



المعهد العالي للدراسات التكنولوجية ببرادس
قسم الهندسة الكهربائية

Rapport de Stage de Fin d'études

LICENCE EN GENIE ELECTRIQUE

Smart Garden

Entreprise d'accueil : Ecole Nationale d'Ingénieurs de Tunis (ENIT)



Réalisé par : Nour TRABELSI

Ahmed Aziz GADDOUR

Encadré par :

Encadreur ENIT : M. Férid KOURDA

Encadreur ISET : M. Faouzi FAZZENI

Année universitaire : 2021/2022

Code : AII9 /2022

Dédicace

A mes chers parents

Les mots me manquent pour exprimer toute la reconnaissance, la fierté et profond amour que je vous porte pour les sacrifices, et les prières que vous n'avez jamais cessé de consentir pour mon instruction et ma réussite. Que Dieu tout puissant vous garde et vous procure santé, bonheur et longue vie pour que vous demeuriez le flambeau illuminant mon chemin.

A mes sœurs et à toute ma famille

Pour votre soutien pendant toutes ces longues années d'études.
Que vous trouveriez dans ce travail, le témoignage d'un amour éternel.

À mes chers ami(e)s et mes camarades en L3All

Pour tous les magnifiques instants que nous avons vécus ensemble

A tous ceux qui me sont chers

Je dédie mon travail...

Trabelsi Nour

Dédicace

A ma très chère mère, affable, honorable, aimable : vous représentez pour moi le symbole de la bonté par excellence, la source de tendresse et l'exemple de dévouement qui n'a pas cessé de m'encourager et de prier pour moi.

A mon père, dont le mérite, les sacrifices et les qualités humaines m'ont permis de vivre ce jour. Qu'il trouve en moi la source de sa fierté. A qui je dois tout.

A ma sœur et mon frère, que dieu les garde et les protège, que leurs chemins soient pleins de succès et de bonheur.

A tous mes amis, qui m'ont soutenu et encouragé durant toute cette période.

A tous mes enseignantes et enseignants, j'espère à voir répondu à vos souhaits de me voir réussir et le témoignage de ma profonde reconnaissance et de mon cœur infini.

A toutes les personnes qui ont participé à l'élaboration de ce travail, à tous ceux que j'ai omis de citer.

Gaddour Ahmed Aziz

REMERCIEMENTS

C'est avec un immense plaisir que nous commençons ce rapport de stage par remercier chaleureusement l'Ecole Nationale d'Ingénieurs de Tunis (ENIT).

Nos remerciements s'adressent en premier lieu à Mr Férid KOURDA, directeur du laboratoire des systèmes électriques et notre encadreur de stage au sein du laboratoire pour son accueil, ses conseils, la confiance qu'il nous a accordée dès notre arrivée au laboratoire et pour l'expérience enrichissante et pleine d'intérêt qu'il nous a fait vivre durant ces mois.

En second lieu, Nous tenons à remercier Mr Faouzi FEZZANI, chef du département électrique et notre encadreur de l'ISET de Radés pour le temps qu'il nous a consacré tout au long de cette période.

Nous remercions également l'ensemble des employés et toute l'équipe professionnelle de l'école d'ingénieur de Tunis pour leur accueil sympathique et leur coopération tout au long de notre stage et aux personnes qui nous ont apporté leur afin de mener ce projet et d'élaborer ce rapport.

Finalement, nous sommes honorés de remercier l'institut supérieur des études technologiques de Rades (ISET RADES) qui nous a donnée l'occasion de vivre cette expérience enrichissante aussi bien sur le plan professionnel qu'humain.

SOMMAIRE

INTRODUCTION GENERALE **1**

CHAPITRE 1 : CADRE DE PROJET **2**

INTRODUCTION.....	3
I. PRESENTATION DE L'ENVIRONNEMENT DU TRAVAIL	3
1. <i>École Nationale d'Ingénieurs de Tunis</i>	3
2. <i>Cahier des charges.....</i>	4
2.1. Présentation générale de projet « Smart Garden » ou le jardin intelligent.....	4
2.2. Mise en situation	4
2.3. Problématique.....	5
2.4. Etude théorique	5
3. <i>Solution retenue.....</i>	7
4. <i>Les étapes du projet.....</i>	7
II. LES DEFIS CENSES ETRE RELEVES	8
1. <i>IOT</i>	8
2. <i>Smart Garden</i>	8
3. <i>Les capteurs.....</i>	8
4. <i>Les cartes électroniques</i>	9
5. <i>Les composants électroniques.....</i>	10
6. <i>Les batteries</i>	10
7. <i>Le BMS.....</i>	11
8. <i>Réseau de communication.....</i>	11
9. <i>Pluviométrie en Tunisie.....</i>	12
10. <i>Calcul du volume d'un réservoir.....</i>	14
CONCLUSION	15

CHAPITRE 2 : ÉTUDE MATERIELLE ET LOGICIELLE..... **16**

INTRODUCTION.....	17
I. ÉTUDE MATERIELLE	18
1. <i>Périphériques d'alimentation.....</i>	18
1.1. Panneaux Photovoltaïque	18
1.2. Batteries Plomb-Acide 12V \ 17,2Ah	19
2. <i>Les Cartes électroniques</i>	19

2.1.	Chargeur PC régulateur 20V	19
2.2.	Chargeur téléphone régulateur 5V	22
2.3.	Protection contre les courts-circuits	24
2.4.	Circuit de la protection contre la haute température	25
2.5.	Carte Relais	26
2.6.	Carte Commande.....	27
3.	<i>Choix des capteurs</i>	28
3.1.	Capteur de température et humidité ambiante.....	28
3.2.	Capteur de niveau (capteur à ultrasons)	28
3.3.	Détecteur de niveau (Flotteur).....	29
3.4.	Humidité du sol	29
4.	<i>Les pré-actionneurs et Les actionneurs</i>	29
4.1.	Les pré-actionneurs	29
4.2.	Les actionneurs.....	30
5.	<i>L'unité de commande</i>	32
5.1.	Choix du microcontrôleur	32
5.2.	Comparaison des différentes cartes à bases de microcontrôleurs	32
5.3.	Présentation du ESP WROOM-32D	33
5.4.	Caractéristique du ESP WROOM-32D	33
5.5.	Réseau de communication.....	33
6.	<i>Bilan du projet</i>	33
6.1.	Calcul énergétique	33
6.2.	Cout approximatif	34
II.	LOGICIELS	35
1.	<i>Proteus ISIS</i>	35
2.	<i>Arduino IDE</i>	36
3.	<i>App Inventor</i>	36
4.	<i>SolidWorks</i>	37
5.	<i>Firebase</i>	37
	CONCLUSION	37
	CHAPITRE 3 : PRESENTATION DE LA SOLUTION PROPOSEE	38
	INTRODUCTION	39
I.	PRESENTATION GENERALE DE NOTRE SOLUTION	39
II.	LA PARTIE HARDWARE	40

