

Republique Algérienne Démocratique et Populaire  
Ministère de l'enseignement supérieur et de la recherche scientifique  
UNIVERSITÉ MOULOUD MAMMARI DE TIZI OUZOU



FACULTÉ DU GÉNIE ÉLECTRIQUE ET D'INFORMATIQUE  
DÉPARTEMENT D'ÉLECTRONIQUE

MÈMOIRE DE FIN D'ÉTUDES PRÉSENTÉ POUR  
L'OBTENTION DU GRADE DE MASTER

Domaine : Science & Technologie  
Filière : Électronique  
Spécialité : Électronique Industrielle

Présenté par :  
M. BOUAKLINE NASSIM & M. BENCHABA AGHILES

Thème :

## CONCEPTION ET RÉALISATION D'UNE MAISON INTELLIGENTE VIA RASPBERRY PI 3

*Mémoire soutenu publiquement le : 01 / 10 / 2018 devant le jury compose de :*

Nom & Prénom	Grade	Université	En qualité de :
M. TAHANOUT	MCA	UMMTO	Président
M. Y. ATTAF	MCA	UMMTO	Examineur
M. K. BENAMANE	MCA	UMMTO	Encadreur

# Sommaire

<b>Remerciement</b>	v
<b>Dédicace</b>	vi
<b>1 La Domotique</b>	<b>2</b>
1.1 Définition	2
1.2 Son fonctionnement	3
1.2.1 Les capteurs :	4
1.2.2 les caractéristiques d'un capteur	4
1.2.3 Les types des capteurs :	4
Les Capteurs actifs :	5
Les capteurs passifs :	6
1.2.4 Les familles des capteurs :	6
Capteur TOR :	6
Capteurs analogiques :	6
Capteurs numérique :	7
1.2.5 Actionneurs :	7
1.2.6 Relation capteurs actionneurs :	7
1.3 Maison connectée :	8
1.4 Domotique et maison connectée	8
1.5 Les fonctions de la domotique	8
1.5.1 La fonction de sécurité	9
1.5.2 La fonction de surveillance	9
1.5.3 La fonction de gestion d'énergie	10
1.5.4 La fonction de scénarisation	10
1.5.5 La fonction de communication	11
1.5.6 La fonction de confort	11
1.6 Les technologies	12

1.6.1	La technologie x10	12
1.6.2	La technologie InOne :	13
1.6.3	Le système Chacon :	13
1.7	Les avantages et les inconvénients :	14
1.7.1	Les avantages :	14
1.7.2	Les inconvénients :	14
1.8	Conclusion	15
<b>2</b>	<b>la carte Raspberry Pi</b>	<b>16</b>
2.1	Présentation du Raspberry Pi :	16
2.1.1	Caractéristiques du Raspberry Pi utilisé :	17
2.1.2	Accessoires du Raspberry Pi	18
	Alimentation	18
	Carte microSD	18
2.1.3	Vus d'ensemble :	19
2.1.4	Que peut on faire avec un Raspberry Pi ?	19
2.1.5	Connecteurs et ports disponible :	19
2.2	Linux	21
2.2.1	NOOBS :	22
2.2.2	Définition	22
2.2.3	Comment ça marche ?	22
2.2.4	Raspbian	23
	Caractéristiques	23
	Raspbian Stretch	23
2.2.5	Installation et démarrage du système	24
2.3	Prendre le contrôle du Raspberry Pi	24
2.3.1	SSH	25
	Activer SSH	25
	Installation de SSH	26
2.3.2	VNC	27
2.4	Python :	27
2.4.1	Point fort du langage :	28
2.4.2	Programmation des E/S avec python	29
2.5	Conclusion	30

<b>3 Conception et Réalisation</b>	<b>31</b>
3.1 Composants	31
3.2 Le DTH11 :	32
3.2.1 Introduction :	32
3.2.2 DTH11 :	32
3.2.3 Caracteristiques	33
3.2.4 Principe de fonctionnement	33
3.2.5 Branchement du DH11	34
3.2.6 Script utilisé	35
3.3 Servomoteur	35
Un servomoteur est composé :	36
3.3.1 Commander un servo-moteur	36
3.3.2 Branchement du servo moteur	38
3.3.3 Script utilisé	38
3.4 PIR C-7288	39
3.4.1 Caracteristiques	40
3.4.2 Fonctionnement	40
3.4.3 Branchement	40
3.4.4 Scripts utilisé	40
3.5 Moteur CC	41
3.5.1 Caractéristiques	42
3.5.2 Branchement	43
3.5.3 Scripts utilisé	43
3.6 Capteur MQ9	44
3.6.1 Fonctionnement	45
3.6.2 Branchement	45
3.6.3 Scripts utilisé	46
3.7 Les LEDs	46
3.7.1 Branchement	46
3.7.2 Script utilisé	47
3.8 Contrôle à distance	47
3.8.1 Apache	47
Installation d'Apache	48
3.8.2 PHP	48
Installer PHP	49
PHP contrôle les GPIOs	49

<b>3.9 Conclusion</b> . . . . .	50
<b>A</b>	<b>52</b>
<b>B Branchements du circuit</b>	<b>56</b>
<b>B.0.1 Branchement du DH11</b> . . . . .	56
<b>B.0.2 Branchement du servo moteur</b> . . . . .	57
<b>B.0.3 Branchement capteur de mouvement</b> . . . . .	57
<b>B.0.4 Branchement moteur CC</b> . . . . .	58
<b>B.0.5 Branchement capteur de gaz</b> . . . . .	58
<b>B.0.6 Branchement éclairage</b> . . . . .	59

# Remerciement

*Nous tenons tout d'abord à remercier Dieu le tout puissant et miséricordieux, qui nous a donné la force et la patience d'accomplir ce Modeste travail.*

*Nos remerciements s'étendent également à notre encadreur M.*

*BENNAMANE KAMEL pour ses bonnes explications qui nous ont éclairé le chemin de la recherche et sa collaboration avec nous dans l'accomplissement de ce modeste travail.*

*Ainsi que Nos vifs remerciements vont également aux membres du jury pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail Et de l'enrichir par leurs propositions. Enfin, on remercie tous ceux qui, de près ou de loin, ont contribué à la réalisation de ce travail.*

# Dédicace

*Je dédie cet humble et modeste travail à mes chers parents qui n'ont jamais cessé de formuler des prières à mon égard, de me soutenir et de m'épauler pour que je puisse atteindre mes objectifs. à mes chers Frères et sœurs et leurs époux (ses) et enfants, pour leur soutien moral et leurs conseils précieux tout au long de mes études. à mon binôme, pour son entente et sa sympathie. à tout mes amis sans exception, qui m'ont toujours apporté leur aide et encouragements. à tous ceux qui ont aidé de près ou de loin à réaliser ce travail. à tous ceux que j'aime, et tous ceux qui m'aiment.*

**Nassim**

*Je dédie ce modeste travail et ma profonde gratitude à ma mère et mon père pour l'éducation qu'ils m'ont prodigué ; avec tous les moyens et au prix de toutes les sacrifices qu'ils ont consentis à mon égard, pour le sens du devoir qu'ils m'ont enseigné depuis mon enfance.*

*A la famille toute entière*

*A mon binôme*

*Ma gratitude la plus profonde à mes amis et frères présents, ou absents. A tout le corps enseignant, administratif, et le personnel de l'UMMTO.*

# Introduction générale

Dans la vie au quotidien, la technologie est de plus en plus présente. Les smartphones et les ordinateurs dominent notre vie quotidienne.

Les volets roulants électriques et les portes de garage automatiques deviendraient dans les années avenir très courant dans notre vie de tous les jours. Puisque depuis quelques années, on observe une nouvelle tendance s'imposer et intégrer l'âme même de nos habitation, qui est la domotique.

Chaque être humain rêve d'avoir un logement encore plus confortable, une bonne température dans chaque pièce, un éclairage adapté, une surveillance à distance...

En domotique, de nombreuses fonctions sont contrôlées via un smartphone ou une tablette, augmentant ainsi le confort de ses résidents. La prise en main est conçu pour être très facile. Ainsi, la domotique n'est pas seulement un jouet pour les passionnés de technologie, mais aussi pour les personnes moins douées, la possibilité de contrôler leur maison. En domotique, deux domaines sont au centre de l'attention : la sécurité et le confort des résidents.

La domotique ouvre non seulement de nouvelles possibilités dans le domaine de l'automatisation de l'habitation, mais constitue aussi et surtout un moyen offert à l'individu de contrôler et de gérer son environnement. Grâce à cette nouvelle technologie, l'habitant sera à même de mieux gérer son milieu de travail et de vie sur le plan de la sécurité, du confort, des communications et des applications ménagères.

Notre projet consiste à réaliser un prototype d'une maison intelligente avec la carte Raspberry Pi qui ne cesse de donner des solutions facile pour le domaine de la domotique. D'abord, on va commencer avec une présentation générale de la domotique et ses fonctionnalités et les différents types de technologies utilisées.

Le second chapitre : est dédiée à la description du raspberry pi 3, son domaine d'utilisation et nous allons faire aussi un petit aperçu sur le langage python qui contrôle les GPIO de cette carte.

Pour finir, nous aborder en détails la réalisation électronique de notre maison intelligente accompagnée de la programmation des objets reliés à notre carte Raspberry à savoir : les LEDs, les moteurs, les capteurs, tout ceci contrôlés via une page web.

# Chapitre 1

## La Domotique

### Introduction :

Tout le monde rêve de pouvoir automatiser plusieurs fonctions de sa maison (allumer la lumière, ouvrir la porte du garage ...) sans bouger le petit doigt. Il s'est ainsi développé un domaine particulier de l'automatique dédié à l'usage de la maison appelé domotique. la domotique existe depuis longtemps et pourtant elle reste très peu connue du public elle reste marginalisée en Algérie, pourtant elle serait d'une grande aide dans la vie au quotidien surtout pour les personnes à mobilité réduite (personnes âgées, handicapées, etc.) Dans ce chapitre on parlera en générale sur la domotique, son fonctionnement, l'appareillage utilisé.

### 1.1 Définition

Le terme est un peu obscur mais sa définition nous éclaire sur les propriétés de la domotique. Ce mot issu du latin "Domus", qui signifie maison, regroupe l'ensemble des techniques et technologies électroniques, informatiques et des télécommunications permettant d'automatiser et d'améliorer les tâches au sein d'une maison. Les appareils de la maison sont intégrés au sein de systèmes qui doivent communiquer entre eux afin de gérer des automatismes. Car la domotique milite avant tout pour l'amélioration du quotidien au sein de notre habitat. la domotique vise à apporter des solutions techniques pour répondre aux besoins de confort, de sécurité et de communication que l'on peut trouver dans les maisons, les hôtels, les lieux publics...etc [2].

À l'origine, la domotique avait donc pour but d'automatiser sa maison : ouverture et fermeture automatiques des volets, ouverture du portail électrique, gestion du chauffage, gestion de l'éclairage, etc. Ainsi avant l'ère des smartphones, il était par exemple possible d'activer son chauffage à distance en passant un coup de téléphone à sa maison, ou encore

en lui envoyant un SMS. C'était tout à fait réalisable. Seulement une telle installation était relativement compliquée à mettre en place et, il faut bien l'avouer, couteuse. Cette époque a malheureusement laissé des traces, puisque pour beaucoup encore aujourd'hui, domotique rime avec cher et compliqué. Pourtant, ce domaine a énormément évolué et il existe de nombreuses solutions simples à mettre en place et tout à fait abordables pour le grand public [2]. La domotique a surtout elle-même évolué, si bien que le terme est quelque peu dépassé. La domotique servait à automatiser sa maison ; aujourd'hui on parle de domotique 2.0, ou de « maison intelligente », pour bien marquer l'évolution de ce monde. Les différents domaines de la maison ne se contentent plus d'être automatisés et pilotables, ils communiquent ensemble, permettant à la maison de réagir selon différents événements [2].



FIGURE 1.1 – Domotique

## 1.2 Fonctionnement d'un système domotique :

Le principe de fonctionnement d'un système domotique contient certains dispositifs pour récupérer les informations pour commander une maison, et pour cela un système domotique fait appel aux capteurs et aux actionneurs.

### 1.2.1 Les capteurs :

Un capteur est un dispositif transformant une grandeur physique à une autre grandeur utilisable, tel qu'une tension électrique ou une intensité ; On fait souvent la confusion entre capteur et transducteur : le capteur est au minimum constitué d'un transducteur. Le capteur est souvent le premier dispositif de la chaîne d'acquisition, il s'agit d'une simple interface entre un processus physique et une information manipulable.

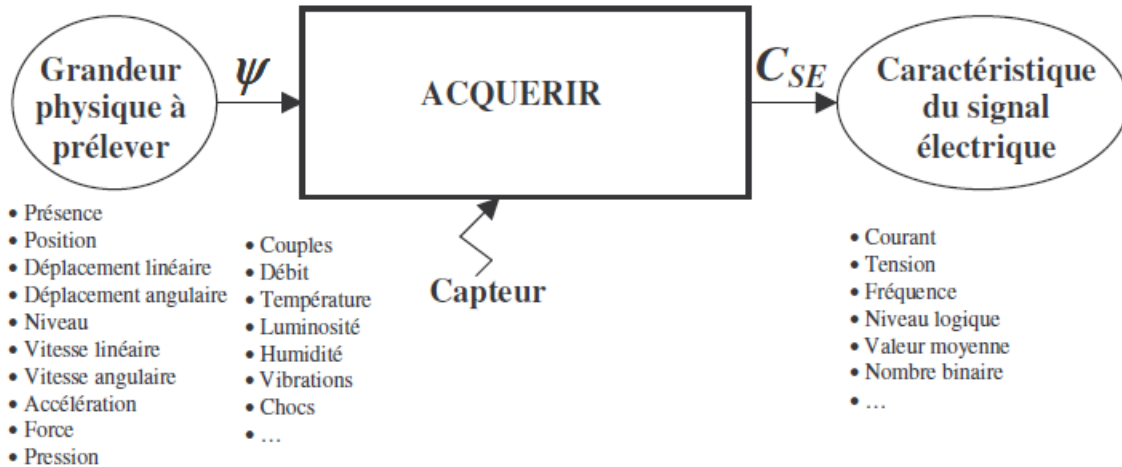


FIGURE 1.2 – Schéma Fonctionnel d'un capteur

### 1.2.2 les caractéristiques d'un capteur

- Etendue de mesure : Valeurs extrêmes pouvant être mesurée par le capteur.
- Résolution : Plus petite variation de grandeur mesurable par le capteur.
- Sensibilité : Variation du signal de sortie par rapport à la variation du signal d'entrée.
- Précision : Aptitude du capteur à donner une mesure proche de la valeur vraie.
- Rapidité : Temps de réaction du capteur. La rapidité est liée à la bande passante.
- Linéarité : représente l'écart de sensibilité sur l'étendue de mesure

### 1.2.3 Les types des capteurs :

Si l'on s'intéresse aux phénomènes physiques mis en jeu dans les capteurs, on peut classer ces derniers en deux catégories :

- Capteurs actifs.
- Capteurs passifs.

**Les Capteurs actifs :**

Fonctionnant en générateur, un capteur actif est généralement fondé dans son principe sur un effet physique qui assure la conversion en énergie électrique de la forme d'énergie à la grandeur physique à prélever : énergie thermique, mécanique ou de rayonnement[5].

