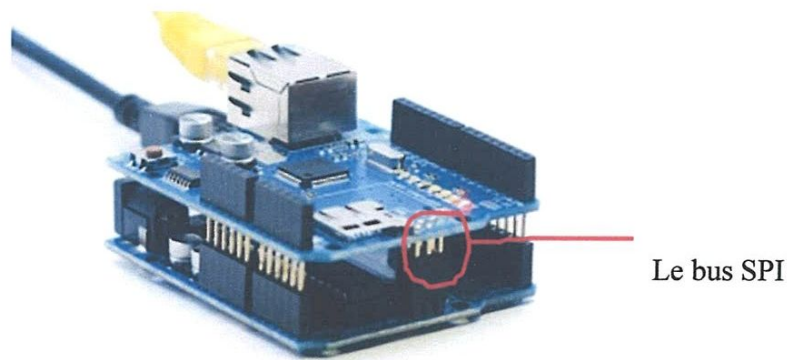


Figure 3.16 : Liaison entre arduino et le shield Ethernet

Le shield Ethernet dialogue avec l'Arduino via le bus SPI. SPI est un protocole de communication commun utilisé par de nombreux appareils différents pour communiquer avec les microcontrôleurs.

Les appareils communiquant via SPI sont dans une relation maître-esclave. Le maître est l'appareil de contrôle (généralement un microcontrôleur), tandis que l'esclave (dans notre cas est le shield Ethernet) reçoit des instructions du maître.

3.5.1. Branchement d'un Arduino sur un réseau local

Le shield Ethernet fait la connexion avec un réseau local, permet de contrôler à distance des systèmes arduino et transmet sur le réseau les états de l'Arduino.

