REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE YAHIA FARES DE MEDEA

Faculté de **T**echnologie

Département du **G**énie **E**lectrique

Mémoire de fin d'études de Master

Filière : Génie électrique
Spécialité : Instrumentation

Smart Garden a base d'un Arduino NANO

Présenté par :

Belaidi Mohamed

Tchikou Sidali

Proposé et Dirigé par :

Dr kalache Mohamed Aissa

Promotion 2022

REMERCIEMENTS

Tous d'abord nous remercions ALLAH, notre créateur de nous avoir donné la force, la volonté et le courage afin d'accomplir ce travail modeste

C'est avec un grand plaisir que je réserve ces lignes en signe de reconnaissance à tous ceux qui ont de près ou de loin contribué à rendre ce travail possible. Même si des fois nos expressions nous trahissent et n'expriment pas réellement nos sentiments les plus sincères.

Ma profonde gratitude va en premier, à Allah le tout Puissant, le miséricordieux de m'avoir donné la force et la patience d'arriver à mettre ce travail à terme. Puis; je voudrais remercier mon encadreur et mon Prof :Dr : Kalach Mohamed Aissa pour son aide et pour la confiance qu'il a mise en moi, qu'il trouve ici toute ma gratitude et surtout ma parfaite considération de l'intérêt qu'il portait à ce travail depuis sa proposition de ce sujet.

Dédicace 1

Je dédie ce travail à mes très chers parents pour tous leurs sacrifices, leur soutien, leur prière tous au long de mes étude et qui ne cessent de me donner avec amour le nécessaire pour que je puisse arriver à, ce que je suis aujourd'hui.

Que dieux vous protège et que la réussite soit toujours à ma portée pour que je puisse vous combler de bonheur.

Mes amis de toute la promotion instrumentation, Ainsi qu'à toutes les personnes qui nous ont aidés à la réalisation de notre travail.

Belaidi Mohamed

Dédicace 2

Tout d'abord, je tiens à remercier DIEU,

De m'avoir donné la force et le courage de mener à

bien ce modeste travail.

Je tiens à dédier cet humble travail à :

A mon très cher père et chère maman (رحمها الله)

A mes sœurs

A mon binôme

A mes meilleurs amis

Tous ceux qui m'aiment et que j'aime .

TCHIKOU SIDALI

الملخص

حاليًا، في هذه الأيام والظروف، يتزايد الاهتمام بالنباتات الداخلية، حيث يمكن استخدامها بعدة طرق. يمكن للبعض أن يعتبرها زخرفة للمنزل أو لزراعة بعض الفاكهة على أي حال.

قررنا إنشاء مشروع يهتم بالنباتات الداخلية من جميع الجوانب. حيث نضع حساسات لقياس رطوبة التربة ودرجة الحرارة. عملنا على تحسين البيئة المحيطة بالمصنع كما لو كان هناك ضوء غير مناسب أو طقس أو إذا كانت درجة الرطوبة غير مناسبة، نضع بعض المراوح والمزيد من الميزات التي سيشرحها التقرير بالتفصيل.

الكلمات المفتاحية : النباتات الداخلية , حساسات , ضوء , طقس, الرطوبة , المراوح .

Résumé

De nos jours et dans ces conditions, l'intérêt pour les plantes d'intérieur grandit, certains peuvent les considérer comme une décoration pour la maison ou pour faire pousser des fruits quand même.

Nous avons décidé de créer un projet qui prendrait soin des plantes d'intérieur sous tous leurs aspects. Où nous mettons des capteurs pour mesurer l'humidité et la température du sol. Nous avons pris des dispositions pour améliorer l'environnement autour du jardin, comme un éclairage inadéquat, les conditions météorologiques ou l'humidité, et avons installé des ventilateurs et d'autres fonctionnalités que le rapport détaillera en détail.

les mots clés : plantes d'intérieur , capteurs , éclairage , météorologiques , humidité , ventilateurs .

Abstract

Nowadays and under such conditions, interest in indoor plants is growing, some can consider them as a decoration for the house or to grow some fruit anyway.

We decided to create a project that would take care of indoor plants in all their aspects. Where we put sensors to measure soil moisture and temperature. We have made arrangements to improve the environment around the garden, such as inadequate lighting, weather conditions or humidity, and have installed fans and other features that the report will detail in detail.

keywords: indoor plants , sensors , lighting , weather , humidity , fans**.**

Liste des abréviations

L'IoT : L'Internet des objets

PWM : Pulse Width Modulation

ICSP : L'Instrument contribuant à la stabilité et à la paix

TTL : Time To Live (durée de vie)

TOR : Tout Ou Rien

E : Entré

S : Sorte

T : Température

USB: Universal Serial Bus.

LED: Light Emetting Diode.

GND: Ground

GPIO : General Purpose Input / Output

SPI : Serial Peripheral Interface

CAN : Controller Area Network

UART : Universal Asynchronous receiver Transmitter

WiFi : Wireless Fidelity

I2C : Inter Integrated Circuit bus

Sommaire

Remerciement .... *i*

Dédicace1.....*ii*

Dédicace2.....*iii*

Résumé…….....*i*v

Liste des abréviations ……v

Sommaire .....….vi

Liste des figures ....……..……..…..…….. vii

Liste des tableaux ……...
vii

Introduction générale

Introduction générale : 1

***Chapitre* 1 Généralité sur la carte arduino nano**

I.1 Introduction : 4

I.2 Historique :.....4

I.3 Définition de la carte Arduino Nano [4] :.....4

I.4 Les broches de l'Arduino Nano :.....5

I.4.1 Alimentation de l'Arduino Nano :.....6

I.4.2 Les broches d'entrées / sorties : 6

I.4.3 Comment utiliser l'Arduino Nano : 7

I.5 Partie programme : 7

I.5.1 l'environnement de la programmation IDE Arduino :.....7

I.5.3 L'utilisation de IDE Arduino [5] :.....8

I.5.4 Les étapes de télé-versement du programme[4] :.....9

I.6 Discussion :.....10

Chapitre* 2 *Généralité sur les systèmes SMART GARDEN et les capteurs

II.1 Introduction : 12
II.2 Définition de la « smart garden » :....12
II.3 Systèmes jardin intelligents & IoT:....13
II.4.Généralités sur les capteurs :....14
II.4.1.Définition :.... 14
II.4.2.Structure d'un capteur:.... 15
II.4.3. Classification des capteurs : 16
II.4.3.1.Capteur passif :.... 16
II.4.3. 2.Capteur actifs : 17
II.4.3.3.Capteur analogique : 18
II.4.3.4.Capteur numérique : 18
II.4.4Rôle du capteur : 19
II.4.5.Paramètres caractérisant les performances d'un capteur :.... 19
II.4.5.1.La sensibilité : 19
II.4.5.2.Etalonnage: 19
II.4.5.3.La résolution:.... 20
II.4.5.4. La sélectivité:.... 20
II.4.5.5.Fidélité :....20
II.4.5.6. Linéarité :.... 20
II.4.5.7. Temps de réponse ou rapidité : 21
II.4.5.8. Réversibilité :.... 21
II.4.6.Grandeurs d'influence :.... 21

Chapitre 3 Simulation d'un Système « Smart Garden »

III.1 Introduction : 24
III.2 Modules du système : 24
III.2.1 Module de Perception et Reconnaissance de situations 24
III.3 Les capteurs et les actionneurs utilisent dans notre projet :....25
III.3.1Capteur de température et d'humidité DHT11 :.... 25
III.3.2.1 Le DHT11 est caractérisé par :.... 25
III.3.2.2 Le câblage :.... 26
III.3.2 Capteur d'intensité lumineuse :.... 27
III.3.2.1 Caractéristiques du capteur :.... 27

III.3.2.2 Le câblage 27
III.3.3 Capteur d'humidité de sol : 28
III.3.3.1 Suggestions et Caractéristiques techniques 29
III.3.3.2 le câblage 29
III.3.4 Le module wifi ESP8622 : 30
III.3.5 Le module wifi ESP8266 est caractérisé par : 30
III.3.6 Module Bluetooth : 31
III.3.6.1 Principe de fonctionnement : 31
III.3.6.2 Le câblage : 31
III.3.6.3 Configuration du module HC-06 : 32
III.4 Schéma globale de câblage : 33
III.5 Schéma bloc : 33
III.5.1 Explication de Schéma Synoptique : 34
III 5.2 Gestion de Ventilation: 35
III 5.3 Gestion de l'irrigation : 35
III 5.4 Gestion d'éclairage : 35
III.5.5 Prise de décision : 36
III.6 Schéma de simulation électrique: 38
III.7 Résultat de simulation : 40
III.8 Réalisation du projet « Smart Garden »: 41
conclusion générale : 43

Liste de figure

Chapitre 1 : Généralité sur la carte Arduino nano

Fig. I.1 : La carte Arduino NANO …….. 5

Fig. I.2 : Interface IDE Arduino ……..…… 9

Chapitre 2 : Généralité sur les systèmes SMART GARDEN et les capteurs

Fig. II.1 : Le principe générale de « *Smart Garden* » …… 13

Fig. II.2: Fonctionnement du capteur ……... 15

Fig. II.3 : Structure d'un capteur …… 16

Fig. II.4 : Exemple d'un signal analogique .....…. 18

Fig. II.5 : Exemple d'un signal numérique …….. 19

Fig. II.6 : Courbe d'étalonnage d'un capteur ……... 20

Fig. II.7: Etendue de mesure et courbe d'étalonnage du capteur ……. 22

Chapitre 3 : Simulation d'un Système « Smart Garden »

Fig. III.1 : Capteur de température et d'humidité (DHT11) ……. 25

Fig. III.2 : Câblage de capteur de température ……..... 26

Fig. III.3 : Capteur de la lumière …….. 27

Fig. III.4 : Câblage de capteur de la lumière BH175 …… 28

Fig. III.5 : Capteur d'humidité de sol …….. 29

Fig. III.6 : Module ESP8622 ……..…… 30

Fig. III.7 : Câblage de module esp8622 ……..…… 31

Fig. III.8: Module Bluetooth ……..……. 32

Fig. III.9 : Câblage de module Bluetooth HC-06 ……... 32

Fig. III.10 : Schéma globale de câblage ……..…… 33

Fig. III.11 : Schéma bloc ……... 34

Fig. III.12: Arbre de décision pour la gestion de l'irrigation ……..... 36

Fig. III.13: Arbre de décision pour la gestion de l'éclairage ……. 37

Fig. III.14: Arbre de décision pour la gestion Ventilation ……..…..… 37

Fig. III.15 : Schéma électrique de simulation..... 38

Fig. III.16 : La valeur de température 27°C est la ventillateur 0.00 …….. 39

Fig. III.17 : La valeur de température 31°C est la ventillateur 0.46 ……... 39

Fig. III.18 : La simulation de contrôle de composant LED ……..……. 40

Fig. III.19 : Schéma electrique de système par logiciel proteus …… 41

Fig. III.20 : La carte en 2D …… 41

Fig. III.21: La carte en 3D ……. 42

Liste de tableau

Tableau. II.1 : Type de matériau utilisé et caractéristique électrique des capteurs passifs ……17

Tableau. II.2 : Capteurs actifs : principes physiques de base ……...17

Introduction générale

Introduction générale :

Actuellement, de nos jours, l'attention portée aux plantes d'intérieur augmente, car elles peuvent être utilisées de plusieurs manières. Certains peuvent le prendre comme décoration pour la maison ou pour faire pousser des fruits de toute façon. Nous avons décidé de créer un projet qui prend soin des plantes d'intérieur sous tous leurs aspects. Où nous mettons des capteurs pour mesurer température, l'humidité du sol et l'humidité de l'air. Nous avons pris des dispositions pour améliorer l'environnement du jardin comme s'il y avait une lumière, un temps ou un degré d'humidité inappropriée. Nous mettons des ventilateurs et d'autres fonctionnalités que cela expliquera en détail. Amélioration de la qualité dans la gestion agricole est nécessaire à cette époque. Avec des nombreux d'incidents de mauvaises récoltes dus à une supervision inadéquate des conditions de terrain. Alors souvent le propriétaire du jardin fait la navette de maison en jardin juste pour vérifier le canal d'irrigation manuellement.

L'application de l'IoT (*Internet Of Thinks*) peut être utilisée dans de nombreux secteurs pour améliorer la qualité de la technologie et faciliter la vie humaine. Les plantes ont besoin de la fertilité du sol pour pousser de façon optimale. Le facteur clé de la gestion de la fertilité des sols réside dans réguler l'équilibre de l'oxygène, de l'eau et des nutriments. L'IoT permet de capter ou de contrôler des objets à distance comme la pompe d'eau et la ventilateur dans l'infrastructure réseau [1] [2].

La gestion de l'eau de chaque plante varie en fonction de l'humidité du sol. L'arrosage est un moyen de drainer ou d'apporter de l'eau au jardin régulièrement. Alors, L'améliorations dans la gestion des plantations sont nécessaires de toute urgence en ce moment. Appliquer la technologie dans le secteur des plantations peut réduire les pertes d'énergie et de temps en utilisant des méthodes d'irrigation efficaces telle que la « *smart garden* » ou jardin intelligent.

Ces recherches veulent résoudre des problèmes de moindre efficacité du mode d'agriculture traditionnel. Le sol sec nécessite une alimentation en eau, la pompe à eau doit être active pour drainer l'eau pour subvenir aux besoins du sol du jardin. La force de cette recherche est que le système de contrôle d'irrigation peut être utilisé partout et à tout moment.

La plupart des méthodes du jardin intelligent vise à créer un système d'irrigation basé sur l'Internet des objets (IoT). Dans ce système, le propriétaire du jardin peut surveiller l'état de l'humidité du sol et contrôler le débit d'eau qui sera arrosé sur les plantes. Le jardin intelligent aide les gens à surveiller automatiquement les paramètres du jardin. Il augmente ainsi les rendements des agricultures et optimiser l'utilisation de l'eau. Le système « smart garden »

utilise un capteur d'humidité du sol qui sera traité par processeur qui est connecté à l'Internet afin qu'il puisse envoyer les données obtenues à partir du capteur.

L'objective de cette thèse est de simuler un système électronique du jardin intelligent à base d'un Microcontrôleur Arduino Nano et des capteurs, ce Arduino Nano permet de contrôler un pompe d'eau pour surveiller l'humidité du sol (irrigation), et contrôler un ventilateur pour surveiller l'humidité du l'air et la température, et contrôler aussi des lampes pour surveiller luminosité.

Chapitre 1

Généralité sur la carte arduino nano

I.1 Introduction

Dans cette thèse on a utilisé Arduino nano comme un moyen pour réaliser un système du jardin intelligent. L'Arduino Nano jeu un rôle important dans cette projet. Pour ce la, on a discute en détail et explique le fonctionnement de cette Arduino dans ce chapitre.

