

**RÉPUBLIQUE ALGÉRIENNE DÉMOCRATIQUE ET POPULAIRE**  
**MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA RECHERCHE**  
**SCIENTIFIQUE**

**UNIVERSITÉ IBN-KHALDOUN DE TIARET**

**FACULTÉ DES SCIENCES APPLIQUÉES**

**DÉPARTEMENT DE GENIE ELECTRIQUE**



# **MEMOIRE DE FIN D'ETUDES**

**Pour l'obtention du diplôme de Master**

**Domaine : Sciences et Technologie**

**Filière : Electronique**

**Spécialité : Electronique des Systèmes Embarqués**

## **THEME**

**Etude et réalisation d'un système intelligent pour  
la commande d'éclairage public**

**Présenté par :**

**BELMEESABIH Ibrahim**

**BENGHENIA Mohamed Amine**

**Devant le jury :**

**Président : NASRI Djilali**

**Examineur : BELARBI Mustapha**

**Encadreur : KOUADRIA Mohamed**

**Promotion 2019/2020**

## ***Remerciement***

Après avoir rendu grâce à **ALLAH** le tout puissant et le Miséricordieux nous tenons remercier vivement tous ceux qui, de près ou le loin ont participé à la rédaction de ce document ; il s'agit plus particulièrement de :

Monsieur **Dr. KOUADRIA Mohamed** pour tout le soutien et l'encadrement qu'il nous a donné.

Nous voudrions également remercier les membres du jury **Dr. NASRI Djilali** et **Dr. BELARBI Mustapha**, pour avoir accepté d'évaluer ce travail et pour toutes leurs remarques et critiques.

Nos remerciements vont nos chers parents, nos sœurs, nos frères, tous nos amis pour leurs encouragements, et tous famille de **BELMESSABIH** et la famille de **BENGHENIA**.

Nous voudrions à cette occasion exprimer notre profonde gratitude à tous nos enseignants qui ont contribué par leur collaboration à notre formation

## دراسة و انجاز نظام ذكي للتحكم في الإضاءة العامة

**الملخص** تجد معظم دول العالم ، وخاصة النامية منها ، مشكلة في ارتفاع استهلاك الطاقة في الإضاءة العامة. ويرجع ذلك إلى عدة أسباب. من بينها أنظمة التحكم القديمة التي لا تستخدم المصابيح من أجل الاستهلاك الأمثل للطاقة. وأعمدة الإنارة تعمل بشكل سيئ ولا تلبي معايير المطابقة.

وعلى ضوء هذه المشاكل قمنا بهذا العمل الذي يهدف الى التحكم الذكي في الإنارة. مما يجعل من الممكن ، بشكل أساسي ، تقليل استهلاك الطاقة. وبالتالي ، يتم التحكم في شدة مصابيح الإضاءة مع مراقبة نسبة الضوء في البيئة ومحتوى الرطوبة في الجو، أيضًا من خلال اكتشاف المارة والضوضاء المحتملة حول مصابيح الشوارع. لذلك يتم التحكم في كل هذه الوظائف بواسطة نظام قائم على لوحة Arduino .

**الكلمات الجوهرية** إضاءة، مراقبة، مصباح، النظام، ذكي، المجلس ، أضواء الشوارع ، اردوينو

### Study and realization of an intelligent system for the control of public lighting

**Abstract** Most countries of the world, especially developing ones, find a problem in the high energy consumption in public lighting. This is due to several reasons, among which the old control systems do not use lamps for optimal energy consumption, and the light poles are poorly operated not respecting the conformity norms.

In light of these problems, we have carried out this work which aims to control an intelligent lighting system, which mainly helps to reduce energy consumption. Thus, the intensity of the lighting lamps is controlled while monitoring the percentage of light in the environment and the moisture content of the atmosphere, also by detecting passers-by and possible noise around public lights. So, all of these functions are controlled by an Arduino board based system.

**Key-words** Lighting, control, lamp, system, smart, board, street lights, Arduino

### Etude et réalisation d'un système intelligent pour la commande d'éclairage public

**Résumé** La plupart des pays du monde, notamment ceux en développement trouvent un problème dans la grande consommation d'énergie en éclairage public. Cela est dû à plusieurs raisons, parmi lesquelles les anciens systèmes de contrôle n'utilisent pas de lampes pour une consommation optimale d'énergie, et les poteaux d'éclairage sont mal exploités ne respectant pas les normes de conformité.

À la lumière de ces problèmes, nous avons réalisé ce travail qui vise à contrôler un système d'éclairage intelligent, ce qui permet, principalement, de réduire la consommation d'énergie. Ainsi, l'intensité des lampes d'éclairage est contrôlée tout en surveillant le pourcentage de lumière dans l'environnement et la teneur en humidité de l'atmosphère, aussi en détectant les passants et l'éventuel bruit aux alentours des lampadaires. Donc, toutes ces fonctions sont contrôlées par un système à base d'une carte Arduino.

**Mots-clés** Eclairage, contrôle, lampe, système, intelligent, carte, lampadaires, Arduino

## **Acronymes**

**PWM** :Pulsed Width Modulation ou modulation de large impulsion

**SRAM** :Static Random Access Memory ou mémoire vive statique

**EEPROM** : Electrically - Erasable Programmable Read -Only Memory ou mémoire morte effaçable électriquement et reprogrammable

**ISCP** : In Circuit Serial Programming

**IDE** : Integrated Development Environment ou Environnement de Développement

**CAO** : Conception Assisté par Ordinateur

**ISIS** : Intelligent Schematic Input System

**PCB** : bloc de contrôle de processus

**LED** : light-emitting diode ou diode électroluminescente

**LDR** : Light Dependent Resistor ou résistance dépendant de la lumière

**PIR** : Passive infrared detector ou capteur Infrarouge Passif

**USB** : Universal Serial Bus

**ADC** : Analog to Digital Converter

**DC** : direct current

## Table des matières

Introduction générale.....	2
I.1 Introduction.....	4
I.2 Histoire de l'éclairage.....	4
I.3 Cadre général de l'éclairage public.....	5
I.4 Principe de fonctionnement .....	5
I.5 Domaines d'utilisation .....	6
I.5.1 Usage industriel.....	6
I.5.2 Usage du secteur tertiaire .....	7
I.5.3 Médical et hospitalier .....	7
I.5.4 Éclairage de sécurité.....	8
I.5.5 Agriculture .....	8
I.5.6 Musées et galeries d'art .....	9
I.6 Organisation générale du système d'éclairage public.....	9
1.6.1 Gestion de system de l'éclairage .....	9
I.7 Nouvelles technologies mises en œuvre .....	11
I.7.1 En Algérie .....	11
I.7.2 Aux pays d'étrangers.....	12
I.8 Conclusion .....	13
II.1 Introduction .....	15
II.2 Définition et domaines d'utilisation .....	15
II.3 Avantages et inconvénients .....	16
II.4 Différents types de carte Arduino.....	17
II.4.1 Cartes à microcontrôleur de type Arduino.....	17
II.4.2 Différentes cartes Arduino.....	18
II.5 Caractéristiques d'une carte Arduino .....	19
II.5.1 Alimentation .....	19
II.5.2 Horloge .....	20
II.5.3 Reset.....	20
II.5.4 Entrées/sorties .....	20
II.1.5 Mémoire.....	21
II.6 Description de la carte Arduino UNO .....	22

II.6.2	Partie logiciel .....	23
II.7	Conclusion .....	27
III.1	Introduction .....	29
III.2	Circuit sans carte Arduino.....	30
III.2.1	Schéma synoptique .....	30
III.2.2	Circuit de détections .....	30
III.2.2.1	Détecteur de bruit.....	30
III.2.2.2	Détecteur de lumière .....	37
III.2.2.3	Détecteur de présence .....	39
III.2.2.4	Détecteur d'humidité .....	42
III.2.3	Partie de puissance.....	44
III.2.3.1	Circuit Gradateur de lumière.....	44
III.2.3.2	Circuit puissance globale .....	46
III.3	Circuit Avec carte Arduino .....	47
III.3.1	Schéma synoptique .....	47
III.3.2	Partie de commande.....	47
III.3.2.1	Capteur de lumière .....	48
III.3.2.2	Capteur de son.....	48
III.3.2.3	Capteur d'humidité .....	49
III.3.2.4	Détecteur de mouvement PIR .....	50
III.3.2.5	La partie d'affichage [18] .....	52
III.3.3	Partie de puissance.....	53
III.4	Etude comparative entre les cas sans et avec Arduino.....	53
III.5	Conclusion.....	54
IV.1	Introduction.....	56
IV.2	Présenter les logiciels qui exploiter .....	56
IV.2.1	Proteus .....	56
IV.2.2	Programmeur Arduino.....	57
IV.3	Simulation de notre projet sans carte Arduino.....	58
IV.4	Simulation de notre projet avec carte Arduino dans Proteus.....	60
IV.4.1	Détecteur de lumière .....	62
IV.4.2	Capteur de son .....	63
IV.4.3	Capteur d'humidité.....	64

IV.4.4 Détecteur de présence.....	65
IV.4.5 Fonction de contrôle de niveau de lumière .....	66
IV.5 Partie Réalisation .....	69
IV.5.1 Matériel utilisés .....	69
IV.5.2 Test et mise en œuvre .....	70
IV.5.3 Problèmes rencontrés .....	72
IV.5.4 Estimation du coût du projet réalisé .....	72
IV.6 Conclusion .....	73
Conclusion générale .....	75
Référence bibliographique .....	76

## Liste des figures

Figure I. 1 Histoire de l'éclairage résumé en images.....	4
Figure I. 2 Eclairage public .....	5
Figure I. 3 Eclairage dans le domaine industriel .....	7
Figure I. 4 Usage médicale.....	7
Figure I. 5 Eclairage de sécurité.....	8
Figure I. 6 Eclairage agricole .....	8
Figure I. 7 Eclairage de galerie d'art.....	9
Figure I. 8 Organisation générale de l'éclairage public .....	9
Figure I. 9 quelques circuits d'entrées .....	10
Figure I. 10 quelque source de l'éclairage .....	11
Figure I. 11 la première utilisation de lampadaires solaires dans Alger .....	12
Figure II. 1 (a) Carte Arduino UNO, (b) Arduino Méga 2560.....	20
Figure II. 2 Carte Arduino " Nano, (d) "Carte Arduino " Leonardo .....	20
Figure II. 3 e) carte Arduino Due, (f) carte Arduino Zero .....	21
Figure II. 4 Description des entrées/sorties de la carte Arduino UNO.....	24
Figure II. 5 Brochage d'une carte Arduino de type UNO [20] .....	26
Figure III. 1 Schéma représente les types des circuits étudiés .....	29
Figure III. 2 Schéma synoptique du montage Sans carte Arduino .....	30
Figure III. 3 Diagramme de bruit de véhicule .....	31
Figure III. 4 Schéma synoptique de détecteur de bruit .....	31
Figure III. 5 Le microphone ELECRET.....	32
Figure III. 6 Chronogramme du circuit Monostable redéclenchable .....	33
Figure III. 7 Le circuit électronique de détecteur de bruit.....	33
Figure III. 8 Schéma synoptique de circuit détecteur de lumière.....	37
Figure III. 9 Capteur LDR photorésistance .....	37
Figure III. 10 Circuit électronique de détecteur de lumière .....	38
Figure III. 11 Schéma synoptique de détecteur de présence .....	39
Figure III. 12 TSOP1738.....	40
Figure III. 13 Circuit d'électronique du détecteur de présence.....	40
Figure III. 14 Schéma synoptique de détecteur de lumière.....	42
Figure III. 15 Capteur d'humidité résistive .....	42
Figure III. 16 Montage de capteur d'humidité .....	43
Figure III. 17 Circuit électronique du détecteur d'humidité .....	43
Figure III. 18 Circuit électronique de gradateur de lumière.....	44
Figure III. 19 Schéma de circuit commande d'un gradateur de lumière.....	45
Figure III. 20 Circuit électronique de la partie puissance .....	46
Figure III. 21 Schéma synoptique du montage avec carte Arduino .....	47
Figure III. 22 Circuit électronique de la partie de commande.....	47
Figure III. 23 Capteur de lumière connecté à la carte Arduino .....	48
Figure III. 24 Circuit de capteur de son connecté à la carte Arduino.....	48
Figure III. 25 Capteur DHT11.....	49



Figure III. 26 Détecteur d'humidité connecté à la carte Arduino .....	50
Figure III. 27 Détecteur de présence PIR .....	51
Figure III. 28 Fonctionnement de capteur PIR.....	51
Figure III. 29 Détecteur de présence connecté à la carte Arduino .....	51
Figure III. 30 Afficheur LCD à logique intégré 02 lignes x 16 colonnes .....	52
Figure III. 31 Circuit électronique de la partie de puissance.....	53
Figure IV. 1 le principe outil de ISIS .....	56
Figure IV. 2 Interface de Arduino IDE .....	57
Figure IV. 3 Circuit électronique de la partie commande (sans arduino) .....	58
Figure IV. 4 Circuit électronique de la partie puissance (sans arduino) .....	59
Figure IV. 5 Organigramme de fonctionnement de notre projet sans carte Arduino.....	59
Figure IV. 6 Les signaux du niveau de luminosité.....	60
Figure IV. 7 Schéma électronique de la partie commande de notre projet (avec arduino).....	60
Figure IV. 8 Circuit électronique de la partie puissance de notre projet (avec Arduino) .....	61
Figure IV. 9 Organigramme de fonctionnement de notre projet avec carte Arduino .....	61
Figure IV. 10 Organigramme de fonctionnement de détecteur de lumière.....	62
Figure IV. 11 Niveau de la lumière.....	62
Figure IV. 12 Organigramme de fonctionnement de détecteur de bruit .....	63
Figure IV. 13 Simulations du Détecteur de bruit .....	63
Figure IV. 14 Organigramme de fonctionnement de détecteur d'humidité .....	64
Figure IV. 15 Simulations détecteur d'humidité avec Arduino .....	64
Figure IV. 16 Organigramme de fonctionnement de détecteur de présence .....	65
Figure IV. 17 Résultats de simulation de détecteur de mouvement.....	65
Figure IV. 18 Organigramme de contrôle du niveau de lumière.....	66
Figure IV. 19 Les signaux du niveau de luminosité des lampes .....	68
Figure IV. 20 La maquette .....	69
Figure IV. 21 Circuit électronique avec Arduino.....	70
Figure IV. 22 l'éclairage dans la nuit.....	70
Figure IV. 23 Détecteur de présence dans la nuit.....	71
Figure IV. 25 Afficheur LCD.....	71
Figure IV. 24 L'éclairage. Cas d'humidité .....	71

## Liste des tableaux

Tableau II. 1 Avantages et Inconvénients de différentes cartes de type Arduino .....	16
Tableau II. 1 Une comparaison de différents types de cartes Arduino .....	18
Tableau II. 3 Les branches entrée/sortie de la carte Arduino .....	21
Tableau II. 4 Description de la carte Arduino UNO .....	23
Tableau II. 5 Les parties d'arduino IDE .....	23
Tableau III. 1 Caractéristiques de LM324 .....	34
Tableau III. 2 Caractéristiques de monostable TC4538BP .....	35
Tableau III. 3 Caractéristiques de transistor BC547 .....	36
Tableau III. 4 Les caractéristiques de relaie SRD-12VDC .....	37
Tableau III. 5 Caractéristiques de H25K5 .....	42
Tableau III. 6 Etude comparative entre les cas sans et avec Arduino .....	54
Tableau IV. 1 Estimation de de projet.....	72

# ***Introduction générale***

## **Introduction générale**

L'éclairage public est en pleine mutation depuis quelques années grâce aux développements de la vie urbaine et sa fonction de base, permettant aux utilisateurs de vivre dans de bonnes conditions de confort. Malgré ça, l'éclairage a un grand problème tel que l'abus de l'énergie électrique et sa mauvaise gestion. [1].

Notre projet consiste de l'étude, la simulation et la réalisation d'un système à base d'une carte ARDUINO permettant la commande automatique de l'éclairage public pour assurer une commande optimale de l'éclairage.

Dans le premier chapitre nous avons présenté l'éclairage dans une façon générale où nous avons touché l'histoire de l'éclairage, principe de fonctionnement, organisation générale ainsi que la dernière technique de l'éclairage en Algérie par rapport aux pays étrangers.

Ensuite, le deuxième chapitre sera consacré à l'étude des caractéristiques, avantages et inconvénients, des différentes cartes Arduino et enfin à la description de cette carte.

Au chapitre III, nous avons fait l'étude théorique des deux circuits (un circuit de contrôle d'éclairage sans carte Arduino et l'autre avec une carte Arduino), avec principe de fonctionnement et les calculs.

Enfin, dans le chapitre IV, nous avons simulé les deux circuits à l'aide du logiciel Proteus et réalisé le deuxième circuit (contrôle d'éclairage Arduino), et présentant la maquette ainsi que sa mise en œuvre.

# ***Chapitre I***

## Généralités sur l'éclairage public

## I.1 Introduction

L'éclairage public est obtenu en utilisant des dispositifs qui convertissent l'énergie électrique en lumière, pour satisfaire le besoin de l'éclairage dans nombreux sites tel qu'éclairage de la route.

Dans ce chapitre, nous avons présenté l'histoire de l'éclairage, le principe de fonctionnement, les domaines d'application et les nouvelles technologies appliquées.

## I.2 Histoire de l'éclairage

En regardant un peu en arrière dans l'histoire de l'éclairage public, on découvre qu'il s'agit de quelque chose relativement récent dans notre société. La véritable mise en place d'un système d'éclairage public est due au perfectionnement des moyens d'éclairer, mais aussi, et en grande partie, à l'arrivée de voitures sur les routes. Depuis les temps anciens déjà, l'homme avait compris qu'il y avait un cycle de jour et de nuit et que durant les heures sombres il était nécessaire d'inventer quelque chose qui pourrait éclairer les objets et les personnes [5].

L'homme s'est toujours soucié d'éclairer son habitat. Cela a commencé par la maîtrise du feu, il y a environ 400 000 ans selon les estimations. Les sources d'éclairage artificielles existent quant à elles depuis plus de 15 000 ans. Au fil des siècles, de nombreuses inventions ont vu le jour : bougie, lampe à huile, éclairage au gaz, lampe à incandescence, lampe à décharge, et depuis le début du 21ème siècle, la LED, de plus en plus utilisée comme moyen d'éclairage [3].

Aux alentours de l'an 1000, un premier éclairage public est mentionné à Cordoba, Al-Andalous.



**Figure I. 1** Histoire de l'éclairage résumé en images

### I.3 Cadre général de l'éclairage public

L'éclairage public consiste, d'une part, dans l'éclairage des voies et des lieux publics pour des raisons de sécurité, d'autre part, dans l'illumination de lieux, de monuments, de bâtiments ou de tout autre élément public ou privé du patrimoine culturel. Ces illuminations peuvent être permanentes, saisonnières ou encore liées à des manifestations particulières [4].

Les réseaux d'éclairage public sont constitués des luminaires, et de leurs supports (candélabres, consoles) et de circuits d'alimentation (conducteurs aériens, câbles souterrains, accessoires) qui peuvent être soit indépendants du réseau de distribution publique d'électricité, soit faire partie de ce réseau [4].



**Figure I. 2** *Eclairage publique*

### I.4 Principe de fonctionnement

L'éclairage public est généralement un éclairage de chaussée offrant une amélioration de la visibilité.

Il est utilisé lorsqu'il y a fréquemment coexistence de piétons et de véhicules, c'est-à-dire essentiellement à l'intérieur des localités, dans les zones bâties et le long des autoroutes et des voies de circulations rapides.

Il s'agit principalement de créer dans ces espaces, des conditions permettant aux usagers de la circulation de s'identifier mutuellement rapidement.

En outre, l'éclairage des rues doit donner un sentiment de sécurité et contribuer à la prévention des accidents, (par exemple collisions aux carrefours et sur les passages pour piétons) [1].

## **I.5 Domaines d'utilisation**

### **I.5.1 Usage industriel**

D'origine fonctionnelle, l'éclairage industriel doit répondre aux normes concernant l'éclairage des postes de travail. Ce type d'éclairage est spécialement adapté aux locaux où il est installé, où les contraintes de volumes, d'empoussièrement et de maintenance sont particulières. Dans les industries où sont effectuées des tâches de mécanique fine et de précision, ainsi que les secteurs de l'électronique, des renforts d'éclairage sur les postes de travail sont installés [5].

L'éclairage utilisé dans les locaux industriels est généralement de conception simple avec une recherche d'efficacité et de facilité d'emploi, munis de source peu consommatrice en énergie, de type fluo ou sodium. Certains secteurs industriels (chimique notamment) nécessitent l'emploi d'appareils protégés. Les industries où le rendu des couleurs est important (imprimerie) nécessitent l'emploi de lampes adaptées. Enfin, certains processus industriels nécessitent l'emploi d'émissions lumineuses particulière type UV ou IR dans des procédés de [5] :

- Durcissement/polymérisation des résines ;
- Accélération photochimique ;
- Chauffage et séchage de peinture ;
- Thermoformage des matières plastiques ;
- Désinfection et stérilisation.





**Figure I. 3** *Eclairage dans le domaine industriel*

### **I.5.2 Usage du secteur tertiaire**

Les lampes fluorescentes sont majoritairement employées dans les luminaires tertiaires de bureau. Décrites à tort comme froides, les lampes fluorescentes permettent, bien disposées, une excellente uniformité d'éclairage. Apparent, suspendu ou encastré, l'éclairage de bureau est souvent complété par des lampes d'appoint pour répondre au besoin de personnaliser la quantité et/ou la qualité de l'éclairage sur chaque poste de travail [5].

### **I.5.3 Médical et hospitalier**

L'éclairage des salles d'opération, de certaines salles d'examen et de soins, ainsi que les salles de soins des cabinets dentaires et des prothésistes dentaires utilise des appareils d'éclairage adaptés (forts niveaux d'éclairages, contrôlent des luminances, spectre des températures de couleur, etc.) [5].



**Figure I. 4** *Usage médicale*

### I.5.4 Éclairage de sécurité

Dans les lieux de travail ou accueillant du public, (magasin, hôtel, bureau, atelier), un éclairage dit de sécurité, ou de secours, est requis par la plupart des réglementations. Ces luminaires spécifiques se mettent automatiquement en fonction, lors des coupures de courant électrique ou dans les situations d'urgences (incendie, évacuation). Ils émettent une lumière relativement faible, mais suffisante ; placés aux endroits stratégiques (changement de direction, porte, escalier, porte de sortie), ils balisent le ou les itinéraires vers la ou les sorties de secours. Les dispositifs d'éclairage de sécurité répondent à des normes de conception strictes.[2]



**Figure I. 5** *Eclairage de sécurité*

### I.5.5 Agriculture

Certains pays se sont fait une spécialité de la culture sous serre avec température et éclairage contrôlés afin d'accélérer le processus de maturation des plantes. Cette culture utilise des lampes émettant dans des longueurs d'ondes spécifiques aux plantes. De même, l'élevage intensif en batterie de la volaille utilise l'éclairage pour accélérer la croissance en raccourcissant le cycle diurne/nocturne [1].