1. Conception des cartes électroniques	40
1.1. Conception de la carte chargeur 5V	40
1.2. Conception de la carte chargeur 20V	40
1.3. Conception de la carte de commande.....	41
2. Câblage des composants	42
2.1. Organigramme de fonctionnement du système	42
2.2. Câblage des batteries	43
2.3. Câblage de l'ESP 32.....	44
2.4. Câblage du module Relais	45
2.5. Capteur température DHT11	46
2.6. Capteur de niveau (capteur ultrason)	47
2.7. DéTECTEUR de niveau (capteur flotteur).....	48
2.8. Pompe à eau	48
2.9. Conception de la vanne motorisée.....	50
2.10. Câblage des lampes	52
III. LA PARTIE SOFTWARE	52
1. <i>Architecture de la plateforme</i>	52
2. <i>La Configuration de Firebase</i>	53
3. <i>L'application Android « ENIT SMART GARDEN »</i>	54
3.1. Développement l'application Android ENIT Smart Garden	54
3.2. Guide d'utilisateur	58
4. <i>Programmation de l'ESP-32</i>	58
4.1. Connexion a un point d'accès WIFI et Firebase.....	58
4.2. Déclarations des variables	59
4.3. Obtention du date et heure	60
4.4. Système d'irrigation.....	61
4.5. Système de gestion des batteries	62
4.6. Système de Gestion du réservoir 1	63
4.7. Commande de la pompe	63
4.8. Système d'éclairage du jardin	64
4.9. Supervision de la température et d'humidité	65
CONCLUSION	65
CONCLUSION GENERALE	66

Liste de tableaux

Tableau 1 : Réseau de communication.....	12
Tableau 2 : Comparaison de différentes cartes à bases de microcontrôleurs	32
Tableau 3 : Caractéristique du ESP32	33
Tableau 4 : Estimation des besoins énergétique journaliers pour Smart Garden.....	34
Tableau 5 : Bilan économique.....	35
Tableau 6 : Câblage des entrées et sorties du microcontrôleur ESP32	41
Tableau 7 : Tableau de connexion des batteries avec ESP32	43
Tableau 8 : Tableau connexion carte relais avec ESP32*	46
Tableau 9 : Tableau de connexion de DHT11 avec ESP32	46
Tableau 10 : Tableau de connexion de HC-SR04 avec ESP32.....	47
Tableau 11 : Tableau de connexion de Flotteur avec ESP32	48
Tableau 12 : Tableau de connexion de Servo avec ESP32	51
Tableau 13 : Tableau de connexion des lampes avec ESP32.....	52
Tableau 14 : Le cycle d'irrigation.....	61

Liste de figures

Figure 1 : Logo ENIT	3
Figure 2 : ENIT SMART GARDEN.....	4
Figure 3 : Diagramme SADT du système	5
Figure 4 : Diagramme Pieuvre	6
Figure 5: Diagramme ombrothermique TUNIS	13
Figure 6 : Tableau climatique de Tunis.....	14
Figure 7 : Schéma explicatif.....	17
Figure 8 : Constitution de l'espace	18
Figure 9 : Panneau Photovoltaïque.....	18
Figure 10 : Batterie.....	19
Figure 11 : Régulateur LM338T	20
Figure 12 : schéma d'application du régulateur LM338T.....	21

Figure 13 : schéma du régulateur LM338T avec ajout des capacités et d'une diode de protection.....	22
Figure 14 : Régulateur 7805.....	23
Figure 15 : schéma du régulateur 7805 avec ajout de diode de protection	23
Figure 16 : circuit de protection contre les courts-circuits pour le régulateur 5V	24
Figure 17 : circuit de protection contre les courts-circuits pour le régulateur 20V	25
Figure 18 : circuit de protection contre une haute température.....	26
Figure 19 : circuit de la carte de Relais	26
Figure 20 : Arduino Module d'alimentation 3,3V/5V	28
Figure 21 : Capteur DHT11	28
Figure 22 : Capteur ultrason HC-SR04	28
Figure 23 : Module relais 4 canaux	30
Figure 24 : Lampes.....	30
Figure 25 : Mini pompe d'eau	31
Figure 26 : Servo moteur MG995 et pièce de raccord	31
Figure 27 : Le microcontrôleur ESP 32	33
Figure 28 : Logo Proteus ISIS	36
Figure 29 : Logo IDE Arduino	36
Figure 30 : Logo MIT APP INVENTOR	36
Figure 31 : Logo SolidWorks	37
Figure 32 : Logo Firebase	37
Figure 33 : Organigramme de fonctionnement du système	42
Figure 34 : Câblage des batteries.....	43
Figure 35 : Organigramme de fonctionnement du système BMS	44
Figure 36 : Architecture de ESP 32D	45
Figure 37 : Relai 1	46
Figure 38 : Câblage de capteur température DHT11	47
Figure 39 : Câblage de capteur de niveau ultrason	48
Figure 40 : Câblage de la pompe.....	50
Figure 41 : Conception de la vanne motorisée	51

Figure 42 : câblage des servos.....	51
Figure 43 : Câblage des lampes.....	52
Figure 44 : Architecture de la plateforme	53
Figure 45 : Firebase d'ENIT SMART GARDEN.....	53
Figure 46 : l'application Android ENIT Smart Garden	55
Figure 47 : blocs App Inventor	56
Figure 48 : base donnée de SMART GARDEN	57

Introduction générale

L'Internet des objets est un système d'appareils connectés, d'ordinateurs et de machines numériques avec des identifiants uniques qui transfèrent des données sur un réseau. Les smartphones, les ordinateurs portables, les capteurs etc.... font tous partie de l'Internet des objets tant qu'ils connectent et partagent des données.

Dans l'usage domestique, il est invoqué dans les domotiques ou maisons intelligentes, les jardins intelligents et même dans l'agriculture intelligente.

Dans le cadre de notre stage de fin d'étude obligatoire de la troisième année licence en électrique spécialité automatisme et informatique industrielle, nous étions amené à travailler sur la création d'un projet intitulé « **ENIT Smart Garden** » dont l'objectif principale est de développer une solution électronique pour connecter un jardin intelligent à un serveur distant à l'aide de plusieurs outils numérique, de petits systèmes embarquées à base des microcontrôleurs, des circuits électroniques et des moyens de télécommunications pour permettre le contrôle et la supervision à distance des équipements électriques, électroniques et informatiques.

Ce projet est piloté par le projet PAQ Promesse Collabora financé par la banque mondiale, qui est en complément du projet PAQ « Smart House » sous la responsabilité de M. Le professeur Férid KOURDA directeur du laboratoire des systèmes électriques à l'ENIT.

Notre rapport va décrire les étapes de la réalisation du système, composé de trois chapitres :

Dans **le chapitre 1**, nous ferons la présentation générale du projet, nous exposerons l'environnement de travail, la problématique, la solution retenue et les défis censés être relevés.

Dans **le chapitre 2**, nous parlerons de la partie Hardware du projet, des matériels nécessaires et des logiciels utiliser.

Dans **Le chapitre 3**, nous mettrons en évidence le travail accompli et son développement tout en citant les parties nécessaires à la réalisation du projet.

Enfin, une conclusion générale et des perspectives suivie d'annexes viendrait clôturer ces travaux.

Chapitre 1 : Cadre de projet

Introduction

Cette partie est consacrée dans un premier temps à la présentation de l'établissement d'accueil ENIT (Ecole Nationale d'Ingénieurs de Tunis) qui a mis à disposition tout le matériel nécessaire pour la réussite du stage. Nous exposerons ainsi le cadre de projet et sa problématique et finalement présenter la solution retenue et les étapes de projet à effectuer.

I. Présentation de l'environnement du travail

1. École Nationale d'Ingénieurs de Tunis



Figure 1 : Logo ENIT

L'École nationale d'ingénieurs de Tunis ou ENIT est une école d'ingénieurs tunisienne, la plus ancienne du pays après l'Institut national agronomique de Tunisie. Dépendante de l'université de Tunis - El Manar, elle se situe au sein du campus d'El Manar à Tunis.