I.2 historique :

En 2005, un projet dont personne n'avait imaginé les conséquences fut lancé à l'Interactive Design Institute Ivrea, à Ivrée en Italie. Une plaque de prototypage fut développée pour donner aux étudiants un moyen simple de concevoir des produits innovants. Un nom revient sans cesse dans ce contexte, celui de Massimo Banzi. C'est l'un des Co-développeurs et cofondateurs de la société Arduino LLC [3]. Pour l'anecdote, le nom Arduino a été emprunté à un bar d'Ivrée où se réunissaient régulièrement les instigateurs du projet. La philosophie qui a motivé ce projet est la simplification et la facilitation de l'accès à l'univers de l'électronique et des microcontrôleurs afin que chacun puisse l'utiliser sans être pour autant un spécialiste de ces domaines. La grande disponibilité et le coût modique de la carte Arduino, ainsi que des composants et modules électroniques, ont également contribué à la popularité rapide de la carte.

I.3 Définition de la carte Arduino Nano [4] :

Arduino est une carte électronique sur laquelle se trouve un microcontrôleur qui peut être programmé pour analyser et produire des signaux électriques.

La carte Arduino Nano est un produit plus spécifique que les cartes Arduino Uno et Arduino Mega. Compacte, elle est parfaite pour les applications un peu plus petites, mais malgré sa petite taille, elle contient une puissance intéressante pour permettre la construction d'objets intelligents et portables .

Il constituée par 14 broches d'entrées/sorties digitales, dont six sont utilisables en PWM, de 8 broches d'entrées analogiques, d'une connectique mini-USB à la place d'un port USB classique, d'un port ICSP et d'un bouton RESET. La figure I.1 montre la carte Arduino Nano.

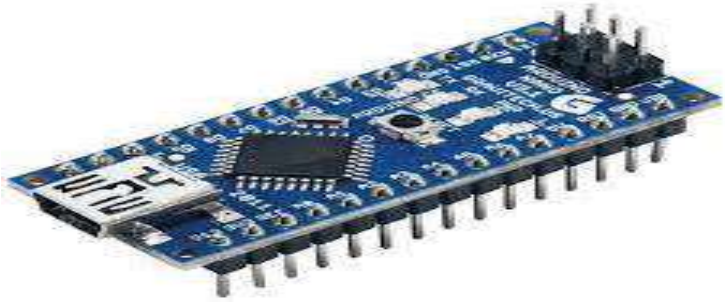


Fig. I.1 : La carte Arduino NANo

Il constitué aussi d'un microcontrôleur. Le microcontrôleur est le cœur de la carte Arduino. Il compose de mémoire morte, mémoire vive, processeur et les interfaces d'entrées/sorties. C'est lui que nous allons programmer . Une fois lancé et alimenté en énergie, il est autonome.

I.4 Les broches de l'Arduino Nano :

Le tableau I.1 montre les différents broches d'entrée et sortie de la carte Arduino Nano.

Catégorie des broches	**Nom des broches**	**Détails**
Alimentation	**Vin**, **3.3V**, **5V**, **GND**	**Vin:** Tension d'entrée sur Arduino lors de l'utilisation d'une source d'alimentation externe (6-12V). **5V:** Alimentation régulée utilisée pour alimenter le microcontrôleur et les autres composants de la carte. **3.3V:** Alimentation 3,3V générée par le régulateur de tension intégré. Le courant maximal consommé est de 50 mA. **GND:** Broches de masse.
Reset	**Reset**	Réinitialise le microcontrôleur.
Broches analogiques	**A0 – A7**	Utilisé pour mesurer une tension analogique dans la plage de 0-5V
Broches d'entrée / sortie	**D0 – D13**	Peut être utilisé comme broche d'entrée ou de sortie. 0V (bas) et 5V (haut)
Liaison série	Rx, **Tx**	Utilisé pour recevoir et transmettre des données série TTL.
Interruptions externes	2, 3	Pour déclencher une interruption.
PWM	3, 5, 6, 9, 11	Fournit une sortie PWM 8 bits.
SPI	10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK)	Utilisé pour la communication SPI.

LED incorporée	**13**	Pour allumer la Led intégré.
I2C	A4 (SDA), A5 (SCA)	Utilisé pour la communication TWI. (I2C)
AREF	**AREF**	Fournir une tension de référence pour la tension d'entrée.

I.4.1 Alimentation de l'Arduino Nano :

Il existe trois méthodes pour alimenter l'Arduino Nano :

- **La prise USB :** Connectez la prise mini USB à un chargeur de téléphone ou à un ordinateur via un câble pour que la carte reçoive l'énergie nécessaire au fonctionnement
- **La broche Vin** : La broche Vin peut être branché avec un 6-12V non régulé pour alimenter la carte.
- **La broche + 5V** : Si vous avez une alimentation régulée + 5V, vous pouvez directement alimenter l'Arduino à la broche + 5V.

I.4.2 Les broches d'entrées / sorties :

Il y a au total 14 broches numériques et 8 broches analogiques sur une carte Nano. Les broches numériques peuvent être utilisées pour interfacer les capteurs en les utilisant comme broches d'entrée ou pour piloter des charges en les utilisant comme broches de sortie. Une fonction simple comme pin Mode() et digital Write() peut être utilisée pour contrôler leur fonctionnement. La tension de fonctionnement est 0V et 5V pour les broches numériques. Les broches analogiques peuvent mesurer une tension analogique de 0V à 5V en utilisant l'une des 8 broches analogiques à l'aide d'une fonction simple, comme analogRead (). En plus de servir leur objectif, ces broches peuvent également être utilisées à des fins spéciales (voir ci-dessous):

- **Les broches de liaison série 0 (Rx) et 1 (Tx):**
  Les broches Rx et Tx sont utilisées pour recevoir et transmettre des données série TTL. Ils sont connectés à la puce série ATmega328P USB à TTL correspondante.
- **Les broches d'interruption externes 2 et 3:**
  Ces broches peuvent être configurées pour déclencher une interruption sur une valeur faible, un front montant ou descendant ou un changement de valeur.
- **Les broches PWM 3, 5, 6, 9 et 11:**
  Ces broches fournissent une sortie PWM 8 bits en utilisant la fonction analogWrite ().

- **Les broches SPI 10 (SS), 11 (MOSI), 12 (MISO) et 13 (SCK):**
  Ces broches sont utilisées pour la communication SPI.
- **La broche 13 LED intégrée :**
  Cette broche est connectée à une LED intégrée.
  Lorsque la broche 13 est HIGH, la LED est allumée et lorsque la broche 13 est BAS, elle est éteinte.
- **L'I2C A4 (SDA) et A5 (SCA):**

Utilisés pour la communication **I2C** à l'aide de la librairie Wire.

- **AREF:**
  Utilisé pour fournir une tension de référence aux entrées analogiques dotées de la fonction analogReference ().
- **La broche RESET:**
  Si cette broche est au niveau bas, le microcontrôleur est réinitialisé.

I.4.3 Comment utiliser l'Arduino Nano :

5 à 10 minutes seront à peine nécessaires pour télécharger votre premier programme sur Arduino Nano.
Tout ce dont vous avez besoin est un câble USB Arduino IDE et votre carte Nano elle-même.

I.5 Partie programme :

La carte Arduino est une carte électronique qui ne sait rien faire sans qu'on lui dise quoi faire. Pourquoi ? Eh bien c'est dû au fait qu'elle est programmable. Cela signifie qu'elle a besoin d'un programme pour fonctionner. L'environnement de programmation open-source pour Arduino peut être téléchargé gratuitement (pour Mac OS X, Windows, et Linux).

I.5.1 l'environnement de la programmation IDE Arduino :

Le logiciel de programmation de la carte Arduino sert d'éditeur de code (langage proche du C). Une fois, le programme tapé ou modifié au clavier, il sera transféré et mémorisé dans la carte à travers de la liaison USB. Le câble USB alimente à la fois en énergie la carte et transporte aussi l'information ce logiciel appelé IDE Arduino.

C'est une interface souple et simple et exécutable sur n'importe quel système d'exploitation Arduino basé sur la programmation en C. La figure I.2 représente divers bouton et fenêtres de l'interface de IDE Arduino.

I.5.3 L'utilisation d'IDE Arduino [5] :

L'IDE Arduino permet :

- d'**éditer un programme** : des **croquis** (*sketch* en Anglais),
  Les programmes sont écrits en langage C
- de **compiler ce programme** dans le langage « machine » de l'Arduino, *la compilation est une traduction du langage C vers le langage du microcontrôleur*
- la **console** donne des information sur le déroulement de la compilation et affiche les messages d'erreur.
- de **téléverser** le programme dans la mémoire de l'Arduino,
  le téléversement (upload) se passe via le port USB de l'ordinateur
  un fois dans la mémoire de l'Arduino, le logiciel s'appelle un **microgiciel**.
- la **console** donne des informations sur le déroulement du téléversement et affiche les messages d'erreur.
- de **communiquer** avec la carte Arduino grâce au **terminal** (ou **moniteur série**).
  pendant le fonctionnement du programme en mémoire sur l'Arduino, il peut communiquer avec l'ordinateur .
  tant que la connexion est active (câble USB, …)

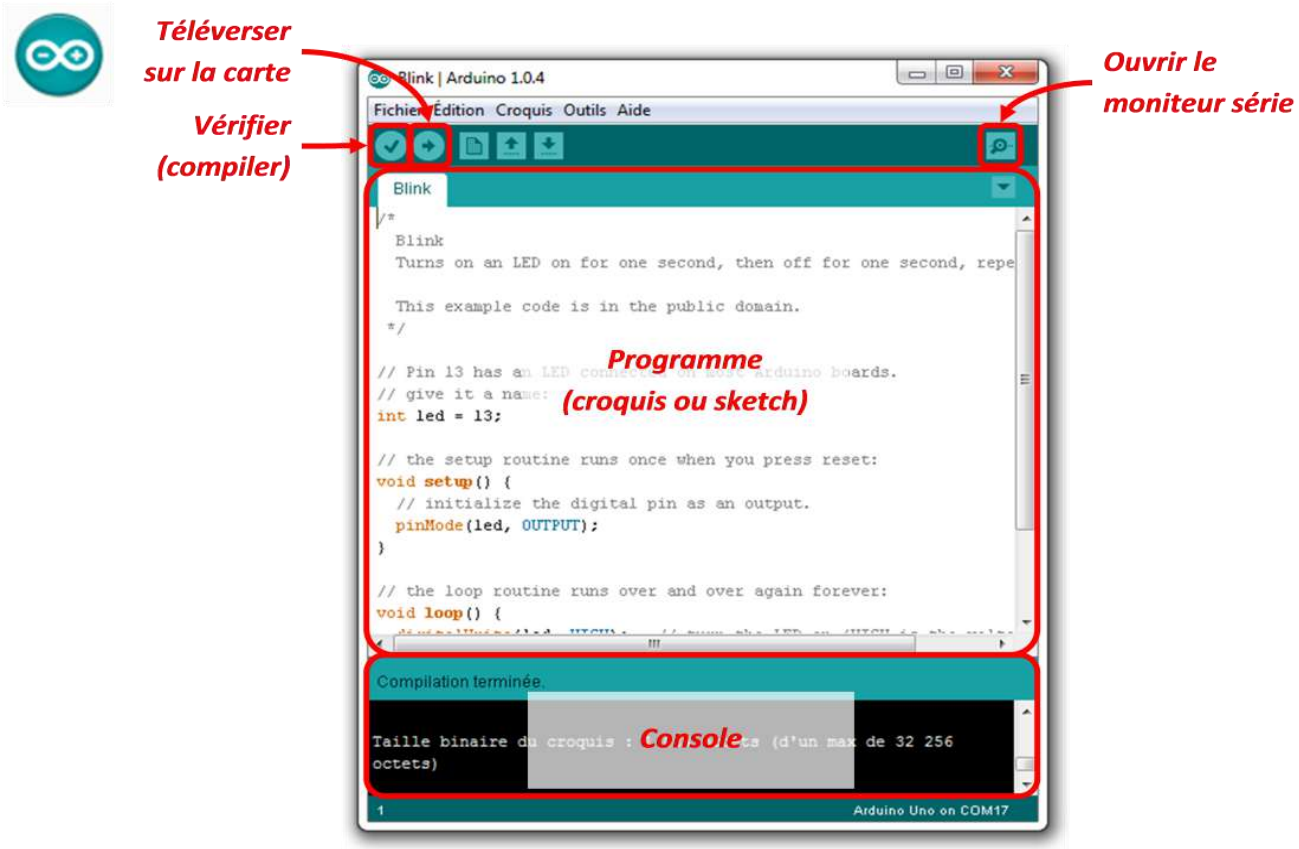


Fig. I.2 : Interface IDE Arduino .

I.5.4 Les étapes de télé-versement du programme[4] :

Une simple manipulation enchaînée doit être suivie afin d'injecter un code vers la carte Arduino via le port USB.

1. On conçoit ou on ouvre un programme existant avec le logiciel IDE Arduino.

2. On vérifie ce programme s'il y a des erreurs avec le logiciel Arduino (compilation).

3. Si des erreurs sont signalées, on modifie le programme.

4. On charge le programme sur la carte.

5. On câble le montage électronique.

6. L'exécution du programme lance automatiquement après quelques secondes.

7. On alimente la carte soit par le port USB, soit par une source d'alimentation autonome (pile 9 volts par exemple).

8. On vérifie que notre montage fonctionne

I.6 Discussion :

Dans ce chapitre, nous avons présentés les caractéristiques de la carte Arduino donnant ainsi les raisons pour lesquelles on l'a choisie, puis nous avons cité des différents types de cette dernière. Ensuite, nous avons expliqué les deux parties essentielles de la carte Arduino ; la partie matérielle et la partie de programmation. Nous avons également expliqué le principe de fonctionnement de la carte Arduino.

Chapitre 2

Généralité sur les systèmes SMART GARDEN et les capteurs

II.1 Introduction :

Dans ce chapitre, nous présentons le système "smart garden" qui est basé sur nombreux capteurs afin d'obtenir les commandes et informations nécessaires pour faciliter le processus de plantation et de production, ainsi que pour préserver le sol et les plantes.

II.2 Définition de la « smart garden » :

La plupart des plantes ont des besoins simples. Il n'y a que trois choses de base que vous devez comprendre avant de décider d'apporter une plante chez vous : la lumière, l'eau et l'air . Si vous pouvez maîtriser ces trois éléments, du point de vue d'une plante, vous pouvez créer un jardin intérieur partout dans le monde et à n'importe quelle saison de l'année.

• Lumière : La plupart des plantes de jardin ont besoin d'au moins six heures de lumière par jour. Mais il faut qu'il y ait une bonne lumière. Si vous placez votre main devant la fenêtre et qu'elle ne projette pas d'ombre, il y a de fortes chances que la lumière ne soit pas suffisante pour que la plupart des plantes vivent une bonne vie. Cependant, vous pouvez toujours compléter les conditions de faible luminosité avec des lampes de croissance.

• Eau : Les plantes ont besoin de conditions proches de celles de leurs habitats d'origine. Une plante qui vit dans le désert aura besoin d'arrosages moins fréquents qu'une plante qui vit dans une tourbière. Savoir quelles conditions d'eau une plante préfère est une bonne première étape pour garder un jardin d'intérieur réussi. Les plantes aux feuilles épaisses et caoutchouteuses sont des accumulatrices d'eau et peuvent généralement survivre avec moins d'eau que les plantes aux feuilles fines et délicates.