**Figure I. 6** *Eclairage agricole*

### I.5.6 Musées et galeries d'art

Depuis les années 1990, la fibre optique est utilisée pour véhiculer la lumière sur un trajet de quelques dizaines de centimètres depuis une source vers l'objet à mettre en valeur, permettant d'obtenir des éclairages ponctuels et discrets, pouvant être élégamment intégrés à une vitrine de présentation, et offrant l'avantage de rayonner très peu d'infrarouge, limitant ainsi le risque d'élévation de température à l'intérieur de la vitrine, néfaste aux œuvres d'art[1]



Figure I. 7 Eclairage de galerie d'art

### I.6 Organisation générale du système d'éclairage public



Figure I. 8 Organisation générale de l'éclairage public

#### 1.6.1 Gestion de system de l'éclairage

Il y a deux types, système manuels et système automatique de gestion de l'éclairage

- Dans le premier système la commande d'éclairage fait manuellement tel que : disjoncteur permet à l'utilisateur d'éteindre, d'allumer, et potentiomètres gradateurs variateurs et boutons poussoirs sont permettent à l'utilisateur d'éteindre, d'allumer et de faire varier l'intensité de l'éclairage.
- Dans le deuxième système la commande d'éclairage se fait par des capteurs, comme capteur de présence pour éclairer les zones des circulations seulement, et cellules photosensibles pour d'allumer, d'éteindre un ou plusieurs luminaires en fonction de la lumière ambiante.



**Figure I. 9** quelques circuits d'entrées

#### a) Gestion centralisée de l'éclairage

La gestion centralisée permet le contrôle, la commande et la gestion horaire et calendrier de la gestion d'éclairage, Elle permet aussi de connaître l'état et les consommations de l'installation d'éclairage [1].

#### b) Les lampes

La lampe est une source d'éclairage, Il est existé deux types de lampes [1] :

- **Les lampes à incandescence** caractérise par un filament brule sont utilisée pour l'éclairage intérieur parce que le rendement lumineux d'elle sont faible.
- **Les lampes à décharge** qui produisent de la lumière grâce à une décharge électrique dans un gaz (néons, les Lampes à vapeur de mercure, les lampes à vapeur de sodium à basse pression et à haute pression).





**Figure I. 10** *quelque source de l'éclairage*

## I.7 Nouvelles technologies mises en œuvre

### I.7.1 En Algérie

La ville d'Annaba a opté à l'afin de 2018 pour des lampadaires solaires spécial bord de mer sur la route côtière menant à la plage. Les lampadaires photovoltaïques s'adaptent à tous les projets urbains : routes, voiries, parkings, chemins piétons, parcs et jardins publics, lieux de vie, places de villages, quartiers, zones isolées, etc. Avec son réseau de partenaires proches de chez vous, Fonroche Eclairage peut répondre à tous les projets d'éclairage urbains en Afrique du Nord [8].



**Figure I. 11** *la première utilisation de lampadaires solaires dans Alger*

### **I.7.2 Aux pays d'étrangers**

Plusieurs technologies sont actuellement développées afin de rendre l'éclairage public intelligent [8]:

- Le « dimming » : cette technologie consiste à abaisser le flux lumineux des lampes en réduisant la tension d'alimentation aux heures de faible fréquentation pour réaliser des économies d'énergie. Le système permet de diminuer le niveau d'éclairage de façon progressive afin de le rendre presque imperceptible par l'utilisateur.
- La télégestion : système de gestion globale de l'éclairage public à partir d'un centre de contrôle, duquel des informations sont transférées à chaque point lumineux de manière à commander leur allumage, extinction ou « dimming » de manière individuelle ou groupée.
- La télémaintenance : généralement complémentaire de la télégestion, permet de transférer les informations de chaque point lumineux vers le centre de contrôle, en particulier pour en connaître les dysfonctionnements, planifier et contrôler le remplacement sur le fondement de leur nombre d'heures de fonctionnement réel.
- Les capteurs de mouvement : les lampadaires sont équipés de détecteurs de mouvement, ce qui leur permet de s'allumer uniquement lorsqu'il y a du passage et donc de faire des économies d'énergie.
- Les lampadaires autoalimentés : ces lampadaires sont alimentés par des sources d'énergie renouvelable (éolienne ou panneau photovoltaïque avec stockage) et peuvent

équiper des lieux nécessitant d'être éclairés mais ne disposant pas d'un réseau d'éclairage public.

- L'horloge astronomique : quand le soir tombe ou que le jour se lève, on allume ou on éteint progressivement ;
- Capteurs de luminosité qui peuvent mesurer le niveau d'éclairement des surfaces, variable avec la clarté du ciel, la pluie, ou – ce qui est essentiel dans une région nordique – la réverbération sur la neige ;
- Mesure du trafic qui transmet des données sur l'encombrement des routes, les vitesses des véhicules. Elles permettent alors d'améliorer l'éclairage dans les périodes de pointe de trafic.



**Figure I. 12** *Eclairage de dernière technologie dans l'étranger*

## Conclusion

Dans ce chapitre nous avons présenté une description générale de l'éclairage public et son fonctionnement, les domaines d'utilisation ainsi que les dernières technologies de l'éclairage public. Malgré les progrès réalisés, la réduction de la consommation d'éclairage public reste un problème pour des nombreux pays.

Dans le chapitre suivant, nous présenterons les différentes cartes Arduino avec leurs caractéristiques. Et la description de la carte Arduino qui sera utilisée pour notre projet.

## ***Chapitre II***

Présentation de la carte Arduino



## II.1 Introduction

La commande des circuits électroniques à base des cartes à microcontrôleur est de plus en plus utilisée ces dernières années. Donc, différentes applications pour divers domaines ont recours à une multitude de cartes de développement, soit pour une autonomie de fonctionnement destiné à des tâches bien précises, soit comme dispositifs d'interface entre les circuits d'entrées et ceux de sorties pour une meilleure efficacité de commande.

Le langage Arduino se distingue des langages utilisés dans l'industrie de l'informatique embarquée par sa simplicité. En effet, beaucoup de bibliothèques et de fonctionnalités de base occultent certains aspects de la programmation de logiciel embarquée afin de gagner en simplicité. Cela en fait un langage parfait pour réaliser des prototypes ou des petites applications [8].

Mais, avant toute application du type de carte Arduino, il est important de connaître ses concepts de base. Donc, dans ce chapitre nous allons considérer les types de cartes Arduino, les caractéristiques ainsi que les différentes parties liées à son fonctionnement.

## II.2 Définition et domaines d'utilisation :

Arduino est un système électronique dont les plans de la carte même sont publiés en licence libre qui est basée sur un microcontrôleur (de la famille AVR), et un logiciel, véritable environnement de développement intégré, pour écrire, compiler et transférer le programme vers la carte à microcontrôleur. Les signaux offerts du microcontrôleur sont présents et disposés dans des connecteurs spécialement utilisés pour l'interfaçage [12].

Elle est très utilisée tant que [1] :

- ✓ son prix est modeste
- ✓ son interface est simple
- ✓ son utilisation est aisée permettant un prototypage ou une conception de l'application en un temps record.

Le système Arduino permet à l'utilisateur de réaliser un grand nombre de projets puisque l'étendue de l'utilisation de l'Arduino est large. Par exemples :

- Pour le contrôle des appareils domestiques
- Utilisation dans la réalisation d'un robot
- Dans des applications conçues pour des jeux de lumières
- Communiquer avec l'ordinateur

- Télécommander un appareil mobile

### II.3 Avantages et inconvénients :

Dans cette partie, nous passons en revue quelques différentes cartes d'Arduino en mettant l'accent principalement sur leurs avantages et leurs inconvénients, Tableau II.1.

Généralement, le principal avantage d'une carte à microprocesseur est le traitement rapide et une interface facile.

Type de carte Arduino	Avantages	Inconvénients
<b>Uno</b>	<ul style="list-style-type: none"> <li>• Nombreux exemples de montage sont disponibles sur internet.</li> <li>• Nombres suffisant de broches d'E/S pour des projets élémentaires.</li> <li>• Vaste choix de <b>shields</b></li> <li>• Bon marché</li> </ul>	<ul style="list-style-type: none"> <li>• Nombre de broche insuffisant pour les gros projets</li> <li>• La mémoire disponible risque d'être un peu juste pour projets ambitieux.</li> <li>• Ne peut pas être utilisée comme hôte USB pour simuler périphérique tel qu'un clavier ou une souris.</li> <li>• Acquiert des modules externes qui augmenteront inévitablement le coût total d'achat</li> </ul>
<b>Nano</b>	<ul style="list-style-type: none"> <li>• Encombrement réduit</li> <li>• Possibilité d'être enfichée directement sur la plaque d'essai</li> </ul>	<ul style="list-style-type: none"> <li>• Il n'est pas possible d'utiliser des <b>shields</b></li> </ul>
<b>Mini Pro</b>	<ul style="list-style-type: none"> <li>• Taille réduite et son faible coût</li> <li>• Fournie sans connecteurs pour être facilement intégrée à tout projet final.</li> </ul>	<ul style="list-style-type: none"> <li>• Ne possède de convertisseur USB / série.</li> <li>• Ne possède pas non plus de sortie 3.3V (disponible sur les autres cartes)</li> <li>• Conçu pour les applications où l'espace est très petit</li> <li>• Conçu pour être intégré en permanence dans un projet.</li> <li>• Destiné à un public plus expérimenté.</li> </ul>
<b>Mega</b>	<ul style="list-style-type: none"> <li>• Nombreuses entrées et sorties pour raccorder des capteurs ou des actionneurs.</li> <li>• Capacité de mémoire suffisante pour les gros projets.</li> <li>• Plus de broches UART (4ports de communication série)</li> <li>• Plus de broches MLI(15sorties numériques peuvent être utilisées comme MLI.</li> <li>• Compatible avec la plupart des shields conçues pour Arduino</li> </ul>	<ul style="list-style-type: none"> <li>• Deux fois plus chère que l'Arduino Uno</li> </ul>

**Tableau II. 1** *Avantages et Inconvénients de différentes cartes de type Arduino*

## II.4 Différents types de carte Arduino

### II.4.1 Cartes à microcontrôleur de type Arduino

- En 2004 : l'Arduino UNO a été développé par des enseignants et élèves d'une école de Design italienne, qui avaient pour vocation de démocratiser la programmation et de la rendre plus libre d'accès.

Les cartes à microcontrôleur de type ARDUINO sont disponibles avec de nombreux types de modules qui peuvent être reliés avec celui-ci. Ces modules sont connus sous le nom de « Shield ». Certains des Shield les plus couramment utilisés sont:

- **Shields** Moteurs et servos : interface dédiée au pilotage des moteurs
- **Shields** Afficheurs : permet de contrôler un afficheur
- **Shields** Relais : pour actionner des dispositifs
- **Shields** Ethernet : pour connecter un projet Arduino à Internet par le biais d'un câble RJ45
- **Shields** WIFI : permettant de connecter la carte à Internet
- **Shields** GSM : permet d'envoyer des SMS, MMS, GPRS et d'établir des communications audios.
- **Shields** Bluetooth : permet de commander un dispositif branché sur ARDUINO

II.4.2 Différentes cartes Arduino

Il existe plusieurs types de cartes Arduino, nous citons :

Carte Arduino	Microcontrôleur ( $\mu\text{C}$ )	Type (bits)	CLK (MHz)	Alim (V)
Leonardo	ATmega32U4	8	16	5
Uno	ATmega328P	8	16	5
Nano	ATmega328P	8	16	5
Due	AT91SAM3X8E	32	84	3,3
Mega	ATmega2560	8	16	5
Zero	ATSAMD21G18	32	48	3,3

Carte Arduino	Flash (KB)	SRAM (KB)	EEPROM (KB)	Digital I/O	Sorties PWM	Analog I/O
Leonardo	32	2,5	1	20	7	12
Uno	32	2	1	14	6	6
Nano	32	2	1	14	6	8
Due	512	96	-	54	12	12/2
Mega	256	8	4	54	15	16
Zero	256	32	-	20	10	6/1

Tableau II. 2 Une comparaison de différents types de cartes Arduino [10]

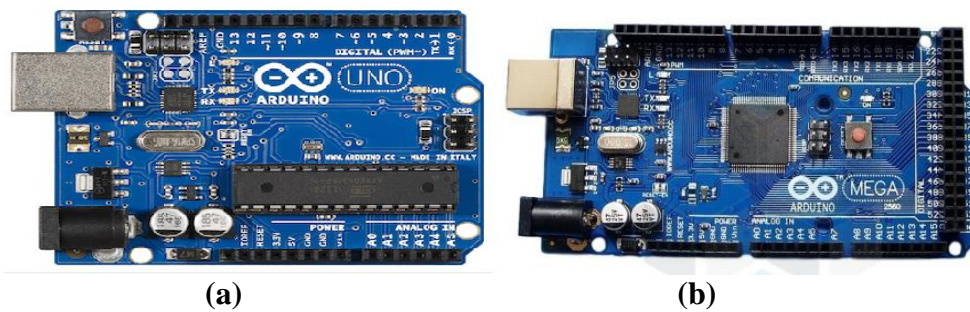


Figure II. 1 (a) Carte Arduino UNO, (b) Arduino Méga 2560

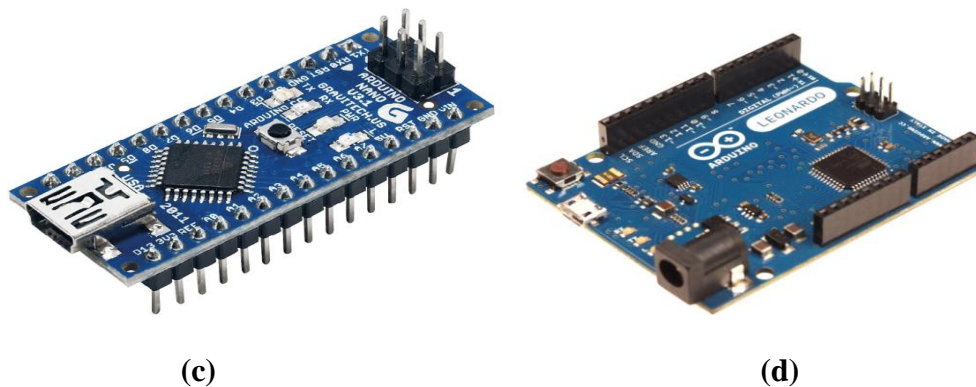


Figure II. 2 Carte Arduino " Nano, (d) "Carte Arduino " Leonardo

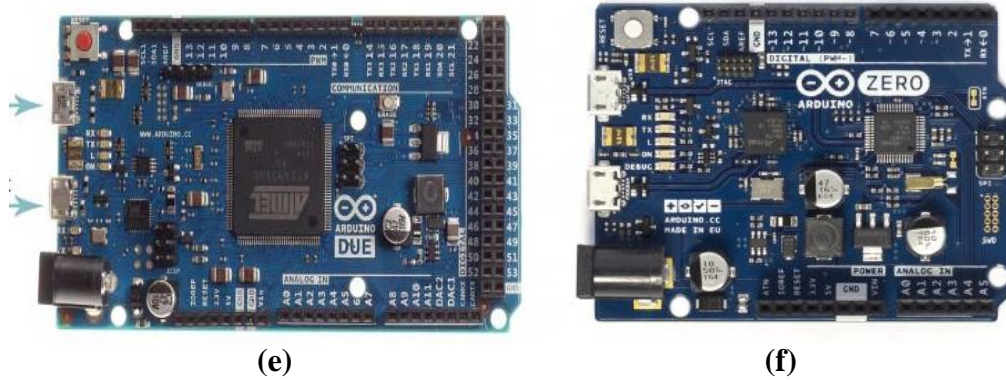


Figure II. 3 e) carte Arduino Due, (f) carte Arduino Zero

## II.5 Caractéristiques d'une carte Arduino

Comme énoncé précédemment, il existe beaucoup de cartes Arduino différentes, mais elles possèdent toutes des éléments en commun [9], à savoir

### II.5.1 Alimentation

Le microcontrôleur présent en général sur les cartes Arduino est alimenté par une tension de 5V.

En fonction de modèle de la carte, la tension peut être fournie

- soit par une des prises d'alimentation présentes sur la carte.
- soit par la prise USB utilisée pour la connecter à un ordinateur.

La valeur de la tension à fournir sur une des prises d'alimentation doit être comprise entre 7 et 12V, mais cette tension n'a pas besoin d'être stabilisée en raison de la présence d'un régulateur de tension sur la carte.

Il peut exister deux prises d'alimentation sur une carte Arduino.

- Il s'agit d'une simple prise jack standard, présente sur les cartes Arduino Uno et Arduino Mega, sur laquelle il suffit de raccorder un bloc d'alimentation fournissant la tension requise pour démarrer sa carte.
- Il s'agit d'une broche nommée Vin présente sur tous les modèles. Il suffit alors de raccorder la branche positive de son alimentation à cette broche, comme on le ferait pour une pile, et la branche négative sur une broche GND.
- La carte possède également deux broches de tensions stabilisées de 3,3V et de 5V. Ces deux tensions sont généralement utilisées pour alimenter des accessoires d'Arduino qui viendront s'ajouter sur l'Arduino, mais elles peuvent être utilisées pour d'autres

circuits.

### II.5.2 Horloge

L'horloge détermine la fréquence ou la rapidité à laquelle les instructions seront exécutées par le microcontrôleur. Ainsi, la fréquence peut varier d'un microcontrôleur à l'autre et est exprimée en hertz (Hz).

Si les PC et Mac peuvent atteindre des fréquences de plusieurs gigahertz, la fréquence est en revanche beaucoup plus faible pour les microcontrôleurs, avec des fréquences de quelques mégahertz seulement.

De plus, Cette fréquence est déterminée par un oscillateur à quartz ou résonateur céramique. Sur les cartes Arduino, l'horloge est généralement cadencée à 16 MHz.[1]

### II.5.3 Reset

La reset est une fonction physique permettant au microcontrôleur, comme son nom l'indique, de réinitialiser son état. Ainsi, la fonction reset permet alors d'interrompre le programme et de le remettre à zéro. Il est toutefois plus aisé d'utiliser cette fonction reset matérialisée sur la carte par un bouton presseur [1].

### II.5.4 Entrées/sorties

Les entrées et sortie sont les moyens que possède le microcontrôleur pour communiquer avec le monde extérieur [1].

- les entrées /sorties sont les broches métalliques qui donnent cette forme familière aux puces électroniques. Ces broches reçoivent ou émettent des signaux logiques qui peuvent alors être interprétés par le microcontrôleur ou d'autres circuits.
- Le nombre de broches peut varier d'un microcontrôleur à l'autre, et même d'une carte Arduino à l'autre.
- Généralement, ces broches fonctionnent sous une tension maximale de 5V et ne peuvent recevoir et fournir que 40 mA de courant.

On peut ranger ces broches par catégorie suivant leur fonctionnalité :

Broche(s)	Fonctions
<b>Digitales</b>	<ul style="list-style-type: none"> <li>- Elles fournissent des données digitales sous forme de signaux logiques.</li> <li>- Elles ne peuvent donc contenir que deux valeurs, une 0 logique correspondant à une absence de tension et une 1 logique correspondant à une tension de 5V.</li> </ul>
<b>Analogiques</b>	<ul style="list-style-type: none"> <li>- Ces broches ne fonctionnent qu'en entrée.</li> <li>- Elles acceptent des tensions comprises entre 0 et 5V. ces tensions sont ensuite utilisées par un convertisseur analogique/numérique qui s'occupe de traduire ces valeurs physiques en données numériques sur 10 bits, soit des données comprises entre 0 et 1023.</li> </ul>
<b>Alimentation</b>	<ul style="list-style-type: none"> <li>- Ces broches servent à alimenter des circuits externes.</li> <li>- Elles permettent également d'alimenter la carte Arduino sans passer par la prise jack présente sur certains modèles.</li> </ul>
<b>Communication</b>	<ul style="list-style-type: none"> <li>- il existe trois ports de communication sur les cartes Arduino : le port série, le port I2C et le port SPI. . Ces ports ne disposent pas de broches.</li> <li>- Ils partagent en revanche leur fonctionnalité avec d'autres broches. Les broches partagées sont différentes d'une carte à l'autre.</li> </ul>

**Tableau II. 3** Les branches entrée/sortie de la carte Arduino

### II.1.5 Mémoire

La plupart des cartes Arduino disposent de trois types de mémoires :

- mémoire flash,
- SRAM
- EEPROM

La **mémoire flash** et l'**EEPROM** sont deux mémoires non volatiles dites mortes, c'est-à-dire qu'elles conservent leurs données même en l'absence de courant électrique.

La **mémoire flash** contient le programme qui sera exécuté par le microcontrôleur lors de chaque mise sous tension.

L'**EEPROM** sert à stocker des variables que l'on souhaite conserver lors d'une mise hors tension à l'autre. Elle peut être utile si l'on souhaite stocker des données globales qui sont susceptibles d'être réutilisées tout au long de la durée de vie d'un projet.

Nous avons choisi la carte Arduino uno, parce qu'elle a tout ce dont nous avons besoin pour tirer pleinement parti du microcontrôleur : 14 entrées / sorties, une connexion USB, un cristal 16 MHz et un en-tête ICSP, Et moins cher que les autres cartes.

## II.6 Description de la carte Arduino UNO

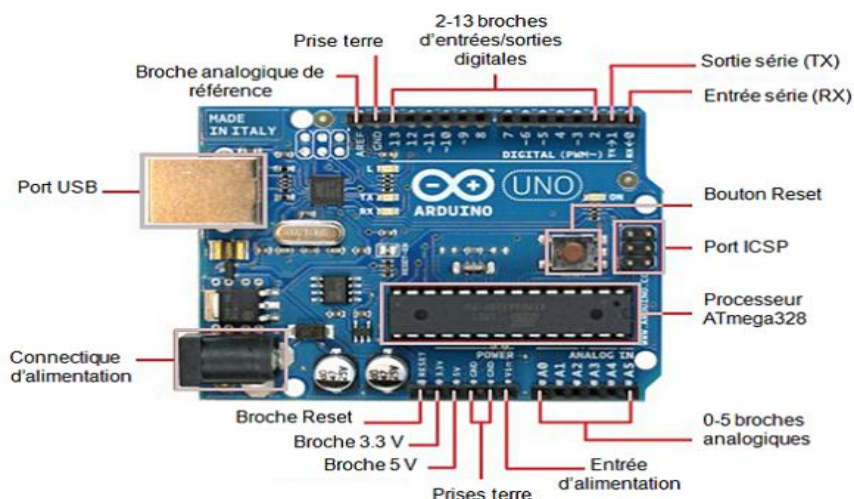
### II.6.1 Partie matérielle

La carte Arduino UNO est le produit populaire parmi les cartes Arduino. Parfaite pour débiter la programmation Arduino, elle est constituée de tous les éléments de base pour construire des objets d'une complexité relativement faible. D'autres cartes sont présentées dans le (tableau II.1).