Situation : Université de Tunis - El Manar

Adresse : Rue Béchir Salem Belkhiria Campus universitaire, BP 37, 1002,

Le Belvédère, 1002, Tunis

Téléphone : 71 874 700

Premier directeur : M. Mokhtar Laatiri

Création : 1968

Directeur 2022 : Pr Kamel BEN SAAD

2 Noms officiels المدرسة الوطنية للمهندسين بتونس : et École Nationale d'Ingénieurs de Tunis

2. Cahier des charges

2.1. Présentation générale de projet « Smart Garden » ou le jardin intelligent

Dans le département Génie électrique est situé notre jardin intelligent. Il a pour but d'offrir un endroit de travail agréable en plein air. Ce projet est piloté par le projet PAQ Promesse Collabera financé par la banque mondiale, qui est en complément du projet PAQ « Smart House » sous la responsabilité de M. Le professeur Férid KOURDA.

2.2. Mise en situation

Notre projet se présente sous la forme d'un jardin qu'on va transformer en un jardin intelligent, c'est la traduction de son nom "SMART GARDEN". La figure 2 illustre l'espace de notre travail.



Figure 2 : ENIT SMART GARDEN

L'année dernière, les étudiants de la promotion précédente de l'ENIT ont pris l'initiative à travailler sur ce projet mais les travaux n'ont pas été réalisés. Des spécialistes ont été installés, une grande table avec des bancs fixes en bois situés sous un toit équipé par 4 lampes LED.

Sur le toit, ils ont installé un panneau voltaïque qui capte l'énergie solaire lié à un boîtier pour gérer et calculer les besoins en électricité .

Sur la table, sont fixés 2 interrupteurs, chacun gère la mise en marche et la fermeture des lampes. A côté des lampes, nous avons prévu la mise en place de deux ports universels, un pour un chargeur téléphone et l'autre pour chargeur de PC.

Tout au long des côtés de cet espace, Ils ont été installés des tubes en PVC pour planter des fleurs qui vont être arrosées d'une façon automatique tout au long de l'année grâce à un programme qu'on va intégrer. Le fonctionnement de l'arrosage est réalisé à l'aide des tubes liés à des bidons de 20 L qui rassemblent l'eau de pluie. La répartition de l'eau se fait par des trous situés tout au long de ce dernier d'une façon équitable.

2.3. Problématique

Les défis à relever dans notre jardin sont :

- **Le problème d'arrosage** : Les arbustes plantés dans le jardin doivent être irriguées de temps en temps d'une manière fréquente sans l'intervention de l'opérateur et ceci aux grés des saisons et de la température.
- **Problème d'alimentation électrique** : Les personnes qui restent dans le jardin doivent profiter d'un chargeur de PC et de téléphone et d'un éclairage la nuit.
- **Problème d'éclairage** : Le jardin doit être équipé des lampes pour aider les étudiants à réviser en fin d'après-midi et fournir une place favorable aux étudiants pour se réunir après les cours.

2.4. Etude théorique

2.4.1. La méthode SADT

La méthode SADT (Structured Analysis and Design Technique) est un outil graphique associé à une méthode d'analyse descendante modulaire et hiérarchisée (Design se traduit ici par conception). Il permet de représenter un modèle (image de la réalité) du système réel.

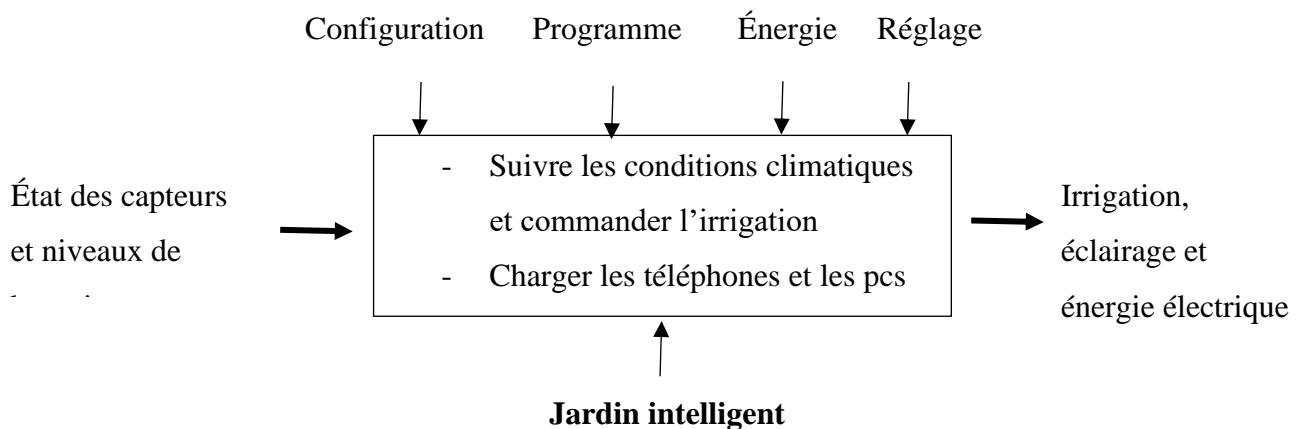


Figure 3 : Diagramme SADT du système

2.4.2. Le diagramme pieuvre

Le diagramme pieuvre ou graphe des interactions est un schéma qui représente la relation entre un produit/service et son environnement. C'est un outil d'analyse utilisé dans le cadre de la méthode APTE (Application aux Techniques d'Entreprise). Le diagramme pieuvre permet de représenter les fonctions de service d'un produit.

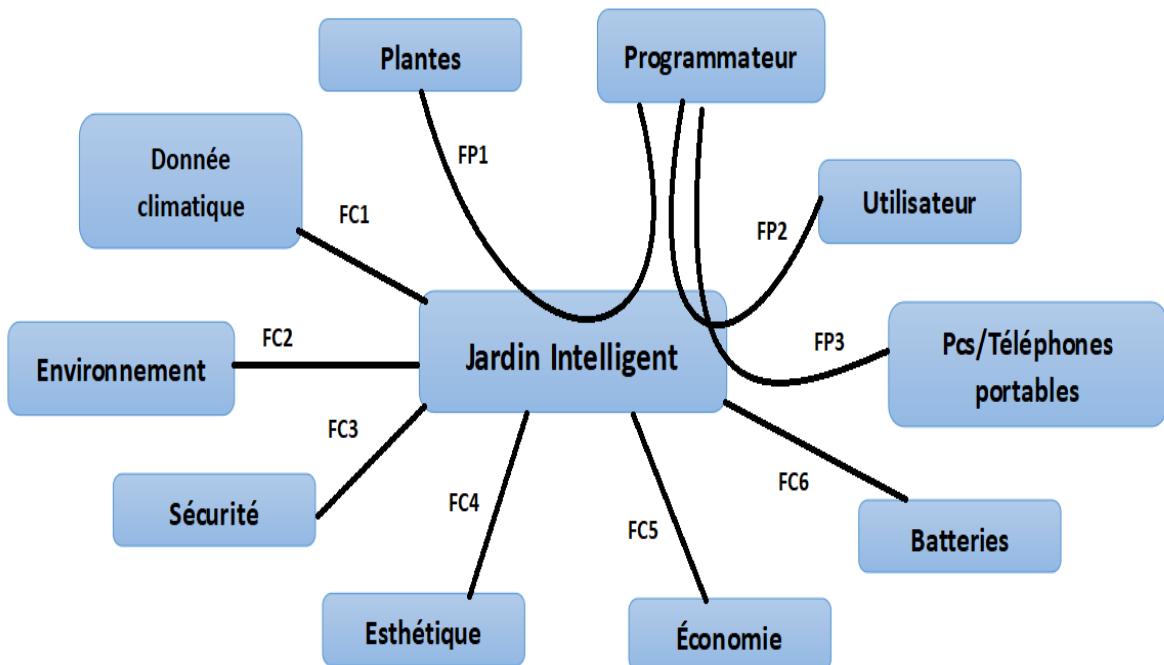


Figure 4 : Diagramme Pieuvre

Recensement des fonctions de services :

FP : Fonction Principale :

- FP1 : Arroser les plantes sans intervention humaine.
- FP2 : Eclairer l'espace pour l'utilisateur.
- FP3 : Charger les PCS et les téléphones portables.