• Air : En tant que sous-produit de la photosynthèse, les plantes produisent de l'oxygène et filtrent les gaz nocifs, comme le formaldéhyde, de votre environnement domestique via leurs feuilles. Pour garder les plantes en bonne santé, vous devez garder leurs feuilles propres et garder l'air autour d'elles en mouvement et humide. Pour ce faire, vous pouvez les placer dans un endroit bien aéré ou les munir d'un petit ventilateur.

Ce système basé sur Arduino pour superviser l'état de température et d'humidité de ma plante et fournir automatiquement ses besoins nécessaires comme l'intensité lumineuse, l'eau et l'air frais pur, et pour ce faire on a besoin des capteurs pour contrôler certains

actionneurs. Par exemple, on contrôlera l'intensité lumineuse en fonction des signaux reçus du capteur de luminosité de la lumière, de la même manière pour l'arrosage on utilisera un capteur d'humidité pour allumer et éteindre une pompe à eau, et un capteur de température/humidité pour allumer et éteindre un ventilateur. Ce système s'appelle le jardin intelligent ou « *smart garden* ». La figure I.1 explique le système « *smart garden* ».

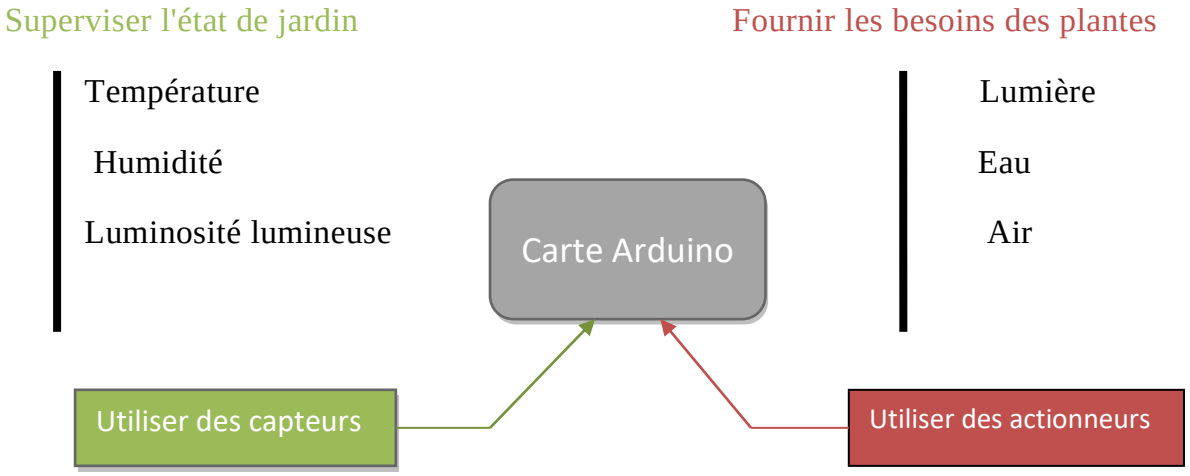


Fig. II.1 le principe générale de « *smart garden* »

II.3 systèmes jardin intelligents & IoT:

Smart Garden est un système d'automatisation a été conçue de plusieurs manières avec la capacité de l'Internet des objets (IoT) intégré avec l'améliorer pour fonctionner plus suffisamment au-dessus des spécifications requises. IoT comprend quatre éléments tels que les personnes, les choses, données et processus. Développer un jardin intelligent IoT système, il existe plusieurs composants qui peuvent être harnachée à cet effet.

- Capteurs : Les capteurs sont très essentiels dans l'environnement IoT parce que ce sont des appareils qui convertissent les signaux électriques en signaux numériques qui sont intégrés dans les systèmes [6]. Nous avons différents types de capteurs tels que: Capteur d'humidité de sol , capteur température et humidité , capteur de luminosité.
- NodeMCU : Le NodeMCU agit comme un hub qui existe en tant que plate-forme IoT open source qui fonctionne sur ESP8266 Wi-Fi d'Espessif Systems et du matériel du module ESP-01 qui est bon marché. Le micro logiciel peut être construit avec un

service de construction cloud, une image Docker, un environnement de construction Linux. Il s'agit d'un microcontrôleur à carte unique composé de 128 Ko de mémoire et de 4 Mo de stockage qui est conçu pour une programmation facile et permet un prototypage facile pour les développeurs [6].

- Arduino : Le logiciel Arduino fournit une meilleure intégration environnement de développement (IDE) pour la programmation le NodeMCU. Arduino est une plateforme multiplateforme application écrite en Java. Ce logiciel comprend diverses fonctionnalités qui incluent le code éditeur, couper et coller du texte, remplacer du texte et recherche, correspondance d'accolades, indentation automatique et coloration syntaxique. La carte dans le logiciel doit être changé en NodeMCU d'Arduino et les bibliothèques pour NodeMCU doivent être incluses dans le logiciel. La carte est testée avec une LED clignotante programme, puis le programme pour le jardin intelligent est écrit.
- Arrosage et irrigation automatiques : L'irrigation peut être considérée comme un processus de canalisation de la source d'eau vers le lit de la plante avec les étapes d'irrigation et les précautions appropriées. L'irrigation goutte à goutte souterraine (SDI) est une méthode d'irrigation répandue qui permet à l'agriculteur de contrôler quand et la quantité d'eau que ses cultures reçoivent. Les systèmes SDI comprennent des capteurs compatibles IoT pour surveiller en permanence les niveaux d'humidité et la santé des plantes [7].

II.4.Généralités sur les capteurs :

II.4.1.Définition :

Un capteur est un dispositif transformant l'état d'une grandeur physique observée en une grandeur utilisable, telle qu'une tension électrique, une hauteur de mercure, une intensité ou la déviation d'une aiguille. On fait souvent (à tort) la confusion entre capteur et transducteur : le capteur est au minimum constitué d'un transducteur [8].

Le capteur se distingue de l'instrument de mesure par le fait qu'il ne s'agit que d'une simple interface entre un processus physique et une information manipulable. Par opposition, l'instrument de mesure est un appareil autonome se suffisant à lui-même, disposant d'un affichage ou d'un système de stockage des données. Le capteur, lui, en est dépourvu.

Les capteurs sont les éléments de base des systèmes d'acquisition de données. Leur mise en œuvre est du domaine de l'instrumentation.

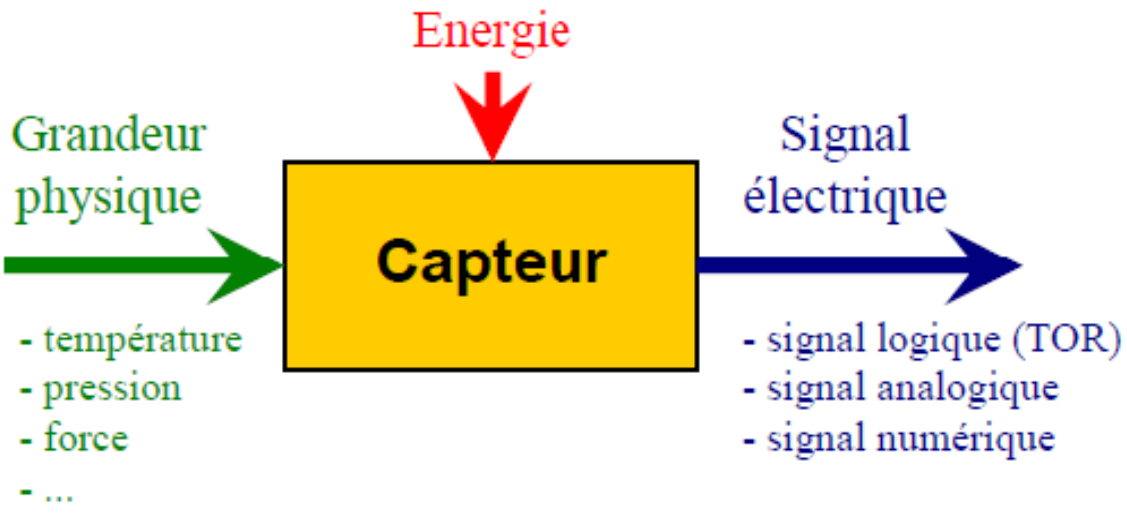


Fig. II.2: Fonctionnement du capteur

Selon le type d'application désirée, le choix d'un capteur adéquat nécessite l'établissement préalable d'un cahier des charges dans lequel sont énumérés :

- le type de l'événement à détecter,

- la nature de cet événement,

- la grandeur de l'événement,

- l'environnement de l'événement.

- etc.….

II.4.2.Structure d'un capteur:

Les différentes parties constitutives d'un capteur sont décrites ci-dessous (figure II.2) :

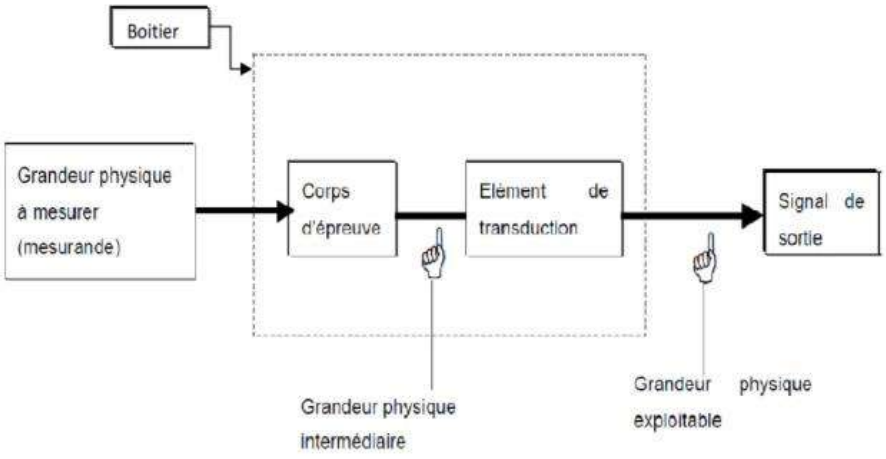


Fig. II.3 : Structure d'un capteur

-Le corps d'épreuve :

Le corps d'épreuve est un corps sensible qui réagit à la grandeur mesurée. Sa fonction est de convertir des grandeurs à mesurer en grandeurs physiques dites mesurables.

- Transducteur :

Le transducteur est un élément sensible fixé au corps d'épreuve. Son rôle et de convertir la grandeur mesurable en grandeur électrique et produisent un signal de sortie.

-Le boîtier :

Le boîtier est un élémenta mécanique de protection, de maintien et de fixation du capteur.

II.4.3. Classification des capteurs :

Suivant aux phénomènes physiques mis en jeux dans les capteurs, on peut classer ces derniers en deux catégories.

II.4.3.1.capteur passif :

Les capteurs dont le signal électrique délivré est une variation d'impédance sont dits passifs car ils nécessitent une source d'énergie électrique extérieure pour fonctionner

On les classe selon leur type d'impédance.

R : Résistance Ω (Ohms)

L : Inductance H (Henry)

C : Capacitance F (Farad)

La variation d'impédance résulte : soit d'une variation de géométrie ou/et sur les propriétés électriques des matériaux.

Grandeur mesurée	Caractéristique électrique sensible	Type de matériau utilisé
Température	Résistivité	Métaux : platine, nickel, cuivre ...
Très basse température	Constante diélectrique	Verre
Flux de rayonnement optique	Résistivité	Semi-conducteur
Déformation	Résistivité	Alliage de Nickel, silicium dopé
	Perméabilité magnétique	Alliage ferromagnétique
Position (aimant)	Résistivité	Matériaux magnéto résistants : bismuth, antimoine d'indium
Humidité	Résistivité	Chlorure de lithium

Tableau. II.1 : Type de matériau utilisé et caractéristique électrique des capteurs passifs

II.4.3. 2.capteur actifs :

On parle de capteur actif lorsque le phénomène physique qui est utilisé pour la détermination du mesurande effectue directement la transformation en grandeur électrique. C'est la loi physique elle-même qui relie mesure et grandeur électrique de sortie. Un capteur actif fonctionne assez souvent en générateur et dans ce cas, la grandeur de sortie est une différence de potentiel.

Grandeur physique mesurée	Effet utilisé	Grandeur de sortie
Température	Thermoélectricité	Tension
Flux de rayonnement optique	Photo-émission	Courant
	Effet photovoltaïque	Tension
	Effet photo-électrique	Tension
Force	Piézo-électricité	Charge électrique
Pression		
Accélération	Induction électromagnétique	Tension
Vitesse		
Position (Aimant)	Effet Hall	Tension
Courant		

Tableau. II.2 : Capteurs actifs : principes physiques de base

On peut classer les capteurs aussi en 2 groupes en fonction de la nature de l'information délivrée en sortie :

II.4.3.3.capteur analogique :

Un signal est dit analogique si l'amplitude de la grandeur physique qu'il représente peut prendre une infinité de valeurs dans un intervalle donné.

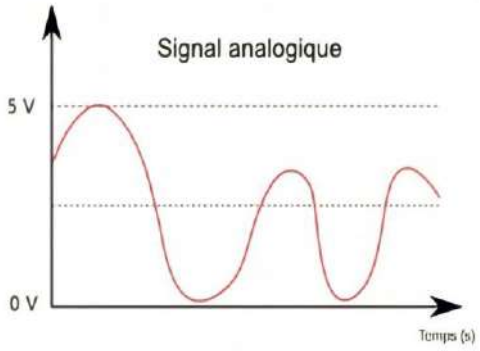


Fig. II.4 : exemple d'un signal analogique

II.4.3.4.Capteur numérique :

Les capteurs numériques donnent en sortie une valeur finie. Par exemple, si une grandeur physique croît de manière linéaire, la sortie du capteur qui va la mesurer donnera soit une information du type " TOR " (Tout Ou Rien), un train d'impulsion ou un échantillonnage.

- Tout ou rien (TOR) : il informe sur l'état d'un système ne pouvant avoir que deux états stables à la sortie du capteur (ouvert/fermé ou bien 1/0).
- Train d'impulsions : chaque impulsion est l'image d'un changement d'état. Par exemple, un codeur incrémental donne un nombre fini et connu d'impulsions par tour.
- Échantillonnage : c'est l'image numérique d'un signal analogique sous la forme d'un " escalier " (on retrouve cette caractéristique en sortie des convertisseurs analogiques / numériques). Pour un intervalle de tension d'entrée, correspond une seule valeur de la sortie.

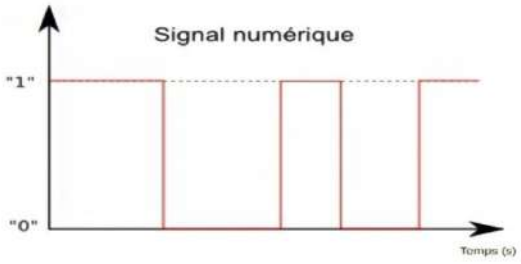


Fig. II.5 : exemple d'un signal numérique

II.4.4Rôle du capteur :

De façon générale, les principaux rôles des capteurs sont :

- ✓ contrôle-qualité des produits dans le domaine industriel.
- ✓ Capteurs de contrôle-qualité des produits finis.
- ✓ Gestion de la sécurité (présence, caméra…).
- ✓ Mesure d'accélération, de vibrations ou de chocs.
- ✓ Gestion de l'environnement (température, d'humidité…).