La carte Arduino UNO est constituée de :

- 14 broches d'entrées/sorties digitales, dont six sont utilisables en PWM, de 6 broches d'entrées analogiques.
- d'une connectique USB.
- d'une connectique d'alimentation,
- Connecteur In-Circuit Serial Programming ICSP pour le téléchargement du programme.
- d'un bouton RESET.
- Microcontrôleur ATmega328 est un microcontrôleur ATMEL de la famille AVR 8bits

La description de toutes les connectiques de la carte Arduino UNO est présentée dans l'image ci-dessous :



**Figure II. 4** Description des entrées/sorties de la carte Arduino UNO



## Les Caractéristiques techniques

CPU	ATMega328
Mémoire flash	32Ko
Mémoire SRAM	2Ko
Mémoire EEPROM	1Ko
Cadencement	16MHz
Alimentation	7 – 12 V ou USB
USB-B	Pour la programmation
E/S numériques	14 proches dont 6 PWM
E/S Analogique	6 proches
Autres E/S	Série, I2C, SPI
Intensité par sortie	40 mA

Tableau II. 4 Description de la carte Arduino UNO

## II.6.2 Partie logiciel

Le code de programme écrit pour Arduino est connu sous le nom de croquis (sketch). Le logiciel utilisé pour développer de tels croquis pour un Arduino est communément appelé Arduino IDE. Cet IDE contient les parties suivantes [21]. L'ensemble de ces points seront traités et vus dans le chapitre 4 :

ARDUINO IDE	Éditeur de texte	C'est là que le code simplifié peut être écrit à l'aide d'une version simplifiée du langage de programmation C ++.
	Zone de message	Il affiche une erreur et donne également un retour sur l'enregistrement et l'exportation du code.
	Texte	La console affiche la sortie de texte par l'environnement Arduino, y compris les messages d'erreur complets et d'autres informations.
	Barre d'outils de la console	Cette barre d'outils contient divers boutons tels que Vérifier, Télécharger, Nouveau, Ouvrir, Enregistrer et Serial Monitor...

Tableau II. 5 Les parties d'arduino IDE

## ➤ Principe d'utilisation d'une carte de développement ARDUINO

Cette partie explore comment une carte ARDUINO peut être utilisée comme un outil pour les travaux d'étude et de recherche [22]. De ce fait, la communication entre le PC et la carte se fait via le port USB, moyennant installation d'un driver adapté (fourni par ARDUINO).



Figure II. 5 Brochage d'une carte Arduino de type UNO [20]

### ▪ Les étapes de programmation ARDUINO

Après la préparation du programme, il y'a des étapes à faire afin de terminer la programmation, qui sont les suivantes :

- ✓ Obtention d'une carte ARDUINO et un câble USB
- ✓ Téléchargement de l'environnement ARDUINO
- ✓ Raccordement de la carte ARDUINO au PC
- ✓ Installation des pilotes du périphérique Série-USB
- ✓ Lancement de l'application ARDUINO

### ▪ Chargement du programme dans la carte

**Étape 1** : lancement du logiciel.



**Étape 2** : ouverture et modification du programme.

**Étape 4** : connexion de la carte au PC avec le cordon USB.

**Étape 5** : transfert du programme vers la carte.



**Étape 6** : vérification du fonctionnement.

### ▪ Structure d'un projet ARDUINO

Un programme ARDUINO est composé de 3 parties:

**Partie1** : déclaration des variables (optionnelle)

**Partie2** : représentée par la **Fonction setup ()**. C'est une partie initialisation et configuration des entrées / sorties. Elle est appelée une seule fois lorsque le programme commence.

**Partie3** : représentée par la **Fonction loop ()**. C'est la partie principale contenant le programme. Elle est répétée indéfiniment en boucle infinie

▪ **Notions du langage ARDUINO**

**Structure**

*Fonctions de base*

- `void setup()`
- `void loop()`

*Structures de contrôle*

- `if`
- `if...else`
- `for`
- `switch case`
- `while`
- `do... while`
- `break`
- `continue`
- `return`
- `goto`

*Syntaxe de base*

- `;` (point virgule)
- `{}` (accolades)
- `//` (commentaire sur une ligne)
- `/* */` (commentaire sur plusieurs lignes)
- `#define`
- `#include`

*Opérateurs arithmétiques*

- `=` (égalité)
- `+` (addition)
- `-` (soustraction)
- `*` (multiplication)
- `/` (division)
- `%` (modulo)

*Opérateurs de comparaison*

- `==` (égal à)
- `!=` (différent de)
- `<` (inférieur à)
- `>` (supérieur à)
- `<=` (inférieur ou égal à)
- `>=` (supérieur ou égal à)

*Opérateurs booléens*

- `&&` (ET booléen)
- `||` (OU booléen)
- `!` (NON booléen)

*Opérateurs composés*

- `++` (incréméntation)
- `--` (décréméntation) (à revoir)
- `+=` (addition composée)
- `-=` (soustraction composée)
- `*=` (multiplication composée)
- `/=` (division composée)
- `&=` (ET bit à bit composé)
- `|=` (OU bit à bit composé)

*Opérateurs bit à bit*

- `&` (ET bit à bit)
- `|` (OU bit à bit)
- `^` (OU EXCLUSIF bit à bit)
- `~` (NON bit à bit)
- `<<` (décalage à gauche)
- `>>` (décalage à droite)

*Pointeurs*

- `*` pointeur
- `&` pointeur

## Fonctions

<p><b>Entrées/Sorties Numériques</b></p> <ul style="list-style-type: none"> <li>• <code>pinMode(broche, mode)</code></li> <li>• <code>digitalWrite(broche, valeur)</code></li> <li>• <code>int digitalRead(broche)</code></li> </ul> <p><b>Entrées analogiques</b></p> <ul style="list-style-type: none"> <li>• <code>int analogRead(broche)</code></li> <li>• <code>analogReference(type)</code></li> </ul> <p><b>Sorties "analogiques" (génération d'impulsion)</b></p> <ul style="list-style-type: none"> <li>• <code>analogWrite(broche, valeur)</code> - PWM</li> </ul> <p><b>Entrées/Sorties Avancées</b></p> <ul style="list-style-type: none"> <li>• <code>tone()</code></li> <li>• <code>noTone()</code></li> <li>• <code>shiftOut(broche, BrocheHorloge, OrdreBit, valeur)</code></li> <li>• <code>unsigned long pulseIn(broche, valeur)</code></li> </ul> <p><b>Communication</b></p> <ul style="list-style-type: none"> <li>• <code>Serial</code></li> </ul>	<p><b>Temps</b></p> <ul style="list-style-type: none"> <li>• <code>unsigned long millis()</code></li> <li>• <code>unsigned long micros()</code></li> <li>• <code>delay(ms)</code></li> <li>• <code>delayMicroseconds(us)</code></li> </ul> <p><b>Math</b></p> <ul style="list-style-type: none"> <li>• <code>min(x, y)</code></li> <li>• <code>max(x, y)</code></li> <li>• <code>abs(x)</code></li> <li>• <code>constrain(x, a, b)</code></li> <li>• <code>map(valeur, toLow, fromHigh, toLow, toHigh)</code></li> <li>• <code>pow(base, exposant)</code></li> <li>• <code>sq(x)</code></li> <li>• <code>sqrt(x)</code></li> </ul> <p><i>Pour davantage de fonctions mathématiques, voir aussi la librairie math.h : log, log10, asin, atan, acos, etc...</i></p> <p><b>Nombres randomisés (hasard)</b></p> <ul style="list-style-type: none"> <li>• <code>randomSeed(seed)</code></li> <li>• <code>long random(max)</code></li> <li>• <code>long random(min, max)</code></li> </ul>	<p><b>Trigonométrie</b></p> <ul style="list-style-type: none"> <li>• <code>sin(rad)</code></li> <li>• <code>cos(rad)</code></li> <li>• <code>tan(rad)</code></li> </ul> <p><b>Bits et Octets</b></p> <ul style="list-style-type: none"> <li>• <code>lowByte()</code></li> <li>• <code>highByte()</code></li> <li>• <code>bitRead()</code></li> <li>• <code>bitWrite()</code></li> <li>• <code>bitSet()</code></li> <li>• <code>bitClear()</code></li> <li>• <code>bit()</code></li> </ul> <p><b>Interruptions Externes</b></p> <ul style="list-style-type: none"> <li>• <code>attachInterrupt(interrupti on, fonction, mode)</code></li> <li>• <code>detachInterrupt(interrupt ion)</code></li> </ul> <p><b>Interruptions</b></p> <ul style="list-style-type: none"> <li>• <code>interrupts()</code></li> <li>• <code>noInterrupts()</code></li> </ul>
--	--	--

## Variables et constantes

<p><b>Types des données</b></p> <p><i>Les variables peuvent être de type variés qui sont décrits cidessous.</i></p> <p><b>Synthèse des types de données Arduino</b></p> <ul style="list-style-type: none"> <li>• <code>boolean</code></li> <li>• <code>char</code></li> <li>• <code>byte</code></li> <li>• <code>int</code></li> <li>• <code>unsigned int</code></li> <li>• <code>long</code></li> <li>• <code>unsigned long</code></li> <li>• <code>float</code> (nombres à virgules)</li> <li>• <code>double</code> (nombres à virgules)</li> <li>• Les chaînes de caractères</li> <li>• objet <code>String</code> <b>NEW</b></li> <li>• Les tableaux de variables</li> <li>• le mot-clé <code>void</code> (fonctions)</li> <li>• <code>word</code></li> <li>• <code>PROGMEM</code></li> </ul>	<p><b>Conversion des types de données</b></p> <ul style="list-style-type: none"> <li>• <code>char()</code></li> <li>• <code>byte()</code></li> <li>• <code>int()</code></li> <li>• <code>long()</code></li> <li>• <code>float()</code></li> <li>• <code>word()</code></li> </ul> <p><b>Portée des variables et qualificateurs</b></p> <ul style="list-style-type: none"> <li>• <b>Portée des variables</b></li> <li>• <code>static</code></li> <li>• <code>volatile</code></li> <li>• <code>const</code></li> </ul> <p><b>Utilitaires</b></p> <ul style="list-style-type: none"> <li>• <code>sizeof()</code> (opérateur <code>sizeof</code>)</li> </ul> <p><b>Référence</b></p> <ul style="list-style-type: none"> <li>• Code ASCII</li> </ul>
--	--

## **II.7 Conclusion**

Dans le présent chapitre, nous avons vu un ensemble de cartes à microcontrôleur pouvant être appliqué dans les systèmes embarqués. A travers ces cartes, nous avons opté pour le choix d'utiliser la carte Arduino UNO et ce, pour un fonctionnement approprié non rencontré avec un système à base de composants électroniques uniquement.

Par conséquent, le chapitre suivant sera consacré à l'étude et la conception du système proposée dans ce travail pour le suivi d'une ligne bien déterminée. Le système à carte Arduino sera comparé à celui non concerné par la carte à travers différentes circuiteries afin de monter la nécessité d'utilisation des circuits programmés.

## ***Chapitre III***

Etude et conception du système de  
commande d'éclairage public

### III.1 Introduction

Dans ce chapitre, nous allons développer les étapes nécessaires pour aboutir à notre objectif et faciliter la réalisation de notre projet, nous allons étudier deux cas de circuit (circuit à basse de carte Arduino et sans carte Arduino), et on compare avec ces deux types.

Le but de notre projet est de contrôler automatiquement l'éclairage des lampes à partir des capteurs : lumière, présence, bruit et humidité (brouillard).

#### Principe générale

Le principe de fonctionnement de notre projet est la commande de l'allumage automatique des lampes à partir des capteurs de : lumière, présence, bruit, humidité (brume).

✓ **Allumage à 0%** : les lampes ne s'allument pas parce que l'éclairage d'environnement est fort (l'éclairage de soleil).

✓ **Allumage à 50%** : Les lampes partiellement allumées, en raison d'un mauvais éclairage ambiant (début du coucher de soleil ou brume, nuages ...).

✓ **Allumage à 100%** : Les lampes fortement allumées, si le bruit est capté ou infrarouge détecte un passage (Voitures, Vélos, Camions, piétons ...).

Pour cela, nous étudierons et réaliserons des circuits en fonction de quatre détecteurs :

- Détecteur de lumière : pour connue le pourcentage de lumières de l'environnement.
- Détecteur d'humidité : pour connue pourcentage d'humidité.
- Détecteur de présence : Pour détecter la circulation des piétons ou des véhicules.
- Détecteur de son : pour détecter le bruit des véhicules.

#### Circuits étudiés du projet

Dans notre travail, deux types de circuits sont considérés : sans et avec carte Arduino. Ce qui nous permettra de comparer les différents composants utilisés, et faire ressortir les avantages et les inconvénients entre les deux circuits étudiés.

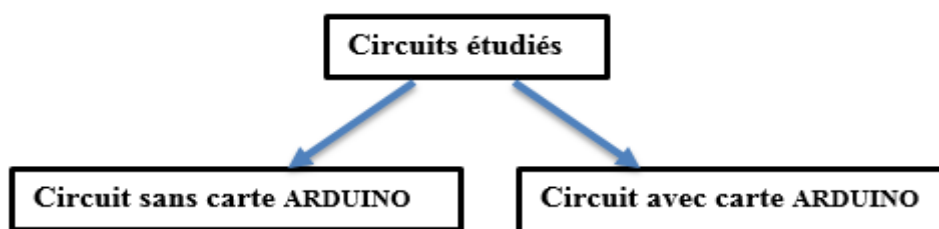


Figure III. 1 Schéma représente les types des circuits étudiés

## III.2 Circuit sans carte Arduino

Il s'agit ici d'étudier les différents circuits électronique sans carte Arduino qui entrent dans la conception du système d'éclairage public.

### III.2.1 Schéma synoptique

Le schéma suivant illustre le fonctionnement synoptique du dispositif général, dans le cas où la carte Arduino n'est pas considérée.

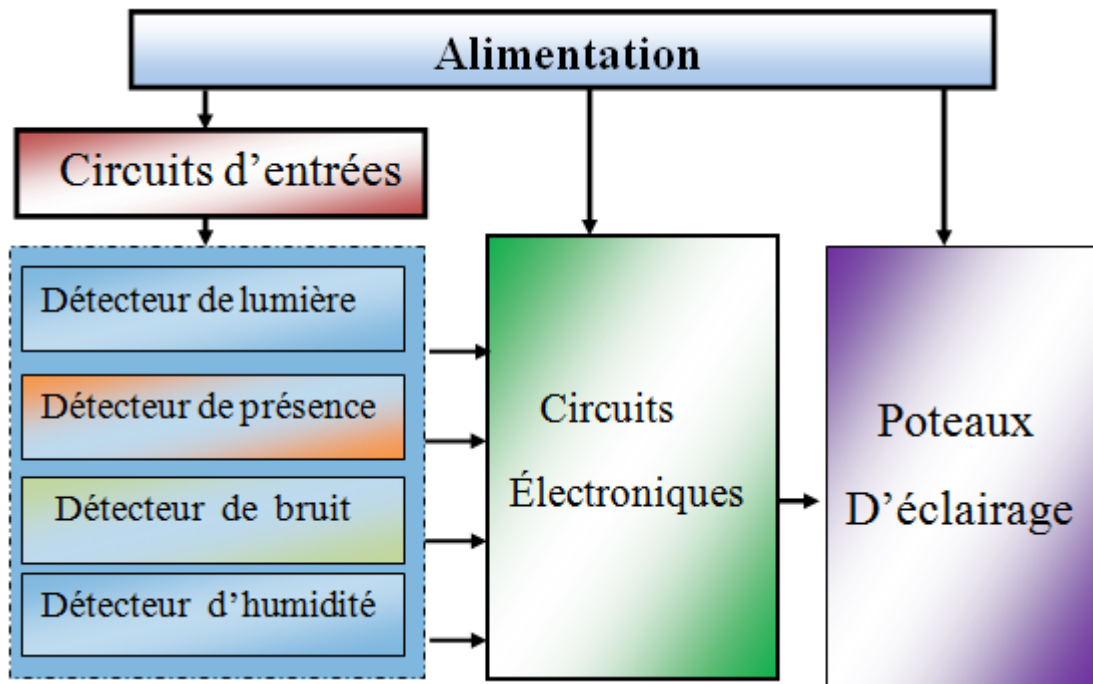


Figure III. 2 Schéma synoptique du montage Sans carte Arduino

### III.2.2 Circuit de détections

#### III.2.2.1 Détecteur de bruit

##### a) Le bruit de véhicule

A la source du bruit routier se trouvent bien évidemment les véhicules mais aussi la route lorsqu'un véhicule l'emprunte. Ces sources peuvent être localisées plus précisément au niveau du moteur du véhicule et du point de contact entre les pneus et la chaussée [15].

- **Le bruit du pneu de véhicule** : Le bruit du pneu lors du déplacement de véhicule trouve son origine principalement dans les vibrations.

Celles-ci varient en intensité en fonction des paramètres suivants :

- La texture du revêtement routier ;
- La vitesse de rotation de la roue ;



- Le profil et la géométrie du pneu ;
- La rigidité du pneu ;

Les vibrations produisent un bruit d'une fréquence comprise entre 500 et 1000 Hz, gamme perceptible par notre oreille. Il existe une autre source de bruit issue de l'échappement de l'air comprimé par le pneu lors du déplacement du véhicule.

- **Le bruit global d'un véhicule en déplacement :**

Le schéma ci-dessous illustre le bruit global produit par une voiture ou un camion circulant sur un revêtement mélange asphaltique dense en fonction de sa vitesse de déplacement. La vitesse de déplacement de la voiture est indiquée sur l'axe horizontal et le niveau de pression sonore (mesuré à 7,5 m du véhicule) figure sur l'axe vertical. Ainsi, une voiture passant sur un revêtement asphaltique dense à une vitesse de 50 km/h produit en moyenne un niveau sonore de 68 dB(A). Dans les mêmes conditions, un poids lourd produit un niveau sonore de 80 dB(A) [16].

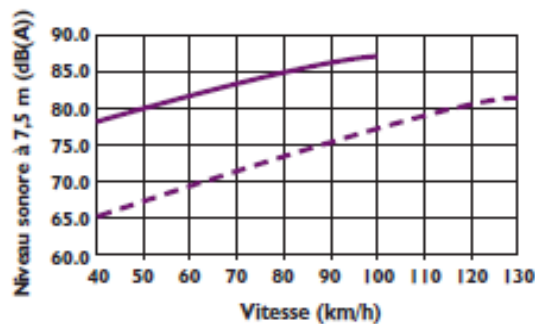


Figure III. 3 Diagramme de bruit de véhicule

**b) Schéma synoptique de circuit de détecteur de bruit**

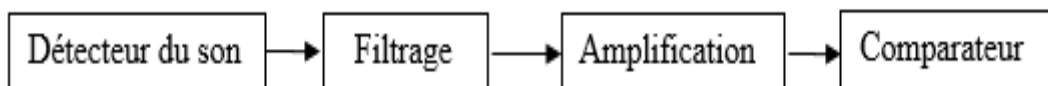


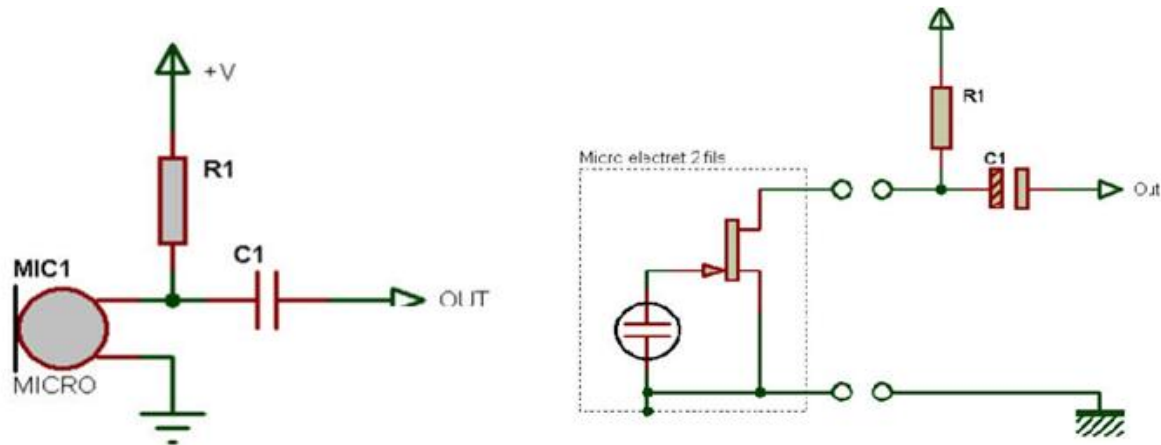
Figure III. 4 Schéma synoptique de détecteur de bruit

✓ **Détecteur du son**

Le microphone qui nous utilise dans ce projet le microphone ELECTRET. Ce dernier est un capteur sonore qui convertit la quantité physique en un signal électrique, caractérise par :

- Voltage d'alimentation est comprise entre 3 Vdc et 10 Vdc.

- Le maximum courant d'entrée est 0.5 mA.
- La sensibilité est 1 V/Pa pour la fréquence de 1 kHz.
- La gamme de fréquences est comprise entre 20 Hz et 20 kHz.
- La zone de détection est omnidirectionnelle.
- Pression acoustique maximale est 110 dB.



**Figure III. 5** Le microphone ELECRET

Un des deux fils du micro est relié à la masse (c'est celui qui est relié au boîtier métallique, fil du bas sur le schéma et sur la photo) et l'autre fil sert en même temps pour l'alimentation du microphone et la sortie. La résistance R1 ( $1\text{K}\Omega < R1 < 47\text{K}\Omega$ ) permet de polariser le transistor FET inclus dans le microphone, alors que le condensateur C1 ( $100\text{nF} < C1 < 10\mu\text{F}$ ) permet de bloquer la tension continue fournie par R1, et de ne laisser passer que le signal audio, qui est alternatif [11].

#### ✓ Filtrage

Le but de l'utilisation du microphone dans notre projet est de détecter le bruit des automobiles, donc le filtrage doit être utilisé pour éliminer les signaux à basse fréquence et laisser passer les signaux à haute fréquence, pour cette raison nous avons utilisé un circuit électronique de filtre passe-haut.

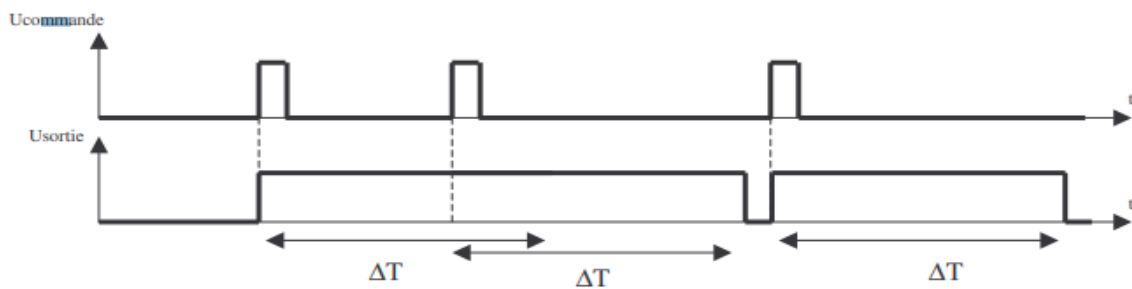
#### ✓ Amplification

L'amplitude de signal qui capté par le microphone est faible (en millivolt), il faut donc amplifier le signal par un circuit électronique à base de l'amplificateur opérationnel.