FC : Fonction Complémentaire :

- FCI : Capter les données climatiques.
- FC2 : Respecter l'environnement.
- FC3 : Favoriser la sécurité publique aux utilisateurs.
- FC4 : Plaire à l'œil.
- FCS : Être écologique.
- FC6 : Être alimenté par des batteries

3. Solution retenue

Les nouvelles techniques émergentes d'internet des objets ont permis de développer de nombreuses nouvelles opportunités et de nouvelles applications dans le domaine des systèmes intelligents. Notre projet, "Smart Garden" est une des utilisations de l'IOT dans le domaine de la domotique.

Notre système peut fonctionner indépendamment de l'opérateur. Il est équipé d'un mécanisme d'irrigation automatique. Un microcontrôleur peut piloter des électrovannes permettront l'arrosage commandé des plantes. Un capteur de température envoie les informations en temps réels vers notre **application** qui permet à l'utilisateur de suivre l'état de son jardin à travers une application mobile.

Pour bien garantir le bien-être des étudiants et d'enseignants qui choisissent de passer leurs temps dans cet endroit, le jardin doit être équipé de deux chargeurs de téléphones portables ainsi que deux chargeurs de PC. Grâce à un système d'éclairage, notre Smart Garden peut être un endroit favorable pour passer une fin d'après-midi et pour réviser, par exemple.

Nous avons constaté, que les étudiants qui arrivent tôt à l'ENIT vont attendre au Smart-Garden. Lors du déjeuner (entre midi et 14h) il n'y a presque plus de place pour s'asseoir.

L'alimentation du jardin est indépendante du secteur, les panneaux photovoltaïques installés permettent d'alimenter quatre batteries en Plomb, deux montées en parallèle et deux en série qui fournissent l'énergie électrique au reste du système. Ainsi on aura deux niveaux de tension, +12V et +24V qui fournissent l'énergie électrique nécessaire.

Pour s'assurer que la tension au niveau de chaque batterie se situe dans une fourchette de 12V \pm 1V un système de BMS : Batteries management System est intégré au sein de notre jardin. Ceci permet l'arrêt immédiat de la consommation électrique en cas d'une baisse de tension au niveau d'une batterie à 11V avec une signalisation de défaut (buzzer, LED qui clignote) ou l'arrêt de charge par le PV si la tension monte à 13 V.

4. Les étapes du projet

Le projet consiste à élaborer un jardin aménagé avec un espace de travail à l'ENIT, il compte d'installer un système intelligent en vue d'assurer :

- La charge de batterie par panneaux photovoltaïque avec une BMS (battery management system).
- L'éclairage de l'espace de travail qui sera muni d'une minuterie électronique.
- La charge de deux téléphones par batteries.
- La charge universelle de deux PC simultanés par batteries.

- La gestion de la récolte d'eau en vue de l'arrosage automatique durant toute l'année du jardin aménagé.
- Réalisation d'une carte commande contient le microcontrôleur ESP32 afin d'assurer la communication à distance WIFI en vue de la récolte des données (niveau des batteries, température et humidité, réserve d'eau, instants d'irrigation ...), commande à distance d'actionneur électrique (pompe, électrovanne, éclairage ...) et visualisation des données de température et niveau d'eau sous formes d'histogrammes.

II. Les défis censés être relevés

1. IOT

L'Internet of Things (IoT) décrit le réseau de terminaux physiques, les « objets », qui intègrent des capteurs, des logiciels et d'autres technologies en vue de se connecter à d'autres terminaux et systèmes sur Internet et d'échanger des données avec eux.

2. Smart Garden

Le projet de jardin intelligent est un jardin devant fonctionner de manière autonome sans intervention humaine. L'objectif est d'automatiser les actions du jardinier de manière à faire pousser les plantes tout en faisant face au changement continu des facteurs externes.

Ses principaux objectifs sont :

- Booster un mode de vie Bio
- Atteindre un haut niveau d'Autonomie
- Manipuler un maximum de paramètres
- Améliorer progressivement une pousse optimale des plantes

3. Les capteurs

L'irrigation automatique suppose la connaissance parfaite des conditions ambiantes afin de contrôler de façon efficace les modules automatiques d'irrigation à savoir la température, la luminosité et l'humidité.

- **La température ;**

La température représente la mesure de l'énergie cinétique moyenne des particules dans un échantillon de matériau, exprimé en degré Celsius. La température est le facteur le plus déterminant de la production agricole. Il existe plusieurs méthodes différentes pour mesurer la

température pour mesurer la température, et ces méthodes diffèrent par le coût et la précision de l'équipement.

- **L'humidité :**

Humidité ambiante

Toute comme la température et la lumière, cela affecte certains mécanismes végétaux et affecte la productivité et la rentabilité des cultures. La réussite des cultures dépend fortement de l'air qui est relatif à l'humidité ambiante. De façon générale, une humidité relative de 40 à 60 % convient à la majorité des plantes.

Humidité du sol

L'humidité du sol joue un rôle important dans le maintien de la vie sur la terre. Sa première utilisation est de permettre la croissance de la végétation. Elle nous renseigne sur le besoin en eau de la plante.

- **Détection de niveau :**

Un capteur de niveau est un dispositif électronique qui permet de mesurer la hauteur du matériau, en général du liquide, dans un réservoir ou un autre récipient. Les capteurs de niveau se divisent en deux types principaux :

Capteurs de niveau qui permettent de mesurer la variation de la hauteur du niveau. Il renvoie un signal proportionnel à la hauteur exacte du niveau du matériau.

Le détecteur de niveau indique uniquement si le niveau atteint ou non une certaine hauteur, on parle ici de mesure ponctuelle. Un détecteur ne renvoie donc pas un signal proportionnel à la hauteur du niveau mais un signal de commutation (type tout ou rien, 0 ou 1).

4. Les cartes électroniques

Une carte électronique est un ensemble de composants tel que des résistances, condensateurs ou circuits intégrés réunis sur une plaque de manière à former un circuit destiné à un usage précis. [1]

La conception réussie d'une carte électronique suit plusieurs étapes indispensables :

La conception sur ordinateur

- Réalisation du schéma
- Router le schéma électrique

Fabrication du circuit imprimé

- Imprimer le typon
- Insolation de la plaque époxy
- La révélation

- Graver le circuit imprimé

Mise en place et soudure des composants

- Les composants traditionnels :

- Percer le circuit
- Souder les composants

- Les composants SMD :

- Dépôt de pâte
- Placement des composants
- Soudure

5. Les composants électroniques

- **Protection inverse de polarité :**

La protection contre les inversions de polarité s'effectue à l'aide d'une diode parallèle inversée par rapport à la tension de service et un fusible rapide censé fondre avant que la diode ne casse.

