

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التعليم العالي والبحث العلمي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جامعة سعيدة – د. الطاهر مولاي –

Université de Saïda – Dr Tahar Moulay –



Faculté de Technologie

Département d'électronique

**MEMOIRE DE FIN D'ETUDES EN VUE DE L'OBTENTION  
DU DIPLOME DE MASTER EN ELECTRONIQUE  
OPTION: ELECTRONIQUE DES SYSTEMES EMBARQUÉS**

**THEME**

---

**Conception d'un système embarqué pour  
l'agriculture intelligente**

---

Présenté par : **Djemama Mohamed Anwar**

Soutenu le 26 Juin 2025, Devant le jury composé de :

|                    |                                               |           |
|--------------------|-----------------------------------------------|-----------|
| BENSAAD Abdellah   | Maître de conférences à l'Université de Saïda | Président |
| BOUARFA Abdelkader | Maître de conférences à l'Université de Saïda | Examineur |
| CHAMI Nadir        | Maître de conférences à l'Université de Saïda | Encadreur |

# Remerciements

Je remercie Dieu et le remercie pour son succès et son aide dans la réalisation de ce projet.

J'adresse mes sincères remerciements et ma reconnaissance à mon professeur superviseur Dr CHAMI Nadir pour ses précieux conseils, sa patience et son soutien continu tout au long de la réalisation de ce travail.

Je remercie également tous les professeurs du département d'électronique pour toutes les connaissances et connaissances qu'ils nous ont fournies tout au long des années d'études.

Je n'oublie pas de remercier mes collègues et amis pour leur coopération et leur soutien.

Je remercie particulièrement ma famille honorable pour leur soutien constant et leurs encouragements à chaque étape. Sans eux, je n'aurais pas atteint où je suis aujourd'hui.

# Table des Matières

|                       |   |
|-----------------------|---|
| Introduction Générale | 1 |
|-----------------------|---|

## Chapitre 1 : Internet des Objets et Technologie LoRa dans l'Agriculture

|                                                                 |    |
|-----------------------------------------------------------------|----|
| 1.1 Introduction                                                | 2  |
| 1.2 Internet des Objets (IoT)                                   | 2  |
| 1.2.1 Définition                                                | 2  |
| 1.2.2 Architecture de l'IoT                                     | 2  |
| 1.2.2.1 Couche de détection                                     | 3  |
| 1.2.2.2 Couche réseau                                           | 3  |
| 1.2.2.3 Couche de traitement des données                        | 4  |
| 1.2.2.4 Couche application                                      | 4  |
| 1.2.3 Avantages de l'IoT.                                       | 5  |
| 1.2.4 Inconvénients de l'IoT                                    | 5  |
| 1.2.5 Domaines d'application de l'IoT                           | 5  |
| 1.3 Évolution de l'Agriculture vers l'Agriculture Intelligente  | 6  |
| 1.3.1 Agriculture 1.0 : Agriculture traditionnelle              | 6  |
| 1.3.2 Agriculture 2.0 : Agriculture mécanique                   | 6  |
| 1.3.3 Agriculture 3.0 : Agriculture numérique                   | 7  |
| 1.3.4 Agriculture 4.0 : Agriculture intelligente                | 8  |
| 1.4 Technologies Modernes utilisées en Agriculture Intelligente | 9  |
| 1.4.1 Système de positionnement global                          | 9  |
| 1.4.2 Canules de détection dans l'agriculture intelligente      | 9  |
| 1.4.3 Robotique et automatisation dans l'agriculture verticale  | 10 |
| 1.4.4 Drones (Smart Planes)                                     | 11 |
| 1.5 LPWAN : Réseaux bas débit et longue portée                  | 11 |
| 1.5.1 Caractéristiques principales                              | 12 |
| 1.6 LoRaWAN                                                     | 12 |
| 1.6.1 Applications agricoles de LoRaWAN                         | 12 |
| 1.6.1.1 Contrôle de l'irrigation                                | 12 |
| 1.6.1.2 Santé du sol                                            | 12 |
| 1.6.1.3 Gestion des actifs                                      | 13 |
| 1.6.2 LoRa                                                      | 13 |
| 1.7 Conclusion                                                  | 14 |

## Chapitre 2 : Matériels et Logiciels du Système d'Agriculture Intelligente

|                           |    |
|---------------------------|----|
| 3.1 Introduction          | 16 |
| 3.2 Logiciel utilisé      | 16 |
| 3.2.1 Arduino IDE         | 16 |
| 3.2.2 Choix de l'Arduino. | 17 |
| 3.3 Microcontrôleurs      | 18 |
| 3.3.1 ESP32               | 18 |
| 3.3.2 Avantages           | 20 |

|                                         |    |
|-----------------------------------------|----|
| 3.3.3 Inconvénients                     | 20 |
| 3.3.4 Caractéristiques..                | 20 |
| 3.3.5 Caractéristiques supplémentaires  | 21 |
| 3.4 Module de communication LoRa SX1278 | 22 |
| 3.4.1 Caractéristiques.                 | 22 |
| 3.5 Capteurs.                           | 23 |
| 3.5.1 Capteur DHT11                     | 23 |
| 3.5.2 Capteur d'humidité du sol FC-28   | 24 |
| 3.6 Actionneurs                         | 25 |
| 3.6.1 Mini pompe à eau                  | 25 |
| 3.6.2 Relais.                           | 26 |
| 3.6.3 Écran OLED SSD1306                | 27 |
| 3.7 Application Mobile – Blynk          | 28 |
| 3.7.1 Blynk.Console                     | 29 |
| 3.8 Conclusion                          | 29 |

## Chapitre 3 : Réalisation et Mise en Œuvre du Système

|                                                     |    |
|-----------------------------------------------------|----|
| 4.1 Introduction                                    | 30 |
| 4.2 Définition et importance de l'irrigation        | 30 |
| 4.3 Application de l'IoT à l'irrigation             | 30 |
| 4.4 Système d'irrigation intelligent proposé        | 31 |
| 4.5 Schéma synoptique du système.                   | 31 |
| 4.6 Architecture générale du système                | 32 |
| 4.7 Fonctionnement du système                       | 33 |
| 4.8 Différentes parties du système                  | 33 |
| 4.8.1 Côté émission                                 | 33 |
| 4.8.2 Côté réception                                | 35 |
| 4.9 Configuration et création d'un projet Blynk IoT | 38 |
| 4.10 Résultat de la réalisation avec Blynk IoT      | 39 |
| 4.11 Conclusion                                     | 40 |
| Conclusion Générale.                                | 41 |
| Annexe                                              | 42 |
| Références Bibliographiques                         | 43 |

# Table des Figures

| N° de Figure | Titre de la Figure                                       | Page |
|--------------|----------------------------------------------------------|------|
| Figure 1.1   | Internet des Objets                                      | 2    |
| Figure 1.2   | Architecture de l'IoT                                    | 3    |
| Figure 1.3   | Infrastructure typique de l'IoT                          | 4    |
| Figure 1.4   | Domaines d'utilisation de l'IoT                          | 6    |
| Figure 1.5   | Agriculture 1.0                                          | 7    |
| Figure 1.6   | Agriculture 2.0                                          | 7    |
| Figure 1.7   | Agriculture 3.0                                          | 8    |
| Figure 1.8   | Agriculture 4.0                                          | 9    |
| Figure 1.9   | Système de Positionnement Global (GPS)                   | 10   |
| Figure 1.10  | Canules de Détection dans l'Agriculture Intelligente     | 11   |
| Figure 1.11  | Robotique et Automatisation dans l'Agriculture Verticale | 12   |
| Figure 1.12  | Drones agricoles                                         | 13   |
| Figure 1.13  | Architecture de LoRa                                     | 14   |
| Figure 2.1   | Logo de l'Arduino IDE                                    | 16   |
| Figure 2.2   | ESP32                                                    | 18   |
| Figure 2.3   | Broches d'E/S de l'ESP32                                 | 19   |
| Figure 2.4   | Module LoRa SX1278                                       | 22   |
| Figure 2.5   | Capteur DHT11                                            | 23   |
| Figure 2.6   | Capteur d'humidité du sol FC-28                          | 24   |
| Figure 2.7   | Mini Pompe                                               | 25   |
| Figure 2.8   | Module Relais 5V                                         | 26   |
| Figure 2.9   | Broches du Module Relais                                 | 27   |
| Figure 2.10  | Écran OLED SSD1306                                       | 27   |
| Figure 2.11  | Icône de l'application Blynk                             | 28   |
| Figure 3.1   | Schéma synoptique du système                             | 31   |
| Figure 3.2   | Câblage de l'ESP32 avec DHT11                            | 33   |
| Figure 3.3   | Schéma du câblage de l'ESP32 avec FC-28                  | 34   |
| Figure 3.4   | Schéma final de l'émetteur (Fritzing)                    | 35   |
| Figure 3.5   | Câblage du module LoRa + ESP32 (récepteur)               | 36   |
| Figure 3.6   | Branchement du Relais avec ESP32                         | 37   |
| Figure 3.7   | Interface de l'application Blynk                         | 39   |

---

## Introduction générale

---

## Introduction Générale

Ces dernières années, les technologies de l'information et de la communication ont connu une évolution rapide, touchant de nombreux secteurs tels que la santé, l'industrie, l'énergie et l'éducation. Le domaine de l'agriculture n'échappe pas à cette transformation, en raison des nombreux défis auxquels il est confronté : pénurie d'eau, dégradation des sols, changement climatique, baisse de rendement et croissance démographique. Face à ces enjeux, l'intégration de solutions technologiques innovantes est devenue essentielle pour améliorer la productivité, optimiser les ressources et assurer une agriculture durable.

L'agriculture intelligente, également connue sous le nom d'agriculture 4.0, repose sur l'utilisation de capteurs, d'actionneurs, de systèmes de communication sans fil et de plateformes de contrôle à distance. Elle permet de surveiller les conditions environnementales (température, humidité, état du sol) en temps réel et de piloter automatiquement ou manuellement les interventions agricoles (irrigation, ventilation, etc.).

Dans le cadre de ce travail, nous avons conçu et réalisé un système embarqué pour l'agriculture intelligente, basé sur l'Internet des Objets (IoT). Le système utilise un microcontrôleur ESP32 pour collecter des données à partir de capteurs de température, d'humidité de l'air et du sol, puis les transmettre à distance via un module LoRa SX1278 à faible consommation énergétique. Les données sont ensuite visualisées en temps réel à l'aide de l'application mobile Blynk, ce qui permet à l'utilisateur de surveiller l'état de son exploitation agricole et de déclencher des actions selon les besoins.

Le choix du protocole LoRa s'explique par sa capacité à transmettre des données sur de longues distances, avec une consommation énergétique minimale, ce qui en fait une solution idéale pour les zones rurales éloignées, souvent mal desservies en électricité ou en réseau Internet.

L'objectif de ce projet est donc de proposer une solution simple, économique et fiable destinée aux petites et moyennes exploitations agricoles. Le système peut être utilisé comme base pour développer des versions plus évoluées, intégrant par exemple l'intelligence artificielle, les énergies renouvelables ou la gestion autonome.

Ce mémoire est structuré en trois chapitres :

- Le premier chapitre présente le cadre théorique du projet, en introduisant les concepts clés liés à l'IoT, à l'agriculture intelligente, au protocole LoRa et aux technologies associées.
- Le deuxième chapitre décrit le matériel utilisé et les composants sélectionnés pour la réalisation du système.
- Le troisième chapitre est consacré à la mise en œuvre pratique, incluant les schémas de câblage, les tests réalisés, et les codes de programmation utilisés.

---

# **Chapitre 1**

## **Internet des Objets et Technologie LoRa dans l'Agriculture**

---



## 1.1 Introduction

L'essor de l'agriculture moderne impose l'intégration de technologies innovantes pour répondre aux exigences croissantes en matière de productivité, de durabilité et de gestion des ressources. Parmi ces technologies, l'Internet des Objets (IoT) occupe une place prépondérante, en permettant la collecte, l'analyse et l'exploitation en temps réel des données issues de capteurs connectés. Couplée à des technologies de communication sans fil comme LoRa (Long Range), l'IoT permet de mettre en place des systèmes agricoles intelligents, capables de fonctionner même dans des zones éloignées et mal desservies en réseau [1][2].

## 1.2 Internet des Objets (IoT)

### 1.2.1 Définition

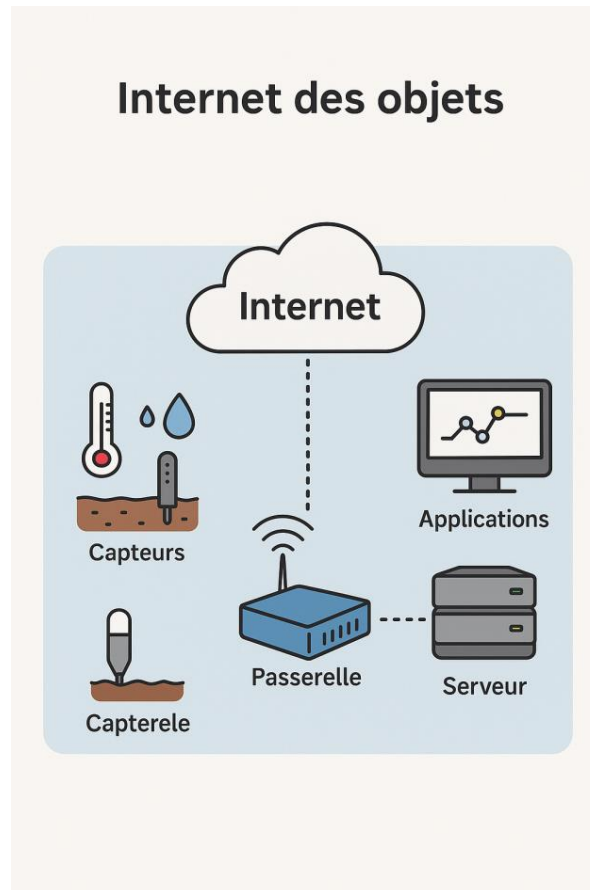
L'Internet des Objets (IoT) désigne un réseau d'objets physiques interconnectés (capteurs, actionneurs, appareils, etc.) capables de collecter et d'échanger des données via Internet. Ces objets sont dotés de technologies telles que le Wi-Fi, LoRa, Bluetooth ou ZigBee, leur permettant d'interagir entre eux et avec leur environnement, sans intervention humaine directe.