II.4.5.Paramètres caractérisant les performances d'un capteur :

Le comportement général d'un capteur est caractérisé par l'évaluation de ces performances via certains paramètres [9] [10], parmi ces derniers les plus utilisés sont les suivants :

II.4.5.1.La sensibilité :

Ce paramètre correspond au changement du signal de mesure lié à la modification ou au changement de la grandeur physique dans l'environnement du capteur.

Elle est définie comme étant la variation du signal de sortie (S out) par rapport à la variation du mesurande (m) (pente de la porti linéaire de la courbe d'étalonnage) et s'écrit:

$$S = \frac{\Delta S_{out}}{\Delta m}$$

Expérimentalement, la détermination de la sensibilité dépend directement de la grandeur mesurée.

II.4.5.2.Etalonnage:

Il donne l'évolution de la sortie en fonction de l'entrée. Il est généralement donné par la courbe en régime permanent .

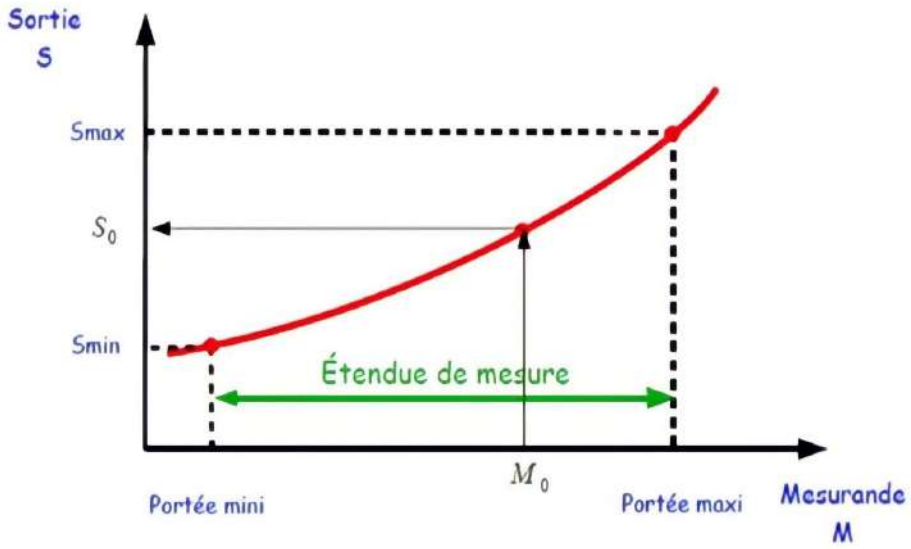


Fig. II.6 : Courbe d'étalonnage d'un capteur

II.4.5.3.La résolution:

La résolution R du capteur est la plus petite variation de la valeur physique mesurée. Ce dernier caractérise le plus petit décalage spectral possible que le capteur peut mesurer de manière significative. Ce paramètre est essentiel pour effectuer très peu d'inspections avec la plus grande précision possible .

II.4.5.4. La sélectivité:

Correspond à sa capacité à détecter une substance parmi d'autres. Elle dépend de la partie sensible du capteur [11].

II.4.5.5.Fidélité :

Il définit la qualité du capteur pour fournir le résultat le plus proche possible lors d'analyses répétées de la même grandeur physique m. L'erreur de fidélité correspond à la dispersion (écart type) σ des résultats autour de la valeur moyenne de n mesures effectuées sur m .

II.4.5.6. Linéarité :

L'erreur de linéarité spécifie l'écart maximal entre la courbe d'étalonnage et une ligne droite appelée « meilleure ligne ». L'écart linéaire est exprimé en tant qu'erreur système en pourcentage de la plage de mesure .

II.4.5.7. Temps de réponse ou rapidité :

C'est la capacité de l'instrument à suivre des changements à mesurer. Il représente le temps nécessaire au capteur pour produire sa sortie dans un certain pourcentage d'écart par rapport à la valeur finale lorsque le mesurande (l'entrée) est soumis à une variation brusque de type échelon .

Le temps de réponse noté tα est tel que α vaut généralement90% . La connaissance du temps de réponse d'un capteur est un élément essentiel lors de la réalisation de mesures.

II.4.5.8. Réversibilité :

Il définit la capacité d'un matériau à revenir à son état d'origine après avoir supprimé l'excitation .

II.4.6.Grandeurs d'influence :

Le vocabulaire international de métrologie nous propose la définition suivante : grandeur qui, lors d'un mesurage direct, n'a pas d'effet sur la grandeur effectivement mesurée, mais a un effet sur la relation entre l'indication et le résultat de mesure.

La température a souvent un effet sur les instruments de mesure et il est important pour le métrologue d'identifier toutes les grandeurs qui influencent la relation entre l'indication et le résultat de mesure.

Hormis le mesurande à caractériser, les grandeurs d'influence sont des "parasites" qui peuvent être d'influencer la grandeur électrique de sortie dans l'opération de détection. Les principales grandeurs d'influence comprennent:

- ✓ Température
- ✓ Pressions, accélérations, vibrations, forces
- ✓ Humidité
- ✓ la position du capteur et sa fixation
- ✓ les rayonnements nucléaires
- ✓ Champs magnétiques variables ou statiques
- ✓ Tension d'alimentation (amplitude et fréquence)

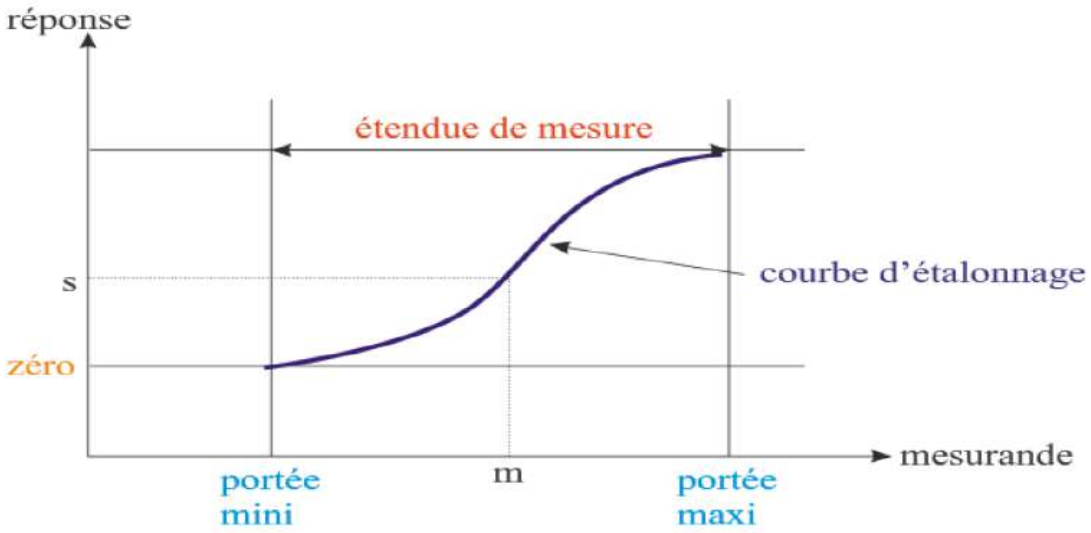


Fig. II.7 : Etendue de mesure et courbe d'étalonnage du capteur

Chapitre 3

Simulation d'un Système « Smart Garden »

III.1 Introduction :

Dans ce chapitre nous simulons d'un système « *Smart Garden* » à base d'Arduino Nano et différents capteur et actionneur. Dans un premier lieu, nous allons présenter le matériel nécessaire à la conception de notre projet, qui sera simulé ; nous expliquerons les différents capteurs et composants qui seront utilisés ainsi que leur mode de fonctionnement.

III.2 Modules du système :

Ce projet est constitué par deux modules :

- **Module de capture des informations :**

Composé d'une carte Arduino Nano et d'un ensemble de capteurs, ce module permet de collecter des données sur le climat intérieur sur le jardin, il s'agit de :

- La température et l'humidité de l'air.

- La luminosité ou l'intensité lumineuse.

- L'humidité du sol.

- **Module actionneur :**

Composé également par les composants suivants :

- Les lampes de lumière pour contrôler l'éclairage dans le jardin.

- Ventilateur pour l'aération du jardin.

- Pompe pour irrigation du jardin.

Les actions à entreprendre par ce module à propos de ces composants sont : allumer (composant) et éteindre (composant).

III.2.1 Module de Perception et Reconnaissance de situations

Le Module de Perception et Reconnaissance de situations est le module qui permet de reconnaître les différentes situations à la base desquelles les décisions (du système ou l'opérateur humain) doivent être établies. Ces situations sont déduites à partir des données brutes fournies par module de capture. Les différentes situations considérées dans le cadre de notre projet sont :

- Pour le sol: sec ou non sec.

- Pour la lumière: suffisant ou insuffisant.

- pour le climat: chaud ou modéré.

III.3 les capteurs et les actionneurs utilisent dans notre projet :

III.3.1Capteur de température et d'humidité DHT11 :

C'est un capteur de Température et d'humidité très apprécié pour sa simplicité de mise en œuvre et son coût peu élevé.

Il ne requiert qu'une résistance de tirage et une alimentation 3V ou 5V pour fonctionner. Sa programmation est facile à l'aide des librairies Arduino, le module numérique DHT11 délivre et reçoit un signal digital sur une entrée/sortie série unique.

Ses deux capteurs analogiques sont une résistance qui détermine le taux d'humidité et une thermo-résistance de type NTC (Negative Temperature Coefficient) afin de mesurer la température[12].

Ce capteur de température et d'humidité basé sur le circuit DHT11 communique avec un microcontrôleur type Arduino ou compatible via une sortie digitale.

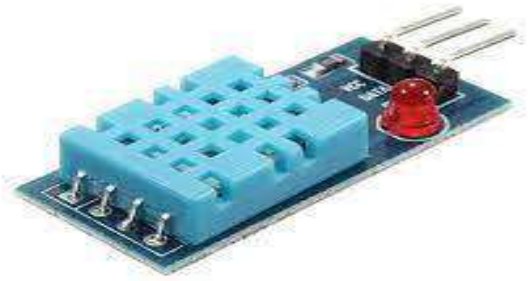


Fig. III.1 : capteur de température et d'humidité (DHT11)

III.3.2.1 Le DHT11 est caractérisé par :

-Alimentation: 3-5.5V DC

-Signal de Sortie: Signal Numérique via single-bus

-Capteur: Résistance Polymere

-Plage de Mesure: Humidité: 20-90%RH; Température: 0-50°C

-Précision: Humidité +-4%RH (Max +-5%RH); Température +-2.0°C

-Résolution: Humidité 1%RH; Température 0.1°C

-Hystérésis : +-1%RH

-Stabilité : +-0.5%RH/an

-Période de mesure: 2s

-Dimensions: 12x15.5x5.5

III.3.2.2 Le câblage :

Le brochage du capteur 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. Celle-ci doit impérativement être reliée à l'alimentation via une résistance de tirage de 4.7K ohms (il s'agit d'une sortie à collecteur ouvert).
- La broche n°3 n'est pas utilisée et ne doit pas être câblée.
- La broche n°4 est la masse du capteur (GND).

La figure suivant présentée le câblage entre circuit Arduino avec un capteur de Température (DHT11)

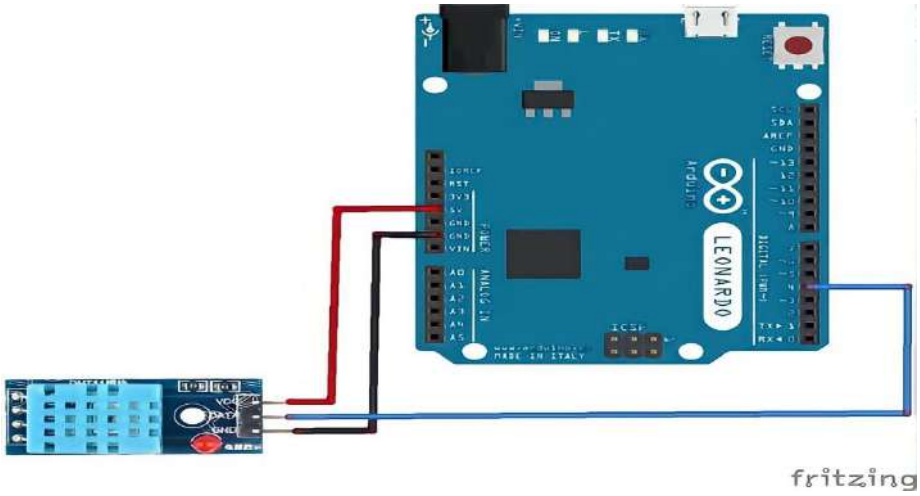


Fig. III.2 : câblage de capteur de température

III.3.2 capteur d'intensité lumineuse :

Le BH1750 est un capteur numérique de lumière ambiante. Il est facile de s'interfacer avec un microcontrôleur, car il utilise le communication I2C protocole. Il consomme une très faible quantité de courant. Ce capteur utilise une photodiode pour sentir la lumière. Cette photodiode contient une jonction PN. Lorsque la lumière tombe dessus, des paires électron-trou sont créées dans la région d'appauvrissement. En raison de l'effet photoélectrique interne, de l'électricité est produite dans la photodiode. Cette électricité produite est proportionnelle à l'intensité de la lumière. Cette électricité est transformée en tension.

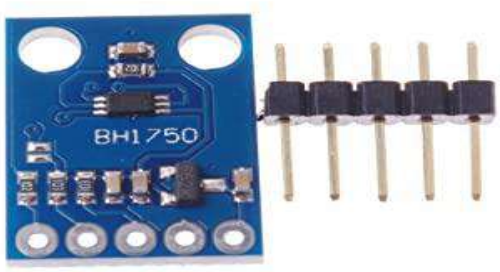


Fig. III.3 : capteur de la lumière

III.3.2.1 Caractéristiques du capteur :

- Alimentation : 3V – 5V
- Dimensions : 13,9 mm x 18,5 mm
- Compteur d'intensité lumineuse
- Sortie entre 0-65535 lux
- Interface I2C
- N'a pas besoin d'éléments externes

III.3.2.2 le câblage

Constitué par 4 bronches :

- GND
- VCC 3,3v / 5v
- SDA (Serial Data Line): ligne de données bidirectionnelle,
- SCL (Serial Clock Line): ligne d’horloge de synchronisation bidirectionnelle.