- **Résistance de rappel :**

Nous sommes entourés d'ondes électromagnétiques. Les ondes de la transmission de la radio, de la télé, etc. le courant électrique aussi émet des ondes électromagnétiques.

Ces ondes électromagnétiques induisent des tensions parasites entre les broches des composants électroniques.

C'est pour cela qu'une broche connectée à rien, n'est pas forcément à l'état 0.

Une résistance de pull up fixe l'état à HIGH (état haut).

Une résistance de pull down fixe l'état à LOW (état BAS).

- **Condensateur de découplage :**

C'est un condensateur destiné à réduire les petites perturbations aux niveaux des pins de connections. Le condensateur relie le canal d'alimentation à la masse pour abaisser son impédance dans la bande passante du signal.

On n'a pas besoin d'une grande précision sur la valeur du composant. Celle-ci dépend de l'intensité qui traverse le canal et de la fréquence du signal.

6. Les batteries

Les batteries rechargeables sont largement utilisées dans les systèmes photovoltaïques autonomes pour emmagasiner le surplus d'énergie et pour alimenter les charges de faible et moyenne puissance. Les batteries de type plomb - acide sont de loin les plus employées en

raison de leur faible coût, leur simple maintenance et leur adaptation à tout type d'application. Ces batteries sont cependant si fragiles devant les phénomènes de surcharge, de décharge profonde et les dérives de température et de courant qu'il faut leur associer un régulateur de charge approprié afin d'assurer leur protection.

L'importance d'un régulateur de charge dans un système photovoltaïque autonome est indispensable. Il doit être réalisé avec le plus grand soin pour répondre aux exigences de coût, de simplicité, de généralité et de fiabilité. [2]

7. Le BMS

Le BMS (Battery Management System) est un système de gestion de batterie qui sécurise et recharge les batteries. Il permet de surveiller la tension basse des cellules, de gérer l'équilibrage à la recharge et de mesurer la température de la batterie. Ces paramètres sont importants afin d'avoir un bon fonctionnement de la batterie et d'augmenter sa durée de vie. L'objectif ici sera de réaliser un gestionnaire de batterie sur table qui puisse par la suite permettre à l'utilisateur de voir le niveau de la batterie de sa voiture et de contrôler la charge et la décharge de la batterie.

La conception d'un BMS commence par le choix de la technologie de la batterie.

8. Réseau de communication

Afin d'assurer une communication entre les différentes parties de ce projet, il est nécessaire de mettre en place une solution qui s'adapte à nos spécifications et à nos besoins. En effet, les réseaux sans fil sont très utilisés aujourd'hui et se manifestent sous plusieurs technologies (Lora, Wifi, ZigBee, Z-Wave, Bluetooth, NFC ...) qui se diffèrent par leurs fréquences, leurs modulation leurs portées et leurs débits.

Tech	Support physique	Portée	Débit	Géolocalisation	Avantages	Inconvénients	Applications
LoRaWan	Sans fil	Longue	Bas	Jusqu'à plusieurs centaines de mètres	Bas débit, faible puissance, prix bas pour l'appareil et l'abonnement, longue portée, économie de la batterie	Utilisation d'un modem spécifique, itinérance en cours de déploiement, ne gère pas la mobilité des objets	Adapté à la télémetrie à bas débit. Ce qui englobe généralement l'IoT des mesures

Wifi	Sans fil	Courte	Très Haut	Jusqu'à plusieurs mètres	Très haut débit, qualité du signal assurée, connexion simple et rapide à la passerelle	Non adapté aux objets uniquement alimentés par des batteries, couverture réseau limitée à une faible zone autour de la passerelle, une passerelle Wifi mal configurée expose le réseau à des failles de sécurité	Adapté aux applications de domotique, au contexte indoor
	Sans fil	Courte	Haut	Jusqu'à plusieurs mètres	Haut débit, faible portée, quasiment intégré dans tous les appareils du quotidien	Si la sécurité n'est pas bien configurée alors l'objet s'expose à des failles de sécurité	Adapté aux technologies portables telles que les oreillettes pour les communications voix et aux raccordements d'équipement à un Pc
	Sans fil	Courte	Moyen	Jusqu'à plusieurs mètres	Technologie peu consommatrice en énergie et s'intègre à bas coût dans les équipements	Achat d'appareils spécifiques car la technologie n'est pas disponible dans les smartphones et PC	Adapté aux applications de contrôle de commandes dans les contextes bureautique et domotique

Tableau 1 : Réseau de communication

9. Pluviométrie en Tunisie

Le climat de Tunis est dit tempéré chaud. L'été, à Tunis, les pluies sont moins importantes qu'elles ne le sont en hiver. Tunis affiche une température annuelle moyenne de 18.7 °C.

Les précipitations annuelles moyennes sont de 444 mm.

Le mois le plus sec est celui de juillet avec seulement 2 mm. Une moyenne de 56 mm fait du mois de janvier le mois ayant le plus haut taux de précipitations.

Aout est le mois le plus chaud de l'année. La température moyenne est de 27.1 °C à cette période. 11.3 °C font du mois de février le plus froid de l'année.