Figure 1.1 : Internet des Objets

### 1.2.2 Architecture de l'IoT

Les quatre niveaux de l'architecture de l'Internet des objets sont la couche applicative, la couche de traitement des données, la couche réseau et la couche de détection.



**Figure 1.2 : Architecture de l'IoT**

#### **1.2.2.1 Couche de détection (Sensing Layer) :**

Dans l'architecture de l'Internet des objets, la couche de détection est la couche initiale et est chargée de collecter des données provenant de diverses sources. Elle est composée de capteurs et d'actionneurs répartis dans l'environnement afin de recueillir des données sur diverses caractéristiques physiques, telles que la température, l'humidité, la luminosité et le son. Ces appareils sont connectés à la couche réseau via des communications filaires ou sans fil [2] [3].

#### **1.2.2.2 Couche réseau (Network Layer) :**

Dans une conception de l'Internet des objets, la couche réseau est chargée de faciliter la connectivité et la communication entre les composants du système. Il se compose des technologies et des protocoles qui permettent aux appareils de se connecter et de parler à l'ensemble d'Internet ainsi qu'entre eux. Le WiFi, le Bluetooth, le Zigbee et les réseaux cellulaires comme la 4G et la 5G sont quelques exemples de technologies réseau souvent utilisées dans l'Internet des objets. Afin de prévenir les accès non désirés, la couche réseau peut également contenir des mesures de sécurité comme le cryptage et l'authentification, ainsi

que des passerelles et des routeurs qui servent d'intermédiaires entre les appareils et l'Internet en général [2] [3].

### 1.2.2.3 Couche de traitement des données (Data processing Layer) :

La couche de traitement des données (ou Data Processing Layer) est un composant central dans l'architecture d'un système informatique ou d'un système embarqué, notamment dans les systèmes IoT ou Big Data. Elle est chargée de transformer, analyser et structurer les données brutes collectées par la couche de perception (capteurs, dispositifs d'acquisition, etc.) avant leur stockage ou leur utilisation.

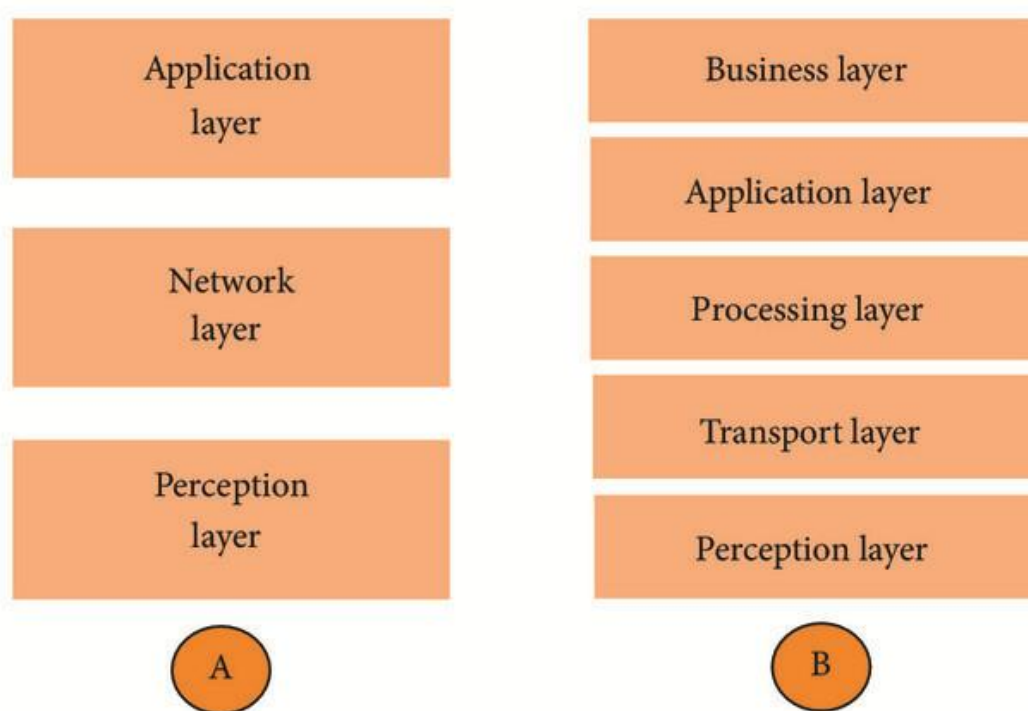


Figure 1.3 : Infrastructure typique de l'IoT

### 1.2.2.4 Couche application (Application Layer):

Dans l'architecture de l'Internet des objets, la couche applicative est la plus haute couche qui communique directement avec les utilisateurs. Elle est chargée d'offrir des fonctions et des interfaces conviviales qui permettent aux consommateurs d'accéder et de gérer les appareils IoT. Divers logiciels et applications, y compris des portails en ligne, des applications pour smartphones et d'autres interfaces utilisateur, sont inclus dans cette couche afin d'interagir avec l'infrastructure IoT sous-jacente. Les services de middleware sont également inclus, ce qui permet un partage et une communication fluides des données entre divers systèmes et appareils IoT. De plus, la couche applicative possède des outils de traitement et d'analyse qui permettent d'examiner les données et de les transformer en connaissances pertinentes. Cela pourrait contenir des fonctionnalités d'analyse sophistiquées

telles que des outils de visualisation de données et des algorithmes d'apprentissage automatique [2] [3].

### 1.2.3 Avantages de l'IoT.

Parmi les avantages de l'IoT est :

- Surveillance en temps réel et prise de décision rapide.
- Réduction des coûts de maintenance grâce à la détection précoce des anomalies.
- Optimisation de l'utilisation des ressources (eau, électricité, engrais...).
- Amélioration de la qualité de service et de la satisfaction client [3].

### 1.2.4 Inconvénients de l'IoT.

- Vulnérabilité face aux cyberattaques en raison du grand nombre de dispositifs connectés.
- Manque de standardisation dans les protocoles de communication et de sécurité.
- Complexité dans l'intégration de systèmes hétérogènes.
- Dépendance à la connectivité réseau et à l'alimentation électrique [4].

### 1.2.5 Domaines d'application de l'IoT

L'Internet des objets (IoT) trouve aujourd'hui de nombreuses applications dans divers secteurs. Dans les villes intelligentes (Smart Cities), l'IoT permet une gestion optimisée de l'éclairage public, de la circulation routière et de la collecte des déchets [5]. Dans le domaine de la santé, il est utilisé pour la télésurveillance des patients ainsi que dans les dispositifs médicaux connectés. L'industrie 4.0 exploite l'IoT pour la maintenance prédictive des équipements et l'automatisation des processus industriels. Le secteur du transport bénéficie de l'IoT pour la gestion de flotte de véhicules et l'optimisation des trajets.

Enfin, dans l'agriculture intelligente, l'IoT facilite l'irrigation automatisée ainsi que la surveillance des cultures et du bétail [6].

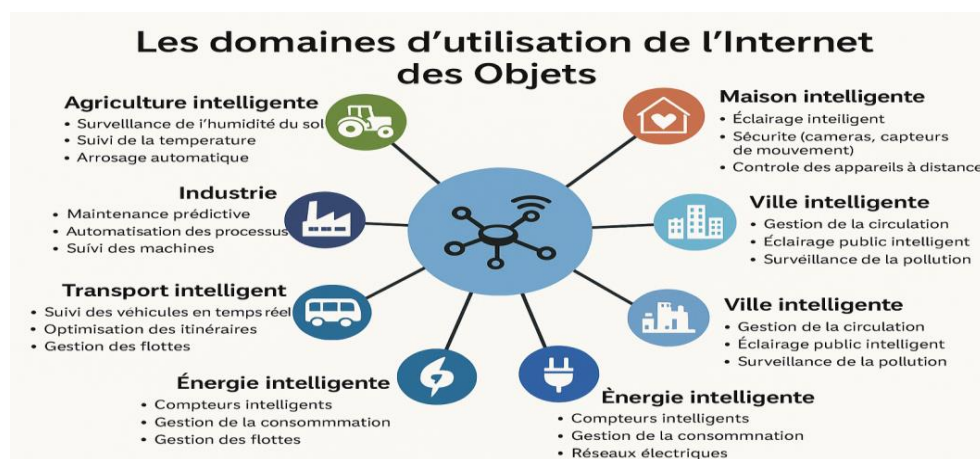


Figure 1.4 : Domaines d'utilisation de l'IoT

## 1. 3 Évolution de l'Agriculture vers l'Agriculture Intelligente

L'agriculture a connu plusieurs phases d'évolution, allant des méthodes traditionnelles à l'intégration des technologies avancées. L'agriculture 1.0 se caractérisait par l'usage de techniques rudimentaires, avec un travail essentiellement manuel ou animal. Avec l'agriculture 2.0, la mécanisation a fait son apparition, marquée par l'utilisation de tracteurs et de moteurs, ainsi qu'un recours accru aux engrais et aux pesticides. L'agriculture 3.0 a introduit une automatisation partielle, avec des systèmes d'irrigation contrôlée et l'utilisation de données numériques pour améliorer la production. Enfin, l'agriculture 4.0 représente l'ère actuelle, où les technologies telles que les capteurs, les drones, l'intelligence artificielle, l'Internet des objets (IoT) et le big data sont intégrées pour permettre une gestion intelligente, efficace et durable des exploitations agricoles [7].

### 1.3.1 Agriculture 1.0 : Agriculture traditionnelle

À ses débuts, l'agriculture était pratiquée selon des méthodes primitives entièrement basées sur le travail humain et animal, avec pour objectif principal de fournir de la nourriture aux humains et d'élever du bétail. Des outils simples tels que faucilles et houes étaient utilisés, et le travail agricole était presque entièrement fait à la main. À ce stade, la productivité est restée faible en raison de l'indisponibilité de machines et de technologies modernes.



Figure 1.5 : Agriculture 1.0

### 1.3.2 Agriculture 2.0 : Agriculture mécanique

La révolution industrielle du XIXe siècle a commencé à changer le visage de l'agriculture avec l'introduction de nouveaux équipements tels que les moteurs à vapeur, et plus tard l'utilisation généralisée des tracteurs et autres machines agricoles. Les engrais chimiques et les pesticides ont également été largement utilisés pour stimuler la production. Bien que cette



période ait accru l'efficacité et la productivité des opérations agricoles, elle a également entraîné des impacts environnementaux négatifs tels que la pollution du sol et de l'eau et la sur exploitation des ressources naturelles et de l'énergie.



**Figure 1.6 : Agriculture 2.0**

### 1.3.3 Agriculture 3.0 : agriculture numérique

Au XXe siècle, avec le développement des ordinateurs et de l'électronique, une nouvelle phase plus sophistiquée de l'agriculture est apparue. L'agriculture 3.0 repose sur l'utilisation de machines automatisées, de robots et de systèmes d'irrigation de précision. L'utilisation de produits chimiques a été réduite, l'utilisation de nutriments a été améliorée et des techniques avancées de lutte antiparasitaire ont été utilisées. Cette initiative a également contribué à améliorer la gestion agricole, à réduire les déchets et à accroître l'efficacité.



**Figure 1.7 : Agriculture 3.0**

### 1.3.4 Agriculture 4.0 : Agriculture intelligente

L'agriculture 4.0 est la révolution contemporaine de l'agriculture, qui s'appuie sur :

- Internet des objets (IdO)
- L'intelligence artificielle
- Analyse de données volumineuses
- Cloud Computing
- Capteurs intelligents

Ces déterminations sont destinées à surveiller les cultures, le sol et le temps en temps réel, permettant aux agriculteurs de les activer. Les dispositifs de surveillance des plantes et du sol sont utilisés avec précision et détectent tout problème à l'étape de détection, ce qui contribue à améliorer la qualité des légumes.

Les outils agricoles intelligents ont également aidé la direction et les gestionnaires à se déplacer davantage, réduisant ainsi le besoin de main-d'œuvre manuelle et d'efficacité dans l'agriculture. Les systèmes d'irrigation intelligents, comme le SIWM, s'appuient sur des modèles d'intelligence artificielle pour éliminer l'eau de l'eau.

L'agriculture intelligente confirme un rôle clé dans la lutte contre le changement climatique et alimentaire, en particulier dans les pays en développement qui affectent la sécurité. Cette technologie permet de produire des produits agricoles tout en réduisant les impacts environnementaux et la consommation de ressources [8].



Figure 1.8 : Agriculture 4.0



## 1.4 Technologies Modernes Utilisées en Agriculture Intelligente

### 1.4.1 Système de positionnement global (GPS) :

Il s'agit d'un système précis qui enregistre les informations de latitude, longitude et altitude. Les satellites GPS transmettent des signaux, ce qui permet aux récepteurs de calculer instantanément leur position et de fournir des informations continues en mouvement. Des renseignements exacts sur l'emplacement permettent aux agriculteurs de localiser avec précision les données sur le terrain, comme la présence d'organismes nuisibles, le type de sol, les mauvaises herbes et d'autres obstacles. Ce système facilite la localisation des différents champs, permettant d'appliquer les intrants nécessaires tels que semences, engrais, herbicides, pesticides et eau à un champ spécifique [8].



Figure 1.9 : Système de Positionnement Global (GPS)

### 1.4.2 Canules de détection dans l'agriculture intelligente :

Afin de comprendre avec précision l'état du sol et des plantes, nous utilisons une gamme de méthodes telles que les rayons lumineux, les ondes électromagnétiques, la conduction électrique et même les ultrasons. Ces techniques nous aident à connaître la structure du sol, le niveau d'humidité, la température, le rapport des nutriments et même l'état de la végétation. Grâce aux techniques de télédétection, nous pouvons distinguer les espèces de cultures, identifier les ravageurs et les mauvaises herbes, détecter tout signe de stress ou de faiblesse dans le sol ou dans les plantes, et suivre également les conditions de sécheresse.