La connexion entre le capteur de la lumière et la carte Arduino est représenté dans la fig.II. suivant :

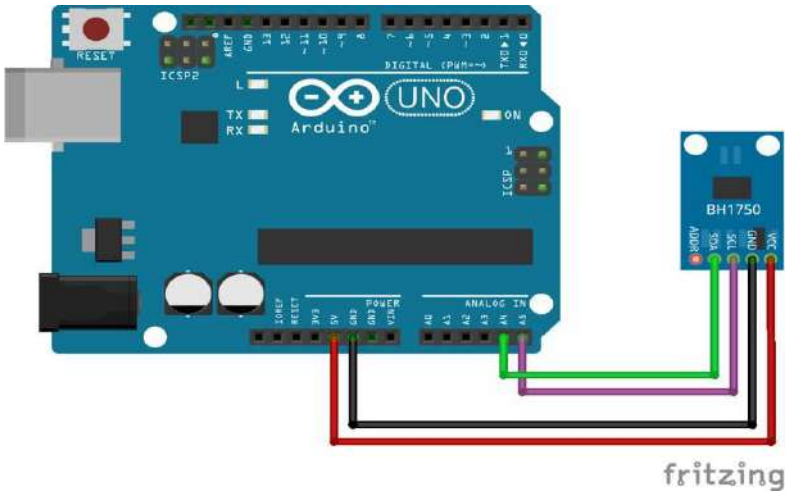


Fig. III.4 : câblage de capteur de la lumière BH1750

III.3.3 Capteur d'humidité de sol :

Le capteur d'humidité est un module électronique principalement utilisé pour détecter la teneur en humidité du sol. La carte de contrôle peut obtenir la valeur d'humidité ou le seuil dans le sol via des broches analogiques ou numériques.

La conductivité électrique de la terre dépend de l'humidité du sol, ce que veut dire que la résistance électrique augmente avec la sécheresse de celui-ci [13]. Raison pour laquelle le module comporte un support qu'on plante directement au sol où sont fixées des électrodes grâce auxquels la résistance du sol est mesurée par conséquent la concentration d'eau. Il a un commutateur pour basculer entre la sortie analogique et numérique. Capable de lire les informations spécifiques d'humidité du sol (analogiques) ou les informations de sol trop humide ou trop sec en fonction du seuil (numérique). Le potentiomètre réglable permet de définir le seuil d'humidité lorsqu'on utilise en numérique. Et Un circuit comparateur d'hystérésis pour une tension de sortie numérique plus stable [13].

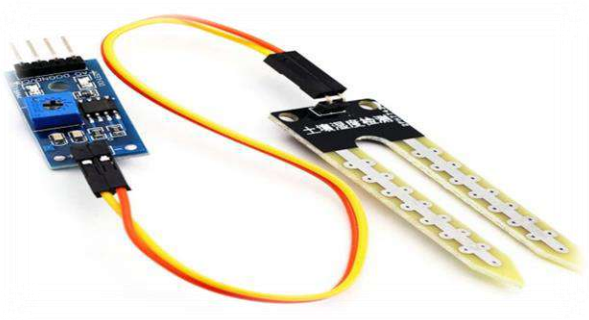


Fig. III.5 : Capteur d'humidité de sol

III.3.3.1 Suggestions et Caractéristiques techniques

• Mesurer l'humidité d'un sol sur plusieurs jours.

• Evaluer l'humidité de plusieurs espèces de plantes afin de déterminer la valeur d'humidité optimale.

• Mesurer l'humidité dans les serres afin de déterminer la quantité d'irrigation.

Caractéristiques

- Plage de Mesure : De 0 à 45 % d'humidité
- Précision : +- 4%
- Résolution : 0.1%
- Courant : 3 mA à 5V DC
- Fréquence par défaut : 10 Hz
- T° d'utilisation : -40 à 60°C

III.3.3.2 le câblage

_ D pour le port de sortie numérique,

– A pour le port de sortie analogique,

– S pour le port de sortie analogique / numérique (défini en fonction du commutateur),

– V et G pour la tension et la masse.

III.3.4 Le module wifi ESP8622 :

Le module Wifi ESP8266 ajoute une fonction de communication par WIFI à la carte Arduino. Il peut ainsi communiquer sans fil à moyenne distance avec n'importe quel autre dispositif Wifi (ordinateur, Smartphone, sur une autre carte Arduino...). La puce ESP8266 nécessite 3.3V tension d'alimentation. Il ne doit pas être alimenté avec 5 volts comme les autres cartes Arduino. NodeMCU ESP-12E carte de Dev peut être connecté à 5V en utilisant le connecteur micro USB ou une broche Vin disponible à bord. Les broches d'E/S de ESP8266 communiquer ou entrée / sortie max 3.3V seulement. Dire que les broches ne sont pas 5V entrées tolérantes. Si vous avez à l'interface avec 5V broches d' E / S, vous devez utiliser le système de conversion de niveau (soit construit vous - même en utilisant la tension de résistance diviseur [14].

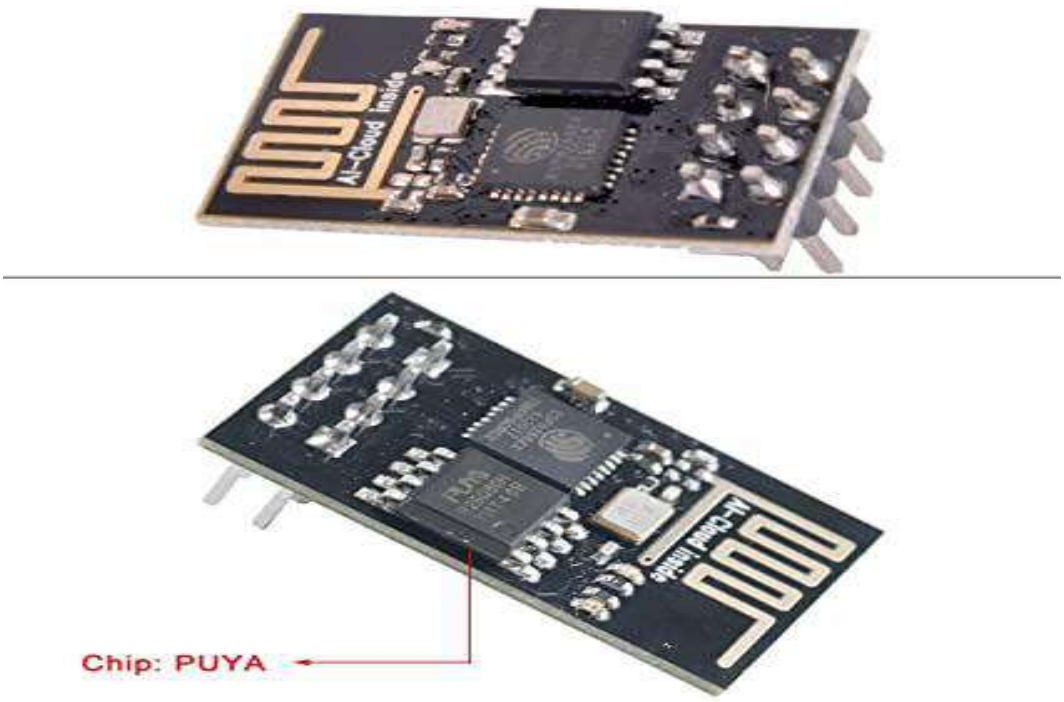


Fig. III.6 : module ESP8622

III.3.5 Le module wifi ESP8266 est caractérisé par :

-Wi-Fi Module - Module ESP-12E similaire à ESP-12 module, mais avec 6 GPIOs supplémentaires.

-Module ESP8266 ESP-12E

- USB intégré Adaptateur UART série (SiliconLabs CP2102)

- Bouton de réinitialisation

-Touche d'entrée (également utilisé pour bootloading)

-Montage en surface, LED rouge contrôlable par l'utilisateur

- régulateur de tension 500mA 3.3V (LM1117)

- Deux entrées d'alimentation protégée par diode (l'un pour un câble USB, une autre pour une batterie)

- Têtes - 2x 2,54 mm en - tête à 15 broches avec accès à GPIO, SPI, UART, CAN et broches d'alimentation

- Alimentation - 5V via port micro USB

-Dimensions - 49 x 24,5 x 13mm

Et pour câblé les modules wifi ESP8266 ont généralement quatre connexions comme indique dans la figure suivant :

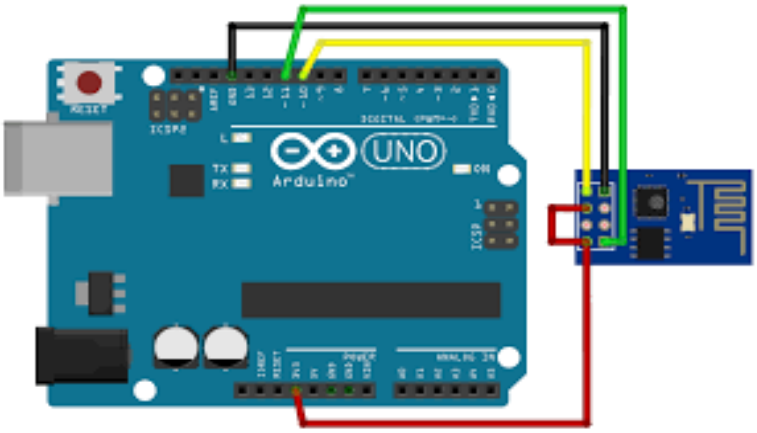


Fig. III.7 : câblage de module esp8622

III.3.6 Module Bluetooth :

III.3.6.1 Principe de fonctionnement :

Le module Bluetooth HC-06 ajoute une fonction de communication par Bluetooth à votre carte Arduino. Il peut ainsi communiquer sans fil à moyenne distance avec n'importe quel autre dispositif Bluetooth (ordinateur, téléphone mobile, second module Bluetooth sur une autre carte Arduino).

III.3.6.2 Le câblage :

Le module Bluetooth HC-06 présente 4 broches pour permettre d'établir la connexion série.

- VCC broche d'alimentation : Typiquement connectée à la broche 5V de l'Arduino.

- GND masse : Typiquement connectée à la broche GND de l'Arduino
- RX broche de réception : Typiquement connecté à la broche de transmission (TX) de l'Arduino
- TX broche de transmission : Typiquement connecté à la broche de réception (RX) de l'Arduino

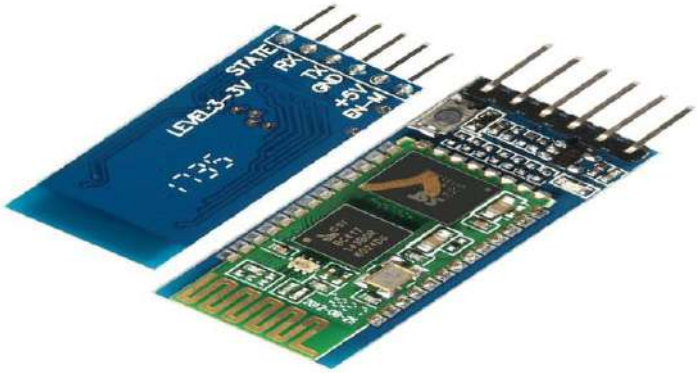


Fig. III.8 : module Bluetooth

III.3.6.3 Configuration du module HC-06 :

La configuration du module Bluetooth peut être intéressante pour vérifier le bon fonctionnement du module et pour modifier ses paramètres notamment lorsque vous utilisez plusieurs modules. Le module doit être alimenté mais non-apparié (la LED doit clignoter).

La **Fig. 9** présentée la connexion de module Bluetooth HC-06 avec l'Arduino :

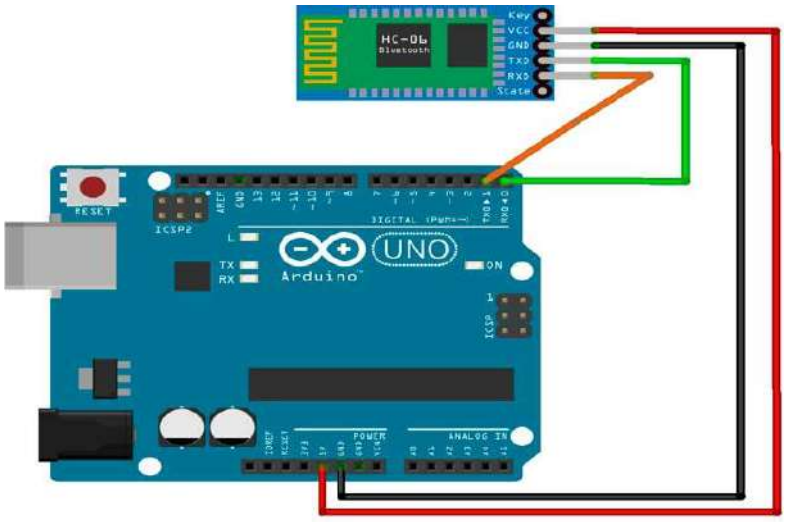


Fig. III.9 : **câblage de module Bluetooth HC-06**

III.4 Schéma globale de câblage :

La Figure III.6 montre comment les composant du système « Smart Gradin » connecte entre eux.

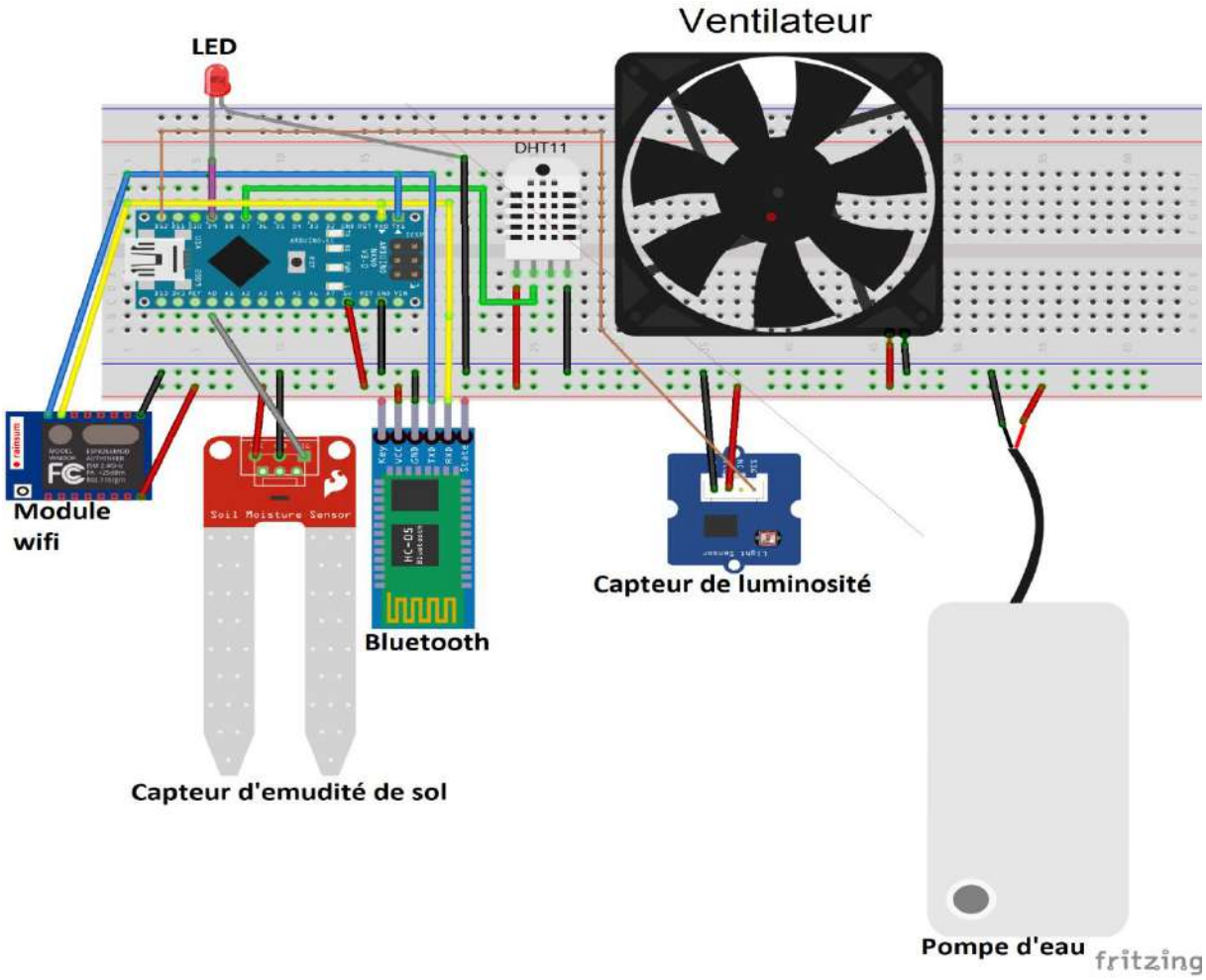


Fig. III.10 : **schéma globale de câblage**

III.5 Schéma bloc :