Mesurande	Énergie propre d'une Mesurande	Principe physique	Grandeur de sortie
Temperature	Énergie thermique	Effet thermoélectrique	Tension
Flux lumineux	Énergie électromagnétique	Effet photovoltaïque	Charge
Force	Énergie mécanique	Effet piézoélectrique	Courant
Pression	Énergie mécanique		Tension
Accélération	Énergie mécanique		Charge
Vitesse	Énergie mécanique	Effet électromagnétique	Tension
Position	Énergie mécanique	Effet hall	Tension

TABLE 1.1 – Tableau des capteurs actifs

**Les effets physique les plus classiques sont :**

- **Effet thermoélectrique** : Un circuit formé de deux conducteurs de nature chimique différente, dont les jonctions sont à des températures  $T_1$  et  $T_2$ , est le siège d'une force électromotrice d'origine thermique  $e(T_1, T_2)$ .
- **Effet piézo-électrique** : L'application d'une contrainte mécanique à certains matériaux dits piézo-électriques (le quartz par exemple) entraîne l'apparition d'une déformation et d'une même charge électrique de signe différent sur les faces opposées.
- **Effet d'induction électromagnétique** : La variation du flux d'induction magnétique dans un circuit électrique induit une tension électrique (détection de passage d'un objet métallique).
- **Effet photo-électrique** : La libération de charges électriques dans la matière sous l'influence d'un rayonnement lumineux ou plus généralement d'une onde électromagnétique.
- **Effet Hall** : Un champ magnétique  $B$  et un courant électrique  $I$  créent dans le matériau une différence de potentiel  $U$  (Tension de Hall).
- **Effet photovoltaïque** : Des électrons et des trous sont libérés au voisinage d'une jonction  $PN$  illuminée, leur déplacement modifie la tension à ses bornes.

**Les capteurs passifs :**

Ils ont besoin dans la plupart des cas d'apport d'énergie extérieure pour fonctionner. Ce sont des capteurs modélisables par une impédance. une variation du phénomène physique étudié engendre une variation de l'impédance. il faut leur appliquer une tension pour obtenir un signal de sortie [5].

Mesurable	Caractéristique électrique sensible	Type de matériaux utilisés
Température	constante diélectrique	Metaux : Plaine, nickel, cuivre...
Flux lumineux	Résistivité	Semi-conducteur
Déformation	perméabilité magnétique	Alliages de nickel, silicium
Position	Résistivité	Matériaux magnéto résistants : bismuth, antimoine d'indium
Humidité	constante diélectrique	Alumine, polymères
Niveau	constante diélectrique	Liquides isolants

TABLE 1.2 – Tableau des capteurs actifs

**1.2.4 Les familles des capteurs :**

Les capteurs sont des composants d'automatisme qui ont pour but de récolter une information sur la partie opérative et de la retransmettre à la partie commande qui pourra ainsi la traiter, et on trouve trois familles essentielle :

**Capteur TOR (tout ou rien) :**

Les capteurs Tout ou Rien (TOR) délivrent une information binaire à la partie commande : l'information adopte l'état 0 ou l'état 1. Chaque état possède une signification dans le contexte du système. On distingue essentiellement les capteurs de type mécanique et ceux de proximité (cellules, inductifs ou capacitifs).

ils sont constitués :

- d'un élément sensible mécanique ou électrique.
- d'un ou plusieurs contacts de type normalement fermé ou de type normalement ouvert. [5]

**Capteurs analogiques :**

La sortie est une grandeur électrique dont la valeur est une fonction de la grandeur physique mesurée par le capteur. La sortie peut prendre une infinité de valeurs continues. Le

signal des capteurs analogiques peut être du type :

- sortie tension ;
- sortie courant ;
- règle graduée, cadran, jauge (avec une aiguille ou un fluide) ; [5]

### Capteurs numérique :

La sortie est une séquence d'états logiques qui, en se suivant, forment un nombre. La sortie peut prendre une infinité de valeurs discrètes. Le signal des capteurs numériques peut être du type :

- train d'impulsions, avec un nombre précis d'impulsions ou avec une fréquence précise ;
- code numérique binaire ;
- bus de terrain ; [5]

### 1.2.5 Actionneurs :

Dans une machine ou un système de commande, semi-automatique ou automatique, un actionneur est l'organe de la partie opérative qui, dès qu'il reçoit un ordre de la partie commande via un éventuel capteur ou pré-actionneur, convertit l'énergie qui lui est fournie en un travail utile à l'exécution de tâches, éventuellement programmées, d'un système automatisé. En d'autres termes, un actionneur est l'organe fournissant la force nécessaire à l'exécution d'un travail ordonné par une unité de commande.

il existe trois types d'actionneur :

- Actionneur électrique.
- actionneur pneumatique.
- actionneur hydraulique.

### 1.2.6 Relation capteurs actionneurs :

Les capteurs sont des informateurs traducteurs, ils détectent une variation de l'environnement de la partie opérative et la traduit en une information interprétable (grandeur électrique) par la partie commande a fin de contrôler les actionneurs qui transforme une forme d'énergie en une autre pour but d'obtenir l'énergie souhaité (énergie mécanique). comme le montre la figure ci-dessous.



FIGURE 1.3 – relation capteurs actionneurs

### 1.3 Maison connectée :

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

### 1.4 Différence entre domotique et maison connectée

La domotique facilite le quotidien mais avec l'intervention de l'Homme (appuyer sur des boutons pour fermer les volets roulants, gérer votre climatiseur avec la télécommande...etc.). La maison intelligente connaîtra déjà certaines de vos habitudes et s'adaptera pour vous, on parle alors de « machine learning » ou d'intelligence artificielle.

### 1.5 Les fonctions de la domotique

L'utilisation de plus en plus importante des Smartphones et des Tablettes contribue à favoriser l'acceptation de la domotique au sein de l'habitat.

Les domaines d'application sont au cœur de la vie quotidienne. Les fonctions suivantes peuvent être réalisées grâce aux technologies intégrées dans la domotique :

- Sécurité,
- Surveillance,
- Gestion de l'énergie,

- Scénarisation des actions,
- Communication,
- Confort.

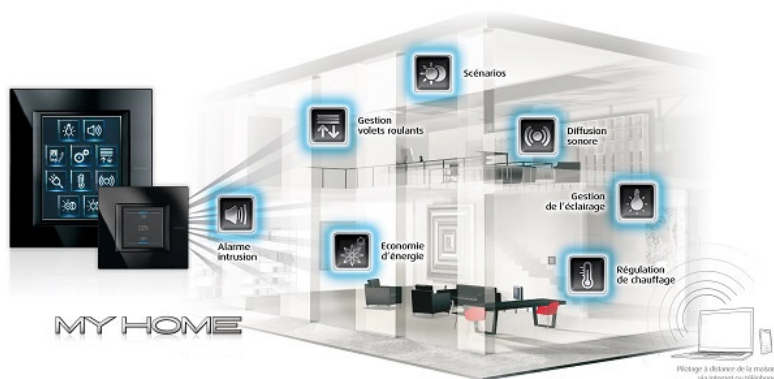


FIGURE 1.4 – Différentes fonctions de la domotique

### 1.5.1 La fonction de sécurité

Contrairement à un système d'alarme traditionnel, une centrale domotique agit sur toute l'installation électrique de l'habitation. Elle dissuade les intrus en simulant une présence par l'allumage aléatoire des éclairages, de la radio ou l'ouverture des volets durant la journée, ... Les accès à un logement ou à un commerce sont contrôlés et enregistrés et, si une intrusion est détectée, la centrale prend les mesures qui s'imposent :

- Sirène,
- Allumage de tous les éclairages de la maison,
- Appel d'un centre de surveillance, d'un voisin ou d'un téléphone mobile.

À partir d'un téléphone, il est possible d'écouter et de s'adresser directement aux intrus grâce à l'Haut-parleur de la centrale. En termes de sécurité, la domotique permet entre autres de :

- Prévenir les risques provenant de l'extérieur (intrusion, cambriolage ...) comme ceux provenant de l'intérieur (accidents domestiques) ;
- Surveiller à distance vos locaux ou votre habitation depuis votre Smartphone, une Tablette ou un ordinateur ;
- Être averti d'un incident (alarmes techniques) par SMS et/ou par courriel.

### 1.5.2 La fonction de surveillance

Différents capteurs détectent les anomalies :

- Inondation,


- Incendie,
- Fuite de gaz,
- Arrêt du congélateur,
- Coupure de courant,
- Vent ou pluie.

La centrale intervient instantanément pour couper les alimentations, remonter les stores, couvrir la piscine, appeler les numéros d'urgence ou faire retentir la sirène si l'occupant est présent.

### 1.5.3 La fonction de gestion d'énergie

La programmation des seuils de température est l'une des principales sources d'économie. La programmation peut être journalière ou hebdomadaire. Des sondes de température renseignent la centrale sur les valeurs de température dans chaque pièce. Elle tient compte de la présence d'un occupant et s'adapte automatiquement à son emploi du temps. Dans un souci de confort un simple appel téléphonique suffirait pour augmenter le chauffage et de préparer l'habitat en prévision de l'arrivée des occupants.

Le délesteur électrique est un petit appareil qui va gérer automatiquement l'alimentation électrique des appareils. Son but est de permettre de réduire le montant de la facture d'électricité en jouant sur le prix de l'abonnement.

Lorsque la consommation risque de dépasser le seuil de puissance souscrite dans le contrat d'abonnement, le délesteur va couper l'alimentation électrique des circuits jugés non prioritaires, mais conservera les circuits prioritaires alimentés. 

- Extinction des éclairages inutiles,
- Réglage de l'intensité lumineuse en fonction de l'activité, ...

La domotique propose ainsi de réduire les consommations énergétiques des bâtiments en adaptant ces consommations aux modes de vie des occupants et à l'environnement extérieur. Cela comprend la régulation de l'éclairage et du chauffage, le traitement de l'air, l'optimisation des ouvrants, la programmation horaire, les commandes à distance, les interrupteurs automatiques pour l'éclairage d'un escalier ou d'un couloir, l'ouverture ou la fermeture d'un volet selon l'ensoleillement...

### 1.5.4 La fonction de scénarisation

Au moment de quitter un habitat ou un commerce, la mise en fonction de l'alarme déclenche une série de contrôles et d'actions, (centralisation des commandes) :

- Fermeture de toutes les lumières,

- Coupure de l'arrivée de gaz,
- Vérification de la fermeture de toutes les fenêtres,
- Allumage de la lumière extérieure durant quelques minutes s'il fait nuit, ...

À partir d'un bouton unique, tous les éclairages du living seront ajustés pour le dîner, une soirée télévision ou la création d'une ambiance lumineuse adaptée à l'activité de l'occupant.

### 1.5.5 La fonction de communication

Aujourd'hui, une centrale domotique sait communiquer

- Par téléphone,
- Par ordinateur (Internet),
- Par des interfaces tactiles,
- Par le biais de microphones permettent une activation par commande vocale associés à des logiciels de reconnaissance vocale.

Ceci permet à une personne de recevoir l'état de son installation et d'émettre des alertes et piloter sa maison de n'importe quel endroit du monde, de son bureau ou de sa voiture. Pouvoir diffuser le son et l'image dans votre logement sont des exemples de scénarios qu'offre la domotique. Les équipements vidéo, home-cinéma, réseau téléphonique et internet sont intégrés dans toutes les pièces de votre habitation. Il est possible de gérer et diffuser ses bibliothèques de musiques et de vidéos dans différentes pièces, de sauvegarder ses données informatiques, d'avoir accès à distance à ses ordinateurs, de faciliter la mobilité et le télétravail. Ces systèmes sont en général indépendants et peuvent être pilotés par les fonctions domotiques.

### 1.5.6 La fonction de confort

Ouvrir le portail sans descendre de voiture, allumer automatiquement la lumière en ayant les bras chargés de paquets, arroser automatiquement le jardin, ouvrir ou fermer les volets ou les stores, programmer une ambiance sonore dans la maison, disposer de la télévision dans plusieurs pièces, déclencher la préparation du café avant le réveil, adapter la température aux conditions extérieures, ... autant de fonctionnalités qui peuvent devenir indispensables.

Un émetteur radio, une télécommande infrarouge, un téléphone, une horloge ou un détecteur sont différents dispositifs de commande qui serviront de moyens pour assurer ces fonctions.

## 1.6 Les technologies utilisées pour la domotique :

Généralement, une installation domotique peut être conçue sur trois principaux types de technologie. Ces technologies peuvent cohabiter, être superposées suivant l'évolution de l'installation dans le temps.

### 1.6.1 La technologie x10

Est une technologie CPL<sup>[1]</sup> créé dans les années 80. Elle est très utilisée aux *Etats-Unis*, mais les modules américains ne sont malheureusement pas compatibles avec le réseau 50HZ (*France, Afrique du Nord*) à cause de la différence de tension secteur/fréquence. En plus de la communication CPL, il existe une partie RF<sup>[2]</sup> qui permet grâce à un traducteur RF/X10 permet de piloter des équipements directement à partir d'une télécommande.

Les inconvénients :

- Lenteur : Les ordres émis en x10 mettent parfois 1s avant d'être exécutés
- Fiabilité : Le manque de retour d'état ne permet pas de contrôler qu'un ordre a été correctement exécuté, et il arrive parfois que certains ordres ne soit pas pris en compte.

Les avantages :

- Moins cher que les autres technologies et facile à installer.



FIGURE 1.5 – Logo X10

---

1. Technologie permettant de faire circuler des informations numériques en passant par les lignes électriques. Les données sont encodées sur le réseau électrique en se basant sur la fréquence du signal électrique (50Hz) et en superposant un autre signal de fréquence plus élevé.

2. Onde Radio (RF) : La technologie radio qui permet d'interconnecter des équipements L'ajout de capteur ou d'actionneur. Plus la fréquence est élevée, plus la vitesse de transmission est grande, mais plus la distance de transmission est faible.

### 1.6.2 La technologie InOne de Legrand :

Est une technologie propriétaire. Elle ne peut s'interfacer directement avec les autres technologies.

La gamme *InOne* regroupe des modules CPL, RF et IR<sup>3</sup>. A noter que tous les modules d'une même catégorie sont compatibles entre eux, mais il faut des modules "traducteurs" pour passer de la radio en CPL ou d'IR en CPL par exemple.

Les inconvénients :

- Impossible d'interagir facilement avec les autres technologies. Même si cette technologie est d'un excellent rapport qualité/prix, elle reste chère par rapport au x10 ou PLCBUS<sup>4</sup>

Les avantages :

- Design très travaillé et Technologie Legrand reconnue pour sa qualité



FIGURE 1.6 – Logo Legrand

### 1.6.3 Le système Chacon :

Est un ensemble domotique basé sur la RF. Tous les émetteurs du système *Chacon* peuvent contrôler les récepteurs. Par exemple, il est possible de contrôler un éclairage à partir d'un interrupteur sans fils, d'une télécommande, d'un détecteur de mouvement.

Les inconvénients :

- Il est parfois nécessaire d'ajouter des relais pour agrandir la distance de transmission des différents modules

Les avantages :

- Très facile à installer
- Le prix des différents modules est d'un très bon rapport qualité / prix.
- Nombreux modules permettant de gérer la maison (lumière, prises...)
- Système évolutif permettant de compléter facilement le système domotique.

---

3. Technologie infra-rouge est connue pour être utilisée dans les télécommandes de nos téléviseurs, A partir d'une émission lumineuse elle est donc parfois utilisée pour faire communiquer des systèmes domotique.