La santé des plantes dépend de nombreux facteurs tels que l'humidité du sol, la quantité de lumière, la disponibilité des nutriments, les précipitations et même la couleur des feuilles. Pour garder les plantes dans leur meilleur état, nous contrôlons la température et l'éclairage, et économisons de l'eau et de l'énergie grâce à un système d'irrigation précis. Les capteurs jouent ici un rôle important ; ils mesurent ces facteurs, et si l'un d'entre eux dépasse la limite requise, il envoie une notification au microcontrôleur afin d'intervenir et d'ajuster à nouveau la situation.



Ces capteurs mesurent des choses comme la température, l'humidité, la qualité du sol, le débit d'air, l'emplacement, le dioxyde de carbone, la pression et la lumière. Il convient bien à l'agriculture car il est puissant, facile à transporter, consomme peu d'énergie et stocke bien les données. Grâce à la disponibilité de capteurs sans fil, nous pouvons recueillir des données importantes sur les cultures plus facilement et avec plus de précision. Nous pouvons les installer dans des outils agricoles modernes ou même dans de grandes machines, au besoin [8].



**Figure 1.10 : Canules De Détection Dans Agriculture Intelligente**

### **1.4.3 Robotique et automatisation dans l'agriculture verticale :**

La robotique et les systèmes d'automatisation jouent un rôle important dans le changement de la façon dont les tâches sont effectuées dans l'agriculture verticale. Ils sont utilisés dans presque tout, de la plantation à la récolte en passant par l'emballage. L'utilisation de ces technologies augmente l'efficacité, réduit la dépendance au travail autant que possible et contribue à réduire le gaspillage.

Cependant, malgré leurs avantages importants, ils font face à des défis tels que le coût élevé de l'installation initiale et la complexité des systèmes d'actionnement et de contrôle [9].



**Figure 1.11 : Robotique et Automatisation dans l'agriculture verticale**

### 1.4.3 Smart Planes (drones) :

Dans le passé, un agriculteur s'appuyait sur sa vue et son expérience pour inspecter ses terres. Aujourd'hui, il a une façon de voir plus de lui, d'en haut, avec une précision étonnante. Les drones agricoles, que certains considèrent comme des petits avions, sont en fait des outils avancés qui recueillent des informations précises sur la Terre et les analysent à l'aide de technologies qui n'étaient pas disponibles il y a quelques années seulement.

L'information ne reste que des images. Le drone analyse tout pendant qu'il est dans les airs, ou stocke des données pour les décomposer plus tard. Celui qui l'analyse est un programme intelligent qui comprend où se trouve le problème et suggère une solution. Le résultat? Des décisions agricoles intelligentes qui économisent l'effort et l'eau et augmentent la production.

Plus précisément, il utilise un système GPS appelé RTK, pas comme le GPS ordinaire, mais beaucoup plus précis – au point qu'il connaît sa position en centimètres! Cela lui donne la capacité de dessiner une carte détaillée qui montre exactement où le sol est sec ou où il a besoin de fertilisation. [10]



Figure 1.12 : Drones agricoles

## 1.5 LPWAN : Réseaux bas débit et longue portée

Les LPWAN (Low Power Wide Area Networks) sont des réseaux de communication sans fil conçus pour transmettre de petites quantités de données sur de longues distances, avec une consommation énergétique minimale. Ils sont idéaux pour les applications IoT, en particulier en milieu rural [10].

### **1.5.1 Caractéristiques principales**

Les LPWAN se caractérisent par leurs capacités de transmettre des informations sur une grande distance (jusqu'à 50 km à l'air libre), ils offrent une excellente pénétration, sont capables de traverser des bâtiments et même de travailler sous terre, ils consomment 300 fois moins d'énergie qu'une puce GSM, ils peuvent fonctionner sur batterie pendant une longue durée (plus de 10 ans), leur coût de production est très faible, quelques euros seulement [11]

## **1.6 LoRawan**

La spécification LoRaWAN est une spécification de réseau étendu à faible consommation d'énergie (LPWAN) conçue pour connecter sans fil des appareils alimentés par batterie à Internet dans les réseaux régionaux, nationaux ou mondiaux, ciblant les exigences IoT critiques telles que les communications bidirectionnelles, la sécurité, la mobilité et les services locaux. LoRaWAN fonctionne sur des bandes de fréquences radio sans licence et est connu pour ses capacités de couverture à longue portée (jusqu'à 15 kilomètres dans les zones rurales), sa faible consommation d'énergie et la possibilité de connecter un grand nombre d'appareils à un seul réseau. L'architecture comprend des périphériques, des passerelles, des serveurs réseau et des serveurs d'applications pour fournir un réseau flexible et évolutif pour une variété d'applications IoT [12].

### **1.6.1 Applications agricoles de LoRaWAN**

#### **1.6.1.1 Contrôle de l'irrigation**

Les agriculteurs peuvent enregistrer avec précision les précipitations et autres conditions météorologiques, et mettre en place des alarmes d'inondation et d'autres alertes en cas de changements dans la qualité de l'eau ou d'utilisation excessive de pesticides [13].

#### **1.6.1.2 Surveillance du bétail:**

Les agriculteurs peuvent mieux surveiller les conditions du bétail, telles que la température, l'oestrus, les maladies, la productivité et l'emplacement, ce qui permet de mieux prévenir la perte ou le vol du bétail.

Les agriculteurs de surveillance peuvent maintenant stocker le stockage, recevoir des alarmes pour les portes et l'équipement et un meilleur contrôle et contrôle de la qualité [13].

#### **1.6.1.3 Santé du sol:**

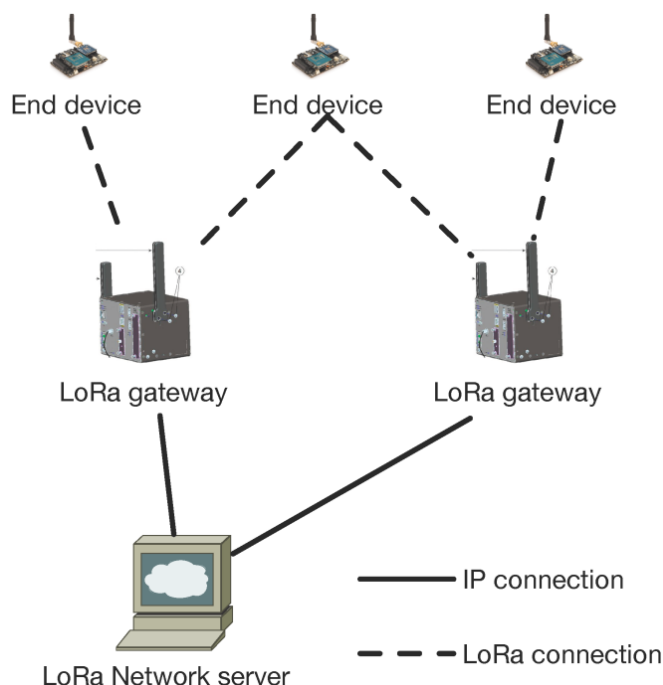
les agriculteurs peuvent désormais planifier et appliquer la bonne quantité d'eau dans les cultures, réduire les déchets et les coûts [13].

#### 1.6.1.4 Gestion des actifs:

Les agriculteurs peuvent comparer des superficies, ajuster la fertilisation, examiner les tendances passées, suivre la qualité du sol de la surface aux racines et améliorer la gestion des cultures à long terme [13].

#### 1.6.2 LoRa:

LoRa, qui signifie "Long String", est un système de communication sans fil longue, faisant la promotion de l'alliance lora. Ce système a pour but d'utiliser de l'équipement à batterie longue, où la consommation d'énergie est d'une importance primordiale. Lora peut généralement se référer à deux couches distinctes : (i) une couche physique utilisant la technique de production des propriétés de la production des propriétés (CSS); et (ii) un protocole mac (loRawan), bien que le système de communication comprend également des approches spécifiques développées par Semtech, permettant des communications le long, faible et faible puissance. Fonctionne dans les bandes ISM 433, 868 ou 915 MHz, selon la région où il se trouve. La charge de chaque transmission peut varier de 2 à 255 octets, et le flux de données peut atteindre jusqu'à 50 kbps lors de l'utilisation de la collecte des canaux. La technique du modèle est une technologie détenue par Semtech. LoRawan offre un accès moyen Le Hearnism de CARRICULTM, permet à de nombreuses extrémités de communiquer avec une porte à modulation lora. Bien que la modélisation lora soit le propriétaire, loRawan est un niveau ouvert en dessous du développement de la bien [14].



**Figure 1.13 : L'Architecture de LoRa**

## **1.7 Conclusion:**

Ce chapitre a exposé les fondements technologiques sur lesquels repose notre projet. L'IoT et les réseaux LPWAN, notamment LoRa et LoRaWAN, permettent de relever les défis de l'agriculture moderne en assurant une surveillance à distance efficace et peu coûteuse. Leur intégration dans des systèmes embarqués ouvre la voie à des solutions agricoles intelligentes, accessibles même dans les zones les plus reculées.

---

## **Chapitre 2**

# **Matériels et Logiciels du Système d'Agriculture Intelligente**

---



## 2.1 Introduction

Dans ce chapitre, nous allons en apprendre davantage sur les différents éléments que nous avons utilisés pour compléter le projet, que ce soit en termes de programmation ou en termes de pièces électroniques. Nous avons d'abord commencé avec l'IDE Arduino, qui est le programme que nous avons utilisé pour écrire et télécharger du code sur la carte ESP32.

Après cela, nous parlerons des microcontrôleurs, en particulier ESP32, et nous expliquerons leurs caractéristiques et fonctionnalités, et nous les comparerons à une autre unité célèbre, l'ESP8266.

Ensuite, nous aborderons le module de communication LoRa SX1278, qui est responsable de l'envoi et de la réception de données entre les appareils, et nous mentionnerons ses caractéristiques.

Enfin, nous parlerons de capteurs tels que le DHT11 pour la température et l'humidité, et le capteur d'humidité du sol FC-28, ainsi que de certains éléments qui exécutent des commandes telles que la pompe, le relais, le ventilateur et l'écran OLED, et nous expliquerons le rôle de chaque élément dans le système

## 2.2 Logiciel Utilisé:

### 2.2.1 Arduino IDE :

Arduino est une plate-forme électronique open source, basée sur un matériel et un logiciel faciles à utiliser. Les cartes Arduino sont capables de lire l'entrée. Vous pouvez compter sur les instructions pour envoyer un message au microcontrôleur. Pour cela, il faut utiliser le langage de programmation Arduino. Au fil des ans, Arduino a réalisé des milliers de projets, utilisés pour fabriquer des instruments scientifiques complexes. Une communauté mondiale de fabricants, d'étudiants, d'artistes et d'artistes crée un processus d'appel qui peut aider beaucoup de mendiants et d'experts. Arduino est né à l'IVREA Institute for Interaction Design comme un outil facile pour le prototypage statique, conçu pour les étudiants sans expérience en électronique et programmation. Une fois devenue une communauté plus grande, la miniature Arduino a maintenu son adéquation et son défi, en produisant diverses pièces, ainsi que des soins 3D



Figure 2.1 : Logo de l'Arduino IDE



### 2.2.2 Choix de L'Arduino

Arduino a été utilisé dans des milliers de projets et applications divers. Le logiciel Arduino est facile à utiliser pour les débutants, mais suffisamment flexible pour les utilisateurs avancés. Fonctionne sur Mac, Windows et Linux. Les enseignants et les élèves l'utilisent pour construire de faibles outils scientifiques, pour démontrer les principes de produits chimiques et physiques, ou pour Utilisez la programmation et robotique. Designers et architectes interactifs, musiciens et artistes utilisent pour l'installation et expérimentent de nouveaux instruments de musique. Les créateurs, bien sûr, utilisent pour effectuer de nombreux projets exposés au créateur de la fatigue. Arduino simplifie même l'utilisation de MicrochStroles, tout en offrant des avantages telque :

- **Prix abordable** : les cartes Arduino sont relativement comparées par rapport aux autres plates-formes MicrochStroler. La version la plus économique du module ardu peut être collectée à la main, ainsi que les modules prédéterminés le coût inférieur à 50 \$.
- **Multiplate** : le logiciel Arduino (IDE) fonctionne sous Windows, Macintosh OSX et Linux. Les systèmes micorontrollés sont limités aux fenêtres.
- **Environnement simple et scolaire** : le logiciel Arduino (IDEA) est facile pour les utilisateurs, tandis qu'il est beau flexible pour les utilisateurs aussi. Pour les enseignants, selon l'environnement de programmation Le tourner, permettant aux étudiants d'apprendre à programmer dans cet environnement pour familiariser avec le fonctionnement d'idées ardues.
- **Software**: Arduino Software est publié sous forme d'open source de source, accessible aux programmeurs expérimentés. La langue peut être étendue à la bibliothèque C ++, et ceux qui souhaitent comprendre les détails techniques peuvent passer par Arduino à la programmation de programmation dans laquelle Est sur le sol. INRÉCÉDÉ Vous pouvez ajouter un code AVR-C dans vos programmes Arduino si vous le souhaitez

## 2.3 Microcontrôleurs

### 2.3.1 ESP32

ESP32 est une plateforme puissante et rentable pour le développement de l'IoT. Développée par Expressive Systems (Shanghai, Chine), elle offre une combinaison performante de fonctionnalités et de capacités pour les applications IoT. ESsp32 présente les caractéristiques suivantes :

- un processeur double cœur,
- une connexion Bluetooth intégrée,
- un grand nombre de broches GPIO pour une utilisation générale et
- une faible consommation d'énergie.