✓ **Comparateur**

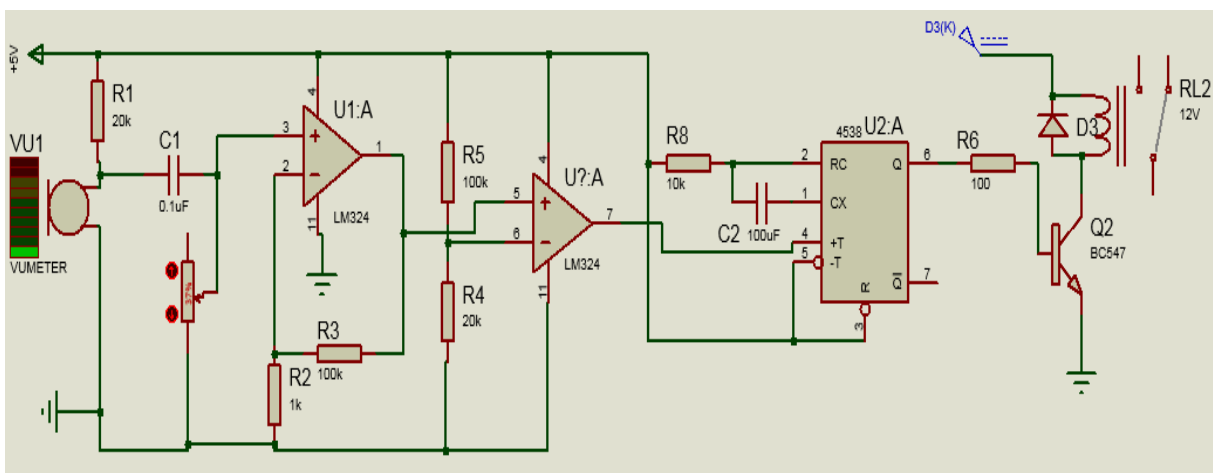
Après l'amplification pour stabiliser la tension de 5V ou 0V, il est nécessaire d'utiliser un comparateur basé sur l'amplificateur opérationnel. Le signal qu'est amplifié est comparé par l'amplitude de référence, si l'amplitude de signal est supérieure à l'amplitude de référence, la sortie est  $V_{cc}$  (5V) sinon la sortie est  $V_{ss}$  (0V). Dans le circuit il n'y a pas de rebouclage de l'entrée + sur la sortie, donc c'est un comparateur à un seuil. La tension qui varie est connectée sur l'entrée +, donc c'est un comparateur non-inverseur.

Ensuite nous avons utilisé un monostable redéclenchable. Lorsque la broche de commande du monostable est activée (par un niveau, un front, une impulsion) la sortie du monostable passe à l'état travail pendant une durée déterminée. Cette durée dépend des composants de la structure électronique et peut également être programmée dans le cas de structure à circuit programmable [14].



**Figure III. 6** Chronogramme du circuit Monostable redéclenchable

c) **Schéma électronique**



**Figure III. 7** Le circuit électronique de détecteur de bruit

### d) Le principe de fonctionnement

1) Premièrement le micro ELECTRET, un fil est utilisé pour l'alimentation (5V) et la sortie du microphone en même temps, le courant d'entrée il devrait être inférieur à 0.5 mA donc on utilise une résistance R1 de 20 K $\Omega$  alors la courant d'entrée est :

$$U = R * I \text{ donc } I = U/R=0.25 \text{ mA}$$

Et l'autre fil est relié à la masse.

2) Ensuite, le signal qui sort du microphone est filtré par un filtre passe-haut

$$G = V_s/V_e = ZR/(ZR + ZC) = R/(R + 1/(jWC)) = jRWC/(1 + jRWC)$$

Alors ce filtre c'est un filtre passe haut d'ordre 1

$$\text{On a } W0 = 2\pi fc \text{ et } 1/(RC) = 2\pi fc$$

$$\text{Donc } W0 = 1/(RC)$$

Calcul de la fréquence de coupure

$$W0 = 2\pi fc, \text{ donc } fc = W0/(2\pi) = 1/(2\pi RC)$$

Dans la simulation la résistance varie de 1% à 100% c'est-à-dire 500  $\Omega$  à 50 k $\Omega$ , alors la fréquence varie de 3.183 kHz à 31.83 Hz

3) Après le filtrage vient une étape d'amplification pour augmenter la tension du signal par l'amplificateur opérationnel non-inverseur, nous utilisons l'amplificateur LM324 qui caractérise par :

TYPE	DESCRIPTION
Nombre de circuits	4
Produit gain-bande passante	1,2MHz
Courant - Polarisation d'entrée	20nA
Tension - Décalage d'entrée	3mV
Courant – Alimentation	1,4mA
Courant - Sortie	40mA
Tension - Alimentation, simple/double ( $\pm$ )	3V ~ 32V, $\pm$ 1,5V ~ 16V

**Tableau III. 1** Caractéristiques de LM324

### Les calculs

On a  $V^+ = V^-$  et  $V^+ = V_e$ , donc  $V^- = V_e \dots (1)$

La loi d'ohm donne  $0 - V^- = R_2 * I \dots (2)$  et  $V^- - V_s = R_3 * I \dots (3)$

(1) et (2) donne  $I = -V_e/R_2 \dots (4)$

(1) et (3) donne  $V_s = V_e - R_3 * I$

Finalement à (3) et (4)  $V_s = (1 + R_3/R_2) * V_e$

Donc le gain de ce cas est  $V_s/V_e = 1 + R_3/R_2 = 101$

4) On passe à la partie de comparateur, La comparaison entre la tension d'entrée et la tension de référence.

Si :  $V_{entree}(V^+) > V_{ref}$  alors  $V_s = 5V$

$V_{entree}(V^+) < V_{ref}$  alors  $V_s = 0V$

Après la comparaison ; on pose qui la lampe est brancher dans la sortie de monostable TC4538BP, si trouve une impulsion (dans ce cas 5V) la lampe est allumée certain temps prédéfini et éteint sinon n'allume pas. Le monostable TC4538BP est caractérisé par :

Catégories	Circuits intégrés Logique – Multivibrateurs
Type logique	Monostable
Circuits indépendants	2
Courant - Sortie, haut/bas	9mA, 15mA
Tension – Alimentation	3V ~ 18V
La résistance externe Rx	$5 K\Omega < R_x < 1000 K\Omega$
Le condensatur Externe Cx	Pas de limites

**Tableau III. 2** Caractéristiques de monostable TC4538BP

Nous avons vu que la bobine du relais nécessite une alimentation par courant continu. Certains relais ont besoin 12V ou plus.

Dans cet exemple, nous allons donc employer d'autres composants pour créer un étage de puissance intermédiaire. Un transistor BC547 fera l'affaire, qu'a caractérisé par :

TYPE	DESCRIPTION
Catégories	Produits semi-conducteurs discrets Transistors
Type de transistor	NPN
Courant - Collecteur (Ic) (max.)	100mA
Tension - Claquage collecteur-émetteur (max.)	45V
Saturation Vce (max.) à Ib, Ic	600mV à 5mA, 100mA
Gain de courant CC (hFE) (min.) à Ic, Vce	200 à 2mA, 5V
Puissance - Max.	500mW

**Tableau III. 3** *Caractéristiques de transistor BC547*

Le collecteur transistor NPN est connectée par la bobine de relie et l'autre extrémité de la bobine est connectée par la source de tension (12v) ; la base de transistor branché par sortie de monostable et l'émetteur branché par la masse. Un autre composant est donc à ajouter dans le circuit à savoir une diode dite "de roue libre" à monter en parallèle sur la bobine de commande du relais, mais dans le sens non passant pour le courant alimentant normalement la bobine. Cette diode dissipera le courant inverse de la bobine au moment de la coupure du courant d'alimentation.

Le transistor fonctionne comme interrupteur ; lorsqu'il n'y a pas de courant dans la base alors il est dans un état de repos (bloquer), et le courant ne circule pas dans la bobine de relie, par contre lorsqu'il y a un courant donc il est dans le mode passant et le circule dans la bobine (champ électromagnétique) et attire le bras de relie, dans le tableau suivant nous avons présenté les caractéristiques de relie.

Catégories	Relais de puissance
Série	SRD
Tension de bobine	12 V CC
Capacité des contacts (courant)	10A
Tension de commutation	250 V CA, 30 V CC – maxi
Tension de relâchement obligatoire	0,6 V CC
Durée de fonctionnement	10ms
Temps de relâchement	5ms
Résistance de bobine	400 Ohms

Tableau III. 4 Les caractéristiques de relaie SRD-12VDC

III.2.2.2 Détecteur de lumière

a) Schema synpotique

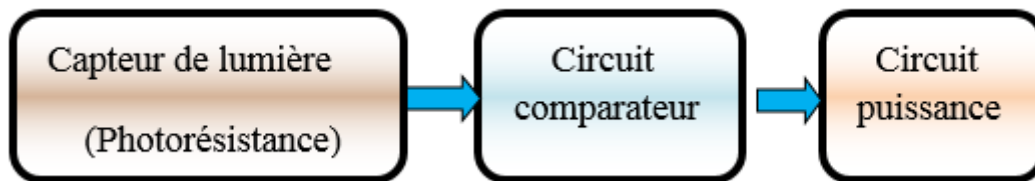


Figure III. 8 Schéma synoptique de circuit détecteur de lumière

✓ Le capteur de lumière (LDR)

Une photorésistance est un composant électronique dont la résistivité varie en Fonction de la quantité de lumière incidente. On peut également la nommer résistance photo dépendante ou cellule photoconductrice.



Figure III. 9 Capteur LDR photorésistance

Une photorésistance est composée d'un semi-conducteur à haute résistivité. Si la lumière incidente est de fréquence suffisamment élevée, les photons absorbés par le semi-conducteur donneront aux électrons liés assez d'énergie pour sauter dans la bande de conduction, les électrons libres ( avec leurs trous d'électron ) ainsi produits abaissant la résistance de l'ensemble incidente est de fréquence suffisamment élevée, les photons absorbés par le semi-conducteur.

### ✓ Circuit comparateur

Le circuit comparateur permet de comparer la tension entre une borne de la photorésistance (qui varie en fonction de la lumière incidente) et de la tension de référence spécifiée.

### ✓ Circuit puissance

Le circuit de puissance se compose d'un relais, qui est ouvert ou fermé en fonction de la quantité de la lumière incidente sur la photorésistance.

#### b) Schéma électronique

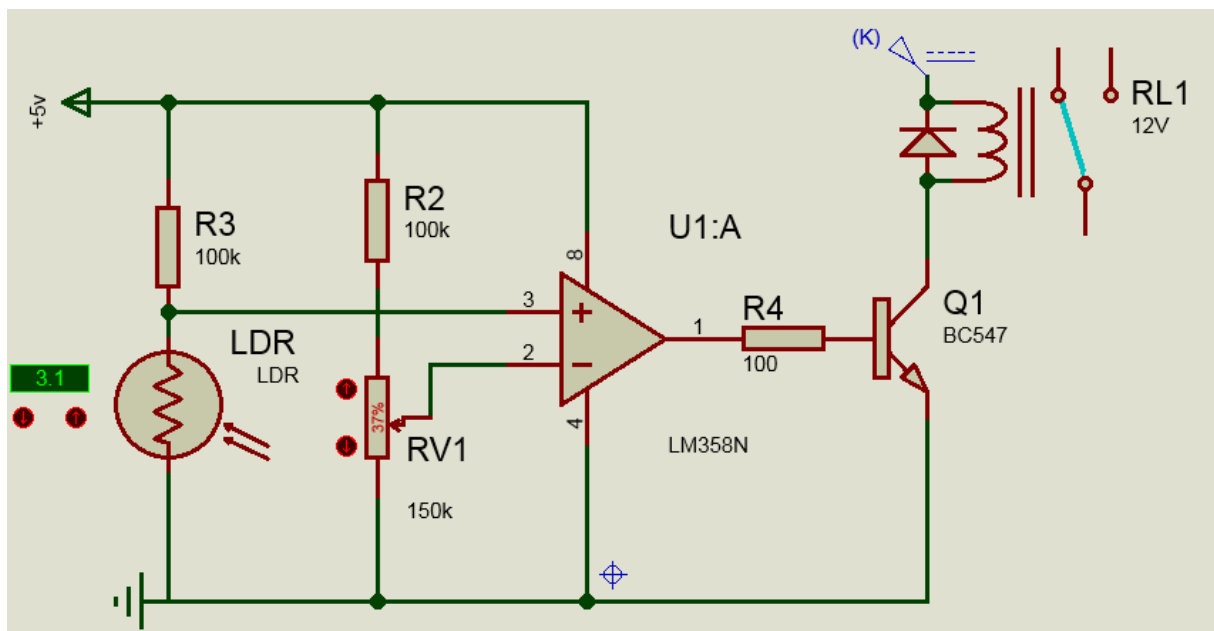


Figure III. 10 Circuit électronique de détecteur de lumière



### c) Principe de fonctionnement

Nous avons utilisé un pont de diviseur de tension constitué d'une photorésistance et d'une résistance R3 pour déterminer la tension  $V^+$  du comparateur opérationnel, et un autre pont constitué d'un potentiomètre RV1 et d'une résistance R2 pour déterminer la tension de référence  $V^-$ , si la tension  $V^+$  supérieur à tension  $V^-$  l'amplificateur opérationnel alimente la base du transistor BC547 à travers la résistance R4, le transistor ouvre ou ferme le relais.

### d) Les Calculs

La photorésistance PGM5506 caractérisée par  $V_{max}=100V$  et  $P_{max}=90mW$ . quand il y a de la lumière la résistance de LDR est dans l'intervalle de  $2K\Omega$  à  $6K\Omega$ , sinon la résistance de LDR est  $0.14M\Omega$ .

- **Calcul du courant qui circule dans LDR**

Selon La loi d'ohm on a  $I(LDR)=P_{max}/V_{max}=90.10^{-3}/100=0.9mA$

- **Calcul de R3**

On a  $V_{cc} - R7 * i(LDR) - V(LDR)$  Alors  $R7 = \left| \frac{V_{cc}-V(LDR)}{i(LDR)} \right| = 100K\Omega$

- **Calcul de la tension de  $V^+$**

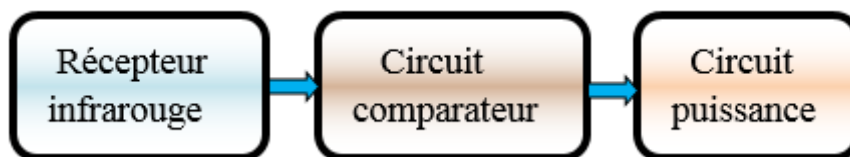
On a  $V^+ = V_{cc} * R(LDR)/(R7 + R(LDR)) = 3V$

Calcul de la valeur de potentiomètre pour tension de référence de 2.5V

$$V^- = \frac{R_v}{R_9 + R_v} * V_{cc} = 2.5V \quad \text{Alors } R_v = 2.5 * \frac{R_2}{(V_{cc} - 2.5)} = 100K\Omega$$

### III.2.2.3 Détecteur de présence

#### a) Schéma synoptique



**Figure III. 11** Schéma synoptique de détecteur de présence

#### ✓ Récepteur infrarouge

Les récepteurs IR (récepteurs infrarouges) sont des dispositifs conçus pour envoyer et recevoir un signal infrarouge d'un appareil à l'autre. Le signal infrarouge est invisible à l'œil

humain. Pour une utilisation correcte, les deux récepteurs doivent se voir physiquement. Idéalement, ils doivent être placés sur une ligne droite mais, de la même manière que les ondes lumineuses, les ondes infrarouges peuvent être réfléchies contre les murs ou le verre.

✓ **Description du récepteur TSOP1738**

Le TSOP1738 appartient à la famille TSOP17xx. Le xx représente la fréquence de modulation, dans le cas du TSOP1738 est 38Khz. IR Émetteur et récepteur est utilisé pour contrôler n'importe quel appareil sans fil. Le récepteur IR, qui détecte les impulsions IR modulées et les convertit en signal électrique.

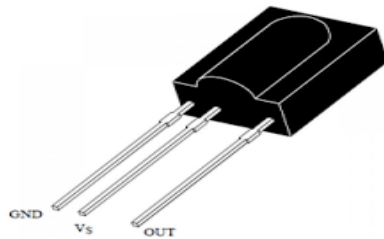


Figure III. 12 TSOP1738

✓ **Circuit comparateur**

Le circuit comparateur permet de comparer la tension entre la borne de la sortie de récepteur infrarouge, qui varie en fonction de présence ou d'absence de rayons infrarouges, et la tension de référence spécifiée.

a) **Schéma électronique**

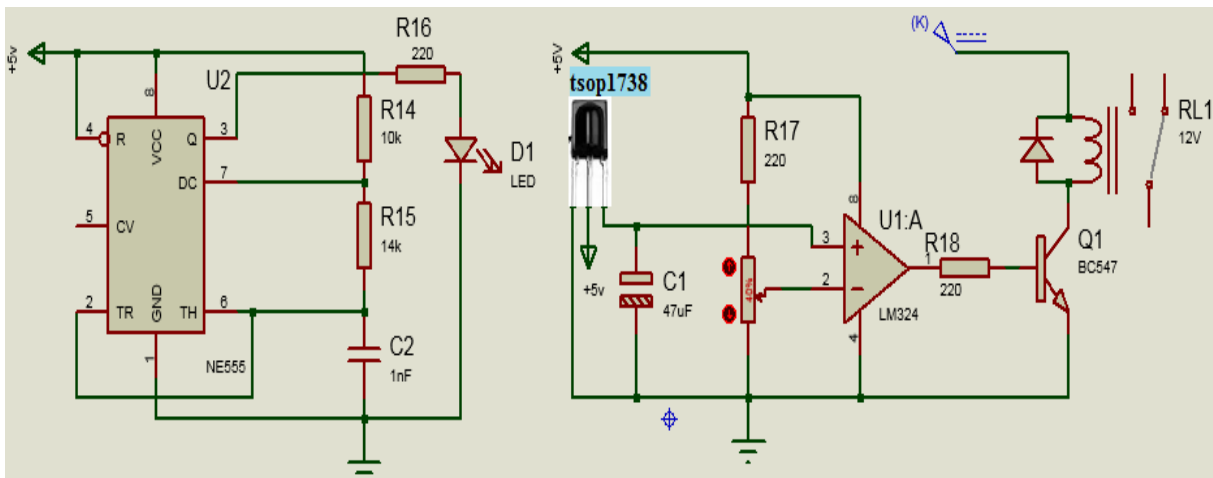


Figure III. 13 Circuit d'électronique du détecteur de présence

## b) Principe de fonctionnement

Ce circuit est divisé en deux parties :

### 1. Partie émettrice (IR)

Cette partie se compose d'un circuit astable à base d'oscillateur NE555 permet de générer un signal de 38KHz pour commander l'émetteur infrarouge.

### 2. Partie réceptrice (IR)

Cette partie se compose de :

- Un pont diviseur de tension, se compose par une résistance R1 (220Ω) et d'un potentiomètre 1 KΩ, permet de fixer la tension de référence.
- Un récepteur infrarouge TSOP1738, le pin de sortie de récepteur est connecté par le  $V^+$  de l'amplificateur, l'amplitude qui réceptionner par TSOP1738 est variable, parce qu'il s'agit de rayons infrarouges tombant sur le récepteur.
- Si les rayons infrarouges existent, l'amplitude de la sortie du récepteur est égal 0.85V, en contraire, s'il n'y a pas les rayons, la sortie de récepteur est 4.85V.
- Si n'est pas l'obstacle, alors Le récepteur, reçoit les rayons infrarouge, Auquel cas l'entrée  $V^+$  sera en état basse ( $V^+=0.85V$ )

$V^+ < V^- \Rightarrow$  la sortie du comparateur est  $-VCC$  (n'existe pas un obstacle).

$V^+ > V^- \Rightarrow$  la sortie du comparateur est  $+VCC$  (il existe un obstacle).

## c) Les calculs

### 1. Partie émettrice (IR)

Calcule des valeurs des résistances et du condensateur pour fréquence de 38KHz :

On a:  $T = 1/f = 1/38.103 = 26.31 \text{ us}$  et  $T = c * \ln(2) * (R14 + 2 * R15)$

On suppose  $R14 = 10 \text{ K}$ ,  $C = 1 \text{ nF}$  et on calcule la valeur de R15 :

$$2 * R2 = (T / (c * \ln(2))) - R1 = (26.31 / (10 - 9 * \ln(2))) - 103 = 27.96 \text{ k} \rightarrow$$

$$R15 = 14 \text{ Kohm}$$

### 2. Partie réceptrice

Calcule de la valeur de potentiomètre pour le potentiel de référence est égal 2.5V

$$V^- = \left( \frac{R_v}{R_{17}} + R_v \right) * V_{cc} = R_v = 220 \text{ ohm}$$

III.2.2.4 Détecteur d'humidité

a) Schéma synoptique

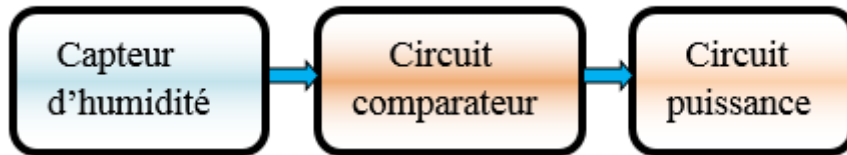


Figure III. 14 Schéma synoptique de détecteur d'humidité

✓ Capteur d'humidité résistive

Un capteur d'humidité résistive permet d'évaluer l'humidité dans l'air à travers d'un simple dipôle se comportant et se connectant comme une résistance.