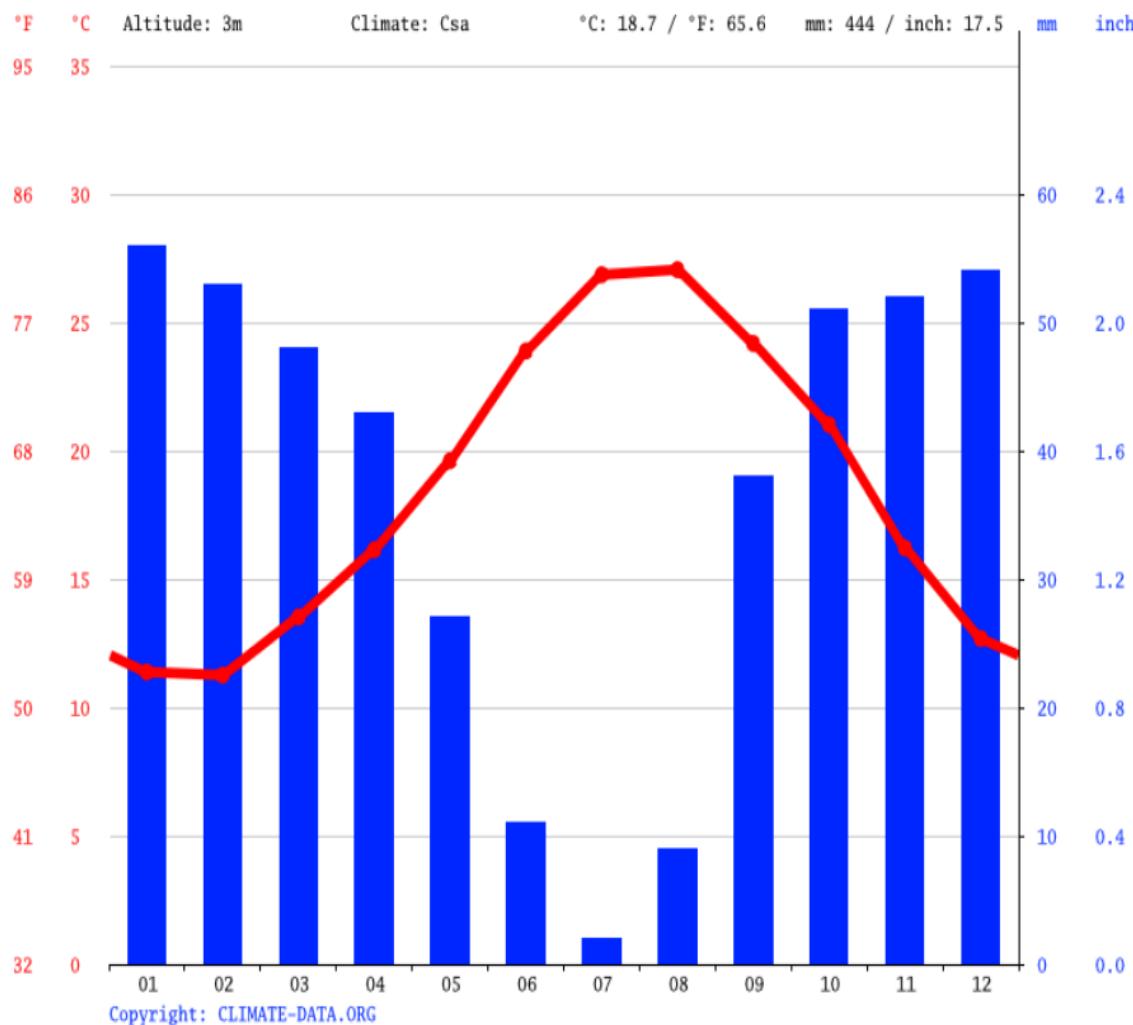


Figure 5: Diagramme ombrothermique TUNIS

Une différence de 54 mm est enregistrée entre le mois le plus sec et le mois le plus humide. Entre la température la plus basse et la plus élevée de l'année, la différence est de 15.8 °C.

L'humidité relative la plus basse de l'année est en juillet (55.98 %). Le mois où le taux d'humidité est le plus élevé est janvier (73.44 %).

Le moins de jours de pluie est à prévoir en juillet (0.33 jours), tandis que les jours les plus pluvieux sont mesurés en janvier (9.33 jours). [3]

	Janvier	Février	Mars	Avril	Mai	Juin	Juillet	Août	Septembre	Octobre	Novembre	Décembre
Température moyenne (°C)	11.4	11.3	13.5	16.1	19.6	23.9	26.9	27.1	24.2	21	16.2	12.7
Température minimale moyenne (°C)	8.4	8	9.7	12	15.1	18.7	21.6	22.2	20.5	17.6	13.1	9.8
Température maximale (°C)	15	15.2	18.1	20.9	24.8	29.6	32.7	32.8	28.9	25.3	19.9	16.2
Précipitations (mm)	56	53	48	43	27	11	2	9	38	51	52	54
Humidité(%)	73%	72%	71%	68%	64%	58%	56%	60%	67%	71%	72%	72%
Jours de pluie (jréé)	7	7	6	6	4	1	0	2	4	6	6	7
Heures de soleil (h)	6.5	7.0	8.5	9.9	11.4	12.5	12.6	11.7	9.8	8.3	7.3	6.5

Figure 6 : Tableau climatique de Tunis

10. Calcul du volume d'un réservoir

La taille de réservoir dépend de plusieurs facteurs spécifiques au projet

- P : la pluviométrie locale en litre par m² et par an.
- S : Surface de récupération des eaux de pluie (surface de toiture ; coefficient de perte)
- Le nombre de jours de réserve dc la cuve
- B : les besoins annuels en litre

La capacité de réservoir est donnée par la formule suivante :

$$\text{Capacité de réservoir (en litre)} = (V+B) / 2 \times (60 / 365)$$

Avec :

- V le volume d'eau récupérable
- B le besoin en cas de pluie en L\an.
- 60 jours le choix du nombre de jours de réserve (Juillet, Aout)

En premier lieu. Nous avons calculé le Potentiel annuel de récupération d'eau de pluie :

$$V= \text{Précipitations} \times \text{Coeff de perte} \times \text{Surface}$$

Avec :

- Coefficient de perte

Coeff toiture ondulée = 0.8

Coeff toiture tuile = 0.9

Coeff toiture plate = 0.6

- Les précipitations annuelles moyennes sont de 444 mm.
- La surface est (3m x 6m) = 18 m²

$$V=448 \times 18 \times 0,8 = 6\,451 \text{ L/an récupéré.}$$

Ensuite, nous avons calculé le besoin annuel d'eau pour arroser les plantes de notre jardin.

La fréquence d'arrosage est pour les mois de :

- Décembre, Janvier, Février, Mars : **I fois/semaine, au total : 1 x 4 x 4 = 16 fois**
- Avril, mai, Juin, Septembre, Octobre, novembre : **2 fois/semaine, au total : 2 x 4 x 6 = 48 fois**
- Juillet, Aout : **3 fois/semaine, au total : 3 x 4 x 2 = 24 fois**

La quantité requise par chaque arrosage est **40 Litre**

Donc le besoin d'eau annuel est : **B=40 x (16+48+24) = 40 x 88=3520 l**

Conclusion

Après avoir présenté l'organisation d'accueil, le cadre du projet, et les défis à relever, nous décrirons dans le chapitre suivant, une étude du matériel et des logiciels.

Chapitre 2 : Étude matérielle et logicielle

Introduction

Dans ce chapitre nous allons développer notre solution en choisissant les équipements adéquats ainsi que les outils nécessaires pour le développement de notre système.

Notre système du jardin intelligent est divisé en 4 parties :

- **Cartes des chargeurs** : Pour adapter la tension d'alimentation de notre système (tensions des batteries +12/+24) à la tension des chargeurs PCS (+20V) et des chargeurs des téléphones portables (+5V).
- **Alimentation du système** : Des batteries alimentées par des panneaux photovoltaïques pour fournir l'énergie électrique à notre système.
- **Armoire de commande** : l'emplacement de la carte commande gérant l'irrigation automatique et l'éclairage de l'espace et le BMS.
- **Application** : Superviser le système et commander l'éclairage et l'irrigation.