L'ESP32 est équipé d'un microprocesseur Xesa LX6 à deux cœurs (Santa Clara, Californie, États-Unis), qui offre une puissance de traitement accrue et facilite l'exécution multitâche et efficace de tâches complexes. L'ESsp32 intègre des interfaces Wi-Fi et Bluetooth

qui simplifient la connexion et la communication avec d'autres équipements ou réseaux. Il prend en charge les différents protocoles Wi-Fi, tels que 802.11 B/G/n, et offre des options de connexion Bluetooth classique et Bluetooth à faible consommation d'énergie. L'Esp32 est doté de nombreuses broches GPIO qui facilitent la connexion et le contrôle de capteurs externes. Ces broches prennent en charge diverses interfaces, notamment SPI, I2C, UART et PWM. L'ESP32 est réputé pour sa puissance, permettant le développement d'applications de gestion de l'énergie événementielle. Il offre des fonctions de gestion du sommeil et de l'alimentation qui contribuent à réduire la consommation d'énergie et sont adaptées aux projets d'alimentation par batterie ou à faible consommation. L'ESP32 se connecte aux écrans, en touchant les écrans LED ou les indicateurs, pour offrir une interface conviviale aux opérateurs et au personnel. L'ESP32 peut être programmé pour différents langages de développement. Le langage de programmation le plus utilisé est C++, et peut être programmé via Arduino ou des plateformes. De plus, l'ESP-IDF (système de développement informatique) offre un ensemble complet de bibliothèques et d'instruments spécifiques à l'ESP32. Certains modules du marché sont basés sur la puce ESP32. Certains modules intègrent des capteurs, ce qui simplifie l'intégration de ces capteurs aux projets IoT. Par conséquent, le module est adapté aux applications telles que le journal de données. Le module comprend une connexion micro-USB permettant la programmation ou l'alimentation [16].



**Figure 2.2 : ESP32**

De plus, le module comprend

- Processor: ESP32 (24 MHz dual core)
- Flash memory: 4 MB
- Built-in microSD card connector
- PSRAM (pseudo-static random access memory): 8 MB
- Built-in Wi-Fi, Bluetooth, USB to serial converter (CP2104 or CH9102F)
- Built-in Li-ion/Li-Po battery charging circuit: TP4054 chip [16].

## ESP32-DevKitC

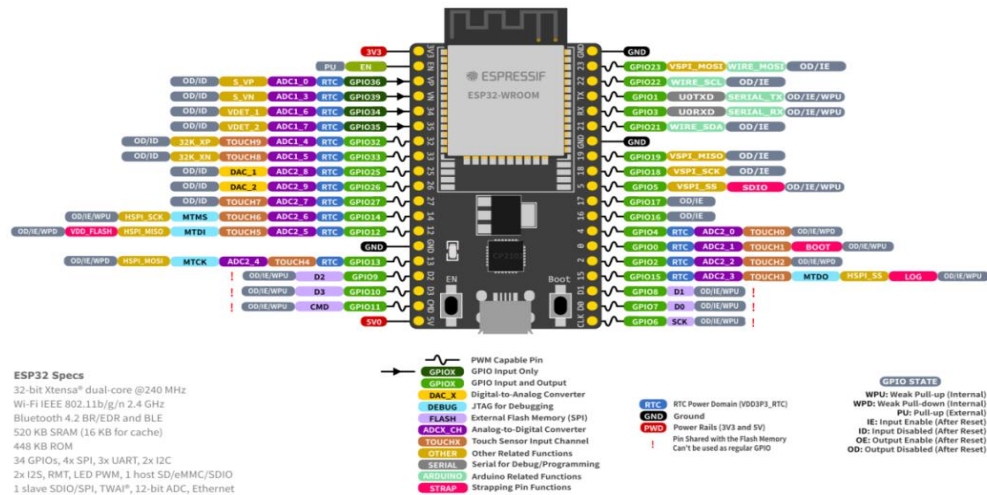


Figure 2.3 'Entrées et les Sorties de ESP32

Ce tableau représente les fonctions des broches d'entrées et des sorties de l'ESP 32

Tableau 1,1 : Broches d'E/S ESP32 [17]

| No. | Name | Type | Function                                |
|-----|------|------|-----------------------------------------|
| 1   | 3V3  | P    | 3.3 V power supply                      |
| 2   | EN   | I    | CHIP_PU, Reset                          |
| 3   | VP   | I    | GPIO36, ADC1_CH0, S_VP                  |
| 4   | VN   | I    | GPIO39, ADC1_CH3, S_VN                  |
| 5   | IO34 | I    | GPIO34, ADC1_CH6, VDET_1                |
| 6   | IO35 | I    | GPIO35, ADC1_CH7, VDET_2                |
| 7   | IO32 | I/O  | GPIO32, ADC1_CH4, TOUCH_CH9, XTAL_32K_P |
| 8   | IO33 | I/O  | GPIO33, ADC1_CH5, TOUCH_CH8, XTAL_32K_N |
| 9   | IO25 | I/O  | GPIO25, ADC2_CH8, DAC_1                 |
| 10  | IO26 | I/O  | GPIO26, ADC2_CH9, DAC_2                 |
| 11  | IO27 | I/O  | GPIO27, ADC2_CH7, TOUCH_CH7             |
| 12  | IO14 | I/O  | GPIO14, ADC2_CH6, TOUCH_CH6, MTMS       |
| 13  | IO12 | I/O  | GPIO12, ADC2_CH5, TOUCH_CH5, MTDI       |
| 14  | GND  | G    | Ground                                  |
| 15  | IO13 | I/O  | GPIO13, ADC2_CH4, TOUCH_CH4, MTCK       |
| 16  | D2   | I/O  | GPIO9, D2 <a href="#">2</a>             |
| 17  | D3   | I/O  | GPIO10, D3 <a href="#">2</a>            |
| 18  | CMD  | I/O  | GPIO11, CMD <a href="#">2</a>           |
| 19  | 5V   | P    | 5 V power supply                        |

### 2.3.2 Avantages :

- **Diversité dans l'utilisation :** L'ESP32 dispose d'une connectivité Wi-Fi et Bluetooth, ce qui lui permet d'être utilisé dans de nombreuses applications IoT, telles que le contrôle des appareils domestiques intelligents ou des appareils portables [18].
- **Traitement puissant :** Il dispose de deux cœurs de traitement à des vitesses allant jusqu'à 240 MHz, ce qui lui permet d'effectuer des opérations complexes telles que le traitement du signal en temps réel [18].
- **Faible consommation d'énergie :** L'ESP32 offre plusieurs modes pour réduire la consommation d'énergie, ce qui le rend adapté aux appareils à batterie [18].
- **Communauté de soutien :** ESP32 a un grand soutien de la communauté des développeurs, ce qui rend facile de trouver des bibliothèques prêtes à l'emploi, des explications et de l'aide en ligne [18].
- **Facilité de programmation :** Il prend en charge l'IDE Arduino et d'autres environnements de développement, de sorte qu'il peut être facilement programmé, même par les débutants [18].

### 2.3.3 Inconvénients

- **Complexité :** Malgré ses avantages, les nombreuses fonctionnalités de l'ESP32 peuvent être un défi au début, surtout pour les débutants [18].
- **Sensibilité énergétique :** Il a besoin d'une alimentation stable, car toute fluctuation de la tension peut entraîner un dysfonctionnement [18].
- **Taille des bibliothèques :** Certaines de ses bibliothèques sont grandes et peuvent consommer une partie importante de la mémoire du périphérique [18].

### 2.3.4 Caractéristiques :

L'ESP32 se compose d'un ensemble de spécifications techniques.

- **Processeur :** Il se compose de deux cœurs Tensilica LX6, et sa vitesse atteint 240 MHz, ce qui lui donne une bonne capacité de traitement des données.
- **RAM :** Il a 520 Ko de SRAM.
- **Stockage (Flash) :** Varie selon la version, généralement 4 ou 16 MB [18].
- **Réseau sans fil (Wi-Fi) :** prend en charge la norme 802.11 b/g/n, ce qui lui permet de se connecter à des réseaux Internet sans fil.
- **Bluetooth :** Inclut la version 4.2 de Bluetooth, ainsi que le support pour la technologie Bluetooth Low Energy (BLE), qui permet de réduire la consommation d'énergie lors du transfert de données [18].
- **Il a jusqu'à 34 broches** qui peuvent être utilisées comme entrée ou sortie (GPIO).
- **Il prend en charge de nombreuses interfaces** telles que : UART, SPI, I2C, I2S, CAN, PWM, ADC et DAC, ce qui lui permet de se connecter à plusieurs types d'appareils électroniques [18].

### 2.3.5 Caractéristiques supplémentaires :

- Il a des capteurs tactiles avec la technologie de capacité électrique.
- Il a un capteur intégré pour mesurer la température.
- Il dispose d'un module de cryptage spécial, qui assure un bon niveau de sécurité dans les opérations de communication [18].

## 2.4 Module communication LoRa SX1278

Le modem LoRa longue portée de l'émetteur-récepteur SX1278 réduit la consommation de courant tout en offrant des communications à très grande distance sur une large plage et une immunité exceptionnelle aux interférences.

Avec sa combinaison de cristal à faible coût et de matériaux peu coûteux, le SX1278 peut atteindre une sensibilité supérieure à -148 dBm en utilisant la technologie unique de modulation LoRa de Semtech. Grâce à sa haute sensibilité et son amplificateur de puissance intégré +20 dBm, il offre le plus grand budget de liaison disponible, ce qui le rend idéal pour toute application où la durabilité ou la portée est nécessaire. Par rapport aux méthodes de modulation conventionnelles, LoRa offre également des avantages notables en matière de protection et de sélectivité, supprimant les limites de conception classiques pour la consommation d'énergie, l'immunité aux interférences et la portée [22].

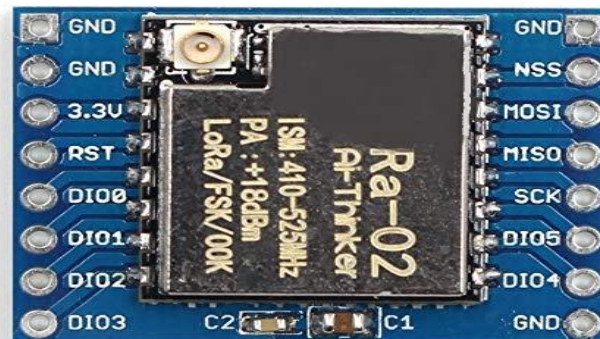


Figure 2.4 : Module LoRa SX1278

### 2.4.1 Caractéristiques :

- LoRa TM Modem : sortie RF constante vs. alimentation en V,
- Haute efficacité de +14dBm PA
- Débit programmable jusqu'à 300 kbps; haute sensibilité : -148 dBm
- Modulation à l'aide de FSK, GFSK, MSK, GMSK, LoRa et OOK
- Synchroniseur bit intégré de récupération d'horloge

- Plage dynamique pour la détection de préambule : 127 dB CAD et RSSI Automatic RF Sense avec un moteur de paquets AFC ultra-rapide pouvant gérer jusqu'à 256 octets avec CRC
- Capteur de température intégré et avertissement de batterie faible [22].

**Tableau 2 .1 Pinout De Module LoRa sx1278 [23]**

| Pin Number | Pin Name | Description              |
|------------|----------|--------------------------|
| 1,2,9,16   | Ground   | Ground pin of the Module |
| 3          | 3.3V     | Input Supply voltage     |
| 4          | Reset    | Resets the Module        |
| 5          | DIO 0    | Digital I/O – 0          |
| 6          | DIO 1    | Digital I/O – 1          |
| 7          | DIO 2    | Digital I/O – 2          |
| 8          | DIO 3    | Digital I/O – 3          |
| 10         | DIO 4    | Digital I/O – 4          |
| 11         | DIO 5    | Digital I/O – 5          |
| 12         | SCK      | SPI – Clock Input        |
| 13         | MISO     | SPI – Data Out           |
| 14         | MOSI     | SPI – Data In            |
| 15         | NSS      | SPI – Chip Select        |



## 2.5 Capteurs

### 2.5.1 Capteur de température et d'humidité DHT11:

Le **DHT11** est un capteur numérique capable de mesurer la température et l'humidité relative de l'air. Il est simple à utiliser, fonctionne entre 3,3 V et 5 V, et fournit une précision modérée ( $\pm 2$  °C pour la température et  $\pm 5$  % pour l'humidité). Son principal avantage est son faible coût, ce qui en fait un capteur adapté aux projets pédagogiques ou de démonstration [24].

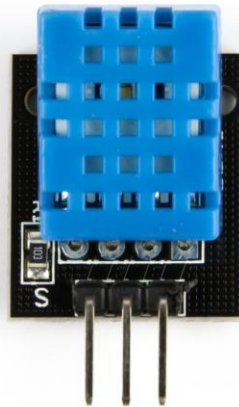


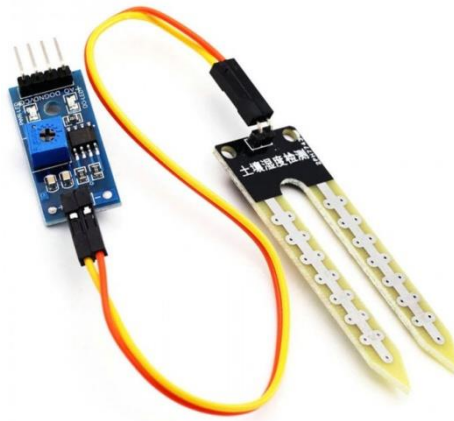
Figure 2.5 DHT11

#### 2.5.1.1 Caractéristiques:

- Capteur numérique de température et d'humidité.
- Idéal pour les projets éducatifs, domotiques ou de prototypage.
- Facile à utiliser avec Arduino, ESP32, etc.
- Tension d'alimentation : **3,3 V à 5,5 V**.
- Courant consommé :
  - **En mesure** : environ **2,5 mA**.
  - **En veille** : inférieur à **100  $\mu$ A**.
- Signal numérique sur 1 fil (protocole propriétaire, type UART simplifié).
- Lecture des données toutes les 1 seconde minimum.
- Communication simple mais lente par rapport à d'autres capteurs.