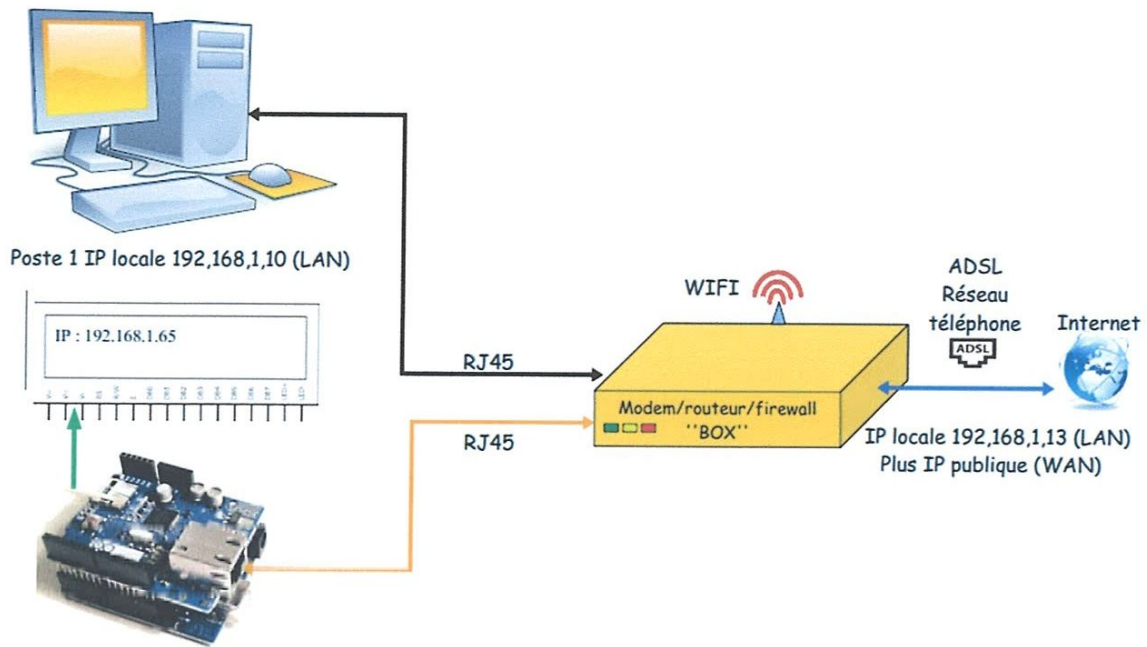


Figure 3.17 : Branchement d'un arduino sur un réseau local

Chaque système est identifié par une adresse IP (Internet Protocol). La configuration de l'adresse IP de shield Ethernet est simple, nous indiquerons l'adresse de notre choix dans le programme de l'Arduino. Cette adresse IP doit être dépend de notre réseau local.

```
#include <SPI.h> .....// intégration de la bibliothèques <SPI.h> pour le protocole spi
#include <Ethernet.h> .....// L'intégration de la bibliothèque <Ethernet.h> pour la connection ethernet
byte mac[] = {
  0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0x53 }; // définition d'une adress mac et IP de notre choix
IPAddress ip(192,168,97, 240); //
```

Figure 3.18 : Configuration du shield Ethernet sur arduino IDE

3.5.2. Lecture et écriture sur le shield via le réseau internet :

Lorsque nous nous connectons à internet nous ouvrons un navigateur qui va se connecter à notre "box" (modem). La requête que nous allons lancer va partir sur internet avec notre IP local plus notre IP publique (variable par manque d'adresses en IPV4). La station qui va recevoir la requête pourra répondre sans problème car elle sait à qui répondre. Pour notre shield Ethernet le problème est différent. Notre adresse IP publique permet à une station extérieure d'avoir accès au routeur de la "box" mais pas à notre Arduino. Il faut faire ce que l'on appelle une redirection de port pour avoir accès à celui-ci. Une ouverture de port permet

de rediriger en interne une requête qui vient de l'extérieur. Cette ouverture de port se fait dans la page de configuration de notre "box". A l'origine ces ouvertures de port servent à des jeux en réseaux, des échanges torrent ou bien à des logiciels tel que Skipe. [17]