Notre système est constitué de plusieurs blocs comme indiqué ci-dessus :

- Bloc d’alimentation,
- Bloc de traitement d’information (Arduino Nano),
- Bloc des capteurs et des actionneurs (ventilateur, pompe d’eau, leds)
- Bloc de communication, (Bluetooth)

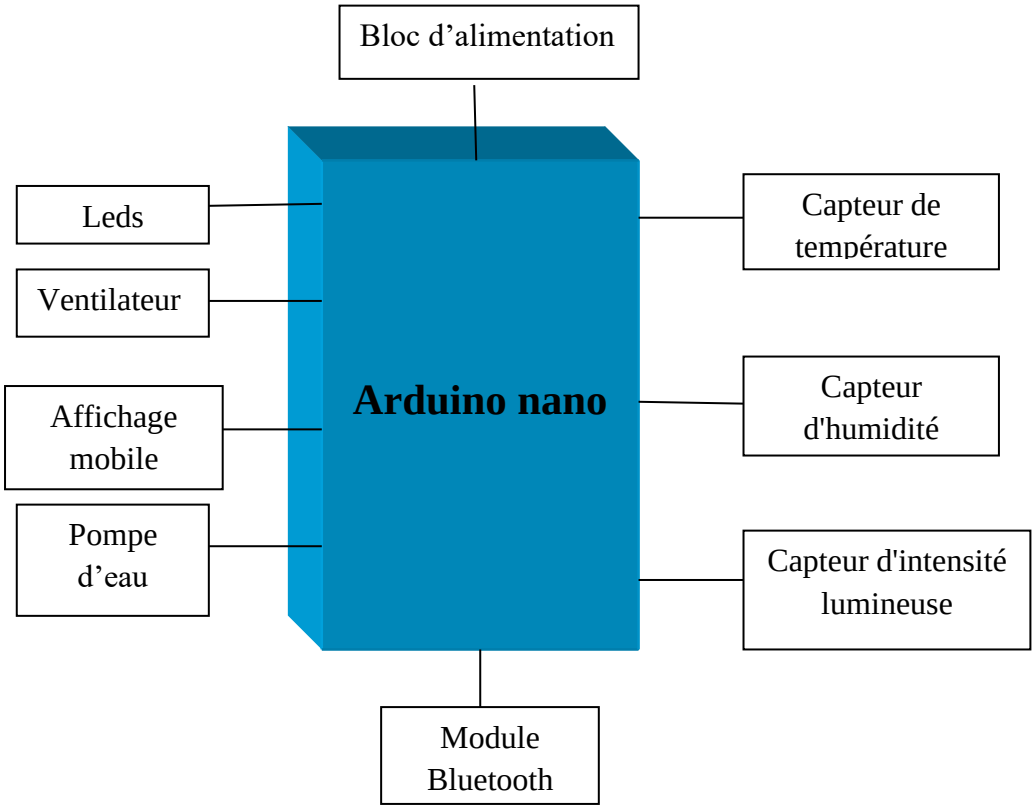


Fig. III.11 : Schéma bloc

III.5.1 Explication de Schéma Synoptique :

- Bloc d’Alimentation :

L’alimentation délivre une tension de 12v et âpres le traitement, elle délivre une tension de 5v, cette tension va alimenter tous les autres blocs.

- Bloc de traitement d’information (Arduino nano):

  C’est le cœur de notre système ,il est basé de carte Arduino, ce bloc reçoit des informations des capteurs et âpres traitement il donne les ordres aux actionneurs.

- Bloc des capteurs et actionneurs :

  Le Bloc des capteur est composé de : un capteur de température, d’humidité, et luminosité. Et le bloc d’actionneur et constitué de ventilateur, pompe d’eau et lampes.

- Bloc de communication (Bluetooth ,wifi):

  C'est un moyen de communication entre le jardin et la personne par téléphone et à distance, car il envoie tous les changements qui se produisent dans le jardin au téléphone, qui donne l'ordre d'éteindre et d'allumer tous les composants situés à distance.

III 5.2 Gestion de Ventilation:

La ventilation du jardin se déroule selon le scénario suivant :
- Les capteurs (capteur de température - capteur d'humidité de sol) enregistrent les valeurs de lumière et d'humidité de l'atmosphère et température, puis envoyées à la carte Arduino.

- Au niveau de la carte Arduino, le programme codée avec un identifiant les valeurs enregistrées, ce qui signifie est ajouté à la température enregistrée et l'humidité de sol.

- En cas de contrôle automatique, module de décision envoie des demandes d'ouverture ou de fermeture du ventilateur.

III 5.3 gestion de l'irrigation :

Les capteurs (capteur de température - capteur d'humidité de l'air - capteur d'humidité du sol) enregistrent les valeurs de température et d'humidité de l'atmosphère et l'humidité du sol, puis envoyées à la carte Arduino. Au niveau de la carte Arduino, le programme codée avec un identifiant les valeurs enregistrées, ce qui signifie que ajouté à la température enregistrée et l'humidité de l'atmosphère et l'humidité du sol (le but de ce processus est de distinguer les données de chaque capteur).
- En cas de contrôle automatique, module de décision envoie des demandes d'ouverture ou de fermeture du robinet pour l'irrigation.

III 5.4 Gestion d'éclairage :

L'éclairage du jardin se déroule selon le scénario suivant :

- Les capteurs (capteur de température - capteur d'humidité de l'air - capteur de luminosité) enregistrent les valeurs de lumière et d'humidité de l'atmosphère et température, puis envoyées à la carte Arduino.

- Au niveau de la carte Arduino, le programme codée avec un identifiant les valeurs enregistrées, ce qui signifie ajouté à la température enregistrée et l'humidité de l'atmosphère et lumière.

- En cas de contrôle automatique, module de décision envoie des demandes d'allumez la lampe et éteignez-la.

III.5.5 Prise de décision :

Tout d’abord, on teste si le sol est sec ou non, si le sol est non sec, la pompe d’eau est immédiatement fermée, Si le sol est sec, il est testé. Si l'humidité est inférieure à 12, la décision d'irrigation est prise pendant 10 minutes. Si elle est supérieure à 15, la décision d'irrigation est prise pour une période de 2 Minutes et sinon la décision d'irrigation est prise pour une période de 5 minutes.

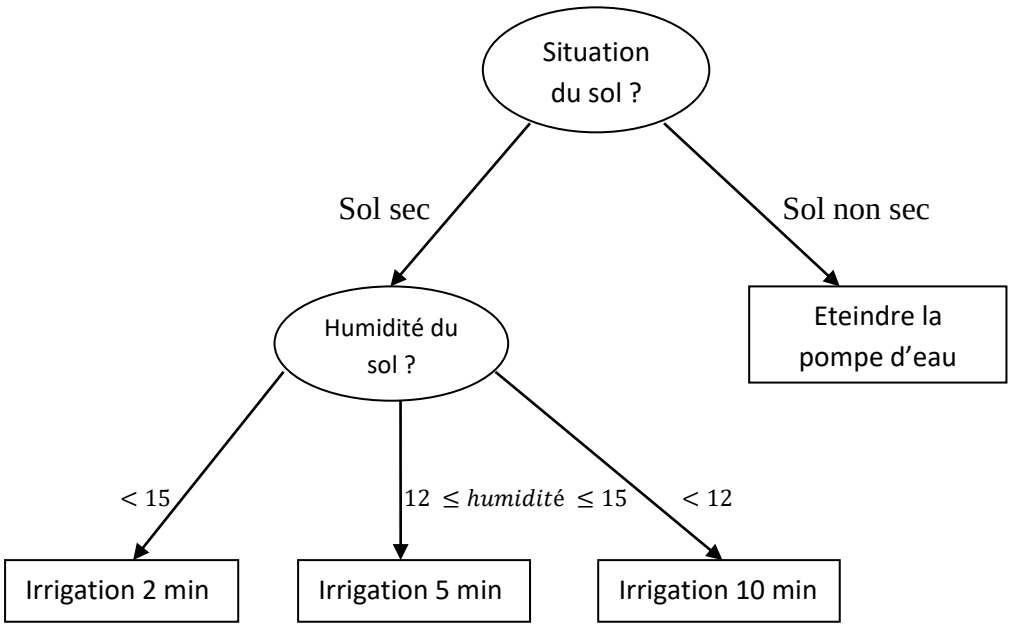


Fig. III.12: Arbre de décision pour la gestion de l'irrigation

Tout d’abord, tester si la lumière suffisant ou insuffisant, dans le cas insuffisant la lampe est immédiatement allumez, dans le cas suffisant, il est testé. Si la densité est inférieure à 12, la décision est allumez la lampe. Si elle est supérieure à 15, la décision éteindre la lampe et sinon rien à faire.

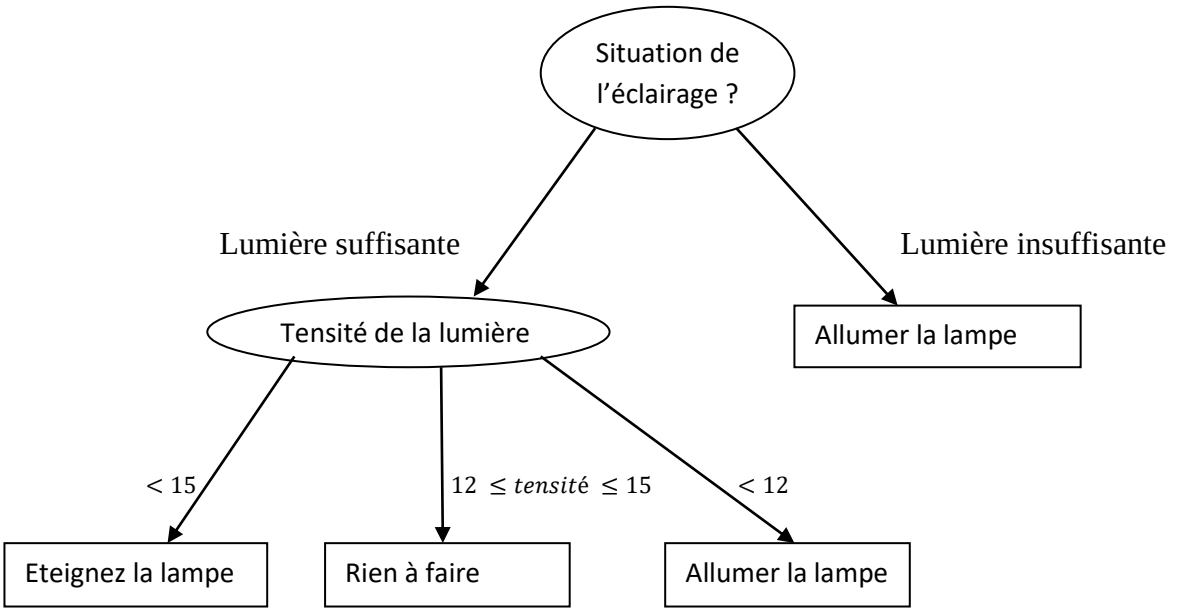


Fig. III.13: Arbre de décision pour la gestion d'éclairage

Tout d'abord, on teste si le climat dans le jardin est modéré ou chaud, dans le cas de modéré la ventillateur est immédiatement éteindre, dans le cas de climat chaud, il est testé. Si la température est supérieure à 15, la décision allumer ventilateur et sinon éteindre ventillateur.

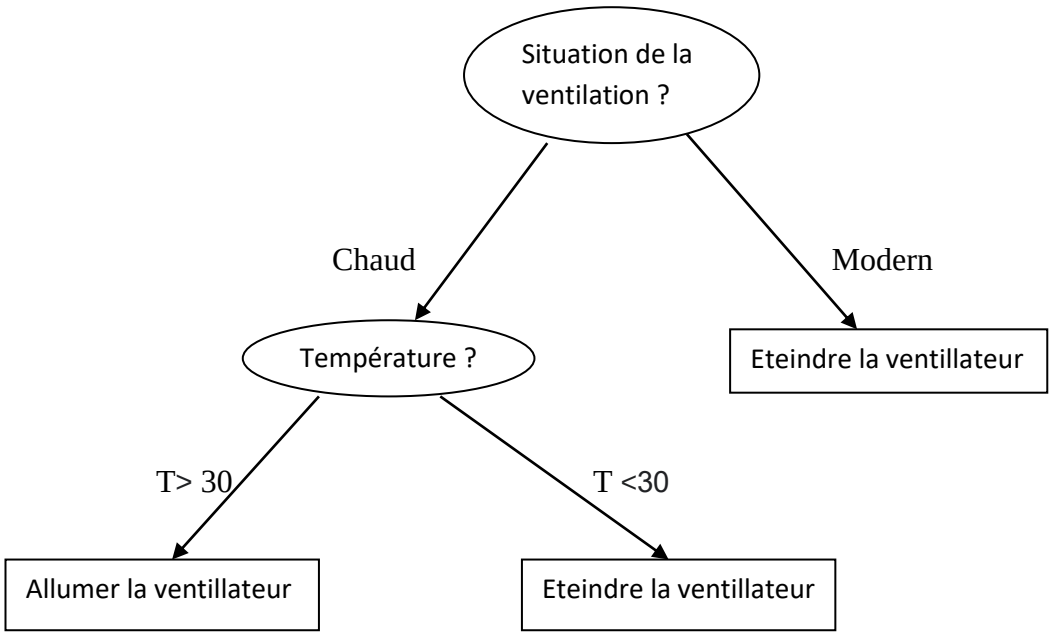


Fig. III.14: Arbre de décision pour la gestion Ventilation

III.6 Schéma de simulation électrique:

Nous allons simuler le projet de « smart garden à travers le logiciel Proteus 8 Professional, qu'il est composé de : (1) un capteur d'humidité du sol, (2) un capteur de luminosité, (3) un capteur de température et humidité DHT11, (3) un moteur qui remplace la pompe d'eau, (4) un ventillateur, (5) une diode LED, (6) un afficheur numérique pour visualiser les valeurs des capteurs, (7) un Arduino NANO. Le rôle de ce dernier est de contrôlé les actionneurs comme le moteur, la ventillateur et LED, grâce à des donné reçue par des capteurs.