### 2.5.2 Capteur d'humidité du sol modèle FC-28:

Le FC-28 est un capteur couramment employé pour évaluer le taux d'humidité du sol. Souvent employés dans les initiatives d'agriculture intelligente et de domotique, ils offrent une surveillance précise des conditions du sol, autorisant ainsi le fonctionnement autonome des systèmes d'irrigation lorsqu'une intervention est nécessaire [25].



**Figure 2.6 : Capteur d'humidité du sol modèle FC-28:**

### 2.5.2.1 Caractéristiques techniques principales

- Dénomination : FC-28, le module de détection d'humidité du sol
- Alimentation : Ce dispositif fonctionne avec une tension variant de 3,3 V à 5 V, ce qui le rend apte à être utilisé avec des puces telles qu'Arduino, ESP32 et d'autres microcontrôleurs.
- Type de sortie : Sortie numérique (D0) : Émet un signal binaire (HAUT/BAS) en fonction du dépassement d'un seuil d'humidité spécifié.
- Sortie analogique (A0) : Offre une mesure exacte du taux d'humidité, exprimée en valeurs de 0 à 1023.
- Genre de capteur : Capteur résistif qui quantifie la variation de résistance électrique due à l'absorption d'humidité du sol.
- Ajustable : potentiomètre inclus pour modifier la sensibilité du seuil.
- Consommation d'énergie : très basse, aux alentours de 20 mA [25].

### 2.5.2.2 Dimensions

- Circuit imprimé : 30 mm sur 15 mm
- Électrode (sonde) : 60 x 20 mm
- Plage de température d'opération : 10-30°C, approprié pour une utilisation domestique et agricole basique.

### 2.5.2.3 Principe de fonctionnement

Le fonctionnement du capteur repose sur le lien entre la résistance électrique et le degré d'humidité du sol. Avec l'accroissement de l'humidité, la résistance entre les pointes de la sonde se réduit, provoquant ainsi une altération du potentiel électrique mesuré en interne.

Un potentiomètre peut être utilisé pour ajuster la sensibilité, permettant de modifier le seuil à partir duquel la sortie numérique (D0) est activée [25].

### 2.5.2.3 Applications pratiques

- Systèmes d'irrigation intelligents : déclenchez automatiquement les pompes à eau en fonction des conditions du sol.
- Surveillance en serre : Suivi constant du taux d'humidité du sol dans les cultures sous serre.
- Initiatives pédagogiques : Fréquemment employé dans les initiatives pédagogiques utilisant Arduino ou ESP32.
- Gestion de l'eau : optimiser l'usage de l'eau dans les pratiques agricoles domestiques et expérimentales [25].

## 2.6 Actionneurs

### 2.6.1 Mini Pompe à Eau

La pompe utilisée est un modèle submersible fonctionnant en courant continu (DC) entre 3 et 6 volts. Elle permet de pomper de l'eau sur une hauteur de 40 à 110 cm, ce qui est suffisant pour irriguer une petite surface agricole. Elle est déclenchée automatiquement par le système dès que l'humidité du sol descend sous un seuil critique [7].



Figure 2.7 : Mini Pompe

#### 2.6.1.1 Caractéristiques:

- Tension continue : 3-6V
- Hauteur de levage maximale : 40-110cm / 15.75« -43.4"
- Débit : 80 à 120 litres par heure
- Le diamètre externe de l'évacuation d'eau : 7.5mm / 0.3"

- Le diamètre interne de la sortie d'eau est de 5mm / 0,2".
- Dimension : approximativement 24mm / 0,95"
- Dimension : Approximativement 45 mm / 1,8 pouces.
- Dimension: approximativement 30mm / 1.2 »
- Substance : plastique technique
- Mode opératoire : conception à courant continu sans balais, guidage magnétique [26].

### 2.6.2 Relais

Un relais est un élément électronique qui autorise la gestion d'une charge électrique à distance en se servant d'un signal électrique de faible puissance. Il se compose d'un enroulement électrique pouvant être déclenché par une faible intensité de courant, ainsi que d'un contact mécanique susceptible d'être ouvert ou fermé en fonction du signal électrique. On utilise fréquemment le relais pour contrôler des équipements électriques comme les éclairages, les moteurs et les pompes, tout en assurant la protection des circuits électroniques en interrompant l'alimentation lors d'une surcharge ou d'un court-circuit. On peut le gérer de diverses façons, par exemple, en se servant d'un microcontrôleur ou d'un interrupteur à commande manuelle [27].

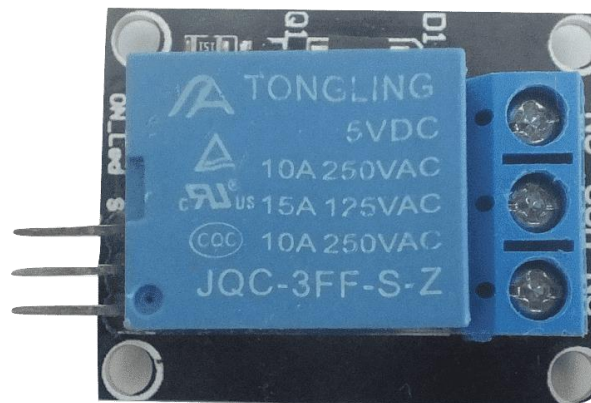


Figure 2.8 Module Relais 5V

#### 2.6.2.1 Principe de fonctionnement

Un relais est un élément que l'on peut assimiler à une vanne : il a la capacité de laisser passer le courant ou de ne pas le faire.

Le relais est équipé d'une bobine électrique qui produit un champ magnétique. Lorsque le champ magnétique est activé, il déplace un élément métallique à l'intérieur du relais, ce qui permet l'ouverture ou la fermeture d'un circuit électrique.

Dans un circuit électrique, un relais fonctionne comme un commutateur. La distinction entre un interrupteur classique et un relais réside dans le fait qu'un interrupteur traditionnel fonctionne mécaniquement, c'est-à-dire en le déclenchant physiquement avec votre doigt, tandis qu'un relais opère via un courant électrique.

Le relais est équipé de trois connexions : COM, NC et NO. NC signifie 'FERMÉ EN GÉNÉRAL'. Cela signifie que lorsque le relais ne reçoit aucun signal d'entrée (valeur LOW dans `digitalWrite()`), le circuit à haute tension branché sera en marche. En revanche, si vous appliquez une tension de 5V au relais, le circuit secondaire se coupera. NO est l'acronyme de 'NORMALEMENT OUVERT'. En d'autres termes, une valeur de 5V appliquée au relais (valeur HIGH dans `digitalWrite()`) interrompra le circuit haute tension, et vice versa [27].

- Le dispositif de relais est doté de trois broches de liaison :
- broche + = Alimentation (3,3V à 5V).
- Broche - = Masse de l'alimentation.
- broche s = broche qui reçoit le signal envoyé par l'Arduino ou le micro:bit [27].

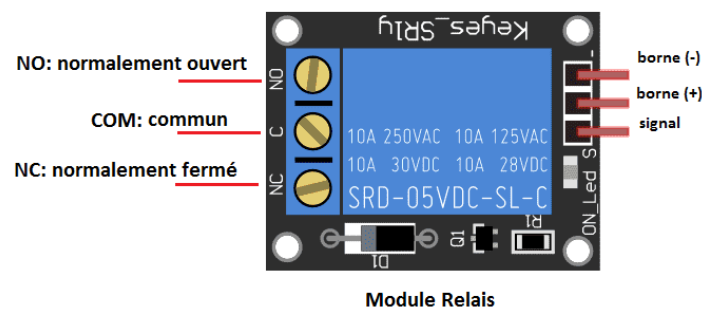


Figure 2.9 : Broches du Module Relais

## 2.6.2 Écran OLED SSD1306

Les écrans OLED (diodes électroluminescentes organiques) sont des écrans modernes utilisés pour afficher du texte et des graphiques dans les projets électroniques. Leur particularité réside dans leur auto-éclairage sans rétroéclairage, ce qui les rend économes en énergie et offre un affichage haute résolution.

L'écran utilise la puce contrôleur SSD1306, qui prend en charge plusieurs protocoles de communication, dont I2C et SPI. Certains modèles permettent même de basculer entre eux selon les besoins [28].



Figure 2.10 : Écran OLED SSD1306

### 2.6.2.1 Spécifications techniques :

- Tension de fonctionnement : 3 V à 5 V.
- Couleur : Monochrome (généralement blanc ou bleu).
- Résolution : 128 x 64 ou 128 x 32 pixels.
- Puce contrôleur : SSD1306.
- Méthode de communication : I2C ou SPI.
- Dimensions de l'écran : Environ 27 x 27 x 4 mm.

### 2.6.2.2 Caractéristiques

- Faible consommation : Sans rétroéclairage
- Haute résolution : Permet d'afficher une plus grande quantité de données que les écrans de texte comme le 1602
- Facile à connecter : Ne nécessite que quatre fils : deux pour l'alimentation (VCC et GND) et deux pour la communication (SDA et SCL en I2C) [28].

## 2.7 Application Mobile – Blynk

Blynk est une plateforme IoT complète qui permet la création d'interfaces mobiles personnalisées pour surveiller et contrôler des systèmes embarqués à distance. Grâce à des widgets graphiques, il est possible d'afficher en temps réel les données envoyées par l'ESP32, de déclencher des actions à distance, ou de programmer des notifications automatiques. Blynk offre également un service cloud pour stocker les données et faciliter l'interconnexion [10].



Figure 2.11 : Icone de l'application Blynk

### 2.7.1 Blynk.Console

L'application en ligne riche en fonctionnalités Blynk.Console sert une variété de types d'utilisateurs. Parmi ses principales fonctions, citons :

- configuration de la plateforme des appareils liés, y compris les paramètres pour les applications.

- Gestion des appareils, des données, des utilisateurs, de l'organisation et de la localisation.
- Gestion et surveillance des appareils via la télécommande [30].

## **2.8 Conclusion**

Ce chapitre a permis de détailler les différents composants matériels et logiciels utilisés dans le projet. Le choix de l'ESP32 comme microcontrôleur principal, combiné aux capteurs DHT11 et FC-28, au module LoRa SX1278, ainsi qu'aux interfaces d'affichage (OLED) et de commande (relais), forme une solution complète, modulaire et évolutive pour un système agricole intelligent.



---

## **Chapitre 3**

### **Réalisation et mise en Œuvre du système**

---

### **3.1 Introduction**

Ce chapitre détaille les différentes étapes de la mise en œuvre du système embarqué proposé pour l'agriculture intelligente. Nous commençons par présenter l'architecture générale, puis les connexions électroniques et l'assemblage des composants. Ensuite, nous expliquons en détail la programmation du microcontrôleur, l'intégration avec la plateforme mobile Blynk, les tests expérimentaux et les performances observées. Enfin, nous discutons des limites du système et des perspectives d'amélioration.

### **3.2 Définition et l'importance de l'irrigation :**

L'irrigation est la pratique agricole de base qui consiste à apporter de l'eau aux cultures de manière artificielle afin d'assurer leur bonne croissance et d'optimiser la production agricole. Elle revêt une importance particulière dans les régions où les précipitations sont insuffisantes ou irrégulières, surtout durant les périodes critiques du cycle végétatif des plantes, comme le printemps et l'été. En l'absence d'irrigation adéquate, les plantes souffrent de stress hydrique, ce qui peut avoir des répercussions importantes sur la qualité et la quantité des récoltes

### **3.3 Application de l'internet des objets à l'irrigation :**

L'intégration de la technologie de l'internet des objets dans les systèmes d'irrigation a transformé les pratiques agricoles en permettant d'améliorer la gestion de l'eau, le rendement des cultures et la durabilité. Les systèmes d'irrigation basés sur l'internet des objets utilisent divers capteurs, tels que des capteurs d'humidité du sol et des capteurs à ultrasons, pour surveiller les conditions environnementales et les niveaux d'eau en temps réel, ce qui permet d'effectuer un contrôle précis de la distribution de l'eau [32]. Ces systèmes utilisent souvent la communication sans fil pour transmettre les données à une unité de contrôle centrale qui utilise des algorithmes intelligents pour optimiser la programmation de l'irrigation et l'utilisation de l'eau, permettant ainsi d'économiser l'eau et de réduire la consommation d'énergie. La possibilité de surveiller et de contrôler à distance les opérations d'irrigation via des interfaces web ou mobiles offre aux agriculteurs la possibilité de prendre des décisions en connaissance de cause et de réagir rapidement en cas de situation critique [32][33].

### **3.4. Système d'irrigation intelligent proposé :**

Compte tenu des défis auxquels le secteur agricole est confronté en matière de pénurie d'eau et de difficulté à la gérer. Nous avons proposé un système visant à optimiser l'utilisation de l'eau tout en économisant du temps et des efforts, grâce à la surveillance du sol et au contrôle à distance de l'irrigation. Ce système implique une communication entre un smartphone et divers autres blocs, tels que des contrôleurs et des capteurs. Et le deuxième système et de contrôle la température et l'humidité grâce au capteur DHT11 via LoRa SX1278 qui ne nécessite pas une connexion internet elle se fait par communication à l'air libre.