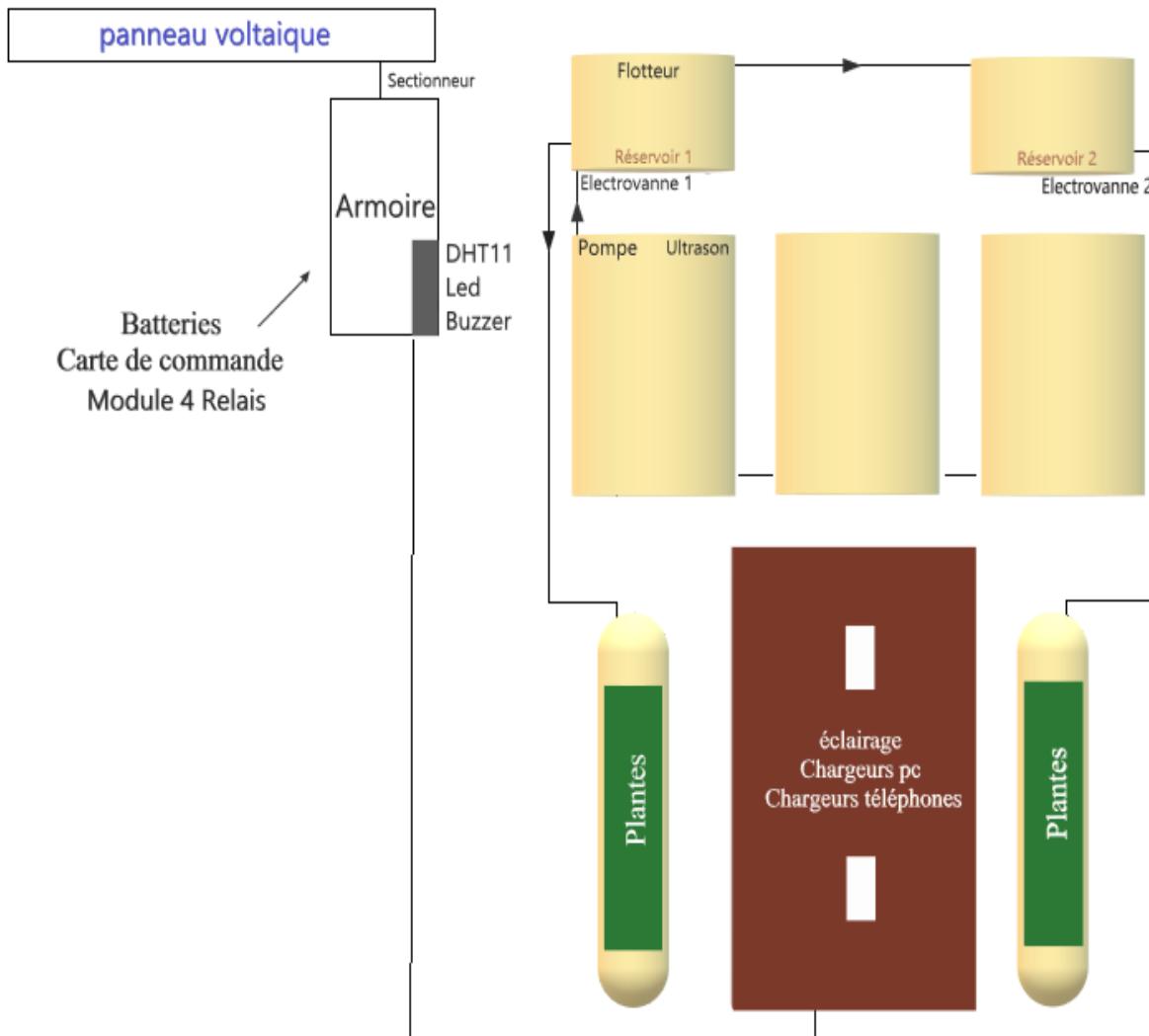


Figure 7 : Schéma explicatif

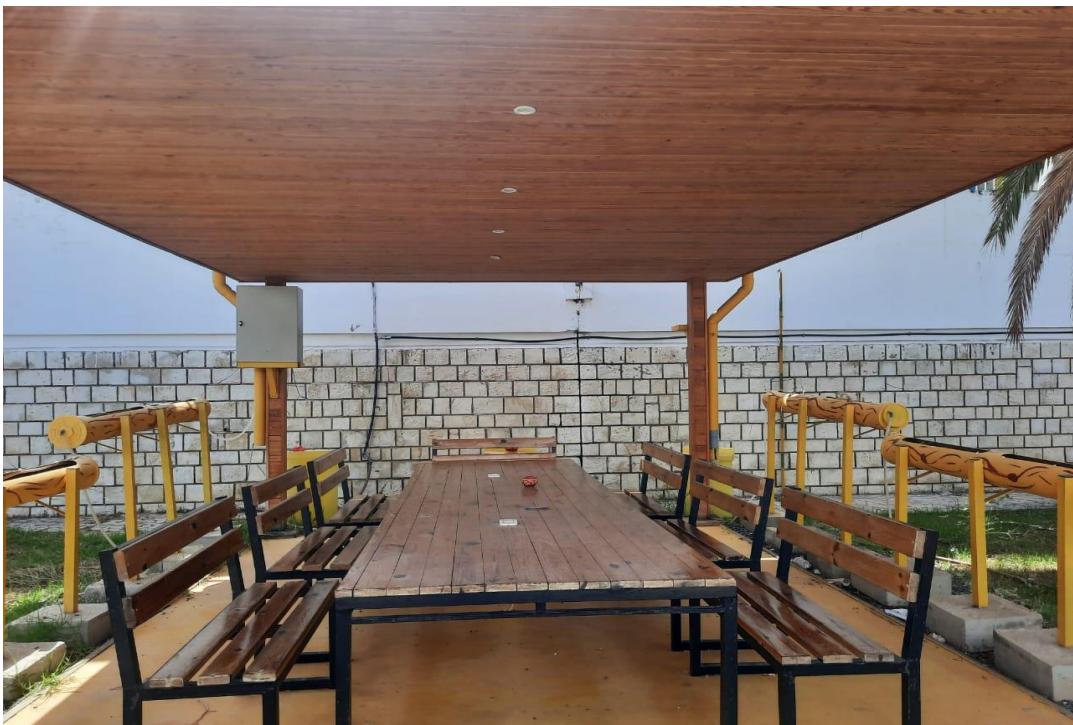


Figure 8 : Constitution de l'espace

I. Étude matérielle

Dans cette partie nous allons développer notre solution en choisissant les équipements adéquats ainsi que les outils nécessaires pour le développement de notre système.

1. Périmétries d'alimentation

1.1. Panneaux Photo1voltaïque



Figure 9 : Panneau Photovoltaïque

Principe de fonctionnement : C'est un assemblage de cellules photovoltaïques dont le matériau est le silicium. Ce matériau exposé à la lumière subit une agitation d'électrons qui permet la circulation d'électrons et donc la génération du courant électrique.

1.2. Batteries Plomb-Acide 12V \ 17,2Ah

Le fonctionnement des batteries se base principalement sur les réactions d'oxydo-réduction ou la transformation de l'énergie chimique en énergie électrique.

Deux électrodes sont placées dans deux électrolytes, l'anode s'oxyde et libère des électrons et la cathode subit une réaction de réduction. La circulation des électrons entre les deux pôles de la pile produit l'énergie électrique.

Et pour recharger la batterie on déclenche la réaction inverse, on refait passer les électrons de la cathode vers l'anode pour reformer des atomes d'ions complets à l'aide des accumulateurs.



Figure 10 : Batterie

2. Les Cartes électroniques

Dans l'étude théorique on va s'adresser aux différents blocs qui constituent les cartes.

Nous allons faire le dimensionnement de chaque bloc tout en se référant aux fiches techniques des composants

2.1. Chargeur PC régulateur 20V

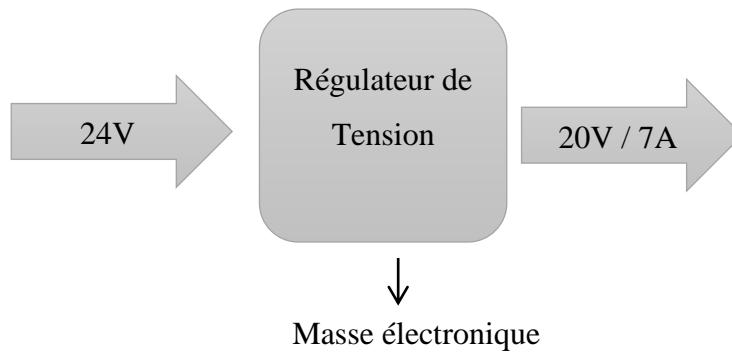
D'après le cahier des charges, on dispose d'une source d'alimentation 24V= à partir des batteries rechargeables et nous devons proposer une tension égale à 20V pour la charge des PC. Nous avons effectué une étude qui concerne les modèles et marques des ordinateurs portables

auprès des étudiants. Nous avons remarqué que certains ordinateurs sont puissants et peuvent absorber une puissance qui atteint 200W. Mais ces ordinateurs peuvent fonctionner avec une puissance inférieure en utilisation normale.