4. Est une autre technologie de domotique basée sur la technologie CPL développée dans la fin des années 90.



FIGURE 1.7 – Logo Chacon

## 1.7 Les avantages et les inconvénients :

Comme tous les projets la domotique contient des avantages et des inconvénients parmi ces derniers on cite :

### 1.7.1 Les avantages :

- Le principal avantage de la domotique est l'amélioration du quotidien au sein de la maison, du point de vue du confort, de la sécurité et de la gestion de l'énergie.
- Ce type d'équipement vous simplifie la vie et optimise votre confort en adaptant votre maison à différents scénarios de la vie quotidienne.
- Il vous permet notamment d'éteindre tous vos appareils électriques et de mettre l'alarme quand vous quittez votre domicile, de régler des ambiances lumineuses (ambiance lecture, ambiance relaxation avec lumières tamisées), de vous réveiller dans un habitat chauffé où le café est déjà prêt.
- La domotique permet aussi de réaliser des économies d'énergie grâce à la gestion automatique du chauffage, de la climatisation et de l'éclairage et à la programmation des appareils électroménagers en heures creuses.
- Elle a pour avantage d'améliorer la sécurité grâce à des alarmes, des systèmes d'ouverture automatique de la porte (reconnaissance vocale, carte magnétique...)
- En cas de tentative d'intrusion dans la maison, un appel téléphonique automatique peut contacter le propriétaire ou une entreprise de sécurité.
- Enfin, ces différentes technologies constituent une aide précieuse pour les personnes dépendantes et handicapées.

### 1.7.2 Les inconvénients :

- Le principal est le prix d'achat et d'installation pour certaines technologies, pourraient rebuter plus d'un qui est beaucoup plus élevé. Mais à long terme les factures d'énergie baisseront.

- Le deuxième inconvénient est le verrouillage qu’offrent certaines marques dans leurs produits ne permettant pas d’avoir un logiciel ouvert.
- Des intrusions via le réseau internet par des personnes tierces par exemple les Cracker (Pirate informatique), pourraient espionner la vie privée ou dérégler l’installation.
- Toujours dans le même contexte avoir les commandes et un accès illimité aux maisons d’autrui, faciliterais un éventuelle actes criminels (cambriolages, autres actes encore plus violent).

## 1.8 Conclusion

Dans ce chapitre nous avons aborder toutes les notions de domotique, le matériel utilisé avec les différentes technologies disponibles.

Dans le chapitre qui va suivre, nous allons voir en détails le Raspberry Pi, ses composants, son domaine d’utilisation. Ainsi que les systèmes d’exploitation qu’elle supporte et le langage informatique utilisé.

# Chapitre 2

## Description et préparation de la carte Raspberry Pi

### Introduction

Le Raspberry Pi est vite devenu un phénomène mondial. Les gens commencent à découvrir le potentiel d'un ordinateur polyvalent à 35€ qui peut servir de station de travail, de *media center*, ou bien encore de contrôleur d'un système domotique<sup>[3]</sup>.

### 2.1 Présentation du Raspberry Pi :

Raspberry est une carte mère d'un mini-ordinateur qui peut être branchée à n'importe quel périphérique (souris, clavier...). Cette carte est fabriquée pour aider à étudier les ordinateurs et pour représenter un moyen d'apprentissage de la programmation informatique en plusieurs langages (python, scratch...). Elle est aussi capable de lire les vidéos à haute définition et même à installer des jeux vidéo. L'intérêt d'utiliser le Raspberry Pi est sa capacité d'interaction avec le monde extérieur et d'exécuter plusieurs variantes du système d'exploitation libre (GNU/Linux, Raspbian Debian ...) et des autres logiciels compatibles.



FIGURE 2.1 – Raspberry pi 3 Model B

### 2.1.1 Caractéristiques du Raspberry Pi utilisé :

Le Raspberry Pi est un nano-ordinateur de la taille, grosso modo, d'une carte de crédit. Ce qui est impressionnant, ce sont ses caractéristiques. Sur un petit bout de carte électronique, il y a un :

- Processeur ARM Cortex A-53 quatre cœurs de 64-bit cadencé à 1,2 Ghz
- 1 Go de RAM (SDRAM)
- Un contrôleur vidéo Broadcom VideoCore IV
- 4 ports USB 2.0
- Un port HDMI
- Un connecteur Jack 3,5 mm / sortie composite
- Une prise Ethernet 10 / 100 Mbps
- WIFI 802.11n
- Bluetooth 4.1
- Bluetooth Low Energy (BLE)
- 40 pins GPIO
- Une interface de caméra (CSI)
- Une interface d'affichage (DSI)
- Un emplacement de carte micro SD
- Alimentation 5V par micro USB

La figure qui suit inventorie les différentes parties d'un Raspberry Pi 3 modèle B,

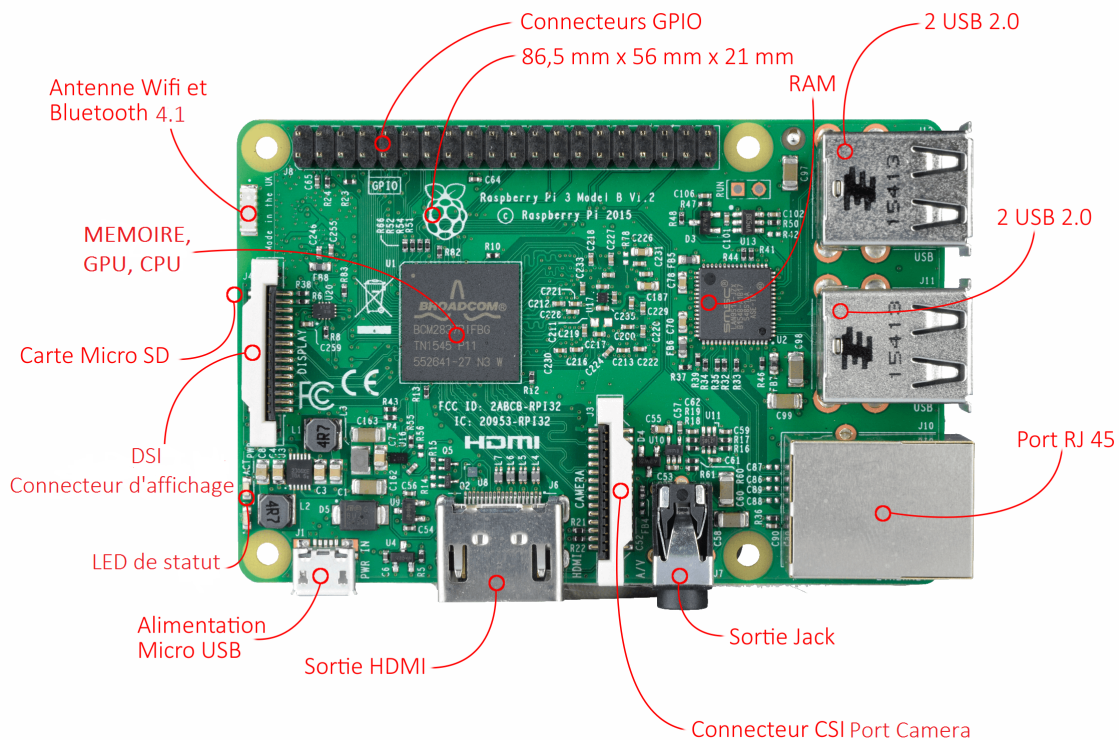


FIGURE 2.2 – Emplacement des différents composants du Raspberry Pi 3 Modèle B

## 2.1.2 Accessoires du Raspberry Pi

### Alimentation

le Raspberry Pi s'alimente sous tension unique de 5 volts, tension sur laquelle il peut consommer jusqu'à 1.8A selon les tâches qu'il exécute. cette alimentation doit être normalement fournie via le connecteur micro USB place dans un angle de la carte.

en effet, l'utilisation d'un chargeur pour Smartphone équipé d'un micro USB qui délivre une tension de 5 volts avec au minimum 1.8A est suffisante pour alimenter notre Raspberry Pi.

### Carte microSD

Pour héberger notre système d'exploitation (Raspbian) nécessaires au bon fonctionnement notre Raspberry Pi, et tous les fichiers nous auront besoin d'espace de stockage c'est pour cela qu'une **microSD** est requise.

La qualité de cette carte, et tout particulièrement sa vitesse, va influencer la vitesse d'écriture des fichiers. Il convient donc de bien la choisir. Pour nous aider des « classes » sont

définies pour nous indiquer la vitesse minimale des cartes en écriture, exprimée en Mo/s. Les cartes vendues en 2010 sont de classe 2, 4, 6 ou 10.

Au niveau de la taille, Raspbian étant très léger, il n'y a pas besoin d'une très grosse carte SD. 4Go suffisent, mais 16 Go sont préférables, afin d'avoir un peu de marge. Pour notre part on a choisit une capacité de 8Go pour notre projet.

### 2.1.3 Vus d'ensemble :

En réalité, et même si par abus de langage on parle du processeur qui équipe le Raspberry pi, il s'agit en fait de ce que l'on nomme aujourd'hui un SoC qui est l'acronyme de *system on chip* soit, mot à mot système sur une puce. Si vous savez ce qu'est un microcontrôleur, vous pouvez très bien considérer qu'un SoC n'est rien d'autre qu'un « gros » microcontrôleur !

### 2.1.4 Que peut on faire avec un Raspberry Pi ?

On peut pratiquement faire avec un Raspberry Pi tout ce que l'on peut faire avec un ordinateur de bureau sous Linux, à quelques exceptions près. Le Raspberry Pi utilise une carte SD à la place d'un disque dur, bien que l'on puisse le brancher à un disque dur USB. On peut modifier des documents bureautiques, surfer sur internet, jouer à des jeux, etc. Le Raspberry Pi est un appareil extrêmement souple qu'on peut utiliser dans toutes sortes de situations, que ce soit pour remplacer un ordinateur de bureau, profiter d'un *media center* ou contrôler un système à l'aide d'un ordinateur embarqué.

### 2.1.5 Connecteurs et ports disponible :

1. **HDMI** : « High Definition Multimedia Interface » permet de relier le Raspberry Pi à un dispositif compatible : écran LCD ou un vidéo-projecteur...
2. **GPIO** : est l'acronyme anglais de "General Purpose Input Output" ce que l'on traduirait pas Entrée/sortie pour usage général. Le connecteur GPIO dispose actuellement de différents type de connexions. Elles sont :
  - De vraies broches GPIO (General Purpose Input Output) avec lesquels vous pouvez allumer/éteindre des LED, etc.
  - Des broches pour une interface I2C qui permet de se connecter sur du matériel en utilisant uniquement 2 broches/pins de contrôle.
  - Une interface SPI<sup>[1]</sup> pour les périphériques SPI, a concept similaire à I2C<sup>[2]</sup> mais

---

1. Element important pour les systèmes embarqués, qui permet une communication entre un maître et un ou plusieurs esclaves.

2. Moyen très pratique pour communiquer avec de multiples périphériques

avec un standard différent.

— Les broches Rx et Tx pour la communication avec les périphériques séries.

De plus, certaines broches peuvent être utilisé en :

- PWM (acronyme de "Pulse Width Modulation" pour modulation de largeur d'impulsion) permettant le contrôle de puissance
- PPM (acronyme de "Pulse Position Modulation" pour modulation positionnelle d'impulsion) permettant de contrôler des servo moteurs.

Raspberry Pi GPIO Header					
Pin#	NAME		NAME	Pin#	
01	3.3v DC Power	⬇	DC Power 5v	02	⬆
03	GPIO02 (SDA1 , I <sup>2</sup> C)	⬇	DC Power 5v	04	⬆
05	GPIO03 (SCL1 , I <sup>2</sup> C)	⬇	Ground	06	⬆
07	GPIO04 (GPIO_GCLK)	⬇	(TXD0) GPIO14	08	⬆
09	Ground	⬇	(RXD0) GPIO15	10	⬆
11	GPIO17 (GPIO_GEN0)	⬇	(GPIO_GEN1) GPIO18	12	⬆
13	GPIO27 (GPIO_GEN2)	⬇	Ground	14	⬆
15	GPIO22 (GPIO_GEN3)	⬇	(GPIO_GEN4) GPIO23	16	⬆
17	3.3v DC Power	⬇	(GPIO_GEN5) GPIO24	18	⬆
19	GPIO10 (SPI_MOSI)	⬇	Ground	20	⬆
21	GPIO09 (SPI_MISO)	⬇	(GPIO_GEN6) GPIO25	22	⬆
23	GPIO11 (SPI_CLK)	⬇	(SPI_CE0_N) GPIO08	24	⬆
25	Ground	⬇	(SPI_CE1_N) GPIO07	26	⬆
27	ID_SD (I <sup>2</sup> C ID EEPROM)	⬇	(I <sup>2</sup> C ID EEPROM) ID_SC	28	⬆
29	GPIO05	⬇	Ground	30	⬆
31	GPIO06	⬇	GPIO12	32	⬆
33	GPIO13	⬇	Ground	34	⬆
35	GPIO19	⬇	GPIO16	36	⬆
37	GPIO26	⬇	GPIO20	38	⬆
39	Ground	⬇	GPIO21	40	⬆




FIGURE 2.3 – Ports GPIO

3. **Port USB 2.0** : Le port « Universal Serial Bus » est un port série qui sert à connecter le Raspberry aux autres périphériques, et il y a 4 ports USB. Le Raspberry Pi fonctionne avec pratiquement n'importe quels claviers et souris USB. Vous pouvez aussi utiliser la plupart des claviers et des souris sans fil (ceux qui marchent avec un dongle<sup>3</sup> que l'on branche sur un port USB).
4. **Port Ethernet** : C'est un port qui correspond au protocole international Ethernet de réseau local à commutation de paquets.

3. Composant matériel se branchant sur les ordinateurs ou les téléviseurs, généralement sur un port d'entrées-sorties. Il peut désigner toutes sortes de matériels comme des périphériques de stockage (clés USB), des clés permettant de se connecter à un réseau Wi-Fi, bluetooth, 3G ou infrarouge.

5. **Prise RCA** : « Radio Corporation of America » est un connecteur électrique utilisé dans le domaine audio/vidéo via jack.
6. **Un slot pour cartes microSD** : Le Raspberry a besoin d'une mémoire externe supplémentaire pour fonctionner. Ce slot permet de connecter la mémoire externe. Vous pouvez utiliser votre propre carte SD dans le Raspberry Pi, mais elle devra être préparée avec une image disque d'un système d'exploitation.
7. **Une prise jack** : C'est une connectique audio-vidéo sa dimension est égale à 3.5 mm.

## 2.2 Linux

Un ordinateur standard fonctionne avec un système d'exploitation comme *Windows*, *OS X* ou *Linux*. C'est ce système qui démarre quand vous allumez votre machine et qui fournit à vos logiciels un accès aux fonctionnalités matérielles de votre ordinateur. Par exemple, si vous codez une application qui utilise Internet, vous pouvez, pour ce faire, utiliser les fonctions fournies par le système d'exploitation. Vous n'avez pas besoin de comprendre et d'écrire du code pour chaque modèle d'interface *Ethernet* ou *Wi-Fi* disponible.

Comme tous les autres ordinateurs, le *Raspberry Pi* utilise un système d'exploitation. Celui fourni par défaut est une version de *GNU/Linux* appelée *Raspbian* qui est idéal pour l'esprit de partage qui règne au sein de la *communauté Raspberry Pi* car il est *libre*<sup>[4]</sup> et *open source*<sup>[5]</sup>. Ce logiciel a été écrit dans le cadre d'un projet communautaire pour ceux qui recherchent une alternative au monopole de *Microsoft Windows* et d'*Apple OS X*.