### 3.5 Schéma synoptique du système

La figure suivante représente le schéma synoptique de notre application

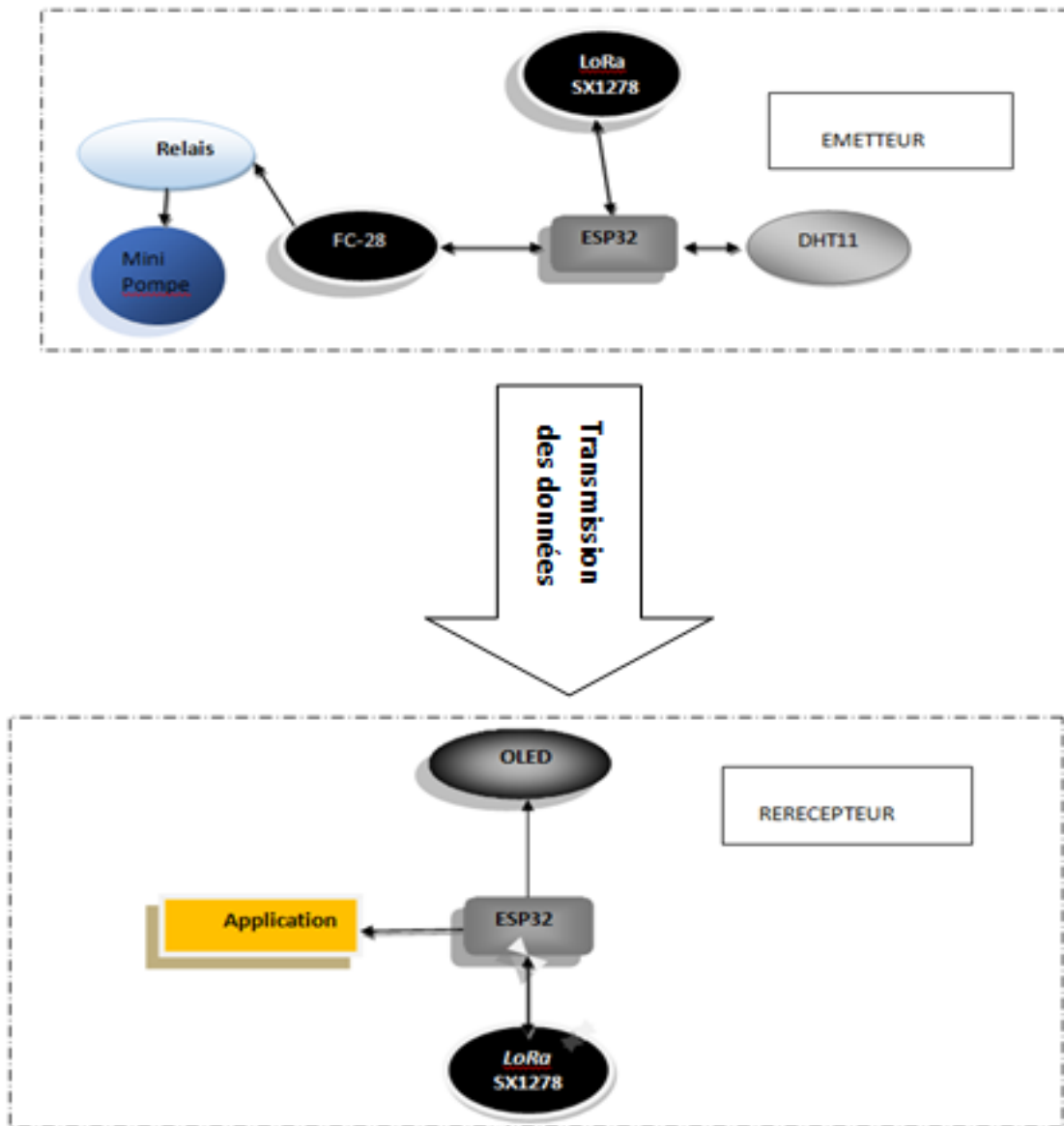


Figure 3.1 : Schéma synoptique du système

### 3.6 Architecture Générale du Système

Le système développé repose sur une architecture distribuée, combinant capteurs, actionneurs, modules de communication et une plateforme mobile. Il permet :

- La collecte des données environnementales à partir de capteurs (DHT11 et FC-28),
- Le traitement local des données via l'ESP32,
- Le contrôle automatique des actionneurs (relais et pompe),

- L’affichage local des valeurs mesurées via un écran OLED,
- La transmission des données via LoRa et Wi-Fi vers une application mobile.

Les modules matériels principaux sont :

- ESP32 : microcontrôleur principal.
- DHT11 : capteur de température et d’humidité.
- FC-28 : capteur d’humidité du sol.
- SX1278 : module LoRa.
- Relais 5V + pompe à eau : système d’irrigation.
- Écran OLED SSD1306.
- Application Blynk pour visualisation mobile.

### **3.7 Fonctionnement du système :**

Le principal objectif du système d’agriculture intelligente que nous avons conçu est de surveiller automatiquement les conditions météorologiques et pédologiques sur le terrain, et d’intervenir au besoin. Nous utilisons d’abord une unité ESP32 pour lire les données du capteur DHT11 (température et humidité), et un deuxième capteur d’humidité du sol.

Après lecture des données, le module LoRa SX1278 les transmet à une station de réception distante. La distance entre les deux stations peut atteindre plusieurs kilomètres grâce à la puissante technologie de transmission longue distance LoRa. D’autre part, le second ESP32, connecté à l’unité LoRa, reçoit ces données et les prépare pour l’affichage dans l’application smartphone .

L’application affiche directement toutes les valeurs : température de l’air, humidité et conditions du sol. En outre, le système comprend un composant de contrôle intelligent qui va au-delà d’une simple surveillance. Par exemple, si le sol est trop sec, ESP32 active automatiquement une pompe pour l’arroser.

## **3.8 Différentes parties du système**

### **3.8.1 Coté émission**

#### **3.8.1.1 Montage de l’ ESP32 avec DHT11**

La figure suivante montre le montage de l’ESP 32 Avec DHT11 pour l’émission de la température et la l’humidité mesurée via LoRa sx1278 qui sont transmis au récepteur qui est composé du LoraSX 1278 et l’écran OLED les valeurs mesurées sont affichées par ce dernier

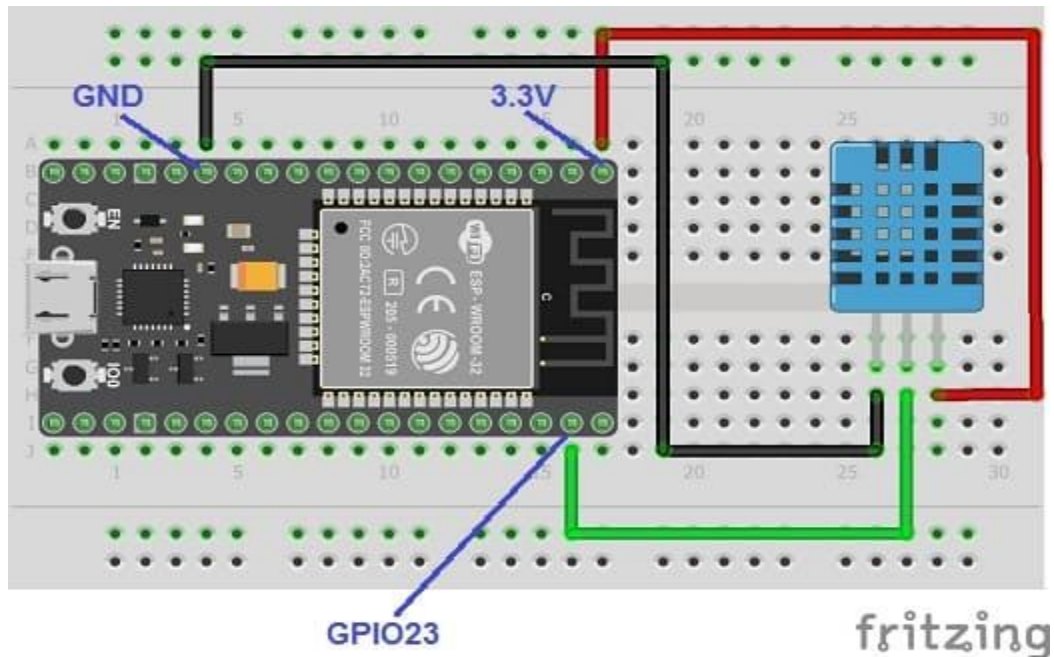


Figure 3.2 : Câblage de l'ESP32 Avec DHT11

Les principales connexions de l'ESP 32 et DHT 11 sont représenté dans le tableau suivant

Tableau 3.1 : Connexion entre ESP32 et DHT11

| Broche ESP32      | Broche DHT11       |
|-------------------|--------------------|
| 3.3V ou 5V        | VCC                |
| GND               | GND                |
| GPIO4             | DATA               |
| entre DATA et VCC | Résistance Pull-up |

### 3.8.2.2 Montage de l' ESP32 avec FC-28 (capteur de humidité du sol)

La figure suivante montre le branchement de l'ESP32 avec le capteur d'humidité du sol

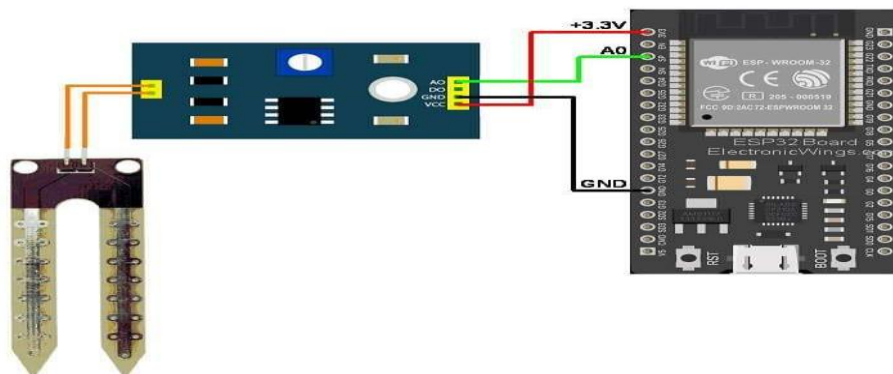


Figure 3.3 Schéma du câblage de l'ESP32 Avec FC-28

Les principales connexions de l'ESP 32 et le capteur d'humidité du sol FC-28 sont représenté dans le tableau suivant

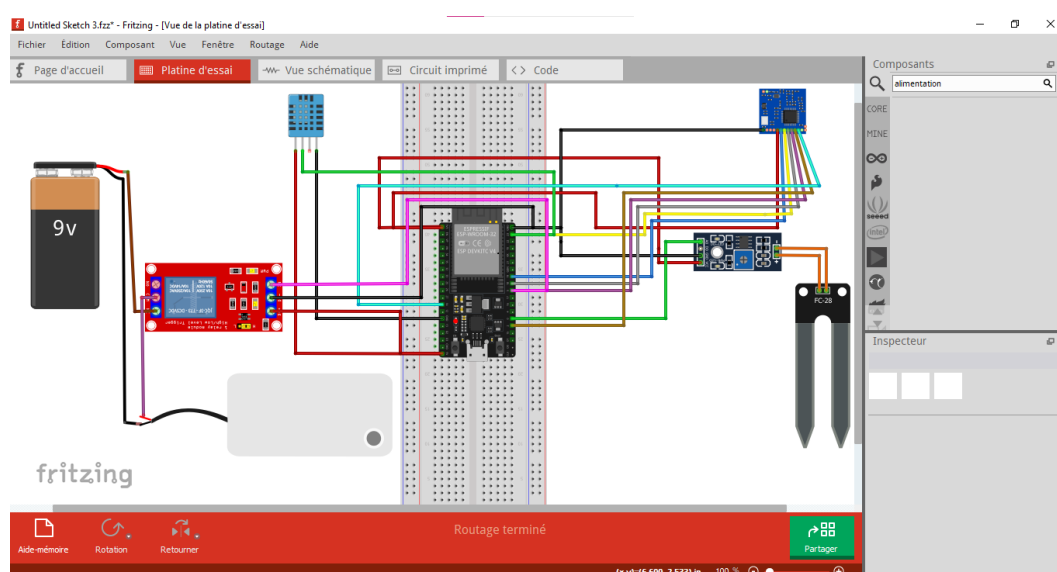
**Tableau 3.2 : Connexion Entre ESP32 et FC-28**

| Broche ESP32 | Broche FC-28 |
|--------------|--------------|
| GND          | GND          |
| GPIO36       | A0           |
| VCC          | 3V3 OU 5V    |

### 3.8.1.3 Montage final de l'émetteur

#### 3.8.1.3.1 Montage sur logiciel FRITZING

La figure suivante représente le montage final de l'émetteur réalisé avec le logiciel Fritzing

**Figure 3.4 Schéma final émetteur**

#### 3.8.1.3.2 Montage réel coté émetteur

La figure suivante représente le montage final de l'émetteur réalisé pratiquement sur une plaque d'essai

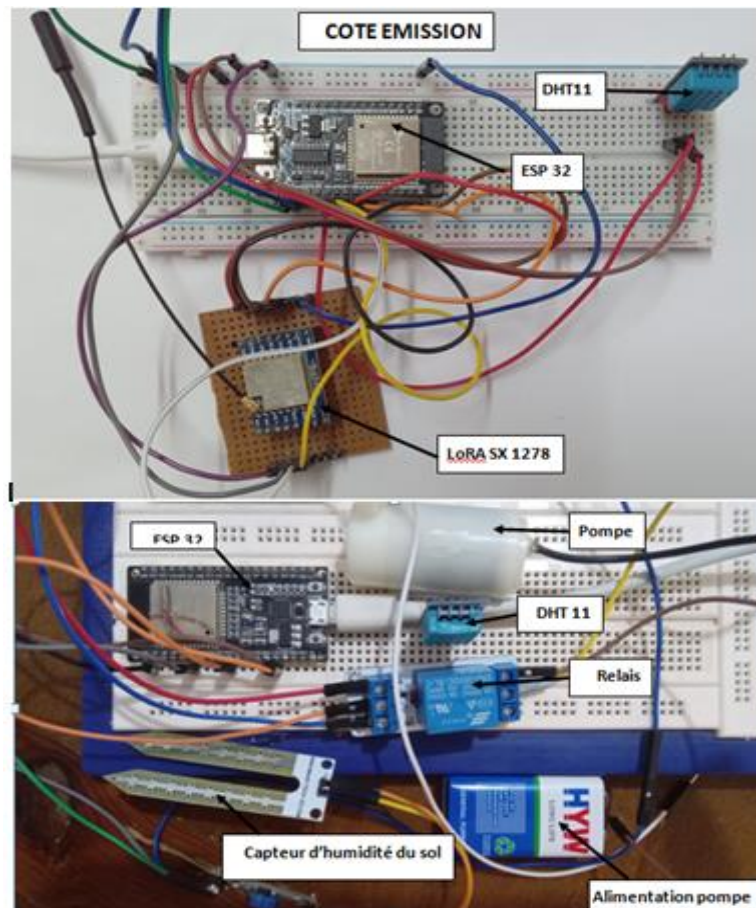


Figure 3.5 : Schéma réel de l'émetteur

### 3.8.2 Coté réception

#### 3.8.2.1 Montage de l'écran OLED et ESP32 :

La figure suivante montre le câblage de l'écran OLED et l'ESP 32

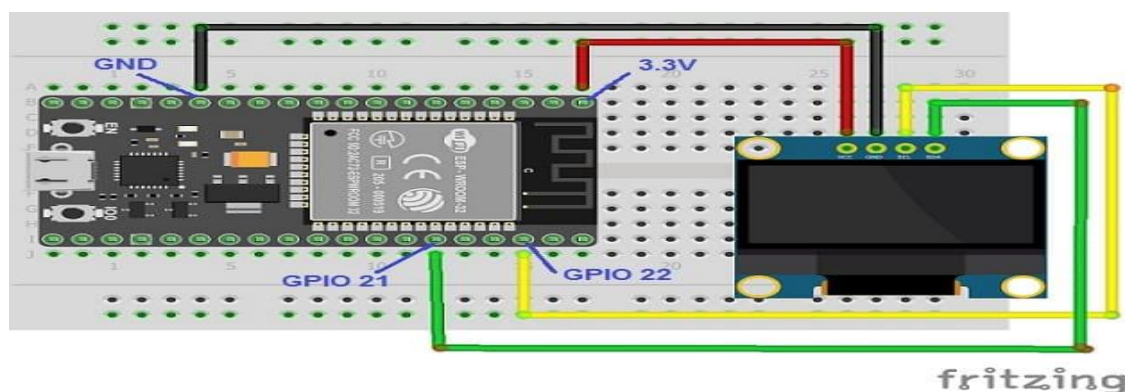


Figure 3.6 : Câblage de l'ESP32 Avec OLED

Les principales connexions de l'ESP 32 et l'écran OLED sont représentées dans le tableau suivant