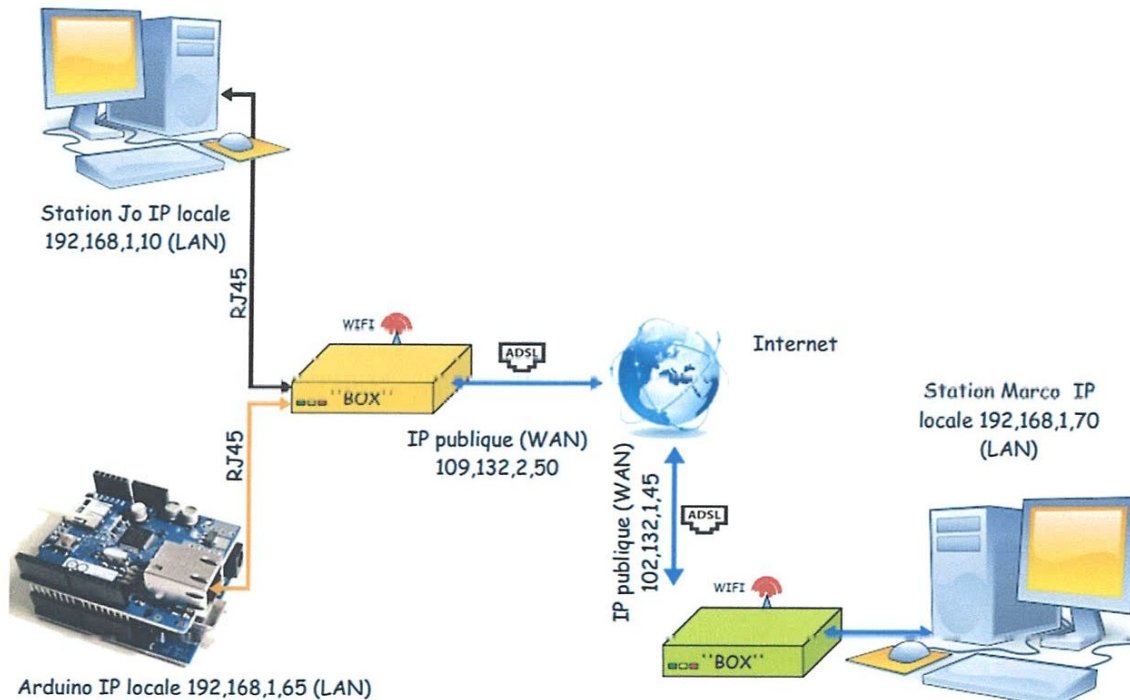


Figure 3.19 : Branchement d'un arduino sur un réseau internet

4. Conception de la page WEB

Le shield Ethernet permet de recevoir des requêtes client, cette opération doit être effectuée à travers une page web à partir d'un navigateur internet. Dans ce cas, l'arduino de notre montage joue le rôle d'un serveur. Le client demande la page web généré par notre arduino en utilisant un navigateur avec la saisie de l'adresse IP du shield ethernet. L'arduino envoie les informations contenues dans la page web au navigateur qui va l'afficher au client. Cette communication entre le navigateur (le client) et le serveur (arduino) se fait grâce au protocole HTTP (HyperText Transfer Protocol).

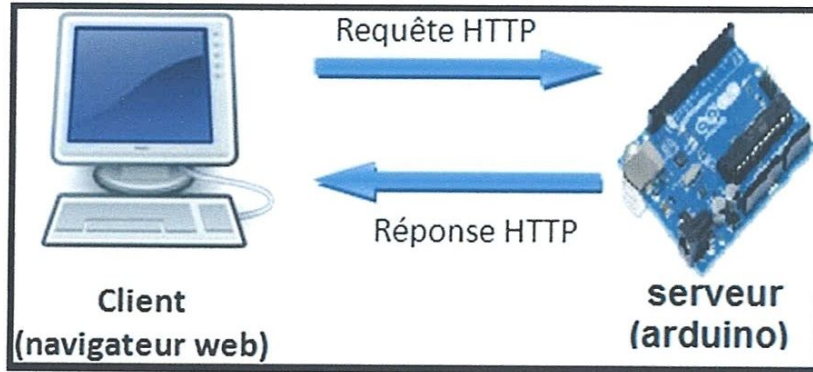


Figure 3.20 : Communication entre le client et l'arduino

- Le navigateur effectue une requête HTTP
- Le serveur traite la requête puis envoie une réponse HTTP
- Une requête HTTP est un ensemble de lignes envoyé au serveur par le navigateur. [18]

La page web est créée on utilise le langage html, le document *html* commence par la balise `<HTML>` et finit par la balise `</HTML>`. Il contient également un en-tête décrivant le titre de la page, puis un corps dans lequel se trouve le contenu de la page. L'en-tête est délimité par les balises `<HEAD>` et `</HEAD>`. Le corps est délimité par les balises `<BODY>` et `</BODY>`. La structure d'un document html doit être comme suit :

```

<HTML>      : ouverture de la page html
<HEAD>      : ouverture de l'en-tête
<TITLE>Titre de la page</TITLE> : la partie du titre
</HEAD>     : fermeture de l'en-tête
<BODY>      : ouverture de la partie body
    Contenu de la page
</BODY>     : fermeture de la partie body
</HTML>     : fermeture de l'HTML [19]

```

La création d'une page web avec arduino est d'autant plus facile à l'aide de la fonction `client.print`, Celle-ci compatible avec le langage HTML. Un exemple est représenté dans l'image ci-dessous.


```
client.println("<html>");
client.println("<head>");
client.println("<TITLE>electronics</TITLE>");.....// le titre de la page
client.println("</head>");
client.println("<body>");
client.println("univ guelma");.....//le contenu de la partie body
client.println("</body>");
client.println("</html>");
```

Figure 3.21 : Présentation de la fonction « *client.println* »

5. Schéma de réalisation de projet :

Notre projet réalisé vise le contrôle à distance et l'affichage de la température et l'humidité d'une maison intelligente, aussi la détection de présence des personnes dans cette maison à travers le réseau internet par un ordinateur ou un Smartphone. Ce circuit de commande se base sur une carte Arduino uno et les capteurs que nous avons mentionné précédemment.