*Linux* est un système d'exploitation *Open Source*. c'est un système d'exploitation complet, basé sur les mêmes concepts robuste d'*UNIX* qui est apparu au début de l'informatique. il a de nombreux partisans fidèles et serviable et s'est transformé en un système d'exploitation puissant et facile à utiliser.

Bien que le système d'exploitation s'appelle *Linux*, plusieurs distributions (appelées aussi *distros*) Linux différentes ont été produites. celles-ci comprennent le même système d'exploitation de base, mais sont livrées avec des applications ou de systèmes de fenêtrage différents. Bien qu'il existe de nombreuses distributions, C'est la *Raspbian* qui est recommandée par la **Fondation Raspberry Pi**.<sup>[4]</sup>

---

4. Une application informatique dont l'accès est illimité par le grand public, tout personne a la possibilité de voir voire le code source (noyau), pouvoir le modifier a sa guise et le redistribuer.

5. Licence qui donne des possibilités d'accès au *code source*, de libre redistribution et de création de travaux dérivés d'un projet informatique. Mis à la disposition du grand public

### 2.2.1 NOOBS :

**Trouver son système d'exploitation grâce à NOOBS** *Lorsque l'on débute avec son Raspberry Pi, on se retrouve rapidement perdu lorsqu'il faut choisir un système d'exploitation sachant que sous Linux, il existe de nombreuses distributions différentes. Heureusement, le logiciel « NOOBS » est là pour nous.*

### 2.2.2 Définition

(« News Out Of the Box Software ») est un logiciel prévu pour tester facilement les systèmes d'exploitations classiques que l'on retrouve couramment sur *Raspberry Pi*. Est un assistant d'installation développé par *Raspberry Pi Foundation*, destiné à simplifier l'installation des utilisateurs débutants dans le **secteur** Linux .

- **Raspbian** : la célèbre distribution classique basée du *Debian*
- **Pidora** : une distribution basée sur *Fedora*
- **Open Elec** : pour transformer le Pi en centre multimédia
- **RaspBMC** : idem qu'*Open Elec*
- et d'autres comme *Arch OS* ou *Risc OS*...

Grâce à ce logiciel, il est possible d'installer plusieurs systèmes d'exploitations sur une seule et même carte **SD** : ainsi, nous pouvons facilement trouver celui qui est le plus adapté à nos besoins sans avoir à télécharger les différentes images nécessaires.

### 2.2.3 Comment ça marche ?

Après le téléchargement et décompression de l'archive NOOBS dans un dossier de notre choix, vient après l'étape de transfert de tout ces fichiers dans une carte *microSD* au préalable bien formatée avant utilisation, on devrait arriver sur cet écran qui permet de choisir parmi les distributions disponibles :



FIGURE 2.4 – Menu NOOBS

Il nous suffit alors de cocher les systèmes que nous souhaitons installer puis de cliquer sur « Install ». Au bout de quelques minutes, tout est installé et dès lors nous pouvons profiter de notre machine !

## 2.2.4 Raspbian

Raspbian est un système d'exploitation *GNU/Linux* spécialement conçu et optimisé pour le *Raspberry Pi*.<sup>[7]</sup>

### Caractéristiques

*Raspbian* est un *mot-valise*<sup>[6]</sup> formé par la fusion des mots "*Raspberry Pi*" et "*Debian*".



FIGURE 2.5 – Raspberry Pi + Debian = Raspbian

### Raspbian Stretch

La dernière version de l'*OS Raspbian* complet, plutôt adapté à la Raspberry Pi 2 et à la Raspberry Pi 3.<sup>[7]</sup>

---

6. Mot formé par la fusion d'au moins deux mots existant dans une langue, exemple : Brexit, de Britain (Grande-Bretagne) et exit.

Il s'agit d'une modification de *Debian* spécifiquement adaptée pour les systèmes sur une puce de type *ARMv6*.

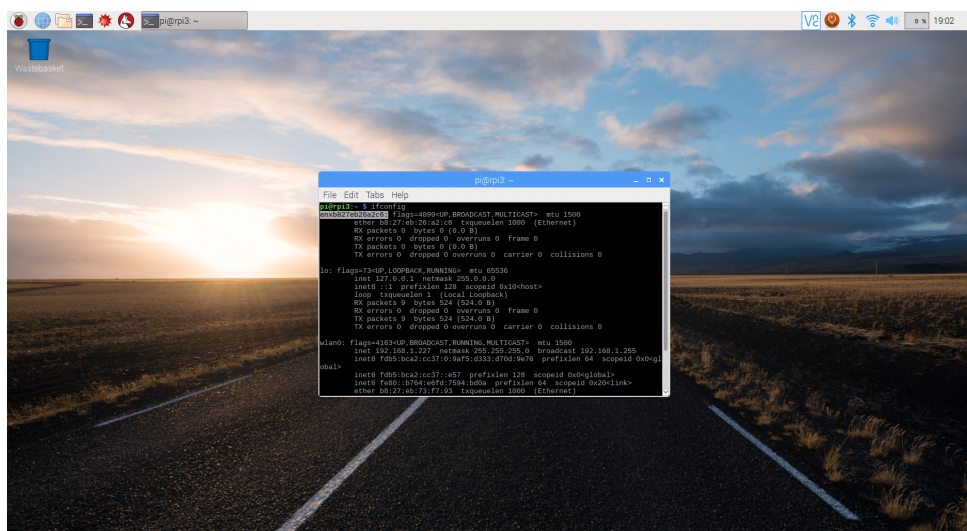


FIGURE 2.6 – Fond d'écran Raspbian

## 2.2.5 Installation et démarrage du système

Dans un premier temps nous aurons besoin de plusieurs choses. Tout d'abord, d'une carte microSD avec raspbian gravé dessus (Les étapes sont décrites dans la sous-section [2.2.3](#), page [22](#), qui fait référence à NOOBS). Par ailleurs il nous faudra également un clavier pour votre Raspberry Pi, et enfin un écran.

Suivez ces étapes pour démarrer votre Raspberry Pi pour la première fois.

- Insérez la carte SD dans son support.
- Branchez un clavier USB et une souris au Raspberry Pi.
- Connectez la sortie HDMI à votre téléviseur ou moniteur.
- Branchez l'alimentation du Pi.

## 2.3 Prendre le contrôle du Raspberry Pi

Brancher la Raspberry Pi à un écran et l'utiliser comme bureau c'est bien, mais pour divers projets ou celle-ci intervient par exemple dans les systèmes embarqués, un écran et tout les autres périphériques (clavier, souris) deviennent vite encombrant, des solutions existent c'est de communiquer avec des commandes et cela à distance avec des différents protocoles et serveurs.

### 2.3.1 SSH

SSH (pour Secure SHell) désigne à la fois un logiciel et un protocole de communication informatiques. Ce protocole possède par ailleurs la particularité d'être entièrement chiffré. Cela signifie donc que toutes les commandes que vous exécuterez via SSH seront totalement secrètes.

SSH permet de se connecter à distance sur une machine en utilisant un compte utilisateur de la dite machine. Pour se faire nous allons dans ce qui suit activer SSH dans notre Raspberry Pi et installer un client SSH dans notre ordinateur.

#### Activer SSH

Par défaut, SSH est installé sur la Raspberry Pi, mais est désactivé pour des raisons de sécurité. La première chose à faire sera donc d'activer SSH sur votre Raspberry Pi.

#### Procédure :

1. Démarrer le "RaspberryPI" sur la distribution "Raspbian".
2. Se connecter au terminal avec CTRL+ALT+T
3. Saisir la commande "**sudo raspi-config**" puis valider.
4. sélectionner "Interfacing Options"
5. Sélectionner l'option "ssh".
6. Sélectionner le choix "Enable".
7. Confirmer le message par "Ok".
8. Terminer la configuration avec le bouton "Finish".

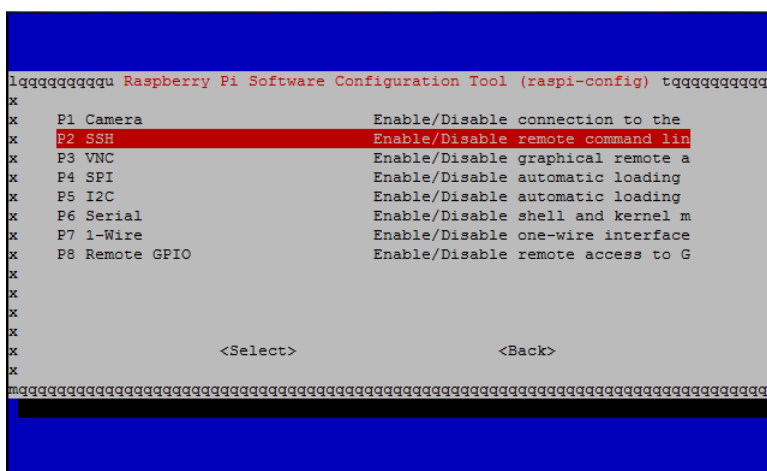


FIGURE 2.7 – Activer SSH

## Installer le client SSH sur votre ordinateur

Pour concrétiser la communication SSH entre nos différentes machines, et après l'avoir activé dans notre Raspberry Pi, reste à l'installer dans notre ordinateur.

**Installer le client SSH sur Linux** Si vous êtes sous Linux, Pour installer un client SSH, rien de plus simple, il vous suffit de lancer la commande suivante :

```
sudo apt-get update && sudo apt-get install openssh-client
```

**Installer le client SSH pour Windows** Étant donné que notre ordinateur fonctionne avec Windows, La solution qui nous est proposée c'est d'installer le logiciel nommé PuTTY.

Développé à l'origine par Simon Tatham pour la plate-forme Windows. PuTTY est un logiciel open source disponible avec le code source et développé et pris en charge par un groupe de volontaires.

Vous pouvez télécharger PuTTY [ici](#).

Ce programme permet de se connecter à distance à des serveurs en utilisant les protocoles SSH, Telnet<sup>7</sup>.

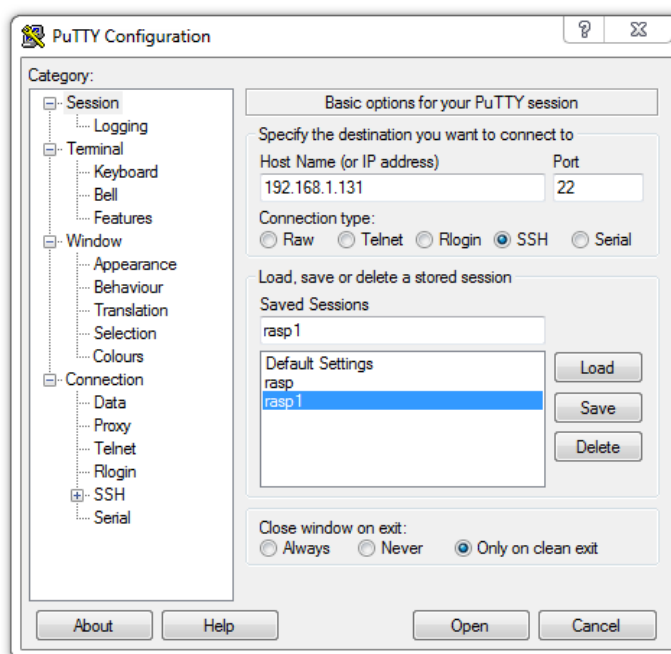


FIGURE 2.8 – Interface PuTTY

7. Protocole utilisé sur tout réseau TCP/IP, permettant de communiquer avec un serveur distant en échangeant des lignes de texte, toute communication est transmise en clair sur le réseau, mots de passe compris.

### 2.3.2 VNC

VNC, ou Virtual Network Computing, est un système permettant la prise de contrôle à distance d'un ordinateur par un autre. Lors de l'utilisation de VNC, deux parties différentes du logiciel sont utilisées. La première partie est le serveur VNC. Celui-ci est installé sur la machine dont l'on souhaite prendre le contrôle (ici le raspberry pi), et il va permettre la connexion et le contrôle par la partie client. La seconde partie est donc le client VNC. Celui-ci est installé sur la machine depuis laquelle on souhaite contrôler le serveur, et il va permettre de traduire nos actions en opérations compréhensibles par le serveur qui va alors contrôler la machine distante depuis notre ordinateur.

Le gros intérêt de VNC, c'est qu'il permet de prendre le contrôle d'une machine distante, tout en affichant le bureau de celle-ci. Nous pouvons donc voir en temps réel ce qui se passe sur votre Raspberry Pi, sans pour autant avoir besoin de la brancher sur un écran.

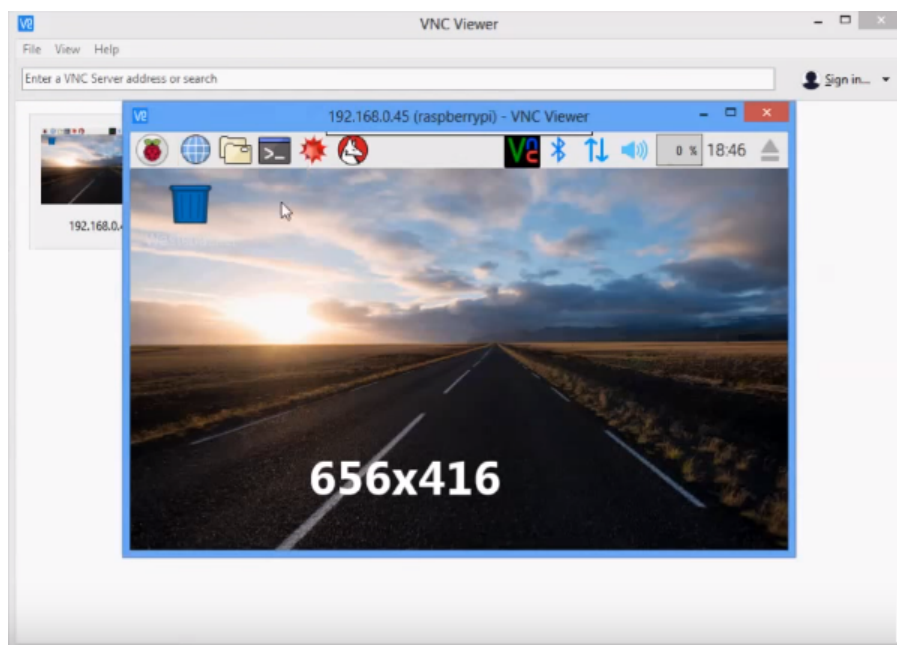


FIGURE 2.9 – Interface VNC

## 2.4 Python :

Nous verrons dans les chapitres qui suivent qu'il est possible de relier ces lignes à une multitudes de composants externes, capteurs, actionneurs ...etc, mais pour faire fonctionner tous ces éléments, il faudra écrire des programmes spécifique qui seront écrit en *Python*.