Tableau 3.4 : Connexion Entre ESP32 et OLED



| Fonction               | Broche OLED | Broche ESP32 |
|------------------------|-------------|--------------|
| Ligne de l'horloge I2C | SCL         | D22          |
| Ligne de données I2C   | SCA         | D21          |
| Alimentation (3.3V-5V) | Vcc         | 3.3V         |
| Masse                  | GND         | GND          |

### 3.7.1.2 Montage ESP32 et LoRa SX1278

La figure suivante montre le montage de LoRa 1278 Avec OLED pour la réception des donnée via LoRa 1278 qui sont transmis de l'émetteur

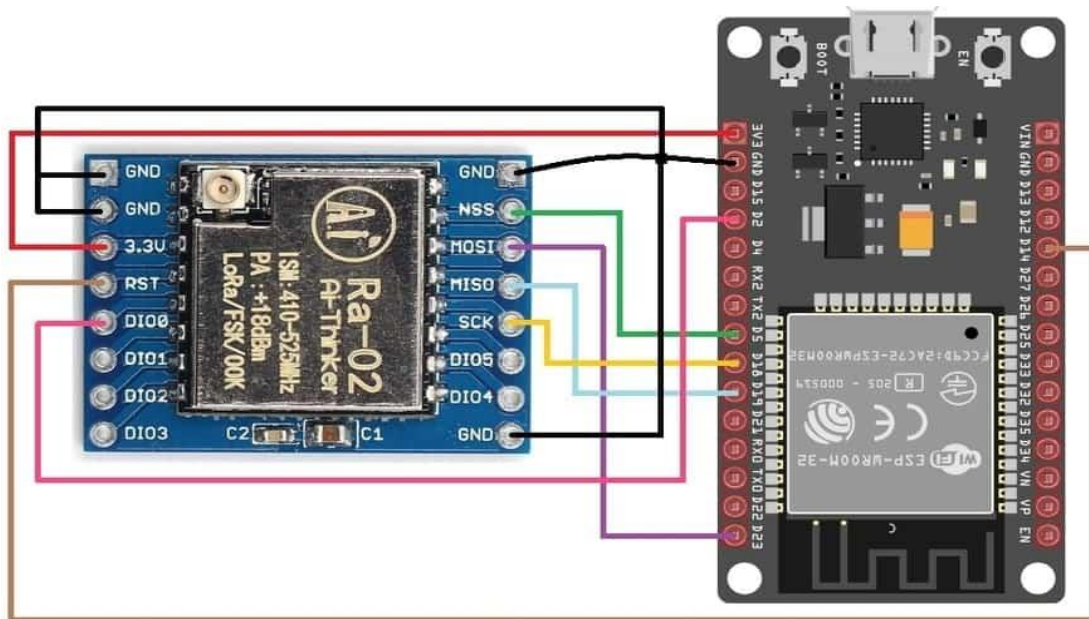


Figure 3.7 : Schéma de Câblage ESP32 Avec LoRa SX1278

Les principales connexions de l'ESP 32 et LoRa sx1278 sont représenté dans le tableau suivant

Tableau 3.2 : Connexion Entre ESP32 et LoRa SX1278

| ESP32 Pins | SX1278 Pins |
|------------|-------------|
| GND        | GND         |
| 3.3V       | VCC         |
| D5         | NSS         |
| D23        | MOSI        |
| D19        | MISO        |
| D18        | SCK         |
| D14        | RST         |
| D2         | DIO0        |

### 3.8.2.3 Montage finale du récepteur sur FRITZING

La figure suivante représente le montage final du récepteur réalisé avec le logiciel Fritzing

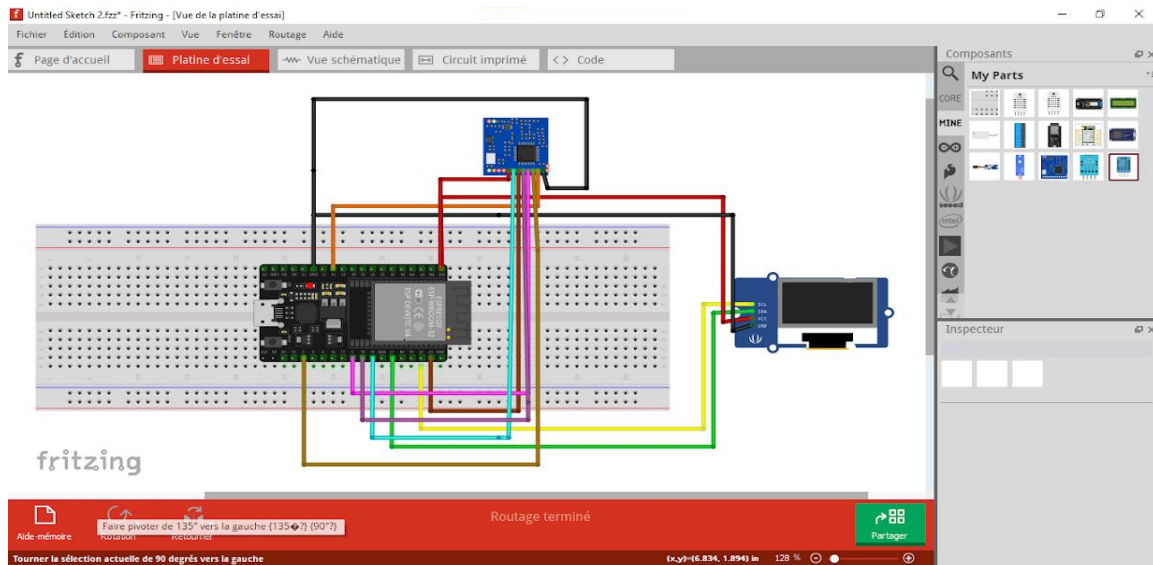


Figure 3.8 : Schéma final récepteur sur FRITZING

### 3.8.2.3 Montage finale du récepteur réel

La figure suivante représente le montage final du récepteur réalisé. L'écran OLED Affiche la température et l'humidité mesurée par le capteur DHT11 de l'émetteur cette dernière est reçue via LoRa SX1278 de la réception qui reçoit les mesures de LoRa SX1278 en émission.

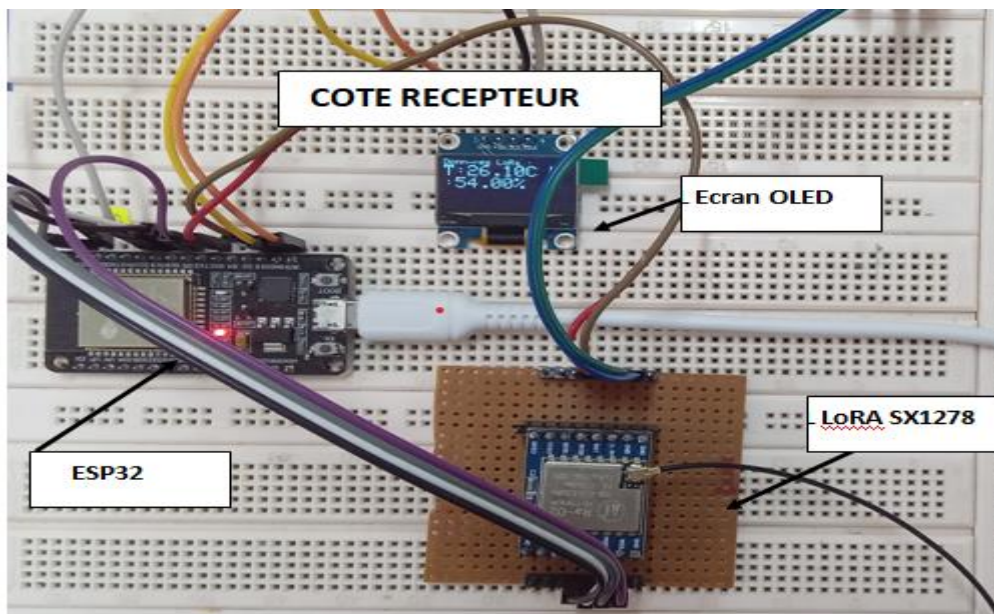
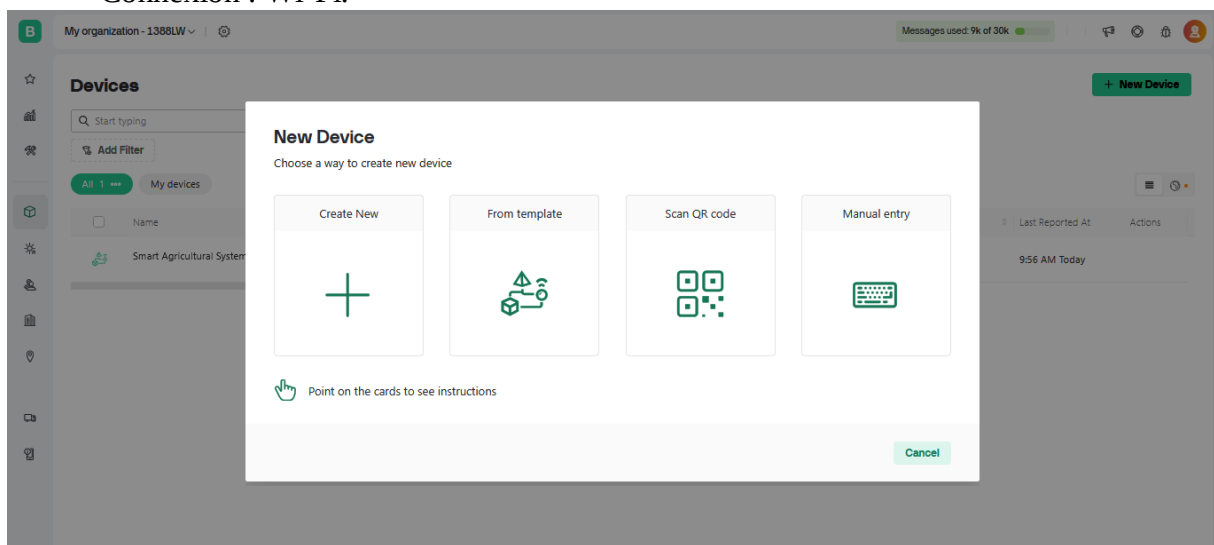


Figure 3.9 : Schéma final récepteur sur FRITZING

### 3.9 Configuration et création d'un projet Blynk IoT :

**Étape 1 :** Création d'un projet dans l'application mobile Blynk IoT.

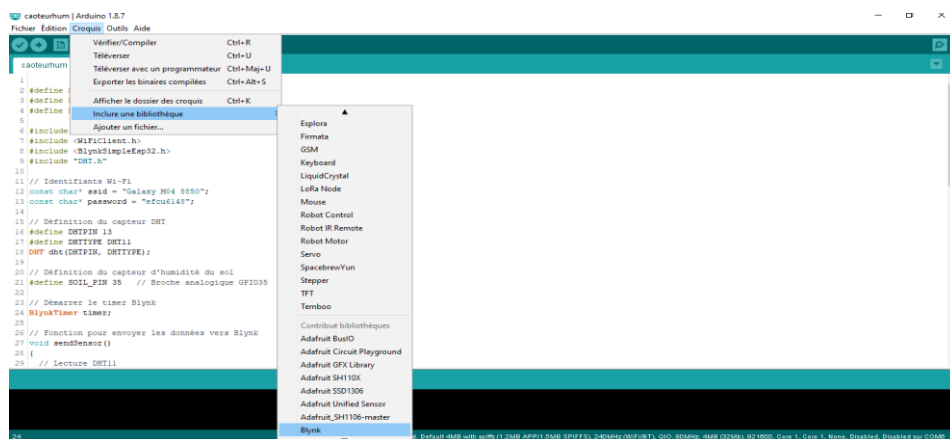
- On a téléchargé l'appli Blynk IoT.
- On a ouvert l'appli Blynk IoT.
- On crée un compte.
- Création d'un nouveau projet :
- On donne un nom.
- Choisissons l'ESP32 comme matériel.
- Connexion : Wi-Fi.