Cette carte de commande Arduino joue le rôle du cerveau intelligent qui permettra de recevoir les commandes et les informations reçu depuis les capteurs, puis les exécuter et envoyer ces derniers par un réseau local à l'aide d'un shield Ethernet bien connecté avec le réseau.

La commande des actionneurs de maison (climatiseur, chauffage, ouverture et fermeture des porte) se fait via des relais qui lier la partie de contrôle (l'arduino) avec les actionneurs. L'affichage de la température et l'humidité est assuré à travers une page web, ce qui permet à l'utilisateur de les inspecter de temps en temps. Le circuit permet aussi d'allumer automatiquement la lumière en cas de présence de personnes dans une pièce et d'éteindre celle-ci en l'absence de mouvement.

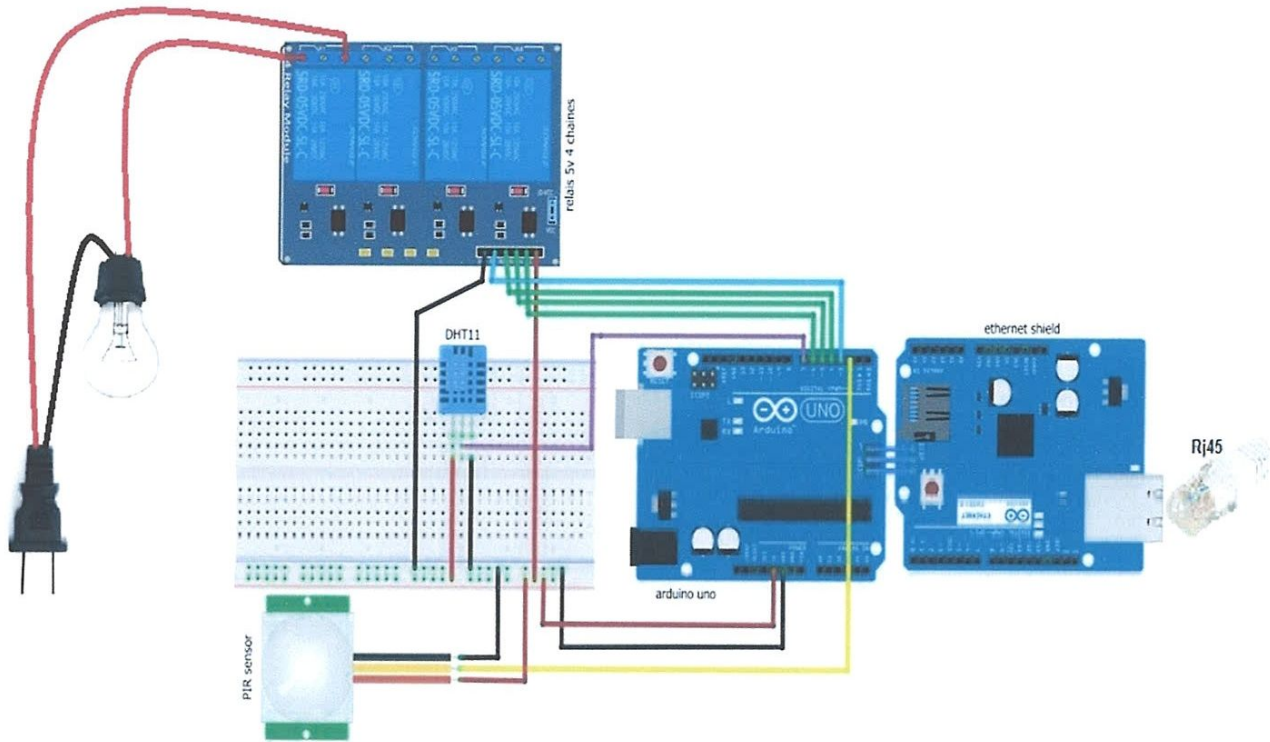


Figure 3.22 : Schéma du projet

Pour améliorer l'embarquement de notre circuit on utilise un relais 5v à 4 chaines qui contient deux bornes d'alimentation (VCC, GND) et 4 bornes d'entrée (IN1, IN2, IN3, IN4), chaque une de ces bornes se branche avec un PIN arduino.