Pourquoi *Python* et pas un autre langage comme *Java*, *C*, *C++*, *Basic*, tous plus connus

au moins du grand public ?

pour au moins quatre raisons :

- *Python* est un langage facile à apprendre par qui n'a jamais programmé. La syntaxe de Python est très simple et, combinée à des types de données évolués (listes, dictionnaires,...), conduit à des programmes à la fois très compacts et très lisibles. A fonctionnalités égales, un programme *Python* (abondamment commenté et présenté selon les canons standards) est souvent de 3 à 5 fois plus court qu'un programme *C* ou *C++* (ou même *Java*) équivalent, ce qui représente en général un temps de développement de 5 à 10 fois plus court et une facilité de maintenance largement accrue.
- Le langage de programmation officielle qui a été pris en charge comme langue d'apprentissage éducatif sur le Raspberry Pi (le langage de prédilection à l'origine du Pi de Raspberry Pi) est *Python*, Python est d'ailleurs installé par défaut dans *Raspbian*.
- Python est un langage de programmation interprété et qu'il est donc très facile d'essayer immédiatement une ou plusieurs instructions dont on est pas sur du comportement.
- De nombreuses *bibliothèques* sont disponibles sur internet pour piloter les lignes d'entrées/sorties du Raspberry Pi en Python.



FIGURE 2.10 – Logo officiel de Python

### 2.4.1 Point fort du langage :

Le grand point fort de *Python* sur *PHP*, *Perl* ou un autre c'est sa lisibilité. Le slogan des fans de Python est : "Life's better without braces", ce qui signifie "La vie est meilleur sans accolades". En Python il n'existe pas d'accolades pour délimiter un bloc de code. En fait, le programmeur Python n'a pas le choix dans la façon de formater son code s'il veut pouvoir l'exécuter sans erreurs. Comme pour un Makefile<sup>8</sup>, une mauvaise indentation produit une

---

8. Un makefile est un fichier (par défaut nommé « Makefile ») contenant un ensemble de directives utilisées par une marque d'automatisation de construction outil pour générer une cible / objectif. Exemple « # » dans

erreur de syntaxe. La syntaxe de Python est plus donc constante et uniforme, voir restrictive. Sa rigidité lui permet d'être plus compréhensible par tous, notamment pour des projets avec plusieurs auteurs et des centaines de lignes de codes sources. Les scripts Python ont tous la même "tête", ils peuvent donc être relus par d'autres plus facilement. Cette remarque n'est pas négligeable surtout lorsque l'on sait qu'un développeur passe au moins autant de temps à lire et comprendre un code qu'à l'écrire !

## 2.4.2 Programmation des entrées / sorties avec python

Les broches marquées *GPIO* peuvent être utilisées comme broches d'entrée / sortie. En d'autres termes, n'importe quelle broche peut être programmée comme une entrée ou une sortie. Dans ce sens on va utiliser plusieurs langages de programmation capable de contrôler ces broches comme le *C*, *Java*, *Bash*... mais dans notre projet on a opté pour le python pour contrôler ces broches.

Le module GPIO est installé par défaut sur les versions les plus récentes de la distribution *Raspbian Linux*. Mais pour les versions plus anciennes, on doit probablement l'installer et effectuer une mise à jour. Pour cela on exécute la commande suivante :

```
"sudo apt-get install python-rpi.gpio"
```

Ensuite on exécute la commande suivante pour la mise à jour :

```
"sudo apt-get update"
```

**Remarque :** avant d'utiliser les broches, vous devez indiquer au module *GPIO* comment votre code y accédera. Le *Raspberry Pi* autorise deux numérotations : celle de la sérigraphie du connecteur de la carte (*GPIO.BOARD*), ou la numérotation électronique de la puce (*GPIO.BCM*). À nous de choisir celle qu'on veut. Pour tester le programme on exécute la commande suivante sur la console : **"sudo python led.py"** Voici maintenant un petit programme *Led-control.py* pour contrôler une LED :

```
1  #!/usr/bin/python
2  import RPi.GPIO as GPIO
3  import time
4
5  GPIO.setmode(GPIO.BOARD) # Utiliser les broches(pin) de la Raspberry Pi
6  #GPIO.setmode(GPIO.BCM) ==> On peut utiliser la numerotation electrique
7  GPIO.setup(13, GPIO.OUT)
8  print("Led on")
9  GPIO.output(13, 1) # Allumer la led
10 time.sleep(1)      # Attendre 1 seconde
```

une ligne d'un makefile commence un commentaire. Ce qui reste de la ligne est ignoré.

```
11 print("Led off")
12 GPIO.output(13, 0) # Eteindre la led
```

Led-control.py

## 2.5 Conclusion

Dans ce chapitre dédié à la présentation du Raspberry Pi, nous avons présenté les divers composants de cette carte, son utilité spécialement pour notre projet sur la domotique, ensuite on a vu comment utiliser les broches GPIO pour contrôler les composants électroniques à l'aide de langage de programmation python.

Dans le chapitre suivant, nous allons expliquer notre projet pratique qui est la réalisation d'un prototype d'une maison intelligente via une page web.

# Chapitre 3

## Conception et Réalisation

### Introduction :

Des que l'on souhaite automatiser une application avec un micro-ordinateur ou un micro contrôleur on s'aperçoit bien vite qu'il passe le plus clair de son temps à communiquer avec le monde extérieur par le biais de ce que l'on désigne sous le nom générique d'ENTRÉES/SORTIES. bien sûr, ces dernières varient fortement d'une application à une autres. Tout ces E/S pourraient constituer les sens du système, puisque elles permettent de capter tout les paramètres d'un lieu donné dont on voudrait mesurer ces constantes.

Cette diversité des E/S la majorité d'entre elles fait appel à des schémas de bases classique et à leurs programmes associés que nous avons regroupé dans la suite de ces chapitres.

### 3.1 Composants

Composants nécessaires de réalisation

- Carte Raspberry Pi 3 Model B.
- Plaque d'essai
- Câbles jumpers M/M et M/F
- Câble Ethernet (RJ45)
- Capteur DHT11
- Capteur de GAZ MQ-9.
- Ponts H L293D.
- Moteur DC (utilisé comme un ventilateur).
- Servo Moteur (utilisé comme une porte).
- Webcam utilisée comme caméra de surveillance.
- LEDs (rouge, jaune, bleu).

- Résistances (dont 3 résistances de  $100\ \Omega$ , deux de  $1\ \text{k}\Omega$  et une de  $4,7\ \text{k}\Omega$ ).
- Un bipueur (Piezo electric Buzzer).

## 3.2 Le DTH11 :

### 3.2.1 Introduction :

Il est nécessaire de connaître certains paramètres dans une maison comme la température, le taux d'humidité, la concentration de certains gaz, etc. Connaître la température permet à un système domotique d'utiliser le chauffage de manière intelligente et donc de réduire sa facture. Quant à l'humidité, un taux trop faible ou trop élevé est nocif pour la santé. La plage idéale est d'ailleurs entre 45 et 55%

### 3.2.2 DTH11 :

Est un capteur dit "*deux en un*" contrairement à d'autre dit spécialisé, il est capable à la fois de mesurer le degré d'humidité et la température d'un environnement bien défini. Ce capteur est capable de mesurer des températures allant de  $0$  à  $+50^\circ\text{C}$  avec une précision de  $\pm 2^\circ\text{C}$  et des taux d'humidité relative de  $20$  à  $80\%$  avec une précision de  $\pm 5\%$ . Une mesure peut être réalisée toutes les secondes. [8]

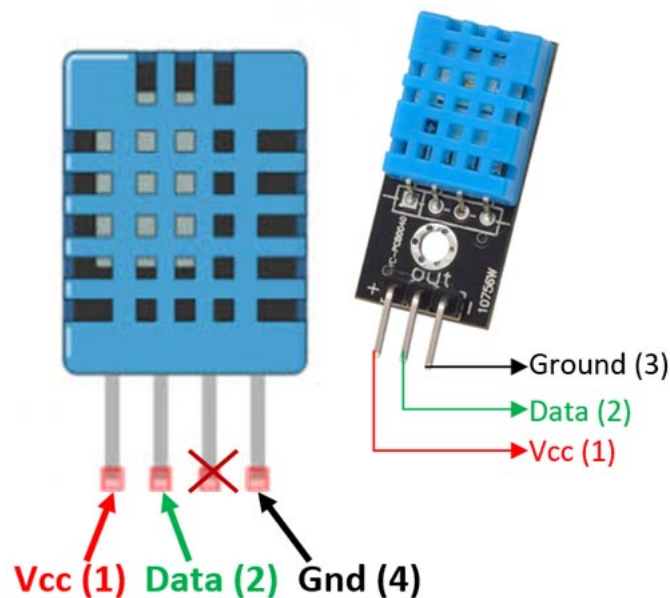


FIGURE 3.1 – DHT11

### 3.2.3 Caractéristiques

Écart de température	0 à +50°C ± 2°C
Écart d'humidité	20 à 80% ± 5%
Écart d'échantillonnage	1 Hz
tension de fonctionnement	3 - 5V
taille du corps	15.1mm*25mm*7.7mm
courant maximal pendant la mesure	2 mA

TABLE 3.1 – Caractéristiques DH11

### 3.2.4 Principe de fonctionnement

Ce capteur. Il est constitués d'un composant de détection d'humidité, d'un capteur de température NTC (ou thermistance) et d'un circuit intégré sur le côté arrière du capteur.

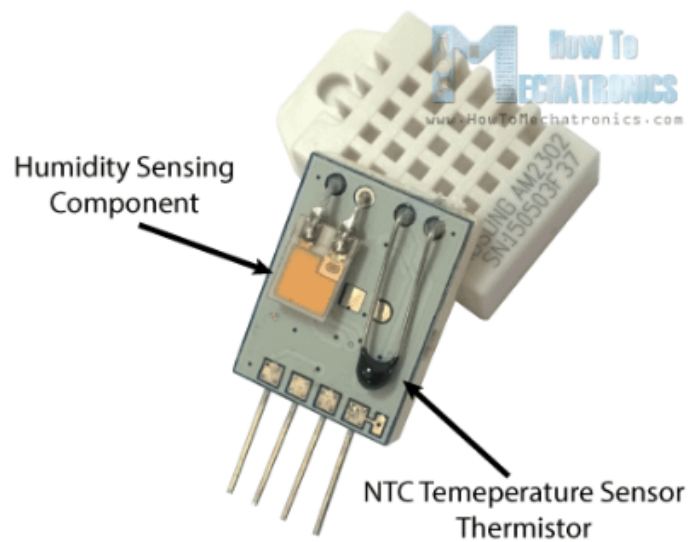


FIGURE 3.2 – Structure interne du DHT22 et du DHT11

Pour mesurer l'humidité, il utilise le composant de détection d'humidité qui a deux électrodes avec un substrat retenant l'humidité entre eux. Alors que l'humidité change, la conductivité du substrat change ou la résistance entre ces électrodes change. Ce changement de résistance est mesuré et traité par le CI qui le rend prêt à être lu par un microcontrôleur. [8]

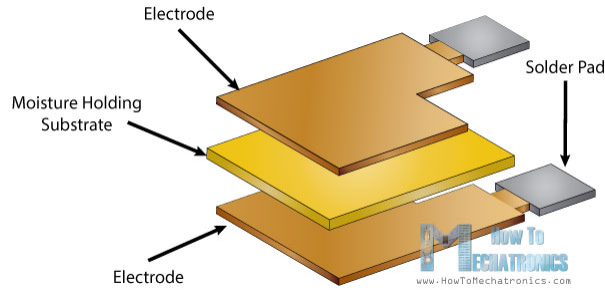


FIGURE 3.3 – Capteur d’humidité

Par contre, pour mesurer la température, ce capteurs utilise un capteur de température NTC ou une thermistance.

Une thermistance est en fait une résistance variable qui change sa résistance en fonction de l’évolution de la température. Ces capteurs sont fabriqués par frittage [8] de matériaux semi-conducteurs tels que des céramiques ou des polymères afin de fournir de plus grands changements de résistance avec de faibles variations de température. Le terme «CTN» signifie «coefficient de température négatif», ce qui signifie que la résistance diminue lorsque la température augmente. [8]

### 3.2.5 Branchement du DH11

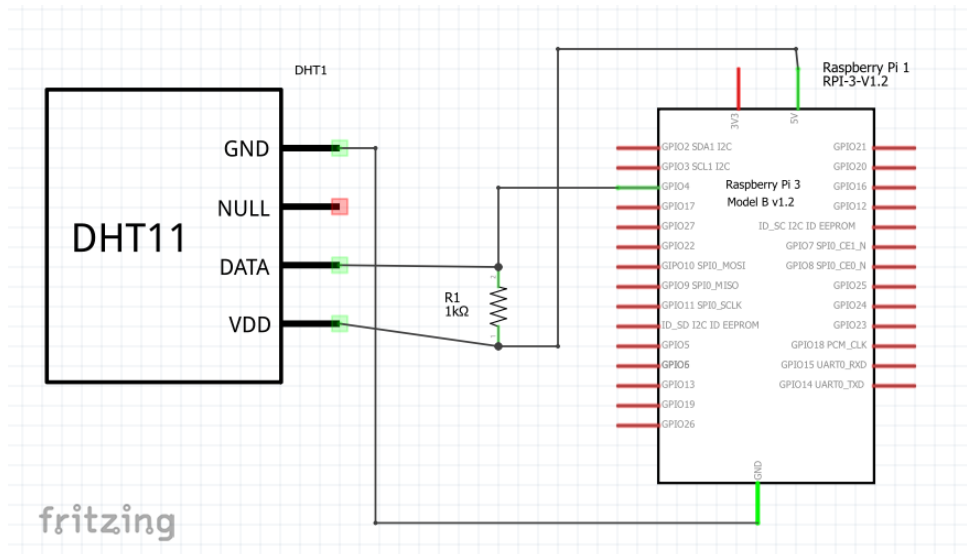


FIGURE 3.4 – Circuit du DHT11

1. Procédé de fabrication de pièces consistant à chauffer une poudre sans la mener jusqu’à la fusion. Sous l’effet de la chaleur, les grains se soudent entre eux (Agglomération des poudres métalliques par chauffage).

### 3.2.6 Script utilisé

```
1 #!/usr/bin/python
2 import sys
3 import Adafruit_DHT
4
5 while True:
6
7     humidity, temperature = Adafruit_DHT.read_retry(11, 4)
8     return humidity
9     print 'Temp: {0:0.1f} C Humidity: {1:0.1f} %'.format(temperature,
    humidity)
```

capH.py

## 3.3 Servomoteur

Un servomoteur (souvent abrégé en « servo », provenant du latin servus qui signifie « esclave ») est un moteur capable de maintenir une opposition à un effort statique et dont la position est vérifiée en continu et corrigée en fonction de la mesure. C'est donc un système asservi. Le servomoteur intègre dans un même boîtier, la mécanique (moteur et engrenage), et l'électronique, pour la commande et l'asservissement du moteur. La position est définie avec une limite de débattement d'angle de 180 degrés, mais également disponible en rotation continue. [\[9\]](#)

Des servo-moteurs, il en existe de plusieurs tailles : des plus petits de quelques grammes, tout en plastique, au plus gros de plusieurs kilogrammes, en métal. L'énergie qui les pilote et les fait n'est pas toujours électrique. [\[10\]](#)



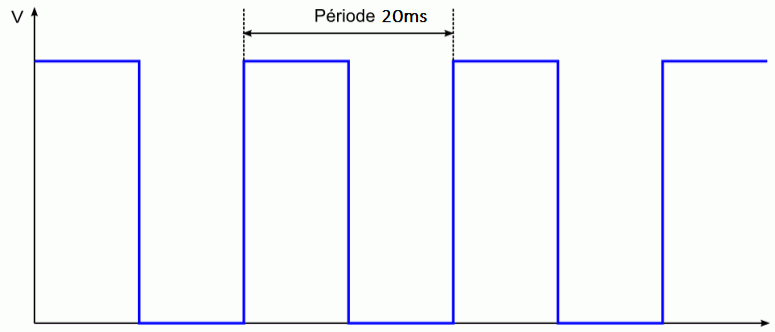


FIGURE 3.6 – Un train d’impulsions de période 20ms

La position du servomoteur est définie par la longueur d’une impulsion. Le servo s’attend à recevoir une impulsion toutes les 20 millisecondes environ. Si cette impulsion est élevée pendant 1 milliseconde, alors l’angle d’asservissement sera nul, si elle est de 1,5 milliseconde, alors elle sera au centre et si elle est de 2 millisecondes, elle sera à 180 degrés.