**Figure 3.10:** Création d'un projet dans l'application Blynk IoT.

**Étape 2 :** Installation de la bibliothèque Blynk dans l'Arduino IDE.

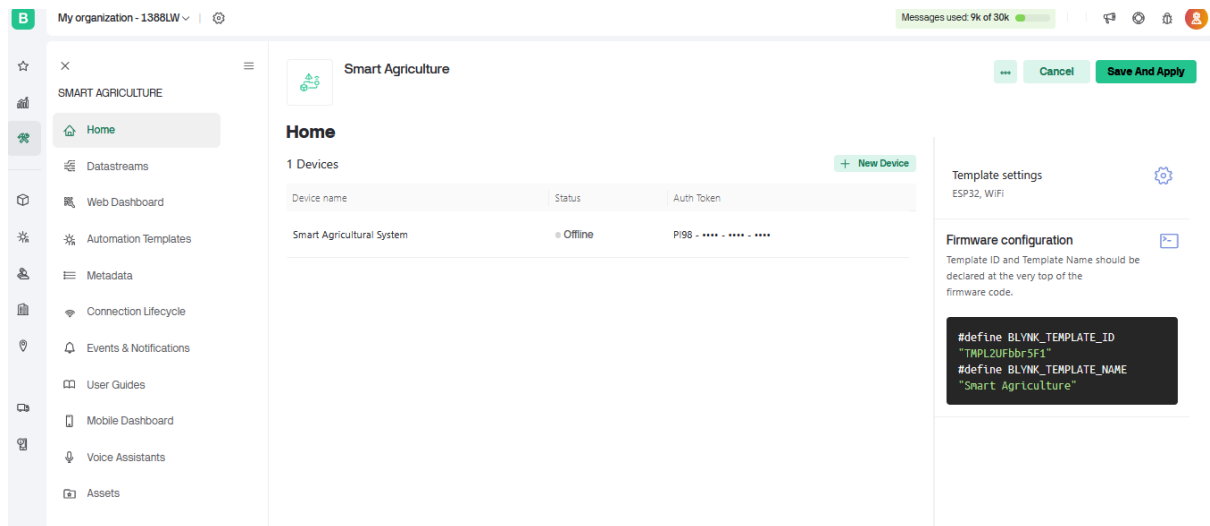
- Nous allons dans **Tools > Manage Libraries**.
- Nous avons cherché Blynk et prouvé Blynk.



**Figure 3.11:** Installation de la bibliothèque Blynk dans l'Arduino IDE.

### Étape 3 : Connexion et programmation de l'ESP32 avec Blynk IoT.

- On a ouvert l'application **Blynk IoT**.
- On clique sur l'icône d'engrenage dans le coin supérieur droit du projet.
- Nous avons sélectionné ensuite "**Appareils**".
- Le jeton d'authentification est nécessaire pour le code. Il faut donc le copier.
- On fusionne le code d'authentification dans le code de l'ESP32.



**Figure 3.12:** Création d'un projet dans l'application Blynk IoT.

## 3.10 Résultat de la réalisation avec l'application Blynk IoT

Une fois que l'application Blynk est connectée à l'ESP32 via Wi-Fi, les valeurs obtenues précédemment sont affichées sur l'interface de l'application sur le téléphone en termes de température, d'humidité, d'humidité du sol et de position de la pompe.

La figure ci-dessous montre le résultat de la connexion de l'application avec le programme. La Figure 3.13.a représente l'interface de l'application Blynk on a l'affichage de la température et de l'humidité en temps réel ainsi que l'humidité du sol qui est supérieur à 30% alors que la pompe est en arrêt. La figure 3.13.b représente l'interface de l'application Blynk qui affiche la température et de l'humidité en temps réel ainsi que l'humidité du sol qui est inférieur à 30% alors que la pompe est en marche.



Figure 3.13 : Notre application Blynk IoT

### 3.11 Conclusion

Ce chapitre a permis de présenter l'ensemble des étapes de la réalisation du système embarqué pour l'agriculture intelligente. Grâce à une combinaison de composants simples et économiques, le système est capable de collecter et transmettre des données environnementales, de prendre des décisions automatiques (arrosage) et d'alerter l'utilisateur à distance.

Le système s'est révélé fiable et extensible. Il constitue une base solide pour le développement de projets plus complexes, intégrant l'intelligence artificielle, l'analyse prédictive ou la supervision de grandes exploitations agricoles.

---

## Conclusion générale

---

## Conclusion Générale

Ce travail s'inscrit dans une dynamique de modernisation de l'agriculture à travers l'intégration des technologies embarquées et des systèmes de communication intelligents. L'objectif principal de ce mémoire était de concevoir et de réaliser un système embarqué capable de surveiller en temps réel les paramètres environnementaux d'une exploitation agricole, et de déclencher automatiquement l'arrosage en fonction de l'humidité du sol.

Dans ce cadre, un dispositif basé sur le microcontrôleur ESP32 a été développé. Il intègre plusieurs capteurs (DHT11 pour la température et l'humidité de l'air, FC-28 pour l'humidité du sol), un module de communication LoRa pour la transmission longue distance, un écran OLED pour l'affichage local, ainsi qu'un relais et une pompe pour le contrôle automatique de l'irrigation. Le système a été enrichi par une interface mobile via l'application Blynk, permettant un suivi à distance par l'utilisateur.

Les résultats obtenus à travers les tests réalisés ont démontré la faisabilité technique du système proposé, sa fiabilité et sa capacité à répondre aux besoins d'automatisation en milieu agricole. De plus, l'utilisation de technologies open source et à faible coût le rend accessible aux exploitations de petite et moyenne taille.

Cependant, certaines limites subsistent, notamment la précision des capteurs choisis, la dépendance à la connectivité Wi-Fi pour l'application Blynk, ainsi que l'absence d'autonomie énergétique. Ces contraintes ouvrent des perspectives d'amélioration, telles que l'intégration de capteurs plus robustes, l'utilisation de l'énergie solaire, ou encore le déploiement de réseaux LoRaWAN avec serveurs locaux pour une plus grande autonomie.

En conclusion, le système conçu représente une solution efficace, évolutive et économiquement viable pour soutenir le développement de l'agriculture intelligente, contribuant ainsi à une meilleure gestion des ressources naturelles et à l'amélioration des rendements agricoles.



---

## **Annexe**

---

## Programme de l'application BLYNK

```
#define BLYNK_TEMPLATE_ID "TMPL2UFbbr5F1"
#define BLYNK_TEMPLATE_NAME "Smart Agriculture"
#define BLYNK_AUTH_TOKEN "Pl986-at87-XblBkTHHTpitBB1D8D5ZG"

#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#include "DHT.h"

// Identifiants Wi-Fi
const char* ssid = "Galaxy M04 8850";
const char* password = "efcu6148";

// Définition du capteur DHT
#define DHTPIN 13
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

// Définition du capteur d'humidité du sol
#define SOIL_PIN 35 // Broche analogique GPIO35

// Démarrer le timer Blynk
BlynkTimer timer;

// Fonction pour envoyer les données vers Blynk
void sendSensor()
{
    // Lecture DHT11
    float h = dht.readHumidity();
    float t = dht.readTemperature();

    if (isnan(h) || isnan(t)) {
        Serial.println("Échec de lecture du capteur DHT !");
    } else {
        Serial.print("Humidité Air : ");
        Serial.print(h);
        Serial.print(" % | Température : ");
        Serial.print(t);
        Serial.println(" °C");

        Blynk.virtualWrite(V0, t); // Température sur V0
        Blynk.virtualWrite(V1, h); // Humidité de l'air sur V1
    }

    // Lecture humidité du sol
    // Lecture humidité du sol
    int soilValue = analogRead(SOIL_PIN);
```

```
// Remplacer 3500 et 1200 par vos valeurs réelles mesurées
int soilPercent = map(soilValue, 3500, 1200, 0, 100);
soilPercent = constrain(soilPercent, 0, 100); // Assurer une limite entre 0 et 100

Serial.print("Humidité du sol : ");
Serial.print(soilPercent);
Serial.println(" %");

Blynk.virtualWrite(V2, soilPercent);

}

void setup()
{
  Serial.begin(115200);
  dht.begin();

  Blynk.begin(BLYNK_AUTH_TOKEN, ssid, password);

  // Lire les capteurs toutes les 2 secondes
  timer.setInterval(2000L, sendSensor);
}

void loop()
{
  Blynk.run();
  timer.run();
}
```

## Programme de l'émetteur

```
#include <SPI.h>
#include <LoRa.h>
#include <DHT.h>

// Définir les broches du capteur DHT11
#define DHTPIN 4
#define DHTTYPE DHT11
DHT dht(DHTPIN, DHTTYPE);

// Définir les broches du module LoRa
#define SS 5
#define RST 15
#define DIO0 16

void setup() {
  // Initialisation du port série
  Serial.begin(9600);
  while (!Serial);

  Serial.println("Initialisation du capteur DHT11...");
  dht.begin();
  delay(2000); // délai nécessaire pour DHT11

  // Initialisation du module LoRa
  Serial.println("Initialisation du module LoRa...");
  LoRa.setPins(SS, RST, DIO0);

  if (!LoRa.begin(433E6)) {
    Serial.println("Échec de l'initialisation LoRa !");
    while (1);
  }

  Serial.println("LoRa initialisé avec succès !");
}

void loop() {
  Serial.println("Lecture des données du capteur DHT11...");

  float temperature = dht.readTemperature();
  float humidite = dht.readHumidity();

  if (isnan(temperature) || isnan(humidite)) {
    Serial.println("Erreur de lecture du capteur DHT11 !");
    delay(2000);
    return;
  }

  // Format des données à envoyer
```

```
String data = "T:" + String(temperature) + "C H:" + String(humidite) + "%";

// Affichage sur le moniteur série
Serial.print("Envoi de données via LoRa : ");
Serial.println(data);

// Envoi via LoRa
LoRa.beginPacket();
LoRa.print(data);
LoRa.endPacket();

delay(5000); // Attente de 5 secondes avant prochaine lecture
}
```

## Programme du Récepteur

```
#include <SPI.h>
#include <LoRa.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);

// LoRa SX1278
#define SS 5
#define RST 14
#define DIO0 26

void setup() {
  Serial.begin(9600);

  // Initialisation de l'OLED
  if (!display.begin(SSD1306_SWITCHCAPVCC, 0x3C)) {
    Serial.println("Échec OLED");
    while (true);
  }

  display.clearDisplay();
  display.setTextSize(1);
  display.setTextColor(WHITE);
  display.setCursor(0, 0);
  display.println("Initialisation...");
  display.display();

  // Initialisation LoRa
  LoRa.setPins(SS, RST, DIO0);
  if (!LoRa.begin(433E6)) {
    display.clearDisplay();
    display.setCursor(0, 0);
    display.println("Échec LoRa !");
    display.display();
    while (true);
  }

  display.clearDisplay();
  display.setCursor(0, 0);
  display.println("LoRa OK");
  display.display();
  delay(1000);
}
```

```
void loop() {
  int packetSize = LoRa.parsePacket();
  if (packetSize) {
    String received = "";
    while (LoRa.available()) {
      received += (char)LoRa.read();
    }

    Serial.print("Reçu : ");
    Serial.println(received);

    // Analyse des données reçues : "T:xx.x H:yy.y"
    float temp = 0.0;
    float hum = 0.0;
    int tIndex = received.indexOf("T:");
    int hIndex = received.indexOf("H:");
    if (tIndex != -1 && hIndex != -1) {
      temp = received.substring(tIndex + 2, hIndex).toFloat();
      hum = received.substring(hIndex + 2).toFloat();
    }

    // Heure (millis converti en secondes)
    unsigned long seconds = millis() / 1000;

    // Affichage OLED
    display.clearDisplay();
    display.setCursor(0, 0);
    display.println("== Donnees Lora ==");
    display.print("Temp: ");
    display.print(temp);
    display.println(" C");

    display.print("Hum: ");
    display.print(hum);
    display.println(" %");

    display.print("Heure: ");
    display.print(seconds);
    display.println(" s");

    display.display();
  }
}
```



## Références Bibliographiques

1. Gubbi, J. et al. (2013). "Internet of Things (IoT): A vision, architectural elements, and future directions". Future Generation Computer Systems.
2. Xu, L.D., He, W., & Li, S. (2014). "Internet of Things in Industries: A Survey". IEEE Transactions.
3. Madakam, S., Ramaswamy, R., & Tripathi, S. (2015). "Internet of Things (IoT): A Literature Review". Journal of Computer and Communications.
4. Roman, R. et al. (2013). "Securing the Internet of Things: New Challenges and Solutions". Computer Communications.
5. Zanella, A. et al. (2014). "Internet of Things for Smart Cities". IEEE Internet of Things Journal.
6. Atzori, L., Iera, A., & Morabito, G. (2010). "The Internet of Things: A survey". Computer Networks.
7. Wolfert, S. et al. (2017). "Big Data in Smart Farming – A review". Agricultural Systems.
8. Zhang, C., & Kovacs, J. (2012). "The application of small unmanned aerial systems for precision agriculture". Precision Agriculture.
9. Nasir, M.H.A. et al. (2018). "Agricultural Robotics: A review of the current trends and future directions". Computers and Electronics in Agriculture.
10. Raza, U. et al. (2017). "Low Power Wide Area Networks: A Survey". IEEE Communications Surveys & Tutorials.
11. Semtech Corporation. "LoRa Technology Overview". [En ligne] [www.semtech.com](http://www.semtech.com)
12. LoRa Alliance. "LoRaWAN Specification". [En ligne] [www.lora-alliance.org](http://www.lora-alliance.org)
13. Vuran, M.C. et al. (2018). "Internet of Underground Things in Precision Agriculture: Architecture and Technology Aspects". Ad Hoc Networks.
14. Arduino.cc, Documentation officielle de l'IDE Arduino.
15. Espressif Systems. ESP32 Technical Reference Manual.
16. NodeMCU ESP8266 Datasheet.
17. Semtech. SX1278 Datasheet and LoRa Technology Overview.
18. Aosong Electronics. DHT11 Sensor Datasheet.