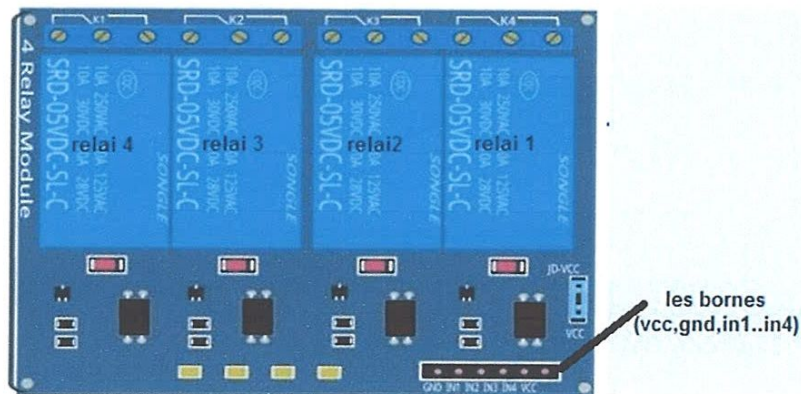


Figure 3.23 : Relais 5v à 4 chaines

6. L'interface de commande et d'affichage :

Ce projet nous permet d'inspecter la température, l'humidité et la présence des personnes ainsi que contrôle et la commande de plusieurs appareils à distance via une interface WEB simplifiée à travers d'un navigateur internet.

Nous avons développé cette interface WEB avec le langage html et le logiciel IDE. Pour consulter cette page web, l'utilisateur propriétaire ou le client doit saisir l'adresse IP du shield celle qui a été configuré dans le programme.

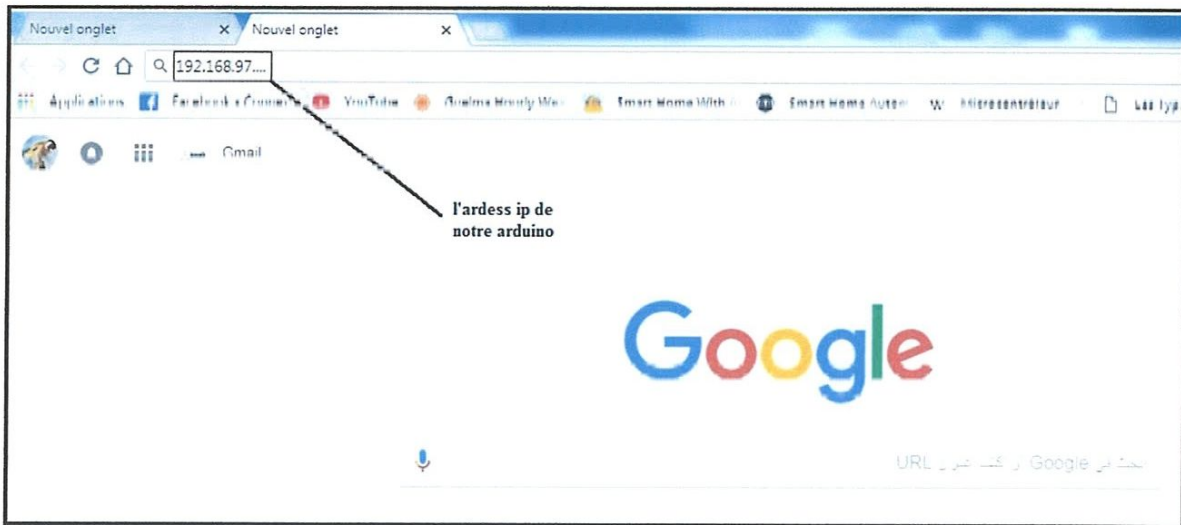


Figure 3.24 : Saisie de l'adresse IP dans le navigateur internet

Après avoir entré l'adresse IP, le panneau de contrôle vient de s'afficher au client comme le montre l'image suivante :