Figure III. 15 Capteur d'humidité résistive

✓ Les caractéristiques de H25K5

Tension nominale	DC 1 ~10V
Température de fonctionnement	0°C ~ 60°C
Gamme d'humidité	20 ~ 90% RH
Temps de réponse	Max. 60 seconde
Hystérésis	3%RH Max

Tableau III. 5 Caractéristiques de H25K5

✓ **Fonctionnement du capteur d'humidité H25K5**

Pour linéarité la tension de sortie, on utilise une thermistance, on réalise le montage suivant :

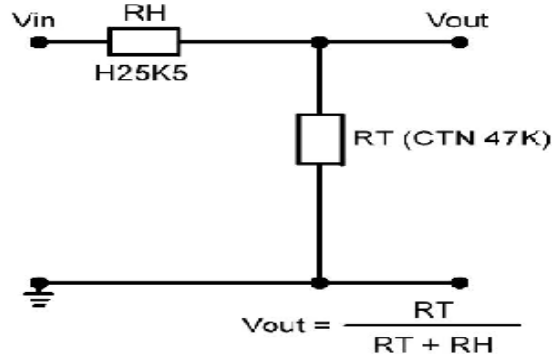


Figure III. 16 Montage de capteur d'humidité

RT = thermistance

RH = capteur d'humidité

$$V_{out} = \frac{RT}{RT+RH}$$

Thermistance CTN : c'est une résistance qui diminue de manière relativement uniforme avec l'augmentation de la température, et vice-versa.

b) **Schéma électronique**

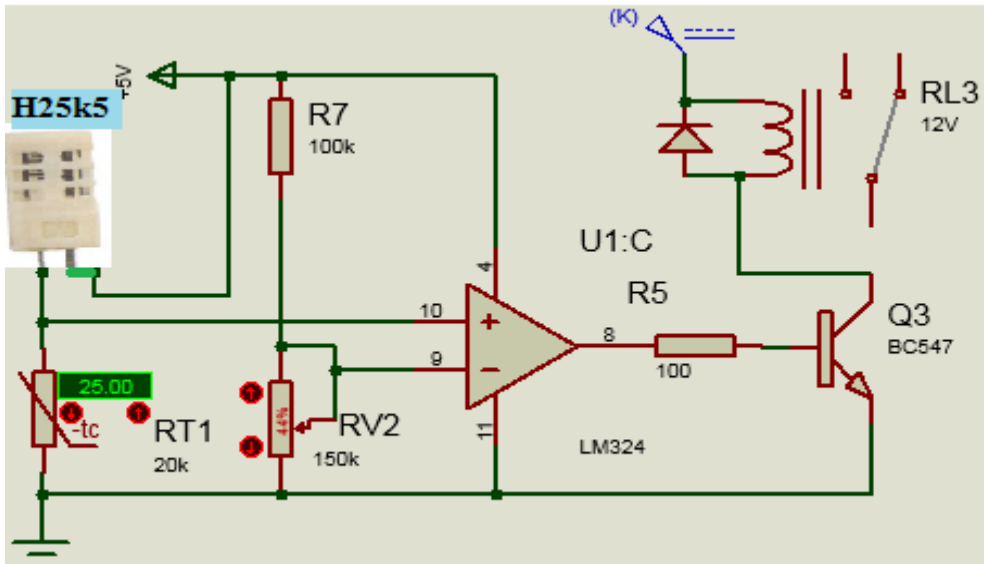


Figure III. 17 Circuit électronique du détecteur d'humidité

### c) Principe de fonctionnement

Nous avons utilisé un pont diviseur de tension composée d'une résistance R7 (100K), et d'un potentiomètre RV2( 150K) pour déterminer la tension de référence et dans l'entrée inverseurs d'amplificateur, et un autre pont constitué d'un capteur d'humidité H25K5 et de thermistance CTN, ils sont utilisés pour déterminer la tension  $V^+$  de l'amplificateur, cette tension est variable, Parce qu'il est basé sur la valeur ohmique de H25K5 et de la thermistance (CTN).

Lorsque l'humidité atteint un certain degré, l'amplificateur est alimenté la base du transistor à travers la résistance R5.

Le transistor fonctionné comme un interrupteur et permet la connexion ou la déconnexion du courant électrique a la bobine du relais.

### III.2.3 Partie de puissance

#### III.2.3.1 Circuit Gradateur de lumière

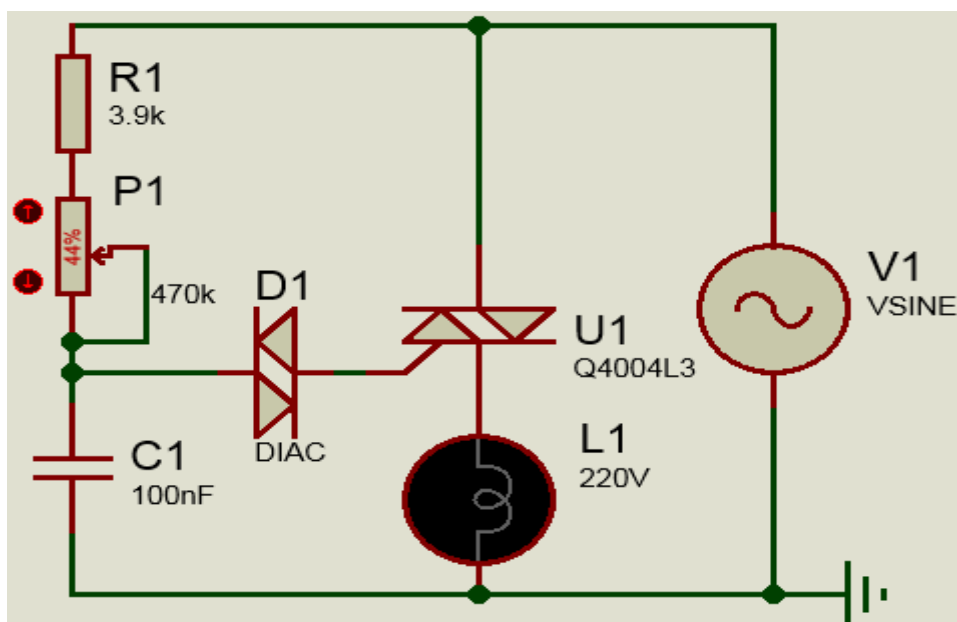


Figure III. 18 Circuit électronique de gradateur de lumière

#### Principe de fonctionnement [19]

Pour commencer, si la tension secteur soit nulle : la gâchette ne reçoit aucun courant, le triac est donc bloqué. Lorsque la tension secteur monte, C1 se charge à travers R1 et le potentiomètre P1. A l'instant où la tension aux bornes de C1 atteint 34V, le diac 32V (c'est le

seul modèle de diac connue !) entre en conduction et la gâchette reçoit un courant qui amorce le triac. C1 se décharge dans la gâchette. La tension aux bornes du diac rendu conducteur s'établit autour de 15V et la tension aux bornes du triac est quasi nulle (1V environ). Pendant le reste de la demi-alternance, le triac reste passant, comme tout thyristor qui se respecte. C1 se décharge alors doucement à travers R1 et P1 puisqu'il y a un quasi court-circuit entre A1 et A2, Lorsque la demi-alternance suivante commence, le triac bloque parce que le courant s'est annulé naturellement lorsque la tension secteur est passée par zéro. Le courant s'annule quand la tension s'annule parce que l'ampoule est résistive et non inductive. C1 se charge maintenant en sens contraire et lorsque la tension à ses bornes atteint -34V, le diac entre en conduction, C1 se décharge dans la gâchette et amorce le triac. Et ainsi de suite... Pour les deux polarités, le comportement est identique, au signe +/- près !

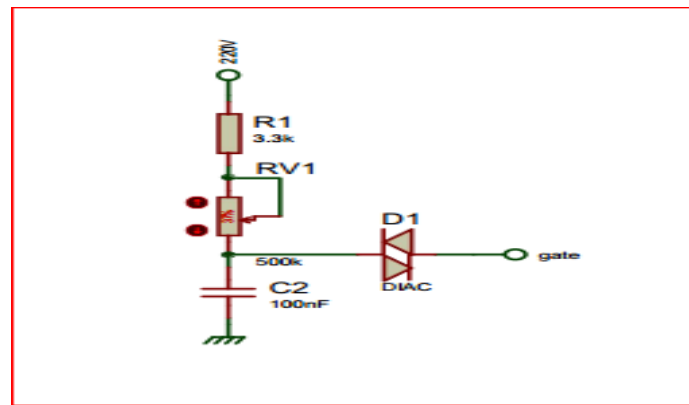


Figure III. 19 Schéma de circuit commande d'un gradateur de lumière

✓ **Choix de composant [19] :**

**R1** : 3.9kOhms/0.25Watt. Valeur minimale pour limiter d'éventuelles pointes de courant sur la gâchette et le diac. On peut choisir 3.3k ou 4.7k sans problème.

**P1** : 470kOhms de préférence. Eventuellement 1MOhm, mais pas moins sinon, même sur la plus grande valeur, l'ampoule brillera toujours un peu.

**C1** : 100nF. La tension à ses bornes ne peut jamais dépasser 35V crête. On peut choisir un modèle 63V ou 100V.

**DIAC** : 32V,2A référence DB3 par exemple. Il n'existe qu'un seul modèle connue.

**TRIAC** : Tension inverse 3V, tension à état bloqué 400V, courant continue direct 50mA.

## III.2.3.2 Circuit puissance globale

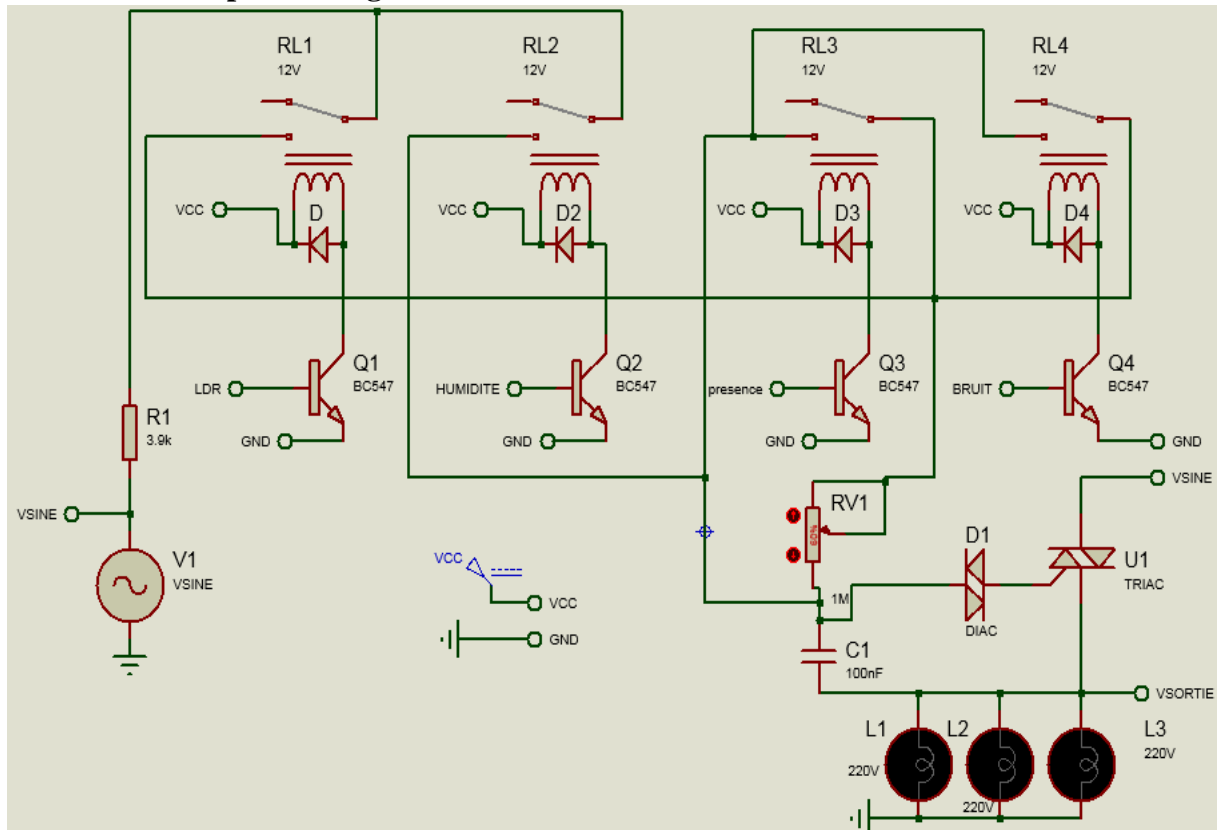


Figure III. 20 Circuit électronique de la partie puissance

✓ **Fonctionnement**

Les quatre détecteurs contrôlent le gradateur au moyen des quatre relais et du potentiomètre, ces relais permettent le réglage du temps de charge et de décharge du condensateur au moyen du potentiomètre afin de contrôler la luminosité des lampes, pour 100% de luminosité le condensateur est chargé directement via la résistance R1, dans ce cas le temps de charge est très petit, le triac est amorcé peu après le début de l'alternance positive ou négative, pour une luminosité de 50% , le condensateur est chargé par la résistance R1 et le potentiomètre P1,dans ce cas le temps de charge dépend de la valeur de le potentiomètre P1 et la résistance R1,ainsi le triac retarde son fonctionnement lors de l'alternance positive et l'alternance négative .



### III.3 Circuit Avec carte Arduino

#### III.3.1 Schéma synoptique

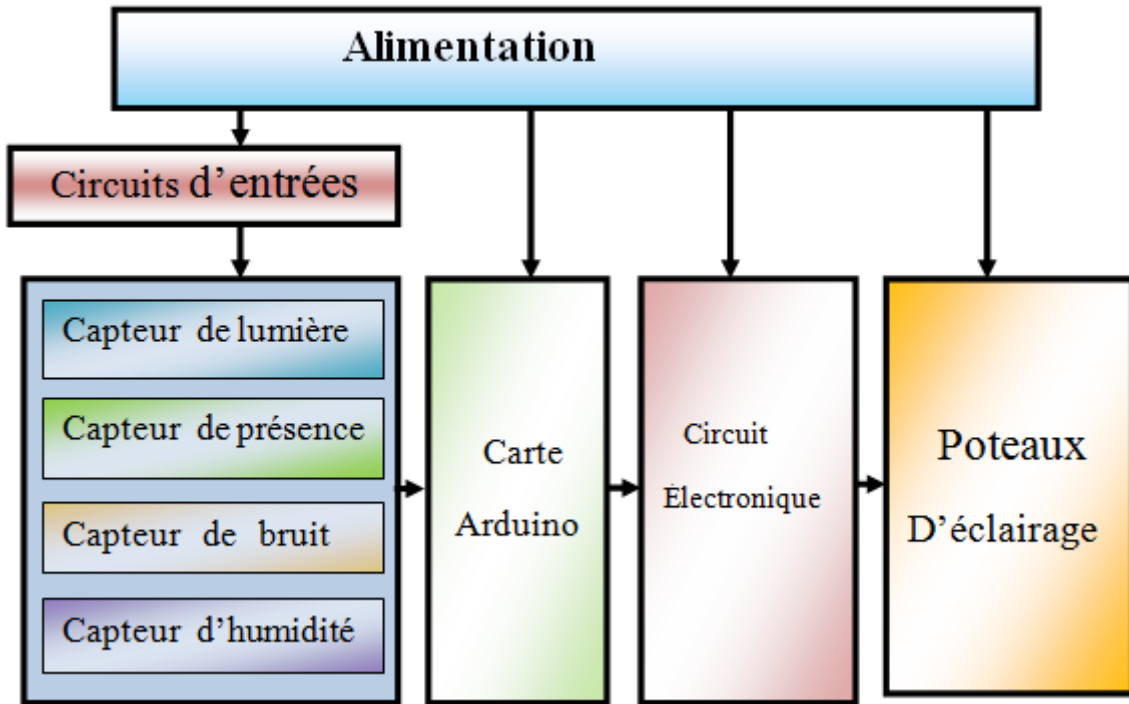


Figure III. 21 Schéma synoptique du montage avec carte Arduino

#### III.3.2 Partie de commande

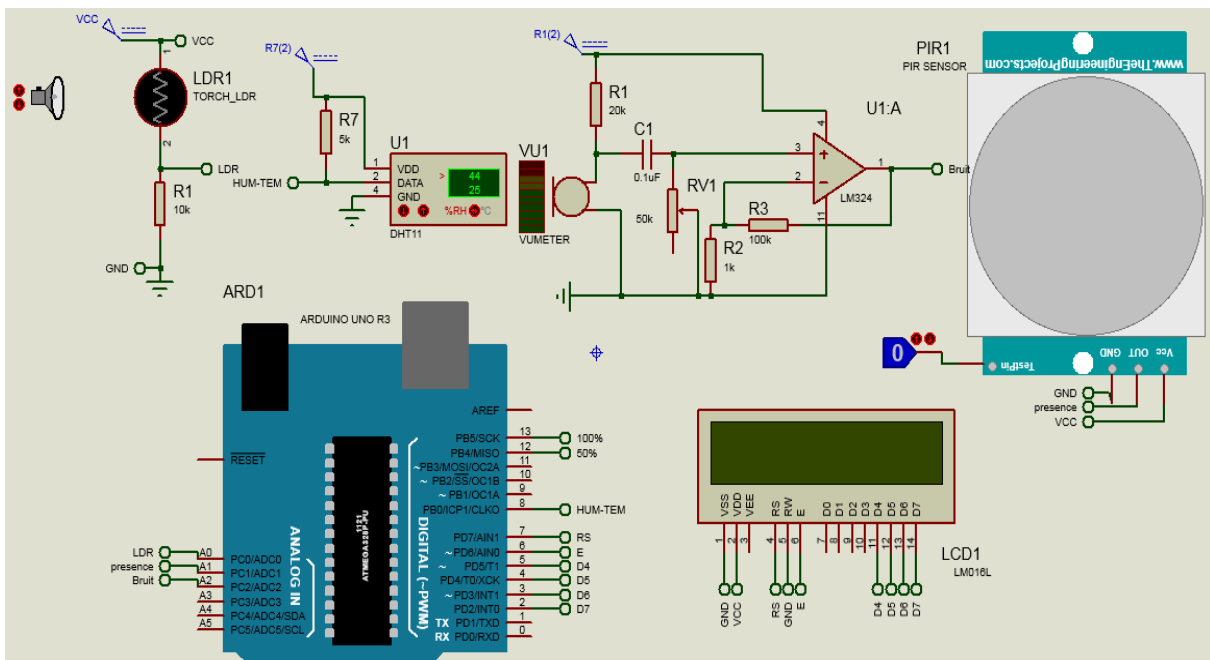


Figure III. 22 Circuit électronique de la partie de commande

III.3.2.1 Capteur de lumière

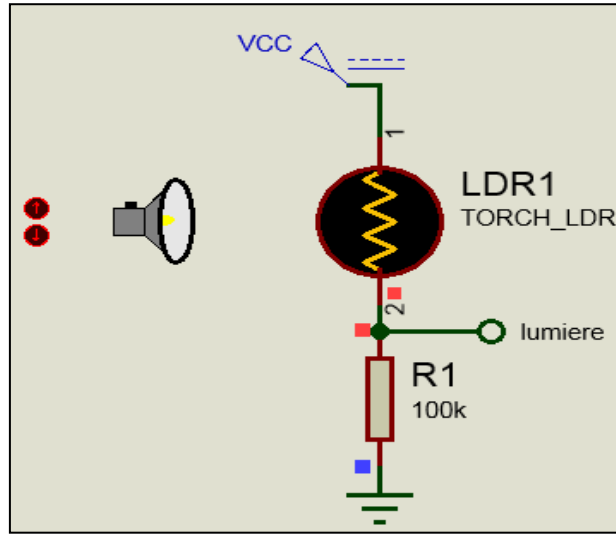


Figure III. 23 Capteur de lumière connecté à la carte Arduino

✓ Fonctionnement

La variation de la valeur ADC dépend de la variation de l'intensité de lumière, la valeur ADC est convertir à pourcentage, la LED activée ou désactivée lorsque dépasser un seuil prédéfini.

III.3.2.2 Capteur de son

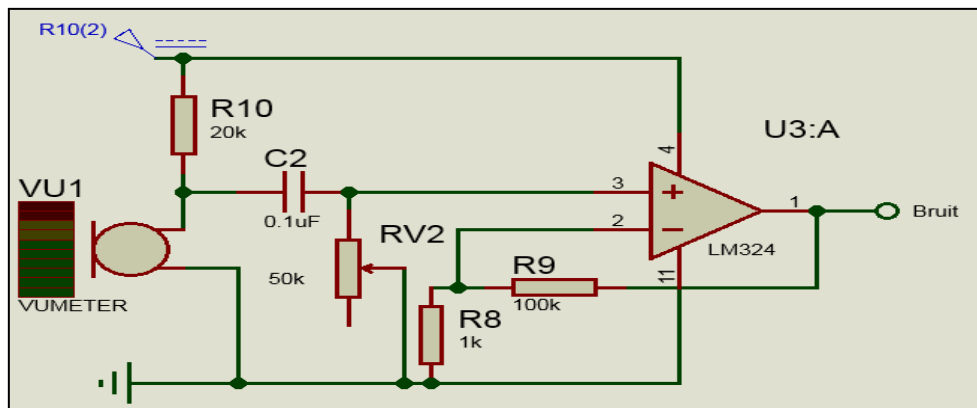


Figure III. 24 Circuit de capteur de son connecté à la carte Arduino

✓ Fonctionnement

Le signal qui capté par le microphone ELECTRET est filtré en suite amplifier par l'amplificateur opérationnel, la sortie de ce dernier brancher dans l'entrée analogique de

l'Arduino UNO, elle convertisse la valeur lue par la broche 8 en fréquence en Hz à l'aide de la bibliothèque freqMeasure.

### III.3.2.3 Capteur d'humidité

#### ✓ Description du DHT11

Le DHT11 est composé de deux parties, un capteur d'humidité capacitif et un capteur de température à base de NTC. Il contient également un circuit électronique élémentaire qui effectue la conversion analogique vers numérique et qui débite un signal numérique proportionnel à la température et l'humidité mesurée par le capteur. La technologie utilisée par le capteur DHT11 garantit une grande fiabilité, une excellente stabilité à long terme et un temps de réponse très rapide.

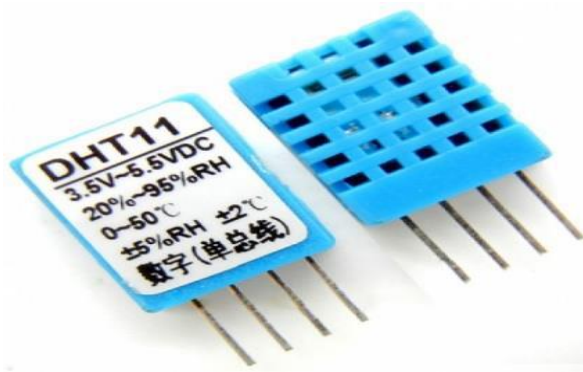


Figure III. 25 Capteur DHT11

#### ✓ Caractéristiques du DHT11

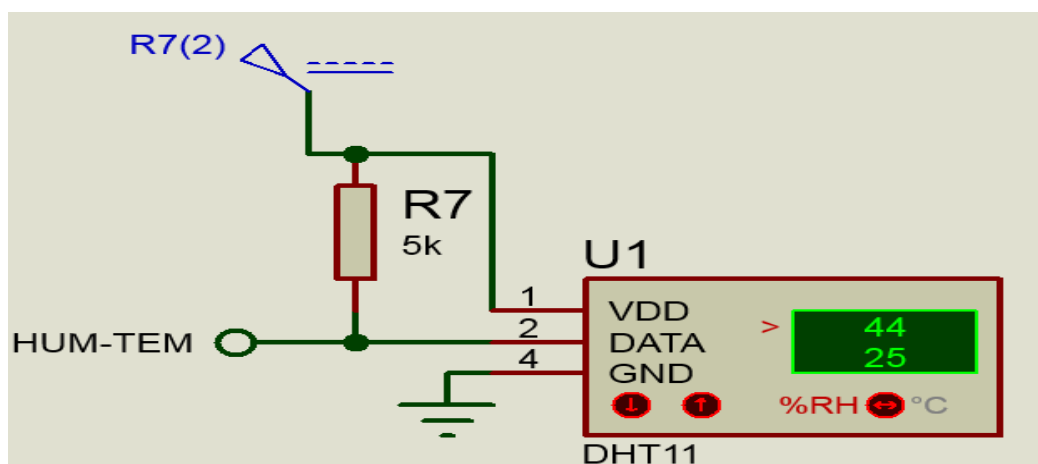
- Alimentation : 3-5.5V DC.
- Consommation : comprise entre 0.5 mA et 2.5 mA.
- Signal de Sortie : Signal Numérique.
- Plage de Mesure : - Humidité : 20-95%RH ; Température : 0-50°C ;
- Précision : Humidité  $\pm 5\%$  RH ; Température  $\pm 2.0^\circ\text{C}$  Résolution : Humidité 1% RH ;  
Température 0.1°C
- Période de mesure : 2s.
- Dimensions : 12x15.5x5.5mm.