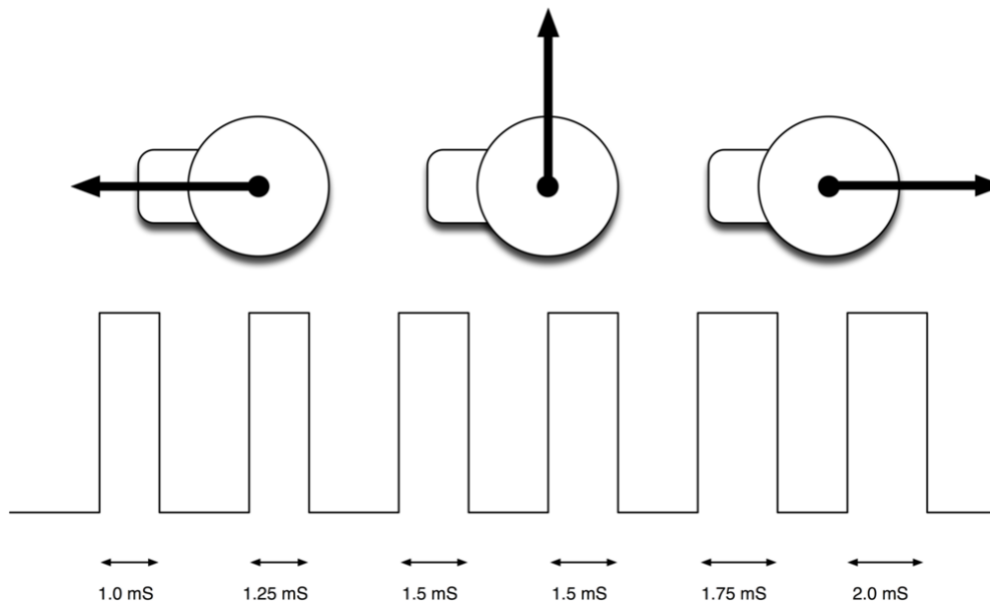


FIGURE 3.7 – Rotation du servo-moteur

### 3.3.2 Branchement du servo moteur

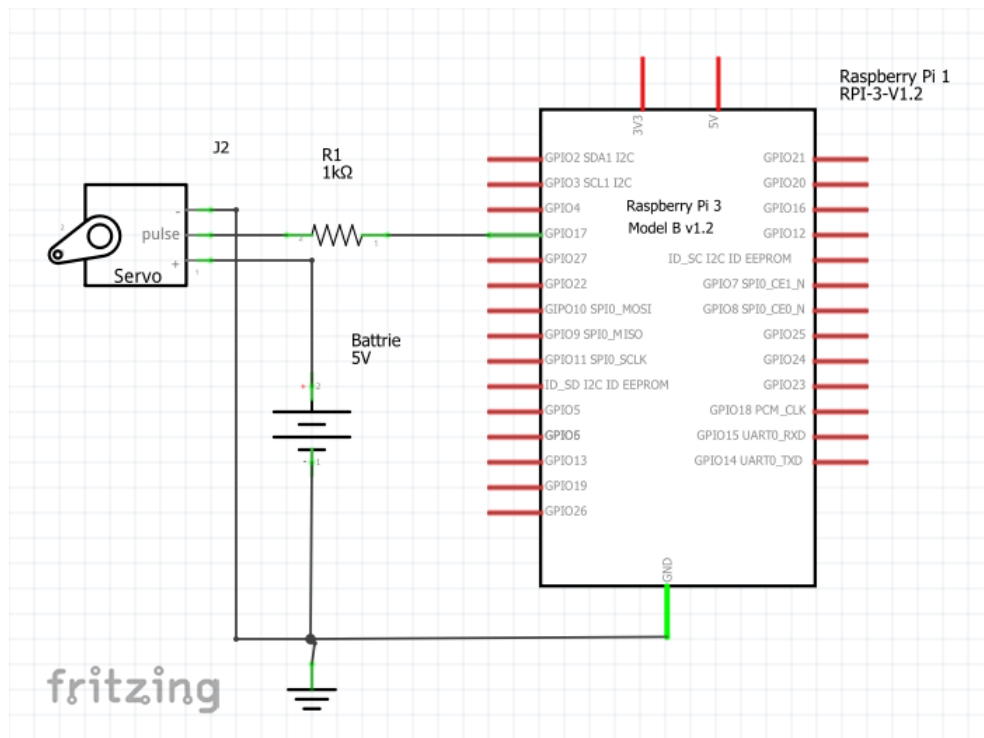


FIGURE 3.8 – Schéma électrique du servo moteur

### 3.3.3 Script utilisé

```

1 import RPi.GPIO as GPIO
2 import time
3 GPIO.setmode(GPIO.BCM)
4 GPIO.setup(17, GPIO.OUT)
5 p = GPIO.PWM(17, 50)
6 p.start(7.5)
7 try:
8     while True:
9         p.ChangeDutyCycle(7.5) #90 degre
10        print "Gauche"
11        time.sleep(1)
12        p.ChangeDutyCycle(12.5) # 180 degre
13        print "Centre"
14        time.sleep(1)
15        p.ChangeDutyCycle(2.5) # 0 degre
16        print "Droite"
17        time.sleep(1)

```

```
18         break
19 except KeyboardInterrupt :
20     p.stop()
21     GPIO.cleanup()
```

servo.py

### 3.4 PIR C-7288

Le capteur PIR est un module pyroélectrique conçu pour détecter la présence d'objets ou d'individus se trouvant dans son champ de vision. Il est généralement utilisé comme détecteur de mouvement ou de proximité. Les capteurs PIR détectent les radiations infrarouges émises par les objets autour de lui. Or la plupart des objets émettent des IR suivant la température de ceux-ci par rapport à la température ambiante. Ces capteurs sont dits passifs, car ils n'émettent pas de radiation (contrairement aux barrières à infrarouge). Les IR sont invisibles à l'œil humain, mais des appareils photos numériques peuvent les détecter.

Dans la vie quotidienne, on rencontre ce genre de capteurs dans les systèmes d'alarme, les allumages automatiques de lampes de jardin, les ouvertures de portes automatiques.

Ce type de capteur peut être utilisé en extérieur pour détecter le passage de personnes ou de voitures. Utilisé en intérieur il peut être un capteur d'appoint pour confirmer les informations données par un autre capteur, ou encore pour réagir aux mouvements d'êtres vivants.



FIGURE 3.9 – Capteur IR

### 3.4.1 Caractéristiques

Source de courant	5-12VDC
Courant de repos	5V 400U
Impulsion de sortie	0.5 sec
Niveau de sortie "élevé"	égale à la tension
Écart de température	-20 ° C to 50 ° C
Gain d'entrée PIR	68dB

TABLE 3.2 – Caractéristiques PIR-7288

### 3.4.2 Fonctionnement

Le capteur PIR convertit ce changement de température en un changement de tension de sortie et ce changement de tension est considéré comme un mouvement détecté.

### 3.4.3 Branchement

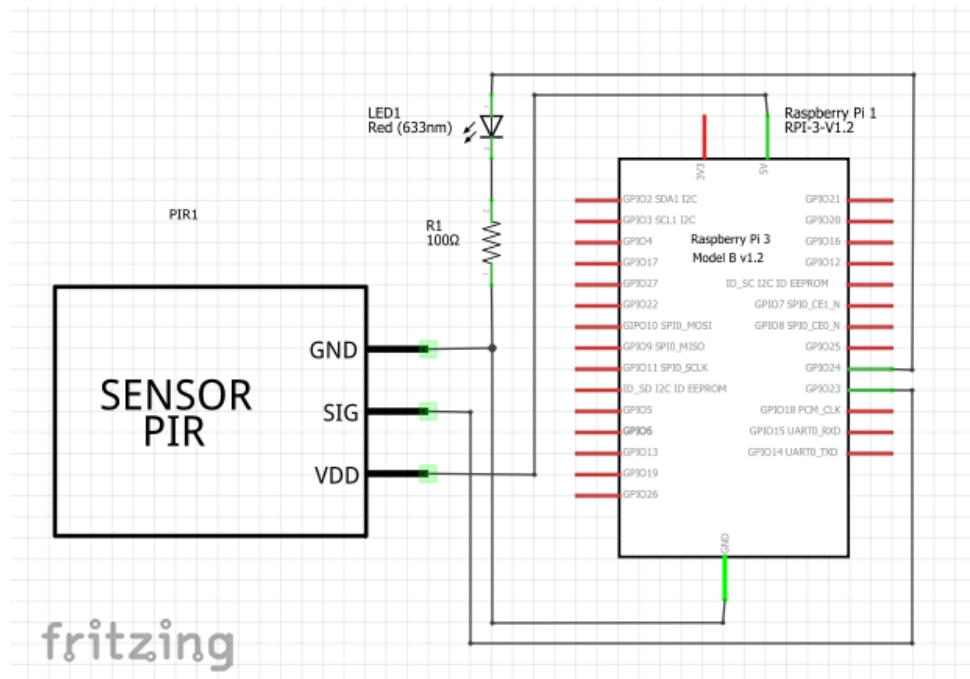


FIGURE 3.10 – Schéma électrique du capteur PIR

### 3.4.4 Scripts utilisé

```
1 #!/usr/bin/python
2 import RPi.GPIO as GPIO
3 import time
4
5 GPIO.setmode(GPIO.BCM)
6
7 GPIO.setup(23, GPIO.IN) #PIR
8 GPIO.setup(24, GPIO.OUT) #LED
9
10 try:
11     time.sleep(2) # to stabilize sensor
12     while True:
13         if GPIO.input(23):
14             GPIO.output(24, True)
15             time.sleep(0.5) # LED alight on for 0.5 sec
16             GPIO.output(24, False)
17             print("Motion Detected... ")
18             time.sleep(5) #to avoid multiple detection
19             time.sleep(0.1) #loop delay, should be less than detection delay
20
21 except:
22     GPIO.cleanup()
```

pir\_py.py

## 3.5 Moteur CC

Un moteur DC est un convertisseur électromécanique permettant la conversion bidirectionnelle d'énergie entre une installation électrique parcourue par un courant continu et un dispositif mécanique ; selon la source d'énergie. Pour faire simple, cela signifie qu'un moteur à courant continu va pouvoir convertir de l'électricité en énergie mécanique. Les moteurs DC ont ainsi la particularité de pouvoir fonctionner dans les 2 sens, suivant la manière dont le courant lui est soumis.



FIGURE 3.11 – Moteur CC

### 3.5.1 Les caractéristiques du moteur courant continu

Le moteur courant continu est caractérisé par une constante de vitesse, et une pente vitesse/couple. Le courant est proportionnel à la charge ; et la vitesse est proportionnelle à la tension d'alimentation. Le moteur courant continu sans fer ne présente pas de couple magnétique résiduel et les agitations électromagnétiques sont dérisoires. Son rendement, qui atteint 90%, surpasse celui des autres technologies de moteurs. Son rotor en forme de cloche donne la possibilité d'accélération très importantes et d'un couple de retenue inexistant. Cela permet d'obtenir des positionnements précis et des vitesses faibles. Le rotor est traditionnellement composé d'un stator à aimant permanent en terre rare de type Al-nico, samarium cobalt ou néodyme fer bore qui expliquent les caractéristiques dynamiques très élevées...

Le moteur à courant continu se compose :

- de l'inducteur ou du stator,
- de l'induit ou du rotor,
- du collecteur et des balais.

**Mise en garde** Il est formellement déconseillé de brancher directement le moteur aux pins GPIO du Raspberry Pi. Les courants assez intenses qui sont nécessaires pour faire tourner le moteur pourraient s'avérer très néfastes pour la santé de notre Raspberry Pi. Nous allons plutôt contrôler le moteur par l'intermédiaire d'un circuit intégré spécialement conçu pour ce genre d'utilisation : le L293D (le pont H).



```

3
4 GPIO.setmode(GPIO.BOARD)
5 GPIO.cleanup()
6
7 Moteur1A = 16 # premiere sortie du premier moteur , pin 16 ==> pin 7(L293D)
8 Moteur1B = 18 # deuxieme sortie de premier moteur , pin 18 ==> pin 2(L293D)
9 Moteur1E = 22 # enable du premier moteur , pin 22 ==> pin 1 (L293D)
10 GPIO.setup(Moteur1A ,GPIO.OUT)
11 GPIO.setup(Moteur1B ,GPIO.OUT)
12 GPIO.setup(Moteur1E ,GPIO.OUT)
13 GPIO.output(Moteur1E ,GPIO.HIGH) # demarre le moteur avec variation de vitesse
14 GPIO.output(Moteur1B ,GPIO.HIGH) # tourne dans un sens
15 GPIO.output(Moteur1A ,GPIO.LOW) # tourne dans un sens

```

moteur\_dc\_start.py

le deuxième son role c est d'arrêter le moteur.

```

1 import RPi.GPIO as GPIO
2 from time import sleep
3
4 GPIO.setmode(GPIO.BOARD)
5 GPIO.cleanup()
6
7 Moteur1A = 16 # premiere sortie du premier moteur , pin 16 ==> pin 7(L293D)
8 Moteur1B = 18 # deuxieme sortie de premier moteur , pin 18 ==> pin 2(L293D)
9 Moteur1E = 22 # enable du premier moteur , pin 22 ==> pin 1 (L293D)
10 GPIO.setup(Moteur1A ,GPIO.OUT)
11 GPIO.setup(Moteur1B ,GPIO.OUT)
12 GPIO.setup(Moteur1E ,GPIO.OUT)
13 GPIO.output(Moteur1E ,GPIO.LOW) # arret du moteur
14 GPIO.cleanup() # renitialise les pin du Raspberry Pi

```

moteur\_dc\_stop.py

## 3.6 Capteur MQ9

Le MQ-9 est un semi-conducteur capteur de gaz très sensible au monoxyde de carbone, au méthane et au GPL. Le capteur pourrait être utilisé pour détecter différents gazes (du CO et des gaz combustibles), il est à faible coût et convient à différentes applications. Le matériau sensible du capteur de gaz MQ-9 est le SnO<sub>2</sub>, qui présente une conductivité plus faible dans l'air pur et la conductivité du capteur augmente avec la concentration des différents gazes.

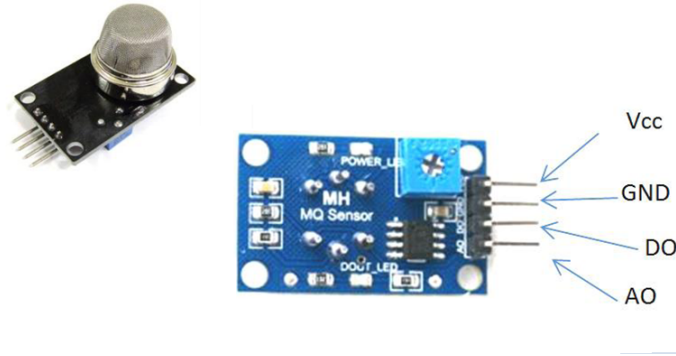


FIGURE 3.14 – Capteur MQ9

### 3.6.1 Fonctionnement

Une fois alimenté, le capteur commence à détecter des variations de niveau de gaz. la transmission se fait de deux manières : soit analogique ou numérique.