Le but de cette simulation est lorsqu'Arduino reçoit une valeur de seuil spécifique de capteur d'humidité du l'air, le ventilateur lance automatique. Si la valeur du capteur de température est inférieure de ce seuil le ventilateur s'arrête. Le seuil est choisit par rapport le type de la plante. Dans notre projet on a choisit cette valeur de seuil 30°C.

Le même principe pour contrôler le moteur (pour surveiller l'humidité du sol) et la diode LED (pour surveiller luminosité). La figure III.15 montre le schéma de simulation dans logiciel Proteus de notre projet. Le code d'Arduino est présenté dans l'Annexe (B).

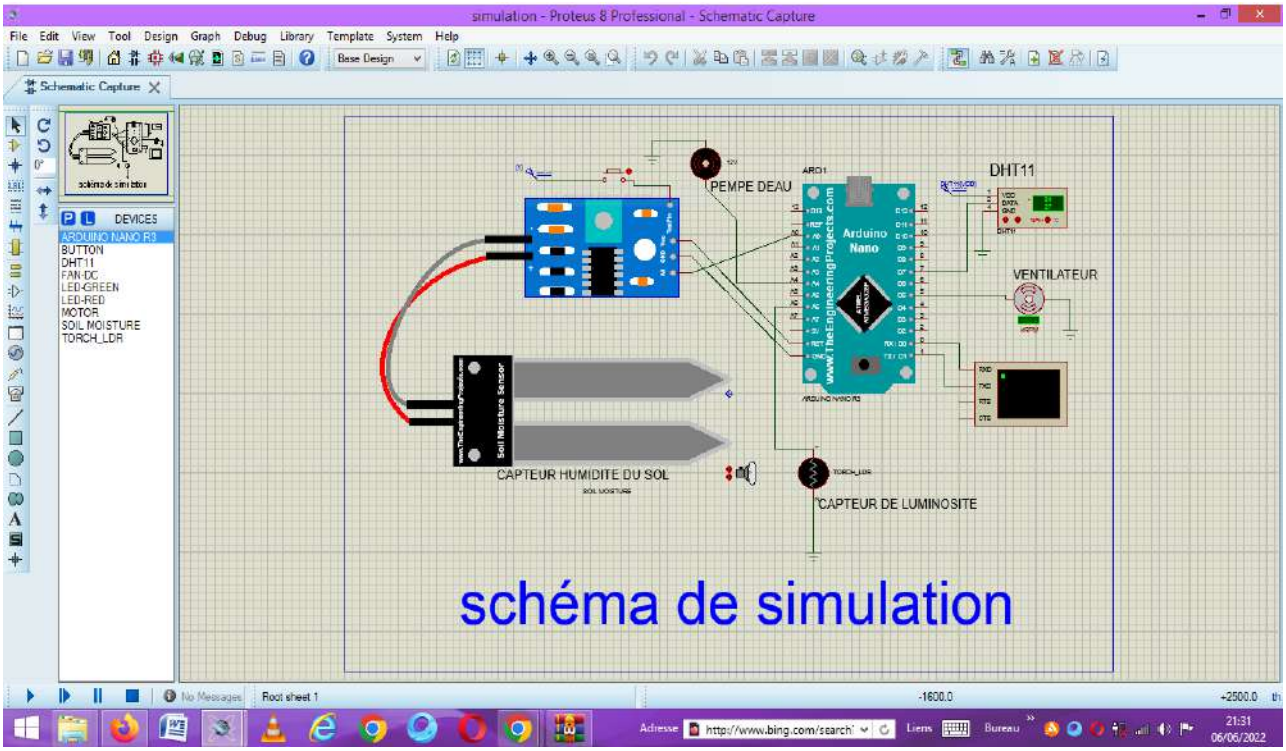


Fig. III.15 : Schéma électrique de simulation

III.7 Résultat de simulation :

La simulation montre que si on mit la valeur de température inférieure à 30°C qu'il capté par le capteur DHT11 la ventillateur s'arrêt, sinon la ventillateur marche. La figure III.16 illustre qu'on a choisi la valeur de température 27°C (Inférieur à 30°C), on remarque que la valeur de ventillateur est 0.00 (ventillateur ne marche pas). Par contre, dans la figure III.17 on fixe la valeur de température 31°C (supérieur à 30°C). On observe que la ventillateur donne la valeur 0.46 (Marche). Par conséquent, on voit que lorsque la température atteindre la valeur de 30°C la ventillateur lance automatiquement.

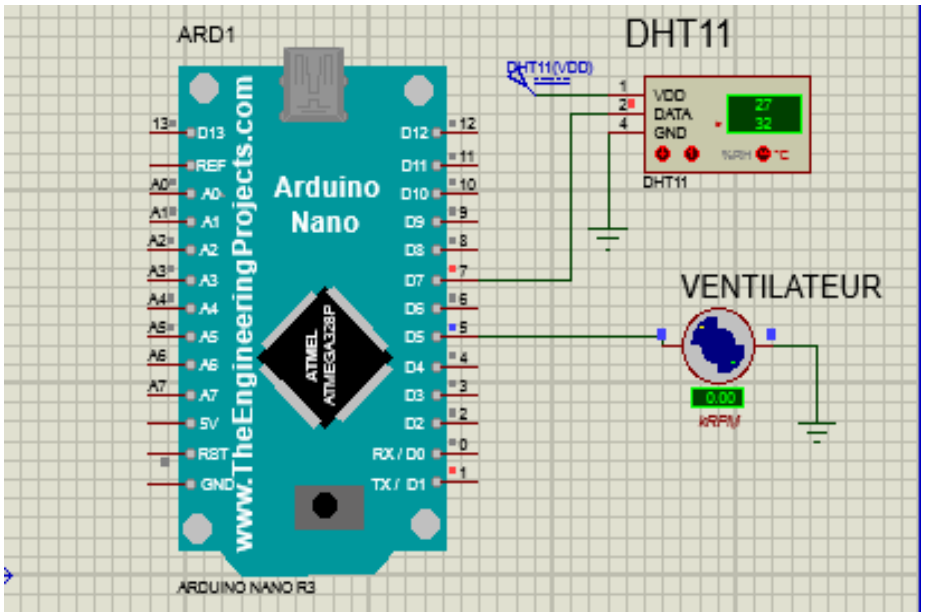


Fig. III.16 : la valeur de température 27°C est la ventillateur 0.00

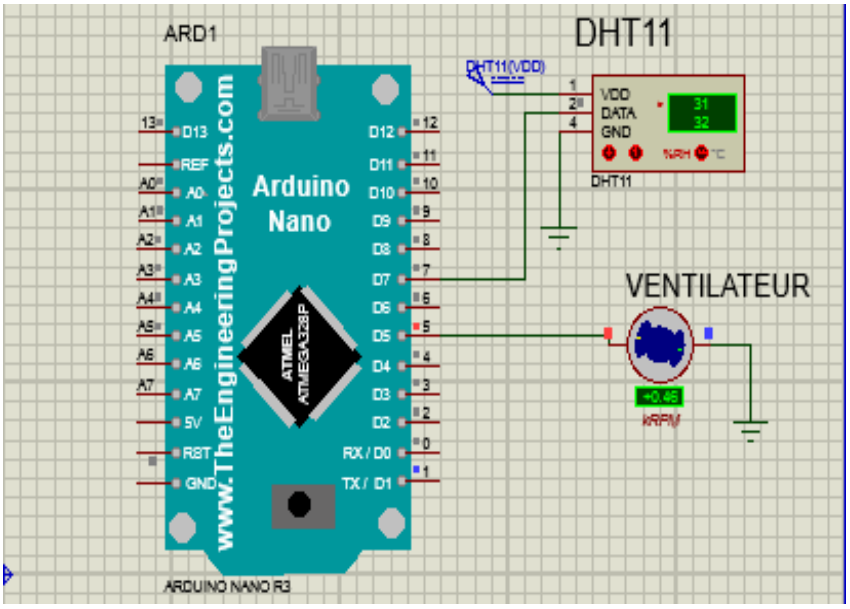


Fig. III.17 : la valeur de température 31°C est la ventillateur 0.46

La figure III.18 montre la simulation de contrôle de composant LED pour donner l'éclairage suffisent du plant. Si la lampe est allumé, le capteur de luminosité LDR capte la lumière, le LED s'éteint automatiquement par Arduino. Si non le LED s'allume.

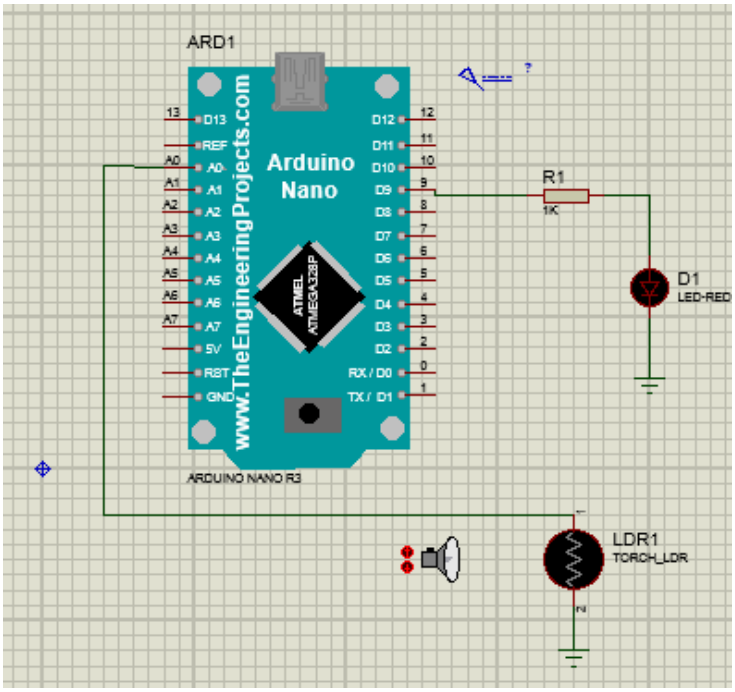


Fig. III.18 : la simulation de contrôle de composant LED

III.8 Réalisation du projet « Smart Garden » :

Dans cette partie on a présenté le schéma éclectique d'un système « smart garden » qui sera convertir en circuit imprimé à la future. Ce schéma est composé de : un Arduino nano, des connecteurs des capteurs J1, J2, et J3, bloc d'alimentation, circuit de la pompe, et circuit de ventilateur et LEDs. le capteur de température et d'humidité de l'air (DHT11) , capteur d'humidité de sol, et capteur de luminosité sont connecté au connecteur J1, J2 et J3.

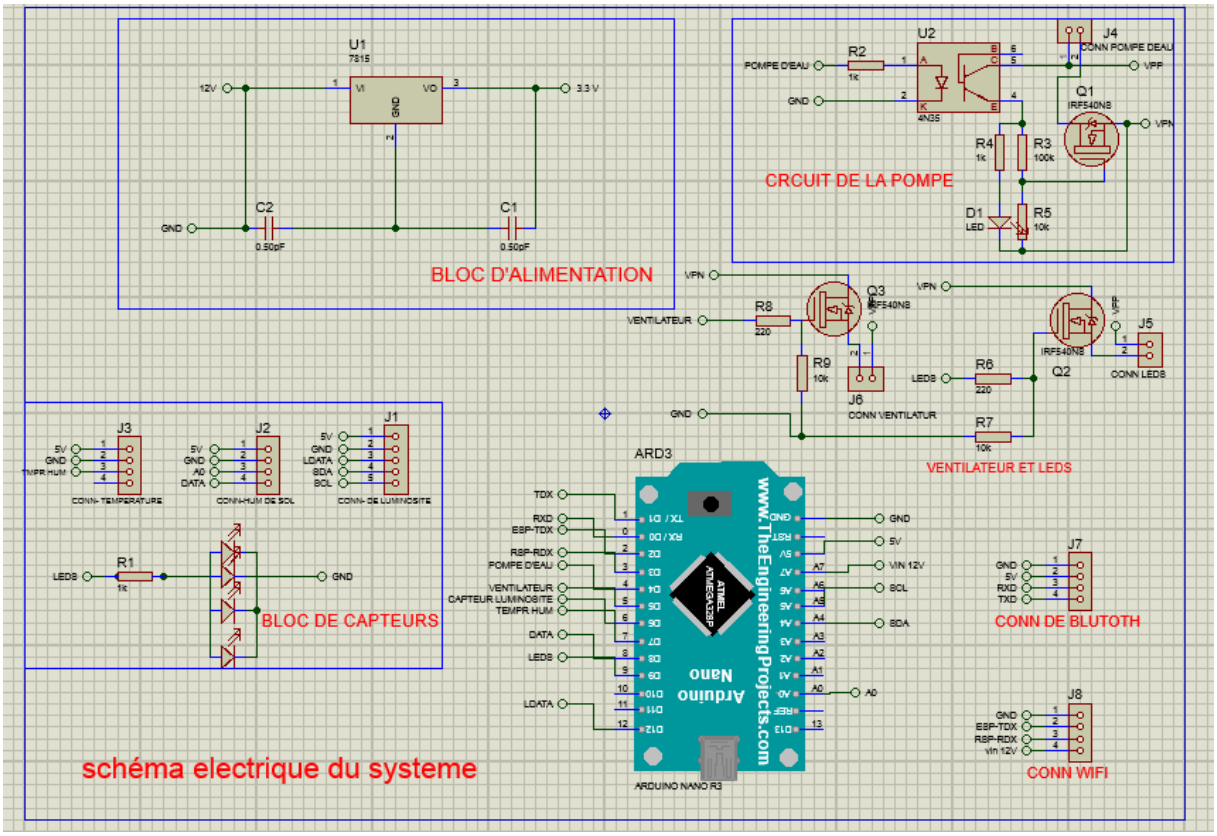


Fig. III.19 : schéma electrique de système par logiciel proteus

La carte de cette simulation en 2D :