✓ **Connectique du DHT11 :**

Le DHT11 est composé de 4 broches espacées de 2,45mm ce qui permet de le brancher facilement avec notre carte de traitement, mais en réalité seulement 3 broches sont utilisées.

Le brochage du capteur DHT11 est 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.
- La broche n°3 n'est pas utilisée et ne doit pas être câblée.
- La broche n°4 est la masse du capteur (GND).



**Figure III. 26** Détecteur d'humidité connecté à la carte Arduino

### III.3.2.4 Détecteur de mouvement PIR

Le capteur infrarouge détecte le mouvement par la radiation naturelle émise par les objets chauds (les personnes, animaux, moteurs de voitures)

#### Caractéristiques [17] :

- Tension d'alimentation : 5V – 20V
- Consommation : 65mA
- Sortie TTL : 3.3V, 0V
- Temps pendant lequel la sortie est Ajustable (3s à 5min)
- Temps de verrouillage  $T_i$  : 0.2 sec à 3s selon fabricant
- Déclenchement : avec redéclenchement, sans redéclenchement
- Sensibilité : moins de 120°, jusque 7 mètres
- Température : – 15 à +70°C

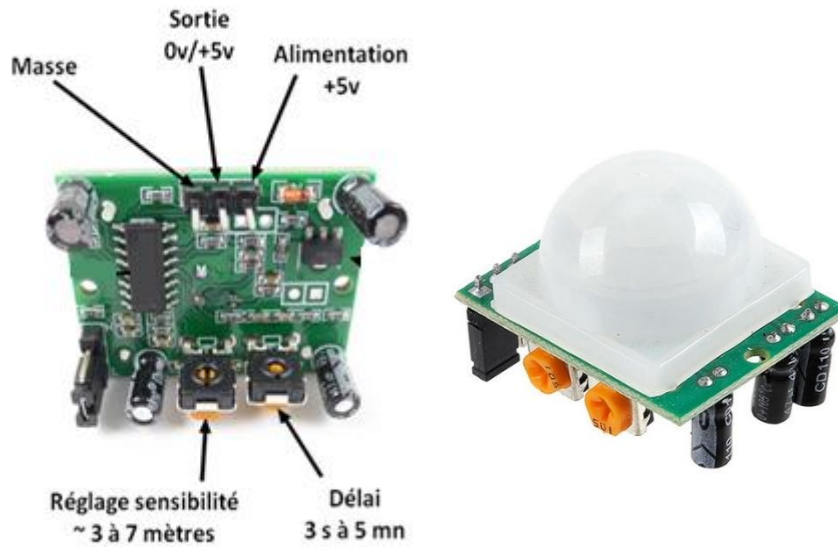


Figure III. 27 Détecteur de présence PIR

Le PIR est un capteur numérique : si un mouvement est détecté, le signal à la sortie du capteur est réglé à un niveau Haut. Sinon, le signal à la sortie du capteur est réglé à un niveau Bas

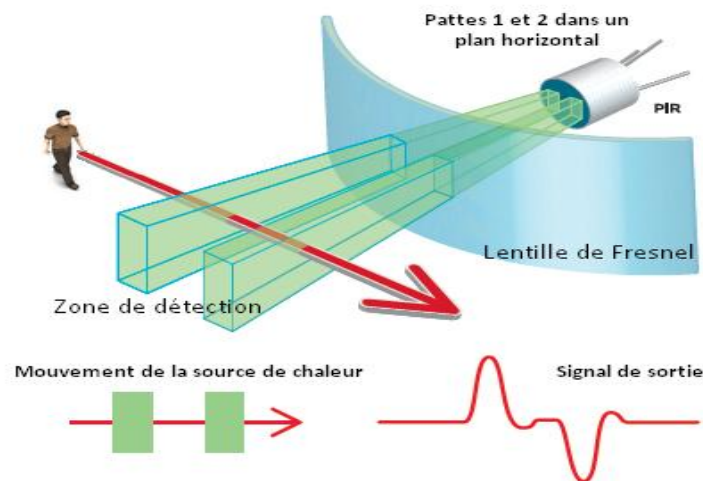


Figure III. 28 Fonctionnement de capteur PIR

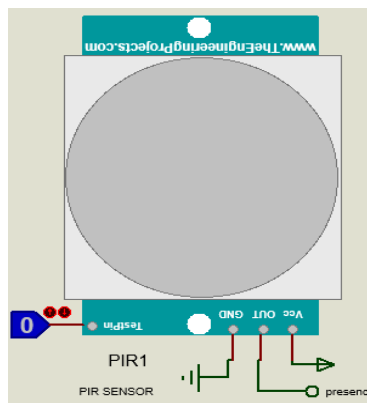


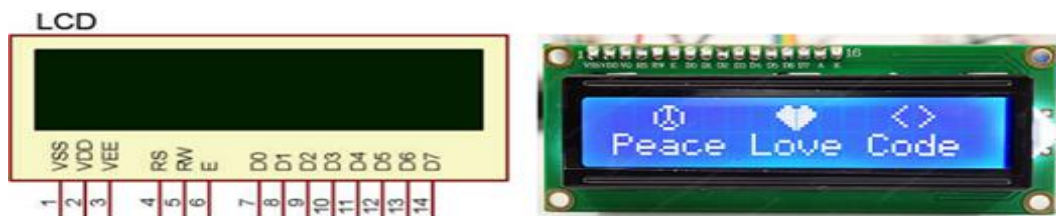
Figure III. 29 Détecteur de présence connecté à la carte Arduino

### III.3.2.5 La partie d'affichage [18]

Les afficheurs à cristaux liquides LCD sont des modules compacts intelligents, et nécessitent peu de composants externes pour un bon fonctionnement. Ils consomment relativement peu de 1 mA à 5 mA, et ils nécessitent une alimentation de 5 V pour pouvoir alimenter leur pilote interne et ainsi permettre l'affichage des caractères sur l'écran.

L'afficheur LCD que nous avons utilisé dans notre réalisation permet d'afficher des caractères ASCII sur 16 colonnes et de 2 lignes.

Ce dernier possède huit bits de données, mais seules quatre sont utilisées. Le composant propose en effet un mode de transmission sur 4 bits en passant les caractères en deux temps. Cela économise quatre lignes de données sur l'Arduino.



**Figure III. 30** Afficheur LCD à logique intégrée 02 lignes x 16 colonnes

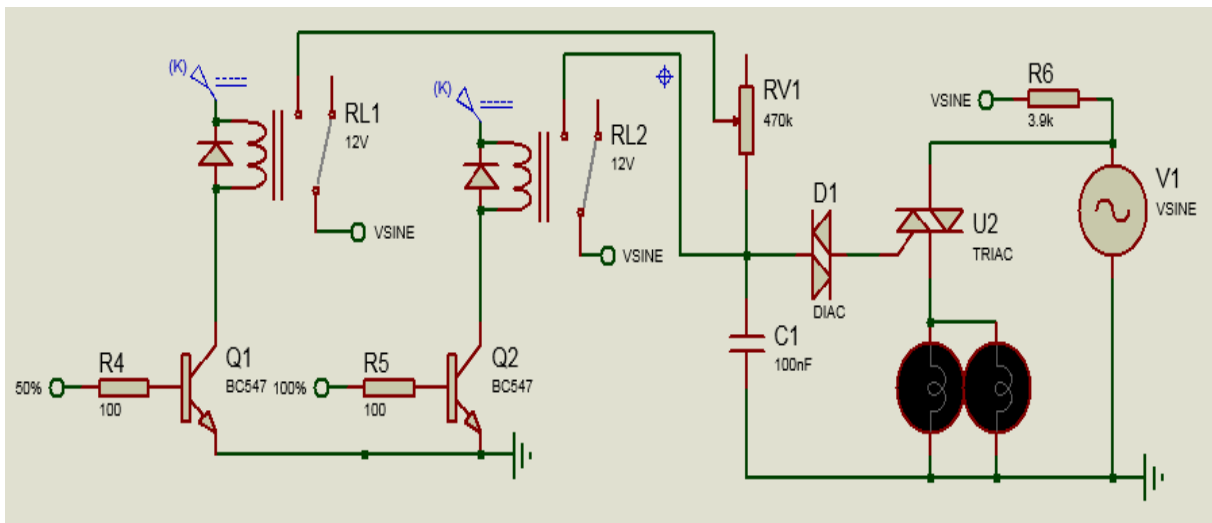
Les différentes broches externes de l'afficheur LCD sont les suivantes :

- ❖ VCC, masse : alimentation de l'afficheur LCD avec 5 V, 0V respectivement.
- ❖ Contraste : entrée permettant de régler le contraste de l'afficheur LCD. Il faut appliquer une tension continue réglable entre 0 V et 5 V à l'aide d'un potentiomètre.
- ❖ VLED : différence de potentiel permettant de commander le rétro éclairage.
- ❖ E (ENABLE) : entrée de validation, elle permet de valider les données sur un front descendant.
- ❖ RS (Register Select) : cette entrée permet d'indiquer à l'afficheur si l'on souhaite réaliser une commande (RS=0) par des instructions spécifiques ou écrire une donnée sur le bus (RS=1).
- ❖ R/W : entrée de lecture (R/W=1) et d'écriture (R/W=0). Lorsqu'on commande l'afficheur LCD, il faut se placer en mode écriture.
- ❖ D7...D0 : bus de données bidirectionnel, il permet de transférer les instructions ou les données à l'afficheur LCD.

Puisque l'afficheur LCD sera commandé par une ARDUINO. Il faut donc penser aux mises en œuvre :

- ❖ Matériel : connexion des broches de l'ARDUINO à l'afficheur LCD.
- ❖ Logiciel : utilisation de sous programmes permettant de commander l'afficheur LCD (initialisation, effacement de l'afficheur, affichage d'un caractère, affichage d'une variable...)

### III.3.3 Partie de puissance



**Figure III. 31** Circuit électronique de la partie de puissance

#### ✓ Fonctionnement

L'Arduino contrôle deux niveaux de luminosité 100% et 50%, le premier niveau via le transistor Q2 et le relais RL2 et le seconde niveau par le transistor Q1 et le relais RL1, les résistances R4 et R5 pour le courant ne pas dépasser le courant de base du transistor (Q1 et Q2) et le courant maximum fourni par la sortie arduino, la résistance R2 pour que le courant ne dépasse pas le courant du collecteur du transistor Q1 et la résistance R3 pour que le courant ne dépasse pas le courant du collecteur du transistor Q2, la partie gradateur a été précédemment expliquée.

### III.4 Etude comparative entre les cas sans et avec Arduino

Comme dans le cas de notre étude nous sommes basés sur deux cas de situations, circuits sans et avec carte Arduino. Le tableau suivant résume les principaux points de la comparaison entre les deux cas de circuits concernés. Nous remarquons que le circuit à

carte Arduino comme il a ses inconvénients, il présente plus d'avantages par rapport au circuit sans carte Arduino.

	<b>Circuit sans carte Arduino</b>	<b>Circuit avec carte Arduino</b>
<b>Coût de fabrication</b>	Moins chère	Plus chère
<b>Les composants utilisés</b>	Utilisation requiert plus de composants	Moins d'utilisation de composants
<b>Maintenance</b>	Difficile (changement des composant)	Facile (correction de code)
<b>Espace réservée</b>	Plus espace	Moins d'espace

**Tableau III. 6** *Etude comparative entre les cas sans et avec Arduino*

### III.5 Conclusion

Dans ce chapitre, nous avons fait l'étude théorique de notre projet avec des calculs, et présenter les composants électroniques que nous avons utilisés pour faciliter la réalisation. Dans le chapitre suivant nous avons fait la partie de simulation ainsi que la partie de réalisation.



## ***Chapitre IV***

Simulations et Réalisations pratiques d'un système de commande d'éclairage publique

## IV.1 Introduction

Dans ce chapitre on va simuler les montages électroniques de commande d'éclairage public avec Arduino et sans carte Arduino, et on va réaliser le premier circuit ainsi présenter les logiciels, matérielles qui exploiter dans notre projet et dans la maquette.

## IV.2 Présenter les logiciels qui exploiter

### IV.2.1 Proteus

Proteus est une suite logicielle destinée à l'électronique. Grâce à ce logiciel, nous pouvons réaliser des schémas électriques et les simuler. Développé par la société Labcenter Electronics, les logiciels incluent dans Proteus, permettent la CAO dans le domaine de l'électronique. Cette suite logicielle est très connue et utilisée dans de nombreuses entreprises et organismes de formation (incluant lycées et universités).

#### A) Présentation de l'outil ISIS

ISIS est principalement connue pour éditer des schémas électriques (assemblage de composants électroniques dont on fixe les valeurs et les références reliés par des connexions électriques (fils)). Par ailleurs, le logiciel permet également de simuler ces schémas, ce qui permet de déceler certaines erreurs dès l'étape de conception. Les circuits électriques conçus grâce à ce logiciel peuvent être utilisés dans des documentations, car le logiciel permet de contrôler la majorité de l'aspect graphique des circuits.

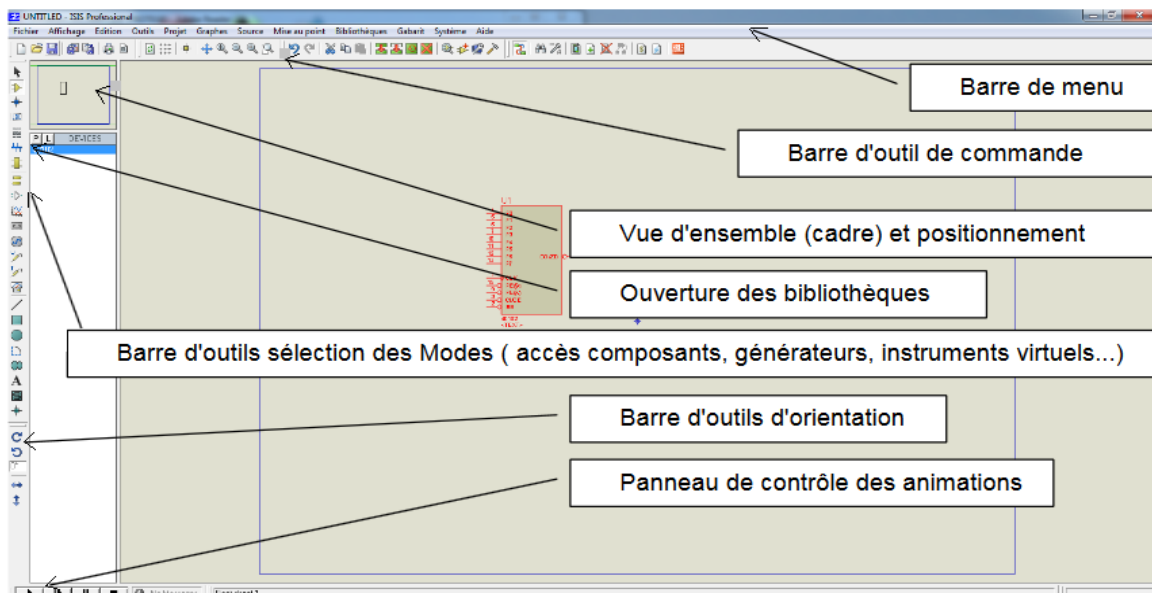


Figure IV. 1 le principe outil de ISIS

## B) Présentation d'ARES

Le logiciel ARES est un outil d'édition et de routage qui complète parfaitement ISIS. Un schéma électrique réalisé sur ISIS peut alors être importé facilement sur ARES pour réaliser le PCB de la carte électronique. Bien que l'édition d'un circuit imprimé soit plus efficace lorsqu'elle est réalisée manuellement par routage, ce logiciel permet de placer automatiquement les composants et de réaliser le routage automatiquement.

### IV.2.2 Programmeur Arduino

Un IDE (environnement de développement) libre et gratuit est distribué sur le site d'Arduino. L'interface de l'IDE Arduino est plutôt simple, il offre une interface minimale et épurée pour développer un programme sur les cartes Arduino.



**Figure IV. 2** Interface de Arduino IDE

- 1- Éditeur de code avec coloration syntaxique
- 2- Barre d'outils rapide
- 3- Barre de menus
- 4- Affichant les résultats de la compilation du code source
- 5- Vérifier le programme
- 6- Compiler et envoyer le programme vers la carte Arduino
- 7- Créer un nouveau fichier
- 8- Charger un programme existant
- 9- Sauvegarder le programme

La structure d'éditeur de code est constituée [1]:

**En tête** : déclaration des variables, des constantes, indication de l'utilisation de Bibliothèques

**Un setup** (initialisation) : cette partie n'est lue qu'une seule fois, elle comprend les fonctions devant être réalisées au démarrage (utilisation des broches en entrées ou en sortie, etc....).

**Une loop** (boucle) : cette partie est lue en boucle ! C'est ici que les fonctions sont réalisées.

On peut ajouter des « sous-programmes » ou « routines » qui peuvent être appelées à tout moment dans la boucle.

### IV.3 Simulation de notre projet sans carte Arduino

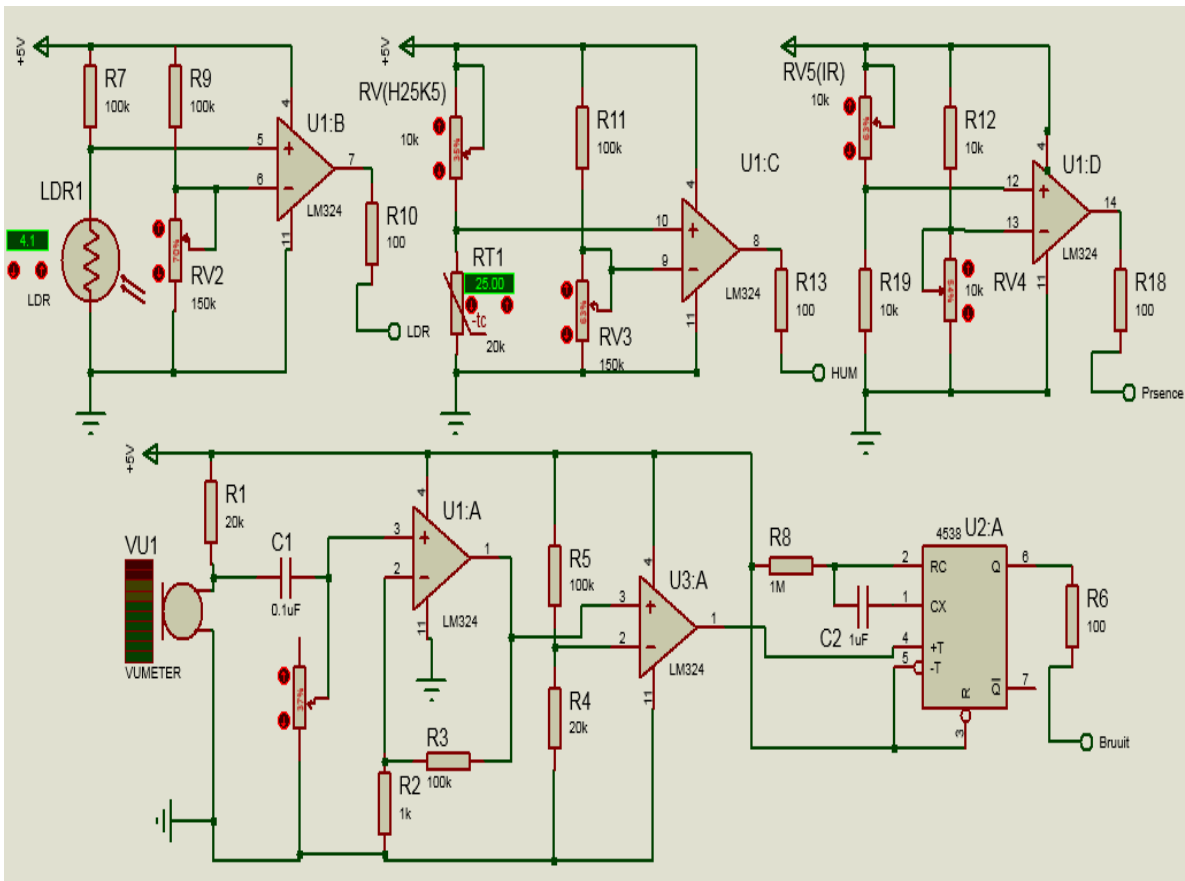


Figure IV. 3 Circuit électronique de la partie commande (sans arduino)

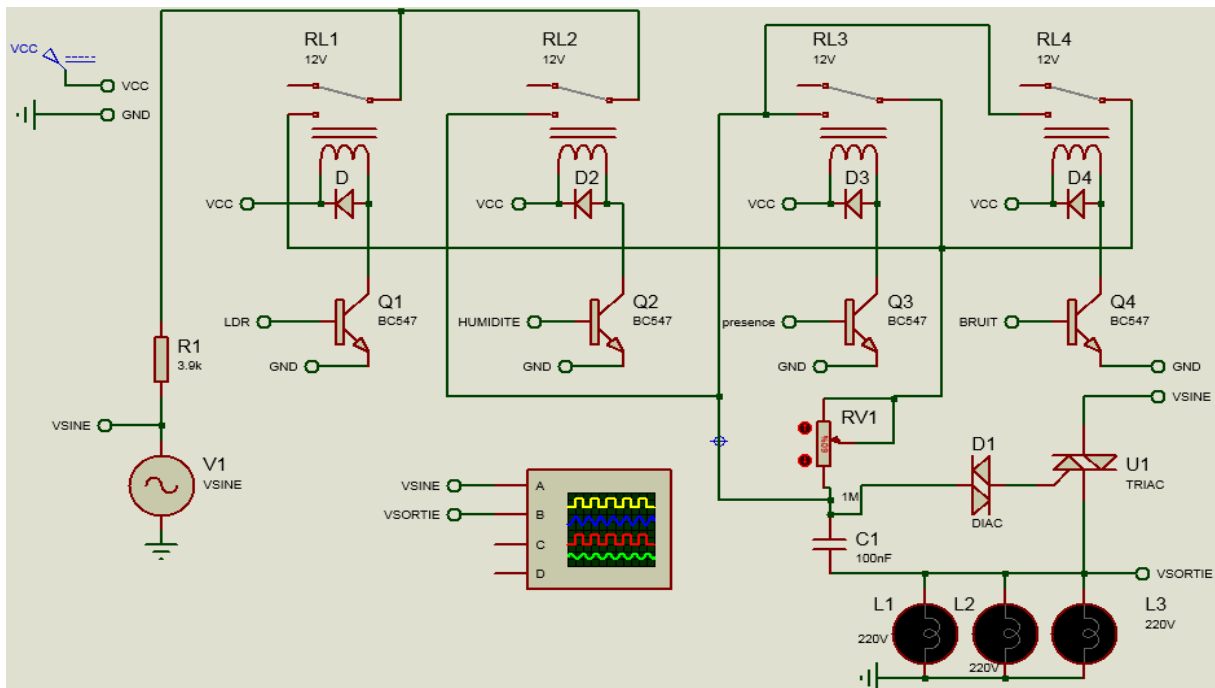


Figure IV. 4 Circuit électronique de la partie puissance (sans arduino)