- En analogique, la tension de sortie varie entre 0 et 5v selon le niveau de gaz.
- En numérique, un potentiomètre permet de régler la sensibilité du capteur qui représente un seuil, si le seuil est atteint, alors la sortie passe en niveau logique 0.

### 3.6.2 Branchement

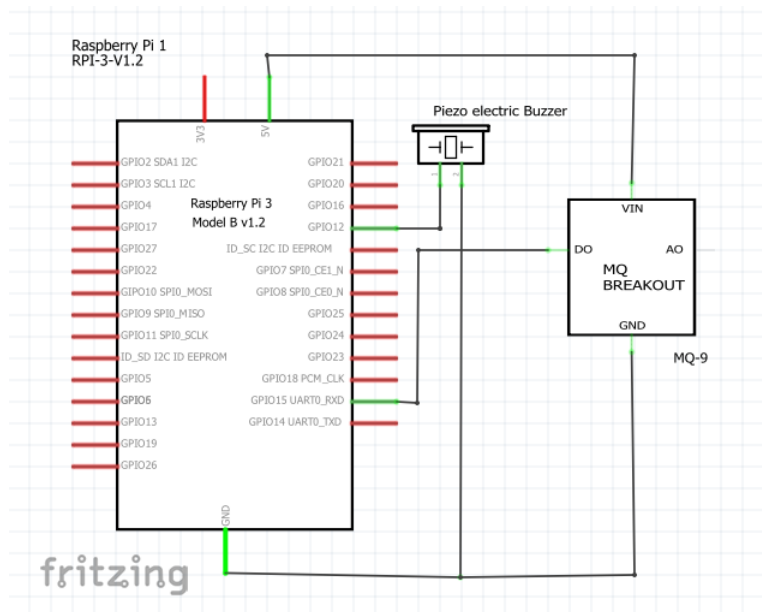


FIGURE 3.15 – Circuit du capteur MQ9

### 3.6.3 Scripts utilisé

## 3.7 Les LEDs

Une diode électroluminescente (abrégé en LED, de l'anglais : light-emitting diode), est un dispositif opto-électronique capable d'émettre de la lumière lorsqu'il est parcouru par un courant électrique. Ces LED feront office d'éclairage dans notre maquette et serviront d'indicateur lumineux pour nos circuits.

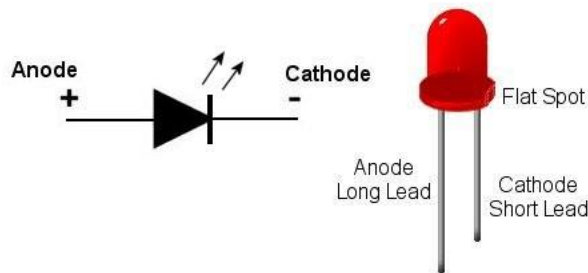


FIGURE 3.16 – Une LED

### 3.7.1 Branchement

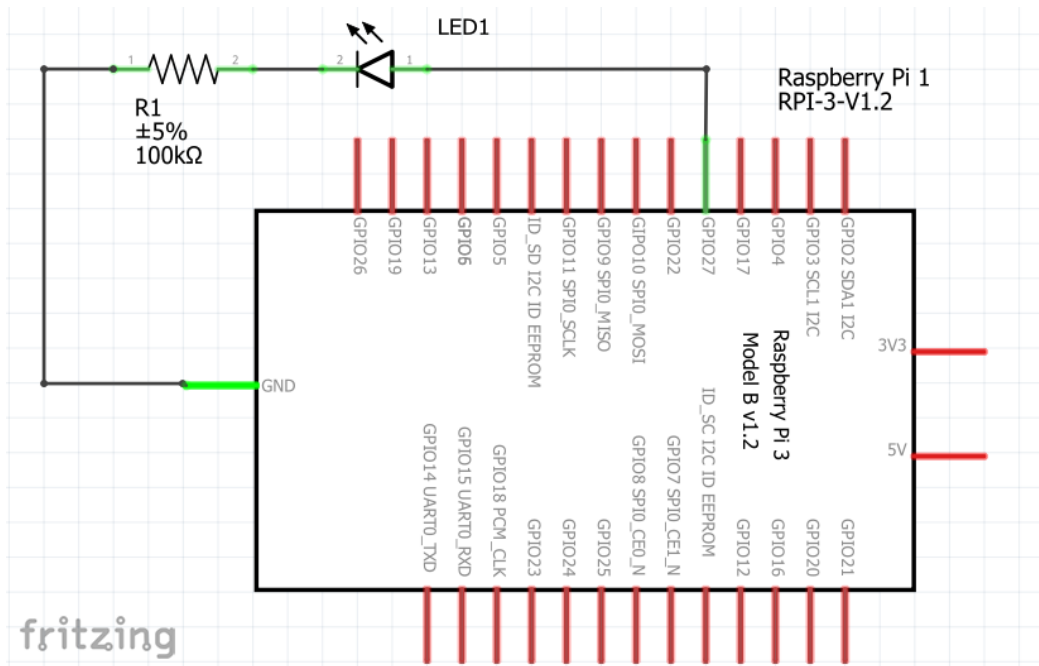


FIGURE 3.17 – Schéma électrique d'une LED

### 3.7.2 Script utilisé

le script pour contrôler la LED est mentionné dans le chapitre 2, sous-section [2.4.2](#), page [29](#)

## 3.8 Contrôle à distance

Pour pouvoir accéder à notre Raspberry Pi à distance via WiFi et contrôler les GPIOs, on a besoin de rendre notre carte un serveur Web capable d'héberger un ou plusieurs sites Internet et assurer donc la communication avec un client et répondre à ses demandes grâce au protocole réseau http. Le plus souvent, un serveur Web utilise d'autres logiciels qui fonctionnent en collaboration avec le serveur HTTP (Apache) comme le serveur de base de données (MySQL, Oracle), et pour communiquer avec un serveur web, on a besoin d'un langage de programmation serveur comme PHP. Dans notre projet, on va opter pour installer la combinaison (Apache, PHP).

### 3.8.1 Apache

Tout d'abord, nous allons installer Apache, qui est le serveur web en tant que tel. Quand on parle de serveur web, on pense souvent à la machine, mais ce terme désigne aussi le logiciel qui permet à la machine d'analyser les requêtes d'un utilisateur (sous forme http), et de retourner le fichier correspondant à la requête (ou une erreur si le fichier n'est pas trouvé, ou la requête est mal formulée). Dans le cadre d'Apache, c'est donc du logiciel que l'on parle.

À l'heure actuelle, Apache est le serveur web le plus utilisé, avec environ 60 % de parts de marché. Apache possède même sa propre licence, utilisée par de nombreux autres projets. De plus, l'utilisation massive d'Apache (devenu le standard des serveurs web), couplée à sa forte popularité, à amener à une formidable abondance de documentation, de cours, et autres livres traitant de son utilisation, depuis l'installation jusqu'à la sécurisation.

Que ce soit pour la Raspberry Pi et Raspbian, ou pour une machine plus généraliste, Apache est donc un choix sûr, et les compétences que nous pourrons acquérir sur le sujet nous seront toujours utiles.



FIGURE 3.18 – Logo Apache

### Installation d'Apache

Avant d'installer le serveur, assurons nous d'avoir une machine bien à jour. Pour cela, on exécute les commandes suivantes :

**sudo aptitude update** ensuite **sudo aptitude upgrade**.

Une fois le Raspberry Pi est mis à jour, nous allons installer le serveur Apache par la commande suivante :

**sudo aptitude install apache2**.

Une fois l'installation terminée, nous pouvons tester qu'Apache fonctionne correctement en nous rendant sur l'adresse du Raspberry Pi. Pour cela, il faut tenter d'accéder au Raspberry Pi depuis le port 80 (ce port n'étant pas encore ouvert depuis l'extérieur, il faudra le faire depuis le Raspberry Pi lui-même). Il nous suffit d'ouvrir le navigateur web du Raspberry Pi (par défaut Midori sous Raspbian), et d'aller à l'adresse « <http://127.0.0.1> ». On obtient alors une page avec un message du genre « It works! » et plein d'autre texte.

Vous pouvez dès à présent utiliser votre Raspberry Pi pour faire un site en HTML, CSS et JavaScript pur, en interne. Cependant, nous voudrions sans doute rapidement permettre des interactions entre le site et l'utilisateur. Par exemple permettre à l'utilisateur de s'inscrire, etc. Pour cela, vous allez avoir besoin de PHP.

### 3.8.2 PHP

Tout d'abord, il faut savoir que le PHP est un langage interprété. Et comme dans le cas des serveurs, l'acronyme PHP peut avoir plusieurs sens. En fait, quand l'on parle de PHP, on peut parler soit du langage, soit de l'interpréteur. Ici, quand nous parlons d'installer PHP, cela signifie que nous allons installer l'interpréteur, afin d'utiliser le langage.

PHP (le langage cette fois) est principalement utilisé pour rendre un site dynamique, c'est-à-dire que l'utilisateur envoie des informations au serveur qui lui renvoie les résultats modifiés en fonction de ces infos. A contrario, un site statique ne s'adapte pas aux informations fournies

par un utilisateur. Il est enregistré sous forme de fichier une fois pour toute, et livrera toujours le même contenu.

PHP est libre, et maintenu par la fondation PHP, ainsi que l'entreprise Zend, et diverses autres entreprises (il est à noter que Zend est aussi l'auteur du célèbre *framework Zend PHP*, très utilisé et reconnu dans le monde de l'entreprise). PHP est un des langages de programmation les plus utilisés, et c'est même le plus utilisé pour la programmation destinée au web, avec environ 79 % de parts de marché .



FIGURE 3.19 – Logo PHP

## Installer PHP

On va encore une fois faire appel à l'administrateur pour installer PHP avec la ligne de commande :

```
sudo aptitude install php5
```

Maintenant pour savoir si PHP fonctionne correctement, ce n'est pas très compliqué, et la méthode est relativement proche de celle employée pour Apache. Nous allons en premier lieu supprimer le fichier « `index.html` » dans le répertoire « `/var/www/html` » par la commande suivante :

```
sudo rm /var/www/html/index.html
```

Puis créez un fichier « `index.php` » dans ce répertoire, avec cette ligne de commande :

```
echo "<? phpinfo();?>" > /var/www/html/index.php
```

À partir de là, le fonctionnement est le même que pour la vérification d'Apache. On tente d'accéder à notre page, par taper l'adresse « `http://127.0.0.1` ».

## PHP contrôle les GPIOs

Nous utilisons dans notre projet le PHP pour contrôler les entrées et sorties de notre maison intelligente. Pour cela, on héberge dans le serveur Apache dans le dossier `/var/www/html/` nos scripts PHP. Chaque script a un rôle pour contrôler une entrée ou sortie reliée avec Raspberry Pi en l'occurrence les LEDs, les moteurs, les capteurs. PHP est capable d'exécuter un

script python grâce à la fonction `exec()`. Mais avant cela, il faut donner les droit d'exécution des scripts pythons par la commande suivante :

```
sudo chmod +x script.py.
```

### 3.9 Conclusion

Dans ce chapitre nous avons aborder la partie électronique de notre projet, en faisant les montages de chacun des composants qui constitue notre maquette. Nous avons pus les tester et les faire fonctionner, en plus c'était l'occasion pour nous de nous perfectionner dans la programmation de langage tel que python ou PHP. Ce était une véritable découverte étant donné que l'univers des Webmaster nous était totalement inconnus jusqu'alors.

# Conclusion générale

Notre projet consistait à réaliser et contrôler la maison intelligente (surveillance, gestion de l'énergie, sécurité ,confort ...).

En premier temps nous avons donné un aperçu général sur la carte Raspberry Pi, le langage python, Avec une description globale des composants utilisés. Après nous avons procédé à l'élaboration de notre maquette (maison miniature) équipée de différents capteurs et actionneurs (moteur dc, capteur infrarouge, servo moteur, capteur de température et l'humidité).

Dans ce projet, nous avons pu réaliser avec succès la maison intelligente "smart home" à base d'une carte Raspberry pi3, et son langage de programmation python.

Notre projet était constitué de :

- Des servo moteur servent à contrôler l'ouverture et fermeture de la porte du garage.
- Un capteur PIR pour détecter le mouvement (présence)
- d'un capteur DHT11 pour contrôler la température et l'humidité dans la maison.
- Une webcam pour surveiller notre maison

Tout cet ensemble de composants était contrôlé à distance via une page web via le serveur apache.

# Annexe A

	Modèle A	Modèle A+	Modèle 1 B	Modèle 1 B+	Modèle Zero	Modèle 2 B	Modèle 3 B
Prix de lancement	25 \$ US	20 \$ US	35 \$ US		5 \$ US	35 \$ US	
Date de lancement	Février 2013	Novembre 2014	Avril-Juin 2012	Juillet 2014	Novembre 2015	Février 2015	Février 2016
SoC	Broadcom BCM2835 (CPU, GPU, DSP, SDRAM, et 1 port USB)					BCM2836	Broadcom BCM2837
CPU	700 MHz ARM1176JZF-S core (ARM11)				1 GHz ARM1176JZF-S core (ARM11)	1 GHz ARM1176JZF-S core (ARM11)	900 MHz quadricœur ARM Cortex-A7
GPU	Broadcom VideoCore IV, OpenGL ES 2.0, MPEG-2 et VC-1 (avec licence), 1080p30 h.264/MPEG-4 AVC high-profile decodeur et encodeur70,71						
SDRAM	256 Mo (intégré avec GPU)		512 Mo (intégré avec GPU) au 15 octobre 2012			1 Go	
Ports USB 2.0	1		2		4		4
Sorties vidéo	HDMI et Composite (Via connecteur RCA ou prise Jack)						
Sorties audio	stéréo Jack 3,5 mm				HDMI		stéréo Jack 3,5 mm
Unité de lecture/écriture	SD/MMC/SDIO	MicroSD	SD/MMC/SDIO			MicroSD	
Connectique réseau	Non		10/100 Ethernet		Non		10/100 Ethernet, Wifi 802.11n, Bluetooth 4.1
Périphériques E/S	8 × GPIO, UART, I <sup>2</sup> C bus, SPI bus avec deux chip selects, I <sup>2</sup> S audio75 +3.3 V, +5 V	17 × GPIO, UART, I <sup>2</sup> C bus, SPI bus avec deux chip selects, I <sup>2</sup> S audio, +3.3 V, +5 V	8 × GPIO, UART, I <sup>2</sup> C bus, SPI bus avec deux chip selects, I <sup>2</sup> S audio75 +3.3 V, +5 V	17 × GPIO, UART, I <sup>2</sup> C bus, SPI bus avec deux chip selects, I <sup>2</sup> S audio, +3.3 V, +5 V			
Système d'exploitation	Debian GNU/Linux, Raspbian OS, Fedora, Arch Linux ARM1, RISC OS, FreeBSD, Plan 9, Kali Linux					idem modèle 1 + Snappy Ubuntu Core83, Windows 10 IoT71	