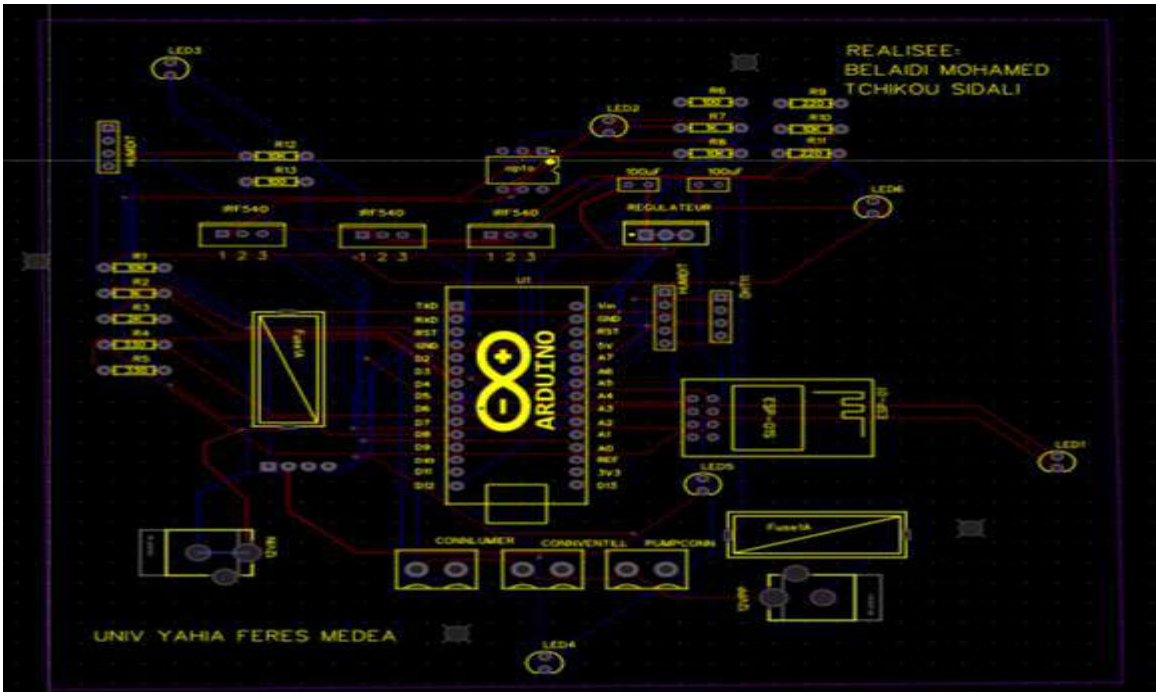


Fig. III.20 : La carte en 2D

La carte de cette simulation en 3D :

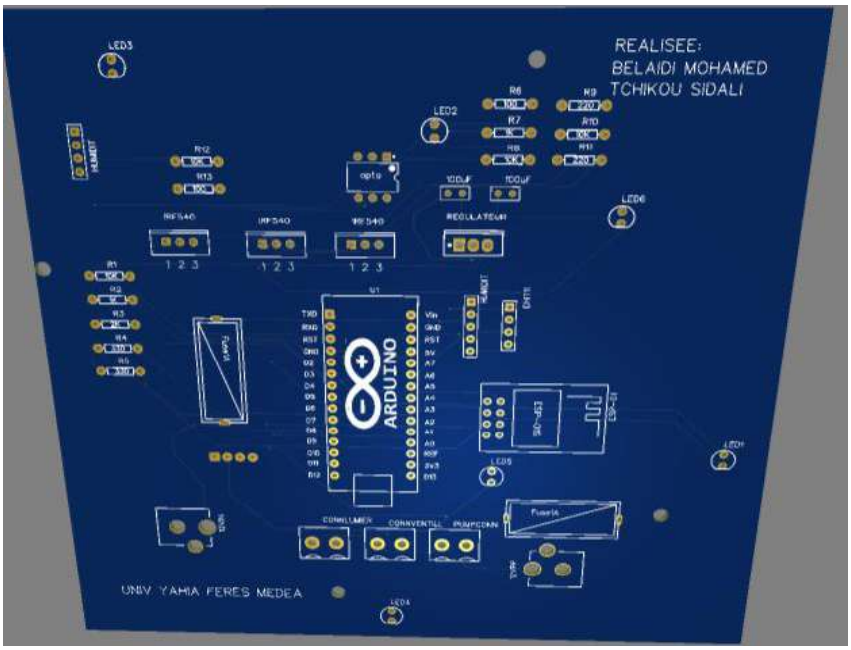


Fig. III.21: La carte en 3D

Conclusion générale

Conclusion générale :

La mise en œuvre du système Smart Garden utilisant l'Internet des objets a été vérifiée pour fonctionner de manière satisfaisante en connectant différents paramètres du sol au cloud et a été contrôlée avec succès à distance via une application mobile. Le système conçu surveille non seulement les données du capteur, telles que l'humidité, la température et luminosité, mais active également d'autres paramètres en fonction des besoins, par exemple, si le niveau d'eau dans le réservoir est réduit à une valeur minimale, l'interrupteur du moteur est allumé automatiquement au niveau d'eau du réservoir atteint la valeur maximale.

Dans cette thèse on a premièrement présenté l'Arduino Nano qui est un élément important de notre projet, en suit on a présenté une généralité sur la « Smart Garden » et les capteurs utilisés. L'objective de notre thèse est de simuler un système de « Smart Garden » par le logiciel proteus, qu'il est basé sur Arduino Nano. Les résultats de notre simulation montrent que les actionneurs comme la ventillateur, LED et pompe à d'eau sont marché automatiquement.

Avec le développement de la technologie des capteurs, le système peut passer au niveau supérieur, ce qui aide les utilisateurs à utiliser leur investissement de manière économique. Si des capteurs de nutriments du sol peuvent être installés, le système peut être modifié pour fournir des engrais au jardin avec précision. Ce système permet d'économiser de la main-d'œuvre et utilise efficacement les ressources en eau disponibles, ce qui conduit finalement à plus de profit. La rétroaction fournie par le système améliorera la mise en œuvre du processus de jardinage.

Annex

A. Les actionneurs

 ventillateur	- Dimensions: 92x92x25mm - tension : 12V DC - Consommation de courant : 0,14A - Vitesse : 3000U/Min - Température d'utilisation : -10...+70°C
 Pompe d'eau	-Dimensions : 20 x 11 x 8.5mm -tension : 12v DC -puissance : 60w -Pression : 0.8 Mpa -Pompage : 5L/min
 led	-Dimensions : 50 x 50 x 10mm -tension : 1.5 V -haute puissance : 117mW 1.2V 20000mcd - type d'alimentation : Câble électrique

B. Code Arduino Nano

```
*/
#include "DHT.h"
// Include the dht library to interface with the DHT sensor
#include <Wire.h>
// IC library to use the light sensor
#include <BH1750.h>
// Import BH1750 library in order to be able to use the light
sensor
//----------------- Define the control pins-----------------
const int Pump=4;
// We've used this pin to control the motor pump
```

```
const int Fan=5;
// Use this pin as a PWM output to control the Fan speed
const int Light=6;
// Use this pin in order to control the LED brightness
const int TempHum=7;
// Input pin for DHT sensor
const int Moisture=8;
// Digital Input pin to read the moisture command signals
const int TestLED=9;
// Use this output pin to test the right soldering on your PCB
by turning on and off the test LEDs
const int Brightness=12;
// Input pin to read the light sensor signals
const int analogMoistue=0;
// Analog Input to read the analog signal from moisture sensor
//-------------------------------------------------------------
#define DHTTYPE DHT11
// DHT 22  (AM2302), AM2321
DHT dht(TempHum, DHTTYPE);
// Temperature and humidity sensor constructor
BH1750 lightMeter;
// Light sensor constructor
char Data='x';
// Store Serial data in this variable
String cmd="";
// Read the full instruction sent from the android app
int flagModeAuto=0;
// flag to activate the auto mode
int sprayCMD=0;
// CMD limit spray
int LightCMD=0;
// CMD limit brightness
int lightLevelPlus=0;
// Variable to control the light brightness
float temperature=0;
// Variable to store the temperature value
float humidity=0;
// Variable to store the humidity value
uint16_t lux=0;
// Variable to read light brightness from the light sensor
//--------------- Start the pin configuration ----------------
void setup()
{
  Wire.begin();
// Initialize the I2C bus (BH1750 library doesn't do this
automatically)
  dht.begin();
// Start the temperature and humidity sensor reading
  lightMeter.begin();
// Start the light sensor reading
```

```
  Serial.begin(9600);
// Set the baudrate up to 9600 BPS to comunicate with the
android app through Bluetooth
  Serial.setTimeout(100);
// Set the time to wait for data before closing the Serial
port (after 100 ms)
  pinMode(Pump,OUTPUT);
  pinMode(Fan,OUTPUT);
  pinMode(Light,OUTPUT);
  pinMode(Moisture,INPUT);
  pinMode(TestLED,INPUT);
  delay(1000);
  digitalWrite(Pump,LOW);
  digitalWrite(Fan,LOW);
  digitalWrite(Light,LOW);
}

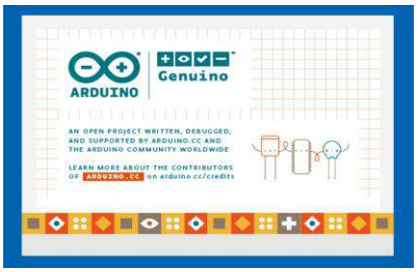
//----------------- Start the Process code ------------------
void loop()
{
  while(Serial.available())
// Read the serial data once available
  {
    delay(10);
    Data=Serial.read();
    cmd+=Data;
  }
  if(cmd=="dt")
// Send the humidity value to the android app
  {
    Serial.print(humidity);
  }
  if(cmd=="dh")
// Send the brightness value to the android app
  {
    Serial.print(lux);
  }
  if(cmd=="db")
// Send the temperature value to the android app
  {
    Serial.print(temperature);
  }
  if(cmd=="o")                                                //
Activate the automatic mode
  {
    flagModeAuto=1;
  }
  if(cmd=="m")                                                //
Disactivate the automatic mode
  {
    flagModeAuto=0;
```

```
  }
  if(flagModeAuto==1)
  {
    autoPump();
    lightBrightness();
    autoFan();
  }
  if(flagModeAuto==0)
  {
    if(cmd=="f")
// Turn ON the FAN
    {
      analogWrite(Fan,255);
    }
    if(cmd=="x")
// Turn OFF the FAN
    {
      analogWrite(Fan,0);
    }
    if(cmd=="l")
// Turn ON the Lights
    {
      analogWrite(Light,255);
    }
    if(cmd=="k")
// Turn OFF the Lights
    {
      analogWrite(Light,0);
    }
    if(cmd=="w")
// Turn ON the Pump
    {
      analogWrite(Pump,255);
    }
    if(cmd=="y")
// Turn OFF the Pump
    {
      analogWrite(Pump,0);
    }
  }
  cmd="";
// Clear the cmd variable to make it available for the next
instruction
  lux = lightMeter.readLightLevel();
// Get the brightness level from the light sensor
  temperature=dht.readTemperature();
// Get the temperature value from the DHT sensor (C)
  humidity=dht.readHumidity();
// Get the humidity value from the DHT sensor (%)
}
//------------ Auto control function for Pump spray ---------
```

```
void autoPump()
  {
    if(analogRead(analogMoistue)<sprayCMD)
    {
      digitalWrite(Pump,HIGH);
      delay(1000);
      digitalWrite(Pump,LOW);
      delay(1000);
    }
  }

//---------- Auto control function for Brightness LED ------
void lightBrightness()
{
  while(lux<LightCMD)
    {
      analogWrite(Light,lightLevelPlus);
// Increase the light brightness
      delay(100);
      lightLevelPlus++;
      lux = lightMeter.readLightLevel();
// Read the light brightness level
    }
}
//--------------- Auto control function for FAN --------------
void autoFan()
{
  if(temperature>30)
  {
    analogWrite(Fan,255);
// Turn ON the fan if the temperature exceed 30C
  }
  else
  {
    analogWrite(Fan,0);
// Turn OFF the fan if the temperature exceed 30C
  }
}
```

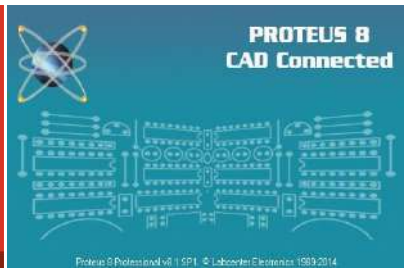
C . logiciel utilisent



IDE Arduino



fritzing



proteus 8

Pour la programmation pour design les schémas pour la simulation

References bibliographiques

[1] M. Suryanegara, A. S. Arifin, M. Asvial, K. Ramli, M. I. Nashiruddin and N. Hayati, "What are the Indonesian Concerns About the Internet of Things (IoT)? Portraying the Profile of the Prospective Market, in IEEE Access, vol. 7, pp. 2957-2968, 2019 doi: 10.1109/ACCESS.2018.2885375.

[2] M. S. Hadi, M. R. Maulana, M. A. Mizar, I. A. E. Zaeni, A. N. Afandi and M. Irvan, "Self Energy Management System for Battery Operated Data Logger Device Based on IoT," 2019 International Conference on Electrical, Electronics and Information Engineering (ICEEIE), Denpasar, Bali, Indonesia, 2019, pp. 133-138, doi: 10.1109/ICEEIE47180.2019.8981406.

[3] Macdonald, Nico (2005-04-18). "Action, Interaction, Reaction". Blueprint. p. 6. Archived from the original on 2003-08-10 . Retrieved 2008-09-23.

[4] Mr Hernando Barrag , pour sa thèse de fin études , avait entrepris le développement d'une carte Arduino, année 2005.

[5] Mr Simon Landrault et Hippolyte Weisslinger, Arduino Nano : Premiers pas en informatique embarquée , Édition du 01 juin 2014.

[6] Thamaraimanalan, T., Vivekk, S.P., Satheeshkumar, G., and Saravanan, P. (2018). Smart garden monitoring system using IOT, Asian Journal of Applied Science and Technology (AJAST), Volume 2, Issue 2, pp 186-192.

[7] . I. Srilikhitha, M. M. Saikumar, N. Rajan, M. L. Neha, and M. Ganesan, "Automatic irrigation system using soil moisture sensor and temperature sensor with microcontroller AT89S52," 2017 Int. Conf. Signal Process. Commun., no. July, pp. 186– 190, 2017

[8] B.Hamidat, Y.Abdeli, '' Etude et conception des capteurs à cristaux Photoniques pour des applications biomédicales'', Université Mohamed Boudiaf - M'sila, 2017.

[9] B.T. Cunningham. Label-free biosensors: Techniques and applications. Cambridge University Press, New York, pp.1-28, 2009.

[10] A. D'Amico and C. Di Natale. "A contribution on some basic definitions of sensors properties". IEEE Sensors Journal, vol.1, n n°3, pp.183-190, 2001.

[11] A .Boughrouat ., « Modification de la sensibilité d'un QCM (Quartz Cristal Microbalance) par le dépôt dessus de films mincesélaborés par plasma à partir de vapeurs de TEOS »Université MENTOURI constantine,2009.

[12] Heather Rhoades, 'How Light Affects The Growth Of A Plant & Problems With Too Little Light', 2018. https://www.gardeningknowhow.com/plantprobl ems/environmental/how-light-affects-thegrowth-of-a-plantproblems-with-too-littlelight.html. [Accessed: 25- Nov- 2018]

[13]. W.Dargie and C.Poellabauer. "Motivation for a Network of Wireless Sensor Nodes," in Fundamentals of Wireless Sensor Networks Theory and Practice, 1st ed. United Kingdom: Wiley, 2010, ch. 1, pp. 3-16.

[14] A. R. Altahtawi, E.Andika, W. N. Harjanto, “Initial Design Development of Wireless Node-based Automatic Irrigation Control Systems and The Internet of Things,” 2018 ISSN: 2085- 2517.