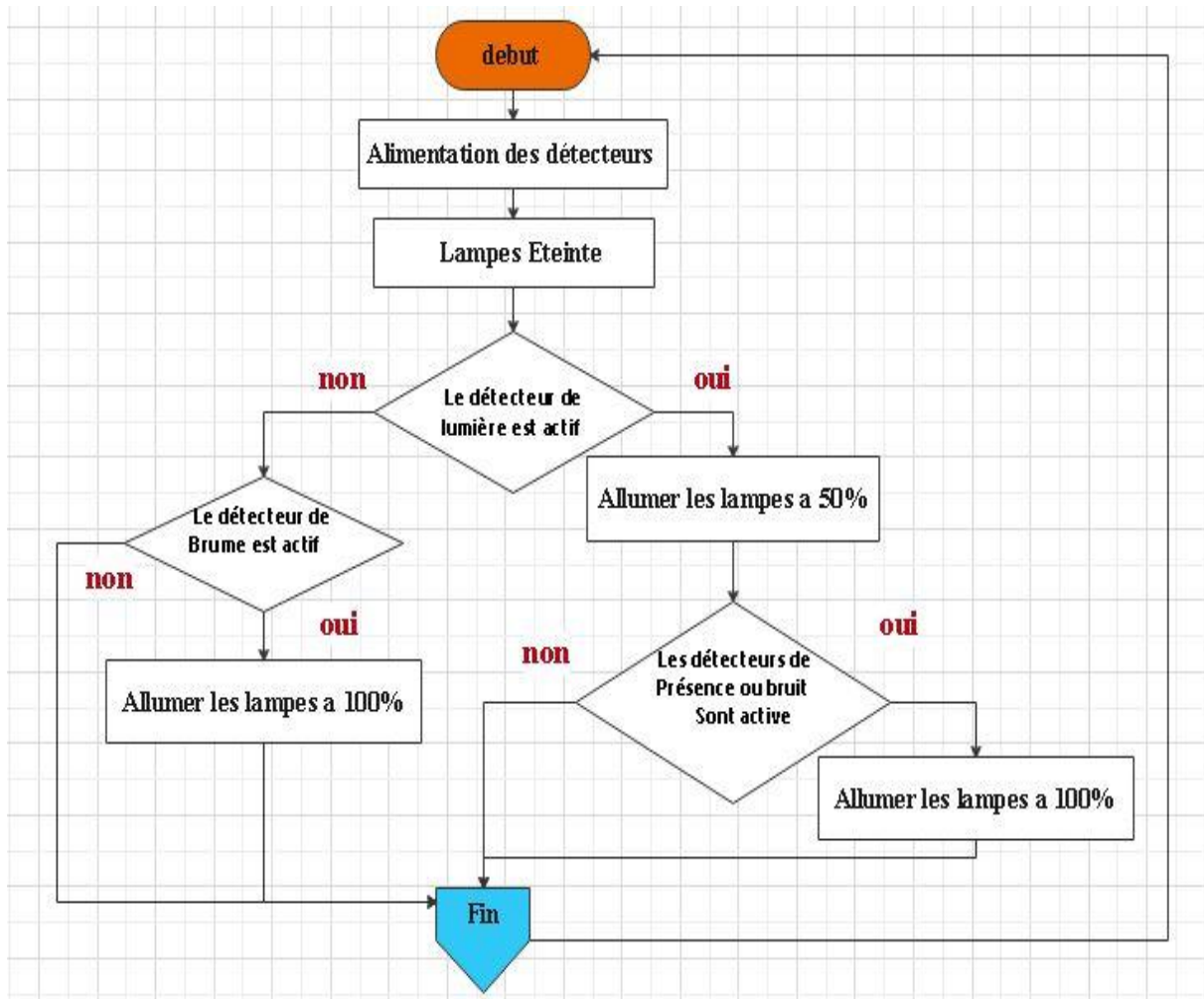


Figure IV. 5 Organigramme de fonctionnement de notre projet sans carte Arduino

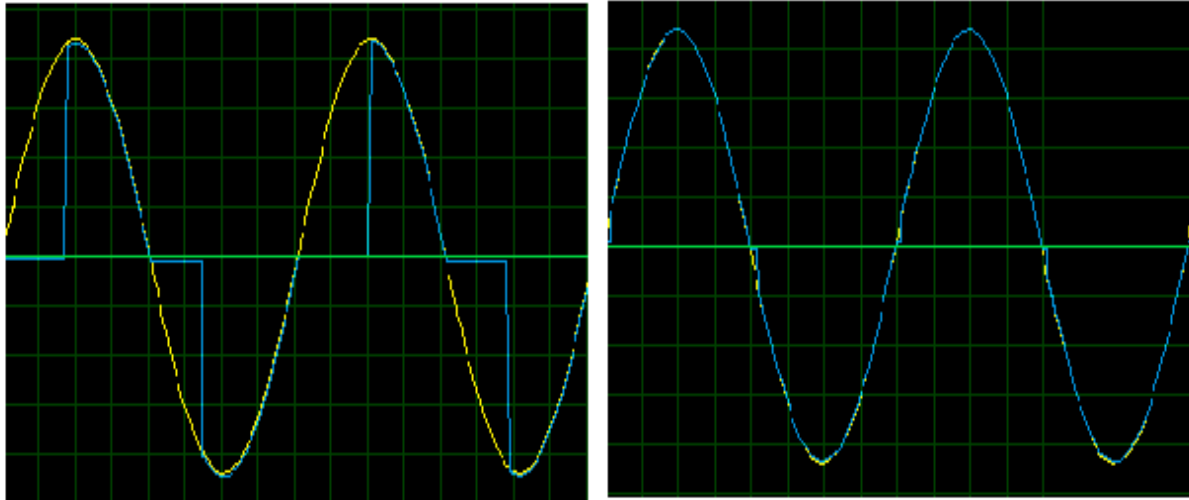


Figure IV. 6 Les signaux du niveau de luminosité

#### IV.4 Simulation de notre projet avec carte Arduino dans Proteus

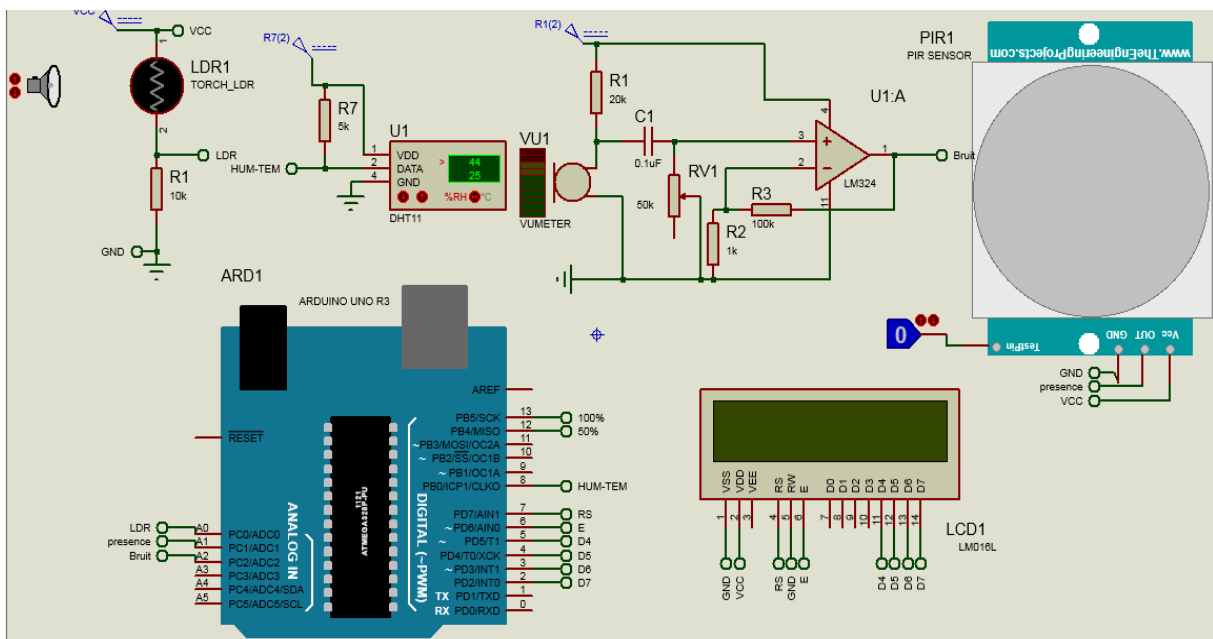


Figure IV. 7 Schéma électronique de la partie commande de notre projet (avec arduino)

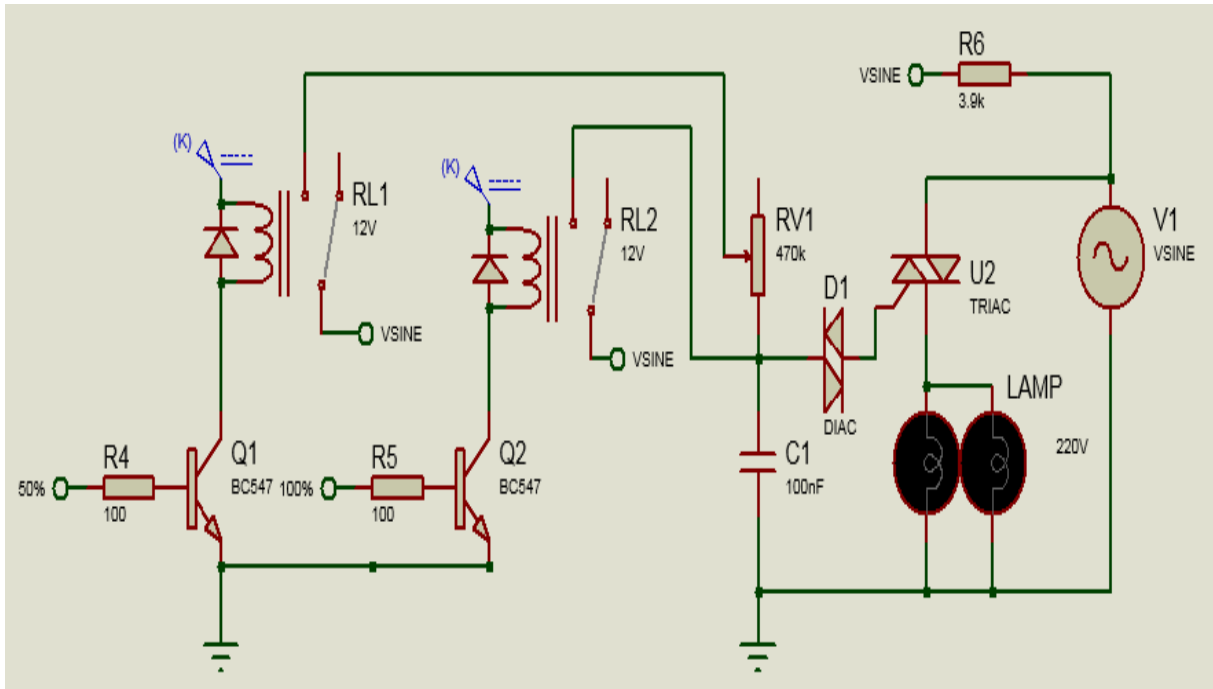


Figure IV. 8 Circuit électronique de la partie puissance de notre projet (avec Arduino)

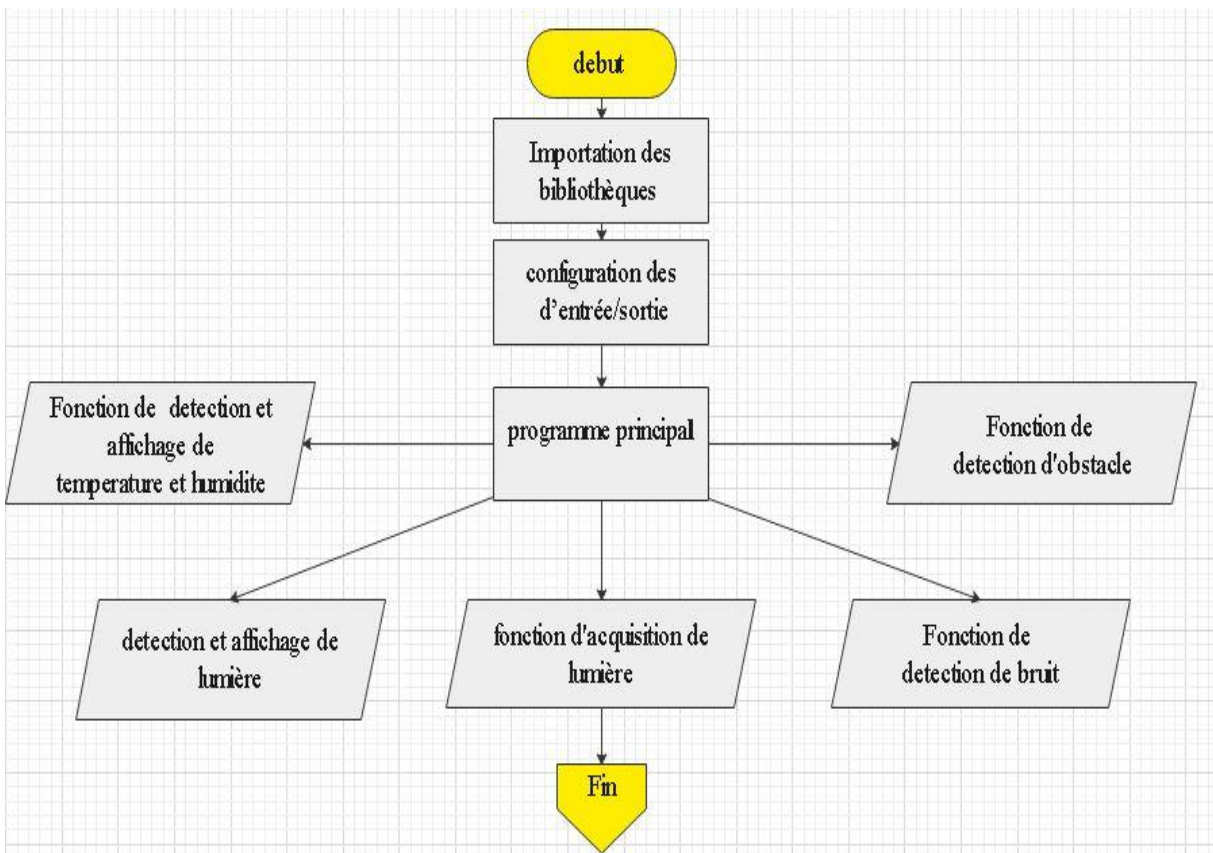


Figure IV. 9 Organigramme de fonctionnement de notre projet avec carte Arduino

### IV.4.1 Détecteur de lumière

#### a) Organigramme

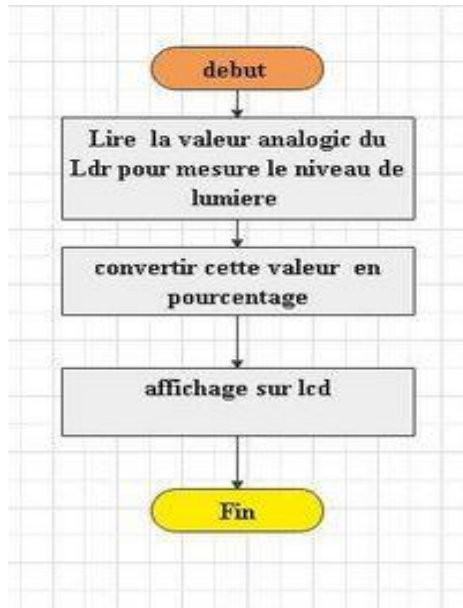


Figure IV. 10 Organigramme de fonctionnement de détecteur de lumière

#### b) Code Arduino

```

void Readlemire()
{
    v_lumiere = analogRead(lumiere);
    v_lumiere=map(v_lumiere,0,1023,0,100);
    lcd.setCursor (0, 0); lcd.print ("niveau delumiere ");
    lcd.setCursor (6, 1); lcd.print (v_lumiere);
    delay(500);
    lcd.clear();
}
    
```

#### c) Résultats de simulation

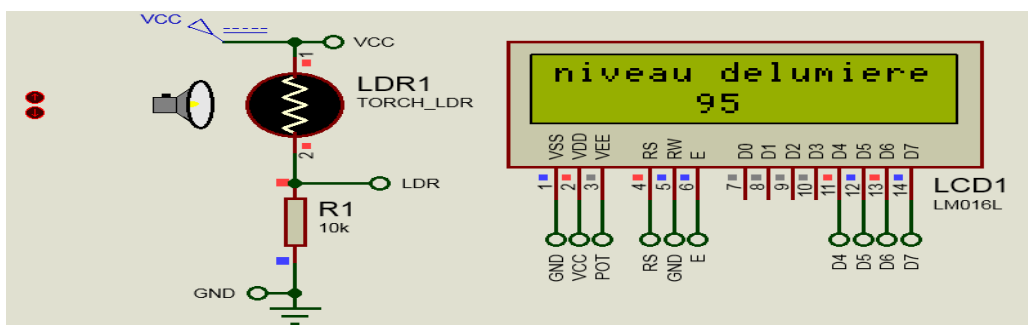


Figure IV. 11 Niveau de la lumière



### IV.4.2 Capteur de son

#### a) Organigramme

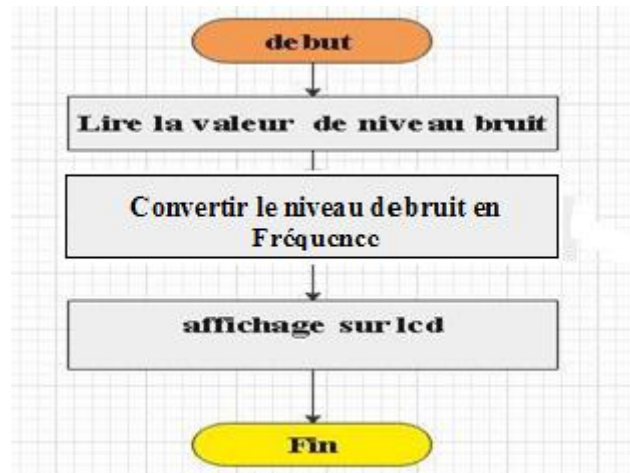


Figure IV. 12 Organigramme de fonctionnement de détecteur de bruit

#### b) Code Arduino

```

void detectbruit()          {
  if (FreqMeasure.available())
  {
    sum = sum + FreqMeasure.read();
    count = count + 1;
    if (count > 3) {
      float frequency = FreqMeasure.countToFrequency(sum / count);
      lcd.setCursor (3,0);lcd.print ("Le frequence:");
      lcd.setCursor (3, 1);lcd.print (frequency);
      lcd.setCursor (8, 1);lcd.print ("Hz");
      frequency_d=frequency;
      frequency=0;
      sum = 0;
      count = 0;
    }
  }
}
  
```

#### c) Résultats de simulation

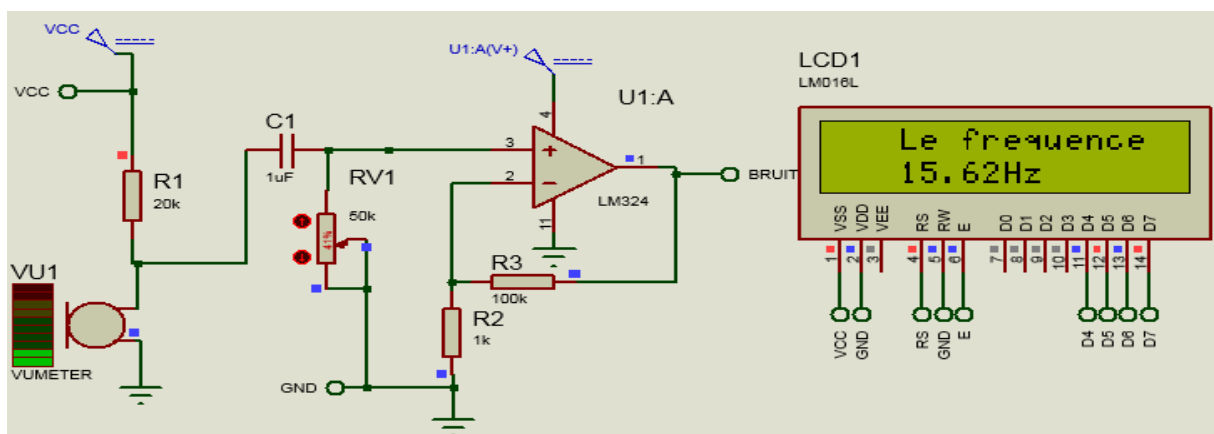


Figure IV. 13 Simulations du Détecteur de bruit

### IV.4.3 Capteur d'humidité

#### a) Organigramme

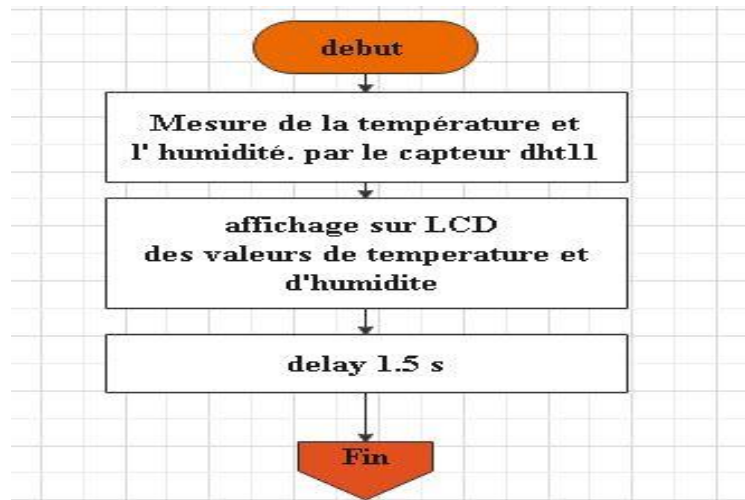


Figure IV. 14 Organigramme de fonctionnement de détecteur d'humidité

#### b) Code Arduino

```

void Readhumdete ()
{
    int err = SimpleDHTErrSuccess;
    if ((err = dht11.read(&temperature, &humidity, NULL)) != SimpleDHTErrSuccess) {
        return; }
    lcd.setCursor (0, 0); lcd.print ("Temperature ");
    lcd.setCursor (0, 1); lcd.print ("l'humidite ");
    lcd.setCursor (14,0); lcd.print ("C ");
    lcd.setCursor (14, 1); lcd.print ("%");
    lcd.setCursor (12, 0); lcd.print ((int)temperature);
    lcd.setCursor (12, 1); lcd.print ((int)humidity);
    delay(1500);
    lcd.clear();
}
    
```