FIGURE A.1 – Différents modèles Raspberry Pi



Ce qu'on peut faire	La commande	Signification
Se déplacer dans un répertoire	cd..	Remonter d'un niveau
	cd/	Retourner a la racine
	cd-	Retourner au répertoire précédent
Lister les fichiers d'un répertoire	ls	Liste le contenu d'un répertoire
	ls -d	Affiche uniquement les répertoires
	ls -a	Liste tous les fichiers du répertoire y compris les fichiers cachés
	ls -F	Affiche les fichiers par type
	ls -X	Affiche les fichiers par type d'extension
Copier un fichier ou un répertoire	cp	Avant de procéder, il demande si il peut écraser le nom du fichier : répondre par Oui(y) ou Non(n)
Supprimer un fichier ou un répertoire	rm	Supprime un fichier ou un répertoire
	rm -d	Permet de supprimer un répertoire qu'il soit plein ou non
	rm -r	Permet de supprimer un répertoire et ses sous répertoires
	rmdir	Permet de supprimer un répertoire
Changer les droits (droit de lecture, d'écriture et d'exécution)	chmod +x	Cette commande change les droits d'un fichier
	chmod 777	Le droit maximal que l'on puisse donner
	chmod 000	Enlève les droits (le contraire de 777)
Autres commandes	man	Affiche le manuel d'utilisation d'une commande
	mv	Cette commande déplace un fichier ou le renomme
	nl	Affiche le nombre de lignes d'un fichier
	ping	Commande qui contacte un hôte par son adresse IP ou son nom
	pwd	Cette commande affiche le répertoire dans lequel on se trouve
	find	Cette commande recherche des fichiers
	arch	Cette commande affiche l'architecture de la machine
	cat	Commande qui affiche le contenu d'un fichier
	clear	Nettoie votre fenêtre de terminal en réléguant tout le texte au dessus
	shutdown now	Sert à arrêter le système
	halt	Permet d'éteindre immédiatement le système
	reboot	Pour redémarrer le système

TABLE A.1 – Quelques commandes utiles à connaître sous Linux

# Annexe B

## Branchements du circuit

### B.0.1 Branchement du DH11

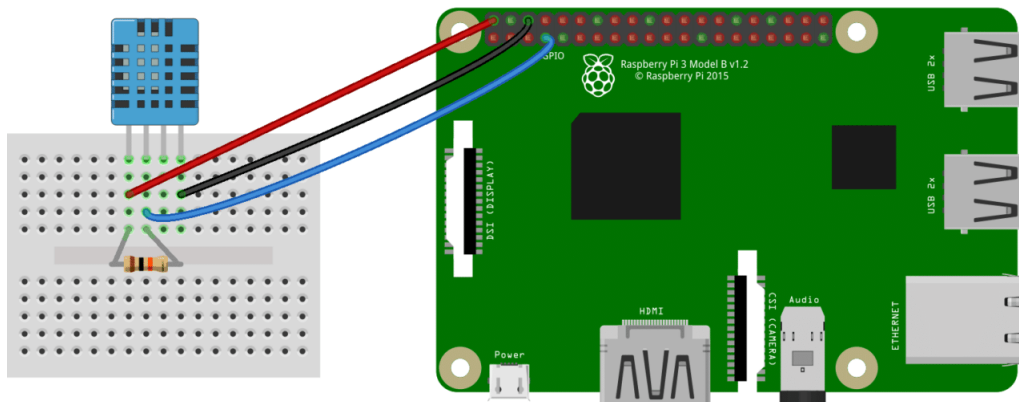


FIGURE B.1 – Branchement DHT11

### B.0.2 Branchement du servo moteur

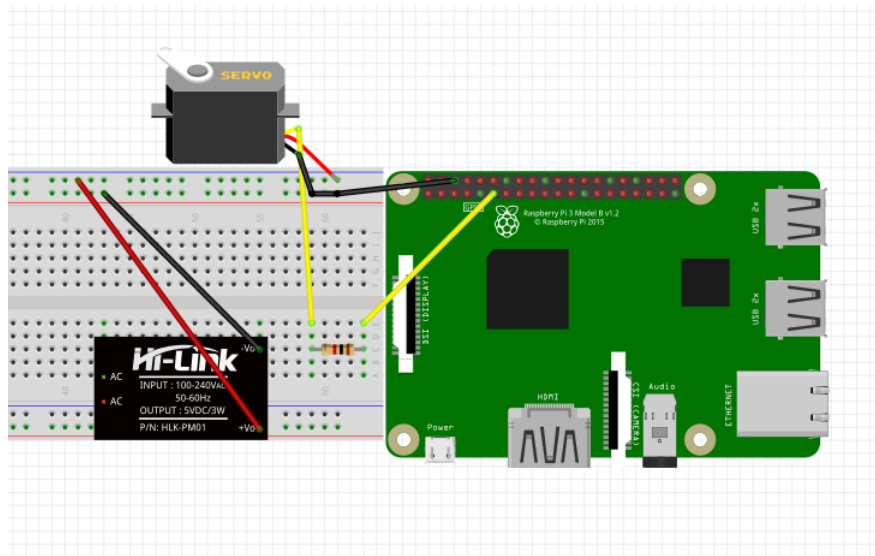


FIGURE B.2 – Branchement du servo moteur

### B.0.3 Branchement capteur de mouvement

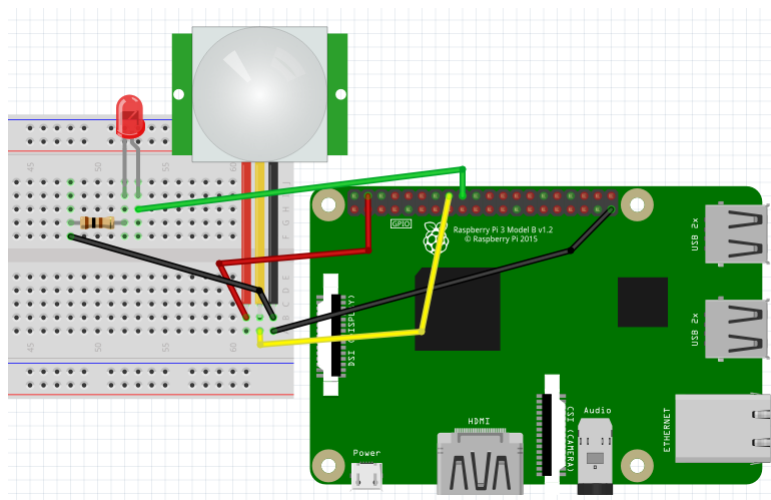


FIGURE B.3 – Branchement du capteur PIR

### B.0.4 Branchement moteur CC

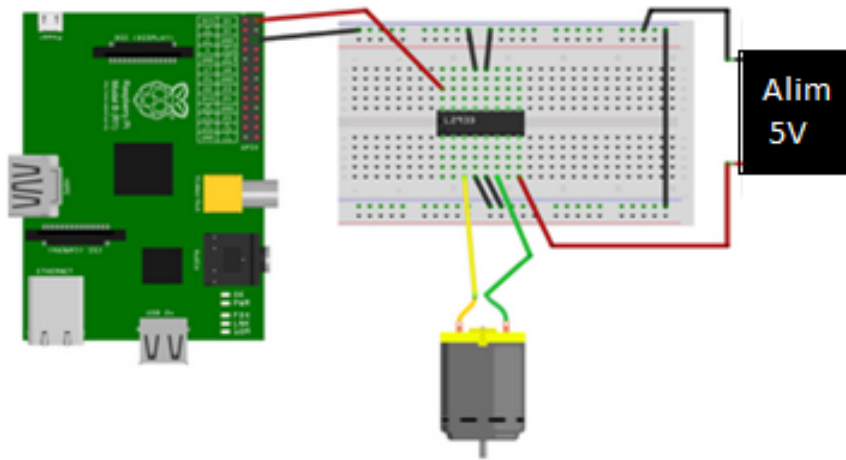


FIGURE B.4 – Branchement du moteur CC

### B.0.5 Branchement capteur de gaz

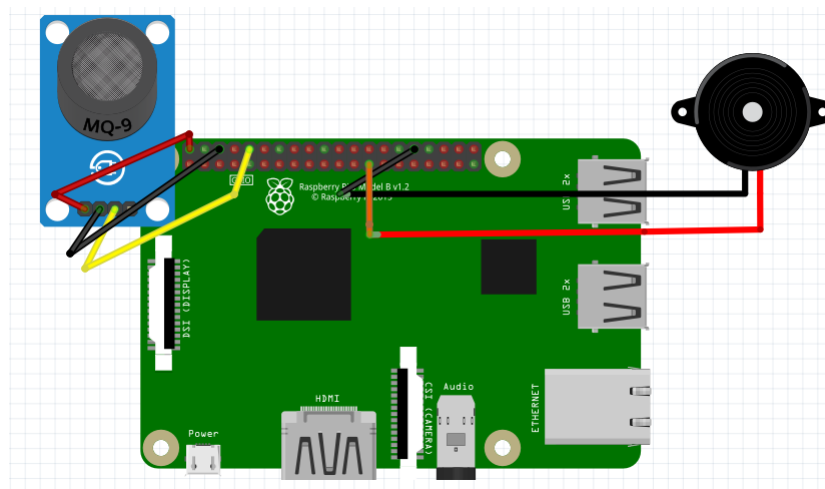


FIGURE B.5 – Branchement du capteur MQ9

## B.0.6 Branchement éclairage

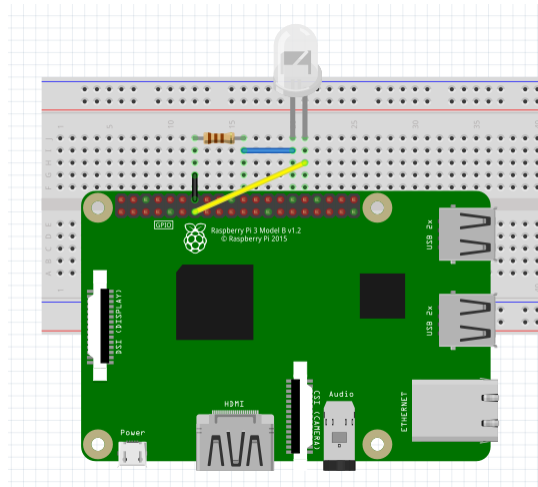


FIGURE B.6 – Branchement d'une LED

# Table des figures

1.1 Domotique . . . . .	3
1.2 Schéma Fonctionnel d'un capteur . . . . .	4
1.3 relation capteurs actionneurs . . . . .	8
1.4 Différentes fonctions de la domotique . . . . .	9
1.5 Logo X10 . . . . .	12
1.6 Logo Legrand . . . . .	13
1.7 Logo Chacon . . . . .	14
2.1 Raspberry pi 3 Model B . . . . .	17
2.2 Caractéristiques du Raspberry . . . . .	18
2.3 Raspbian . . . . .	20
2.4 Menu NOOBS . . . . .	23
2.5 Raspbian . . . . .	23
2.6 Fond d'écran Raspbian . . . . .	24
2.7 Activer SSH . . . . .	25
2.8 Interface PuTTY . . . . .	26
2.9 Interface VNC . . . . .	27
2.10 Logo officiel de Python . . . . .	28
3.1 DHT11 . . . . .	32
3.2 Structure interne du DHT22 et du DHT11 . . . . .	33
3.3 Capteur d'humidité . . . . .	34
3.4 Circuit du DHT11 . . . . .	34
3.5 Servo moteur . . . . .	36
3.6 Un train d'impulsions de période 20ms . . . . .	37
3.7 Rotation du servo-moteur . . . . .	37
3.8 Schéma électrique du servo moteur . . . . .	38
3.9 Capteur IR . . . . .	39

<a href="#">3.10 Schéma électrique du capteur PIR</a> . . . . .	40
<a href="#">3.11 Moteur CC</a> . . . . .	42
<a href="#">3.12 le L293D</a> . . . . .	43
<a href="#">3.13 Schéma du moteur CC</a> . . . . .	43
<a href="#">3.14 Capteur MQ9</a> . . . . .	45
<a href="#">3.15 Circuit du capteur MQ9</a> . . . . .	45
<a href="#">3.16 Une LED</a> . . . . .	46
<a href="#">3.17 Schéma électrique d'une LED</a> . . . . .	46
<a href="#">3.18 Logo Apache</a> . . . . .	48
<a href="#">3.19 Logo PHP</a> . . . . .	49
<a href="#">A.1 Différents modèles Raspberry Pi</a> . . . . .	52
<a href="#">A.2 Schema Raspberry Pi 3B</a> . . . . .	54
<a href="#">B.1 Branchement DHT11</a> . . . . .	56
<a href="#">B.2 Branchement du servo moteur</a> . . . . .	57
<a href="#">B.3 Branchement du capteur PIR</a> . . . . .	57
<a href="#">B.4 Branchement du moteur CC</a> . . . . .	58
<a href="#">B.5 Branchement du capteur MQ9</a> . . . . .	58
<a href="#">B.6 Branchement d'une LED</a> . . . . .	59

# Liste des tableaux

1.1 Tableau des capteurs actifs . . . . .	5
1.2 Tableau des capteurs actifs . . . . .	6
3.1 Caractéristiques DH11 . . . . .	33
3.2 Caractéristiques PIR-7288 . . . . .	40
A.1 Commandes Linux . . . . .	55

# Bibliographie

- [1] Tero Karvinen, Radons G, Sokolov IM (2014) Les capteurs pour Arduino. Dunod, Paris
- [2] Tiphaine Bodin.(2018). « La domotique, principe et fonctionnement ». In : Linternaute.com [En ligne]. Disponible sur : <https://www.linternaute.fr/bricolage/guide-maison-et-jardin/1335625-la-domotique-principe-et-fonctionnement/>
- [3] SIMON MONK (2014) Programmer un Raspberry Pi pltm : initiation avec python, page1. DUNOD, Paris
- [4] SIMON MONK (2014) Programmer un Raspberry Pi pltm : initiation avec python, page 18. DUNOD Paris
- [5] C. Tavernier, « Arduino applications avancées ». Version Dunod.
- [6] « electriciteguide », <https://www.electriciteguide.com/actualites/le-delesteur.htm>
- [7] <https://raspbian-france.fr/telechargements/>
- [8] Dejan Nedelkovski. (2016). « DHT11 & DHT22 Sensors Temperature and Humidity Tutorial using Arduino ». In : howtomechatronics.com [En ligne].<https://howtomechatronics.com/tutorials/arduino/dht11-dht22-sensors-temperature-and-humidity-tutorial-using-arduino/>
- [9] PoBot. « servomoteur - PoBot ». In : pobot.org[archive], [www.pobot.org](http://www.pobot.org) (consulté le 12 janvier 2017)
- [10] Jean-Noël Rousseau. (2016). « Programmez vos premiers montages avec Arduino ». In : openclassrooms.com[Cours], sur <https://openclassrooms.com/courses/programmez-vos-premiers-montages-avec-arduino/donnez-du-mouvement-a-vos-montages-avec-un-servo-moteur>
- [11] Jean-Noël Rousseau. (2016). « Programmez vos premiers montages avec Arduino ». In : openclassrooms.com[Cours], sur <https://openclassrooms.com/courses/programmez-vos-premiers-montages-avec-arduino/>

donnez-du-mouvement-a-vos-montages-avec-un-servo-moteur\T1\

guillemotleftServo-Moteur\T1\guillemotright, sur[http://wiki.mchobby.](http://wiki.mchobby.be/)

[be/index.php?title=Servo-Moteur](http://wiki.mchobby.be/index.php?title=Servo-Moteur)