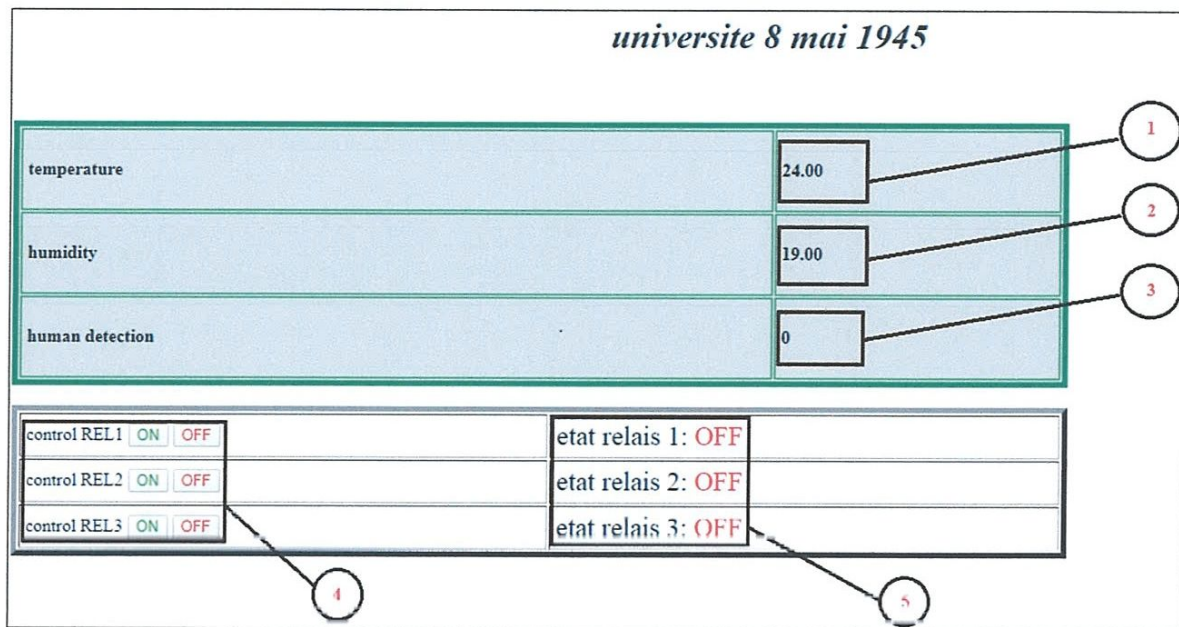


Figure 3.25 : Le panneau de contrôle

L'interface affichée est constituée de deux tableaux, le premier pour l'affichage des mesures des capteurs et l'autre pour le contrôle des relais qui sont connectés avec les actionneurs. On trouve ainsi :

- La zone (1) pour l'affichage de la température mesurée par le dht11
- La zone (2) pour afficher l'humidité mesurée par le dht11
- La zone (3) pour la détection des personnes (1 pour la présence d'un mouvement ou 0 lors de l'absence de mouvement).
- La zone (4) affiche les boutons de commande des relais on/off.
- La zone (5) affiche les états des différents relais.

7. Conclusion :

Comme nous l'avons vu dans ce chapitre, Nous présentons les différentes parties de réalisation de notre Project et nous expliquons comment nous avons procédé au système domotique

Dans ce chapitre, nous venons de montrer l'utilisation de la carte arduino uno et son shield Ethernet dans le système domotique.

Conclusion générale :

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

Dans ce cadre, nous avons fait la conception et la réalisation d'un système domotique, qui permet aux utilisateurs de surveiller et de commander les différents dispositifs domestiques à distance à travers le réseau Ethernet à partir d'un téléphone portable ou un ordinateur. Le système est géré par un microcontrôleur de type ARDUINO sur lequel sont connectés plusieurs accessoires : un capteur de température et d'humidité, un détecteur de mouvement utilisé pour un éclairage automatique et un circuit contient quelques relais pour commuter des prises électriques afin d'allumer ou éteindre les objets s'y branchant. Le circuit réalisé fonctionne comme un serveur WEB permettant d'afficher la température et l'humidité de la pièce, et en plus de commander certains relais.