#### d) Résultats de simulation

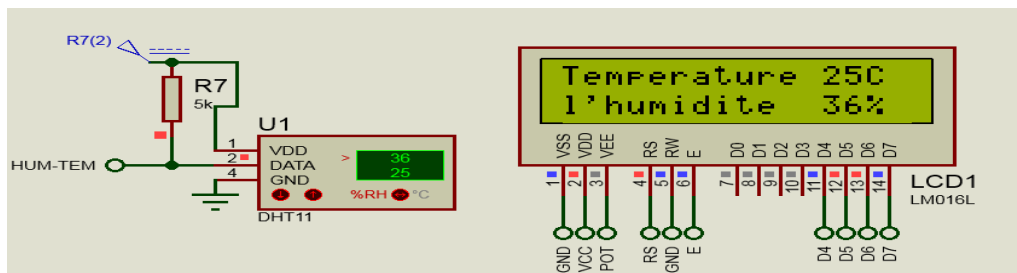


Figure IV. 15 Simulations détecteur d'humidité avec Arduino

#### IV.4.4 Détecteur de présence

##### a) Organigramme

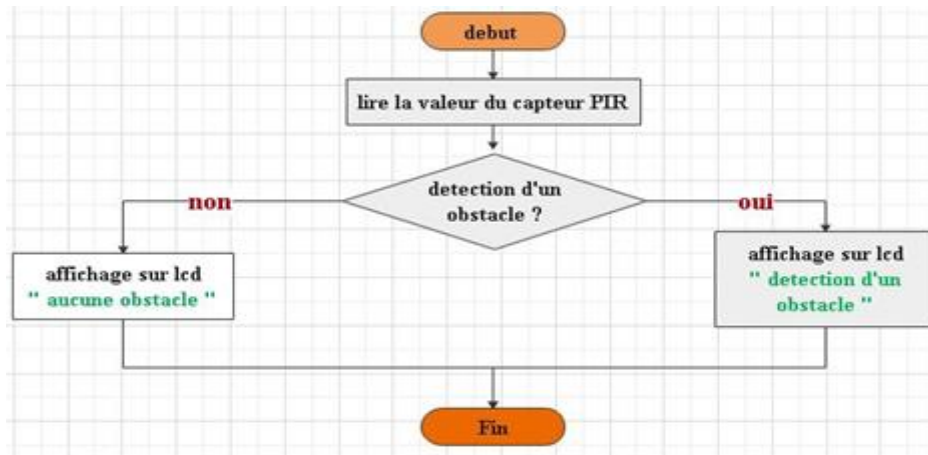


Figure IV. 16 Organigramme de fonctionnement de détecteur de présence

##### b) Code Arduino

```

void detecteOBS()
{
  v_presence =digitalRead(presence);
  if(v_presence =HIGH)
  {lcd.setCursor (0, 0); lcd.print (" obstacle !!!! ");}
  else
  {lcd.setCursor (0, 0); lcd.print (" aucune obstacle ");}
  lcd.clear();}
}
    
```

##### c) Résultats de simulation

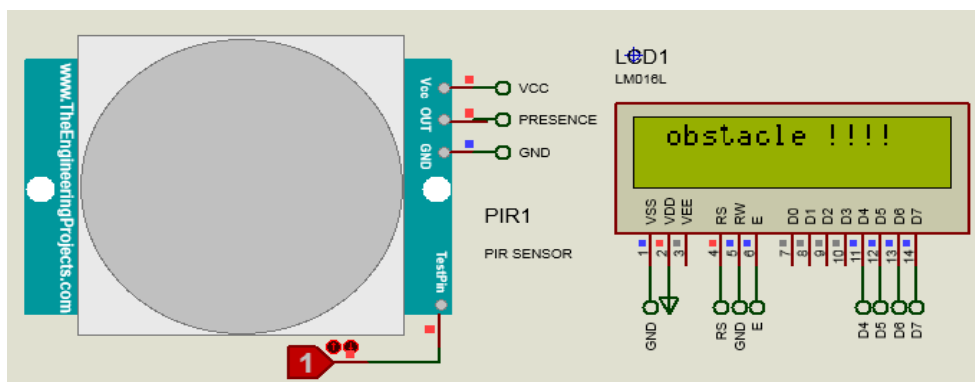


Figure IV. 17 Résultats de simulation de détecteur de mouvement

IV.4.5 Fonction de contrôle de niveau de lumière

a) Organigramme

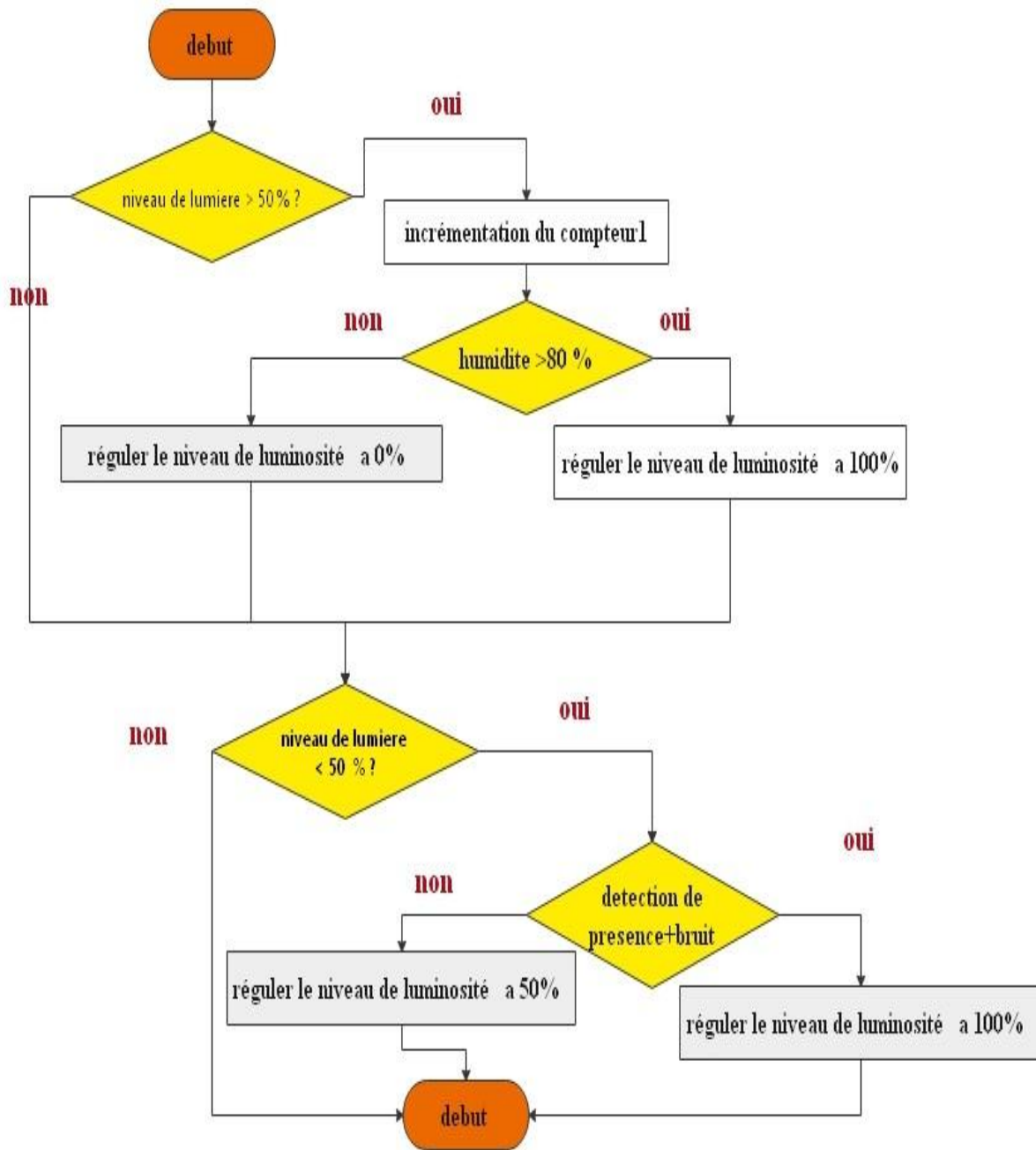


Figure IV. 18 Organigramme de contrôle du niveau de lumière

### b) Code Arduino

```

void light_Adjust()
{
    if( v_lumiere>50) {
        if(humidity>80){
            digitalWrite(nivau2,LOW);
            digitalWrite(nivau1,HIGH);}
        else
        {
            digitalWrite(nivau1,0);
            digitalWrite(nivau2,0);
        }
    }
    if (v_lumiere<50)
    {
        if((frequence_d>500) | (v_presence==HIGH))
        {
            digitalWrite(nivau1,HIGH);digitalWrite(nivau2,LOW);
        }
        else
        {
            digitalWrite(nivau2,HIGH);digitalWrite(nivau1,LOW);
        }
    }
}
}

```

---

### c) Code de Programme principal

```

#include <SimpleDHT.h>
#include <LiquidCrystal.h>
#include <FreqMeasure.h>
LiquidCrystal lcd(7,6,5,4,3,2);
#define pinDHT11 9
SimpleDHT11 dht11(pinDHT11);
byte temperature = 0;
byte humidity = 0;
#define lumiere A0
#define presence A1
#define bruit 8
#define nivau1 11
#define nivau2 10
int v_lumiere = 0 ;
int v_presence =0 ;
int v_bruit =0 ;
double sum=0;
int count=0;
float frequency, frequence_d ;

```

```
void setup()
{
  lcd.begin(16,2);
  pinMode( lumiere, INPUT);
  pinMode(presence,INPUT);
  pinMode(bruit,INPUT);
  pinMode(nivau1,OUTPUT);
  pinMode(nivau2,OUTPUT);
}
void loop()
{
  Readlemire();
  Readhumdete();
  detecteOBS();
  detectbruit();
  light_Adjust();
}
```

#### d) Résultats de simulation

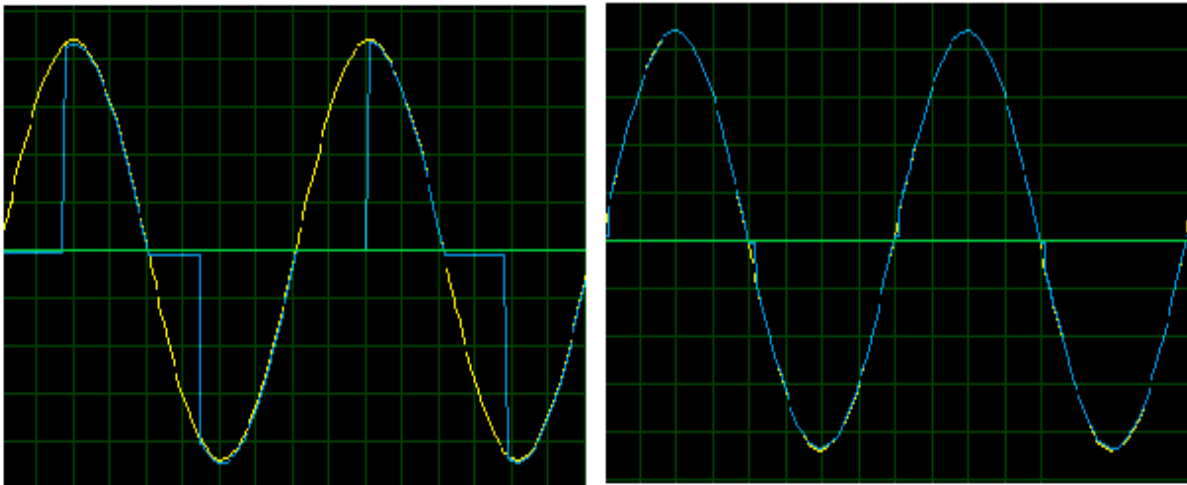


Figure IV. 19 Les signaux du niveau de luminosité des lampes

## IV.5 Partie Réalisation

### IV.5.1 Matériel utilisés

✓ **Les composants exploités**

- Un Arduino UNO
- Les résistances (100  $\Omega$  ,220  $\Omega$  ,1 K $\Omega$  ,10 K $\Omega$  ,20 K $\Omega$  ,50 K $\Omega$  et 100 K $\Omega$ )
- Les condensateurs (0.1 uF, 1 uF, 47 uF)
- Les LED blanches (3)
- Les microphones ELECTRET (3)
- Une résistance LDR
- Un récepteur infrarouge TSOP1738
- Un capteur humidité DHT11
- Emetteur infrarouge

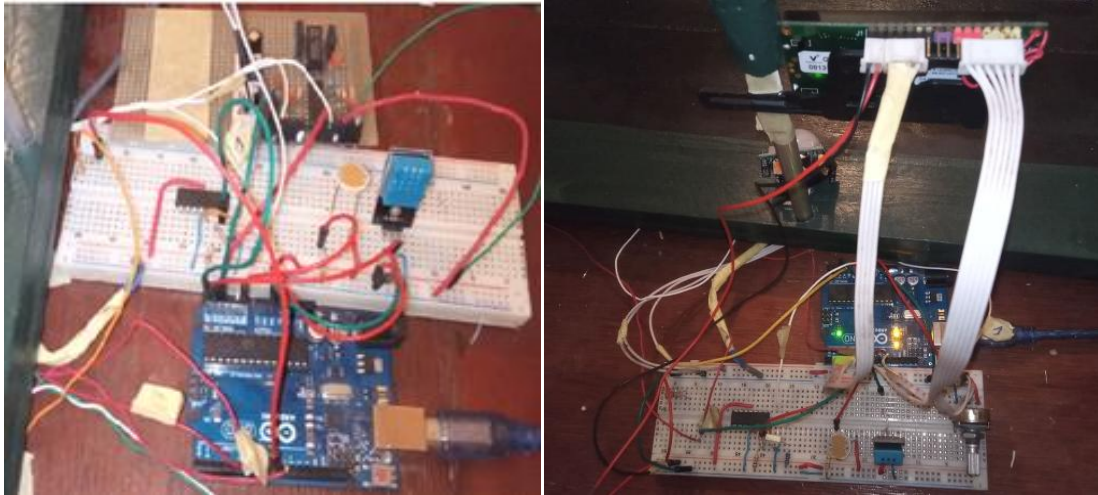
✓ **Les matériaux utilisés pour fait la maquette**

- Une plaque de bois (largeur 40cm, longueur 80cm et hauteur 1cm) est utilisée comme route
- Deux autre plaque (largeur 5cm, longueur 80cm et hauteur 1.5cm) sont utilisées comme Trottoirs
- Trois tubes (longueur 12 cm, diamètre 0.5 cm) sont utilisés comme poteaux d'éclairage public
- Trois LED blanches utilisé comme des lampes d'éclairage
- Nous avons fait les supports des lampes par polyester
- Deux bombes de peinture pour peindre la route, les trottoirs, et les supports



**Figure IV. 20** *La maquette*





**Figure IV. 21** *Circuit électronique avec Arduino*

#### IV.5.2 Test et mise en œuvre

Lorsque le pourcentage de lumière de l'environnement est inférieur à 50%, l'éclairage est automatiquement activé avec une intensité de luminance de 50%.



**Figure IV. 22** *l'éclairage dans la nuit*



Lorsque le capteur PIR détecte un mouvement (personne, animal, voiture) dans la nuit, les lampes s'allument automatiquement à 50%.



**Figure IV. 23** Détecteur de présence dans la nuit

En cas de brouillard ou d'humidité élevée, l'éclairage public est éclairé à 50%, quel que soit le pourcentage de lumière ambiante.

L'écran LCD affiche certains paramètres atmosphériques tels que la température et l'humidité



**Figure IV. 24** L'éclairage. Cas d'humidité



**Figure IV. 25** Afficheur LCD

### IV.5.3 Problèmes rencontrés

Dans ce projet nous avons exploité le microphone comme détecteur de son de véhicule avant d'accéder au poteau d'éclairage public au lieu d'utiliser un capteur sonore comme KY-038, malheureusement ne trouvons pas cette carte, nous avons essayé de simuler et réalisé des cartes sonores mais elles n'ont pas bien fonctionné comme il faut.

### IV.5.4 Estimation du coût du projet réalisé

Nous donnons le coût d'achat de composants, fournitures et matériaux spécifiques à la réalisation de notre projet.

	Désignation	Prix unitaire (DA)	quantité	Total ligne
<b>Composants électroniques</b>	<b>Arduino UNO</b>	<b>2500.00</b>	<b>1</b>	<b>2500.00</b>
	<b>DHT 11</b>	<b>800.00</b>	<b>1</b>	<b>800.00</b>
	<b>LDR</b>	<b>50.00</b>	<b>1</b>	<b>100.00</b>
	<b>PIR</b>	<b>800.00</b>	<b>1</b>	<b>800.00</b>
	<b>Resistance</b>	<b>5.00</b>	<b>10</b>	<b>50.00</b>
	<b>Micro ELECTRET</b>	<b>80.00</b>	<b>3</b>	<b>240.00</b>
	<b>Condensateur</b>	<b>10.00</b>	<b>7</b>	<b>70.00</b>
	<b>Lm 324</b>	<b>30.00</b>	<b>1</b>	<b>30.00</b>
	<b>TC 4538</b>	<b>100.00</b>	<b>1</b>	<b>100.00</b>
	<b>TRIAC</b>	<b>100.00</b>	<b>1</b>	<b>100.00</b>
	<b>Transistor BC547</b>	<b>50.00</b>	<b>2</b>	<b>100.00</b>
	<b>LCD 16*2</b>	<b>800.00</b>	<b>1</b>	<b>800.00</b>
	<b>Relais</b>	<b>100.00</b>	<b>2</b>	<b>200.00</b>
<b>Matériaux utilisés</b>	<b>Peinture</b>	<b>220.00</b>	<b>2</b>	<b>440.00</b>
			<b>Total</b>	<b>6330.00</b>

Tableau IV. 1 Estimation de de projet

## **IV.6 Conclusion**

Dans ce dernier chapitre nous avons réalisé un système de commande d'éclairage public intelligent, et aussi un système qui détecte quelques paramètres importants tel que la Présence, La lumière et l'Humidité.

Le système d'éclairage public a été programmé par ARDUINO.

## ***Conclusion générale***

## Conclusion générale

L'objectif de notre projet était d'étudier, de simuler et de réaliser un système de commande intelligent pour l'éclairage public.

D'abord, l'étude bibliographique menée a permis de se rendre compte de l'importance du domaine de l'éclairage public vis-à-vis du poids des nouveautés technologiques. Se passer de ce développement permettrait le fonctionnement des systèmes sans avoir recours à des améliorations.

Pour cela, plusieurs simulations ont été effectuées pour tester le fonctionnement des différents circuits étudiés. Dans ce cas, nous avons exploité le logiciel de conception assistée par ordinateur pour l'électronique « Proteus ». Ce qui nous a permis de saisir la réalisation de nos schémas électronique, donc de mieux se rattraper pour renforcer nos prérequis dans ce domaine.

De plus, nous avons saisi l'occasion de réaliser plusieurs programmes correspondant à diverses fonctions d'usage données dans le but de capter la lumière, l'humidité, le bruit et la présence, ainsi d'afficher quelques paramètres atmosphériques comme l'humidité et la température. Donc, chacune de ces fonctions a été programmée dans les cas sans et avec Arduino, Ce qui nous a permis d'obtenir un même résultat, mais pour deux situations différentes. Cela représente un avantage pour mieux comprendre la différence.

Par la suite, une maquette a été réalisée. Cela, nous a donné un point avantageux et motivant, et qui nous a poussé à avoir un certain savoir-faire dans le domaine de notre spécialité. Bien sûr, durant le processus de réalisation de notre système, on a rencontré pas mal d'obstacles, voire l'indisponibilité de quelques composants, ainsi que la contrainte liée au manque de notre compétence sur l'application de certains composants rencontrés très nouveaux pour nous, mais à l'issue de ce projet, on a pu comprendre leur importance et leur utilisation pratique.

Le travail développé dans le cadre de ce mémoire ouvre des voies d'amélioration que l'on peut classer comme suit :

- Application du système réalisé avec d'autres types de microcontrôleurs.
- Développement du système réalisé en se basant sur le principe des objets connectés

## Référence bibliographique

- [1] Mehalaine Nourelhouda, mémoire de fin d'études présenter pour obtenir le diplôme de master, Étude et réalisation d'un système intelligent pour la commande d'éclairage publique et surveillance de quelques paramètres atmosphériques
- [2] Sébastien Point, Exigences de conception des blocs autonomes d'éclairage de sécurité, Techniques de l'Ingénieur
- [3] <http://phozagora.free.fr>
- [4] <http://www.smartgrids-cre.fr>
- [5] <https://fr.wikipedia.org/wiki/Éclairage>
- [6] MAZARI Abdellatif et BELLAOUEDJ Ilyès, mémoire master, Etude et Réalisation d'un thermomètre à diode
- [7] Nicolas GOILAV Geoffrey LOI «Arduino » page (12-18.)
- [8] <http://www.smartgrids-cre.fr>.
- [9] Hachemi khaoula : Etude et réalisation d'un système d'alarme à base d'une carte Arduino, mémoire de fin d'étude pour master, Université Oum El Bouaghi, (2017)
- [10] <https://www.locoduino.org/spip.php?article234>
- [11] <http://www.zpag.net/Electroniques/Radio/AlimMicElect.htm>
- [12] <https://www.cours-gratuit.com/cours-arduino/support-de-cours-carte-arduino-uno-pdf>
- [13] <https://circuitdigest.com/microcontroller-projects/arduino-sound-level-measurement>
- [14] [http://jj.gueneugues.pagesperso-orange.fr/S\\_SI/cours/Fonction%20monostable.pdf](http://jj.gueneugues.pagesperso-orange.fr/S_SI/cours/Fonction%20monostable.pdf)
- [15] vademecum\_f4\_fr.pdf
- [16] IBGE, A-Tech & FIGE, 1997
- [17] <https://www.framboise314.fr>
- [18] M. MUSTAPHA BELARBI : Contribution à l'étude d'un générateur solaire pour site autonome, Thèse de Doctorat Es-Sciences, université des sciences et de la technologie d'Oran MOHAMED BOUDIAF.
- [19] [www.astuces-pratiques.fr](http://www.astuces-pratiques.fr)
- [20] [technologuepro.com](http://technologuepro.com)

[21] Leo Louis, 'Working principle of arduino and using it as a tool for study and research', International Journal of Control, Automation, Communication and Systems (IJACACS), Vol.1, No.2, April 2016.

[22] <https://www.technologuepro.com/microcontroleur-2/arduino/programmer.html>