En effet, notre système permet de piloter de façon simple et confortable l'ensemble de équipements électriques notamment l'éclairage, les climatiseurs, les ouvrants...etc. 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 on rajoutant d'autres relais et modifier le code pour incorporer les nouvelles fonctionnalités.

L'apport de notre projet se résume surtout dans la découverte d'un nouveau domaine, la domotique, qui consiste d'un domaine vaste et innovant. Le projet réalisé nous a permis de prendre certaines connaissances sur les réseaux locaux. D'autre part, de se familiariser avec les méthodes de développement qui nous ont permis d'améliorer nos compétences et nos acquis, notamment la programmation des microcontrôleurs ARDUINO avec l'IDE ainsi que la conception des pages WEB par le langage HTML.

En fin, nous pouvons signaler que ce travail n'est qu'une simple application dans le domaine de la domotique, il peut être plus autonome, plus pratique, et assez évolutif vu le pas géant et les progrès actuels réalisés dans les domaines de la technologie et de la communication.

Bibliographie

- [1] <http://www.cea.fr/comprendre/Pages/nouvelles-technologies/essentiel-sur-domotique-maison-connectee.aspx>. Date d'accès : 15/05/2018
- [2] <http://www.guide-domotique.com/la-domotique-cest-quoi-et-ca-sert-quoi>. Date d'accès : 17/05/2018
- [3] Wissem Heni, Imen Hmaied « Mise en place d'une plateforme de télécommande des équipements électrique à distance 'Smart House' » mémoire de Licence Université virtuelle de Tunis 2011.
- [4] http://igm.univ-mlv.fr/~dr/XPOSE2007/aessaidi-ndiop_LA-DOMOTIQUE/reseaux.htm. Date d'accès: 20/05/2018
- [5] <https://www.commentcamarche.com/contents/1113-ethernet>. Date d'accès : 05/06/2018
- [6] <https://www.commentcamarche.com/contents/222-environnement-client-serveur>. Date d'accès : 25/05/2018
- [7] « Les automatismes des bâtiments, La domotique, Le maintien à domicile » Catalogue SIRLAN technologies, Eybens, France, <http://www.sirlan.com>
- [8] Cours en électronique numérique « Le microcontrôleur », Section Génie électrique, Lycée Sarda Garriga, lile, France 2016
- [9] Support de cours « Microcontrôleurs » 1^{ère} Année Master, Département de génie électrique, Université Mohamed Khider de Biskra, 2013-2014
- [10] <https://www.gotronic.fr/art-carte-arduino-uno-12420.htm>. Date d'accès : 03/05/2018
- [11] Abdoune Fahim et Saadi Brahim, « Réalisation d'un système embarqué d'acquisition de donné pour l'analyse et la diagnostique d'un système photovoltaïque » Mémoire de Master, Université Abderrahmane Mira – Bejaia, 2015/2016.
- [12] Djafri Menad et Chelouche Djalal, « Etude et Réalisation d'une Carte Arduino » Mémoire de Master en électronique, Université A.Mira de Bejaia 2016.
- [13] <https://www.carnetdumaker.net/articles/utiliser-un-capteur-de-temperature-et-dhumidite-dht11-dht22-avec-une-carte-arduino-genuino/#le-capteur-dht22-et-son-petit-frere-le-dht11>. Date d' accès : 02/06/2018
- [14] Capteur Température/Humidité DHT11, livre APPRENDRE LE LANGAGE C EN S'AMUSANT AVEC ARDUINO. Vendredi, 25 Mars, 2016
- [15] <https://www.framboise314.fr/scratch-raspberry-pi-composants/module-detecteur-de-presence-pir/> date d' accès : 29/05/2018.

- [16] <http://www.circuitbasics.com/setting-up-a-5v-relay-on-the-arduino/> date d' accès :
01/06/2018
- [17] Johan Rungette, document support de la formation « Arduino applications distantes »
Académie d'Aix Marseille.
- [18] <https://www.commentcamarche.com/contents/520-le-protocole-http> date d' accès :
03/06/2018
- [19] <https://www.commentcamarche.com/contents/491-structure-d-un-document-html> date
d' accès : 03/06/2018