Pour cela nous avons décidé de dimensionner nos chargeurs pour délivrer une puissance maximale égale à 140W. Soit 20V / 7A.

2.1.1. Schéma bloc et choix du régulateur

Pour les chargeurs des ordinateurs portables, la figure ci-dessous montre le schéma bloc de la régulation :



Pour assurer la régulation de tension, nous avons choisi le régulateur LM338T (figure 10) pour ces nombreuses raisons :



Figure 11 : Régulateur LM338T

- Courant maximale supporté : 7A.
- Courant nominal de sortie : 5A.
- Régulation thermique spécifique.
- Protection contre les courts circuits.
- Boîtier TO220 a une surface de dissipation accessible.
- Tension différentielle entrée-sortie entre 3V et 40V. Pour notre cas, la tension différentielle est de 4V en fonction nominal.
- Tension de sortie fixe même si la tension d'entrée varie.

2.1.2. Calcul de la tension de sortie

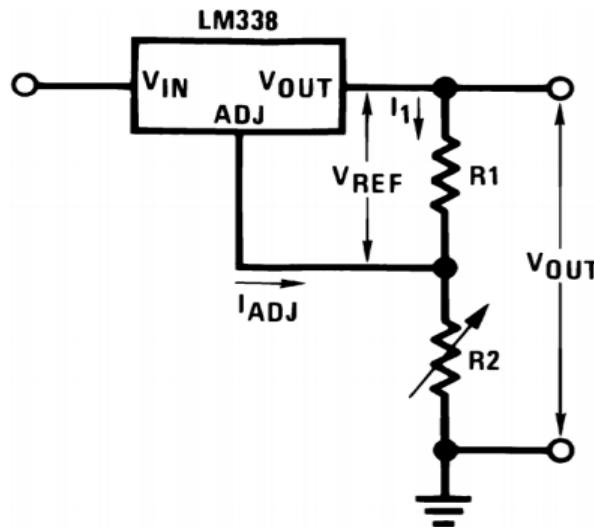


Figure 12 : schéma d’application du régulateur LM338T

L’expression mathématique de la tension de sortie est la suivante :

$$V_{out} = V_{ref} \left(1 + \frac{R2}{R1}\right) + I_{adj} \cdot R2$$

D’après la fiche technique, $I_{adj} = 50 \mu A$, $V_{ref} = 1.25 V$. Pour obtenir une tension de sortie $V_{out} = 20V$, on choisit $R1 = 220\Omega$ et $R2 = 3.3K\Omega$

$$V_{out} = 1.25 \left(1 + \frac{3300}{220}\right) + 5 \cdot 10^{-5} \cdot 3300$$

D’où $V_{out} = 20 V$ est obtenue avec un réglage précis de la résistance variable R2

2.1.3. Ajout des capacités externes

D’après la fiche technique, nous avons besoin d’ajouter des capacités qui permettront d’éliminer les ondulations de la tension d’entrée qui seront amplifiées à la sortie. Donc, on ajoute une capacité céramique $C1 = 220nF$ et une capacité chimique $C2 = 10uF$. Il faut placer les capacités le plus proche que possible du régulateur pour réduire les tensions parasites.

Le schéma du régulateur devient celui de la figure 7 muni d’une diode de protection 1N4001.

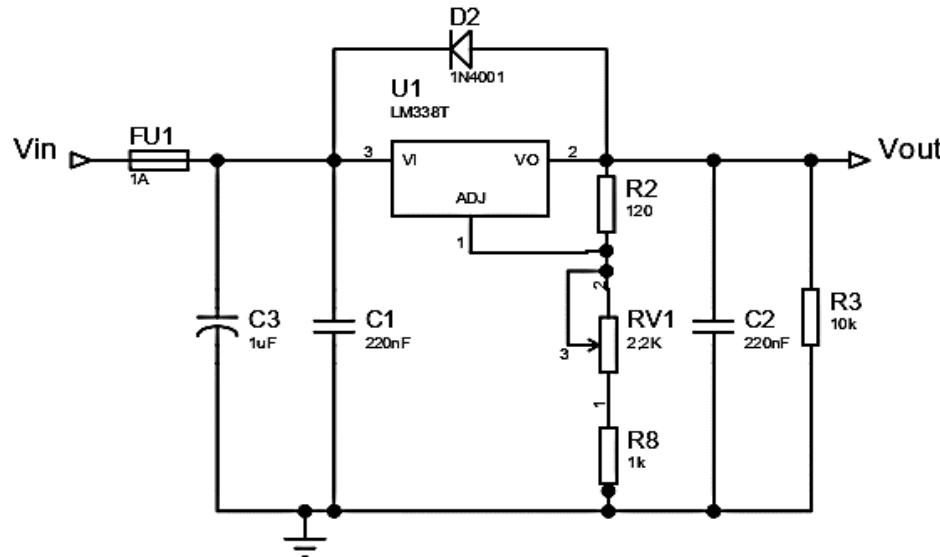


Figure 13 : schéma du régulateur LM338T avec ajout des capacités et d'une diode de protection

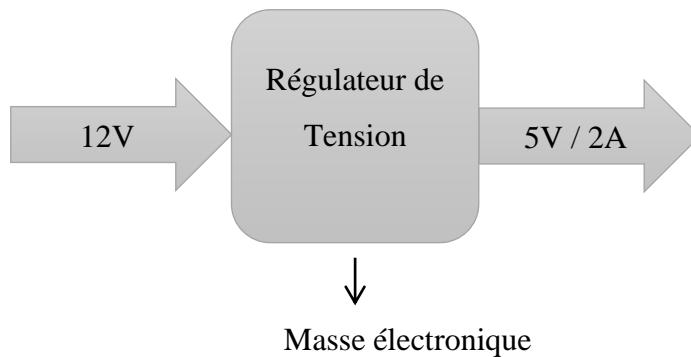
2.2. Chargeur téléphone régulateur 5V

D'après le cahier des charges, on dispose d'une source d'alimentation 12V à partir des batteries rechargeables et nous devons proposer une tension égale à 5V pour la charge des téléphones. Nous avons effectué une même étude qui concerne les modèles et marques des téléphones auprès des étudiants. Nous avons remarqué qu'il y a des chargeurs de smartphones qui ont une puissance pouvant aller jusqu'à 25W. Mais ces chargeurs peuvent fonctionner avec une puissance inférieure en utilisation normale.

Pour cela nous avons décidé de dimensionner nos chargeurs pour délivrer une puissance maximale égale à 10W. Soit 5V / 2A. [4]

2.2.1. Schéma bloc et choix du régulateur

Pour les chargeurs des ordinateurs portables, la figure ci-dessous montre le schéma bloc de la régulation :



Pour assurer la régulation de tension, nous avons choisi le régulateur 7805 (figure 14) pour ces nombreuses raisons :



Figure 14 : Régulateur 7805

- Courant nominal de sortie : 1.5A avec un pic de 2A.
- Tension de sortie fixe égale à 5V
- Protection contre les courts circuits.
- Protection contre la surchauffe.
- Boîtier TO220 a une surface de dissipation accessible.

2.2.2. Montage de régulation

Pour le régulateur 5V, nous prenons la même démarche que nous avons faite pour le régulateur 20V. Nous prévoyons d'insérer des capacités afin d'éliminer les ondulations des tensions parasites. Nous insérons aussi une diode 1N4001 de protection contre les courants de retours éventuels qui au lieu d'augmenter la tension de sortie, seront absorbés en entrée.

La figure 15 présente le montage qui réponds aux besoins précis :

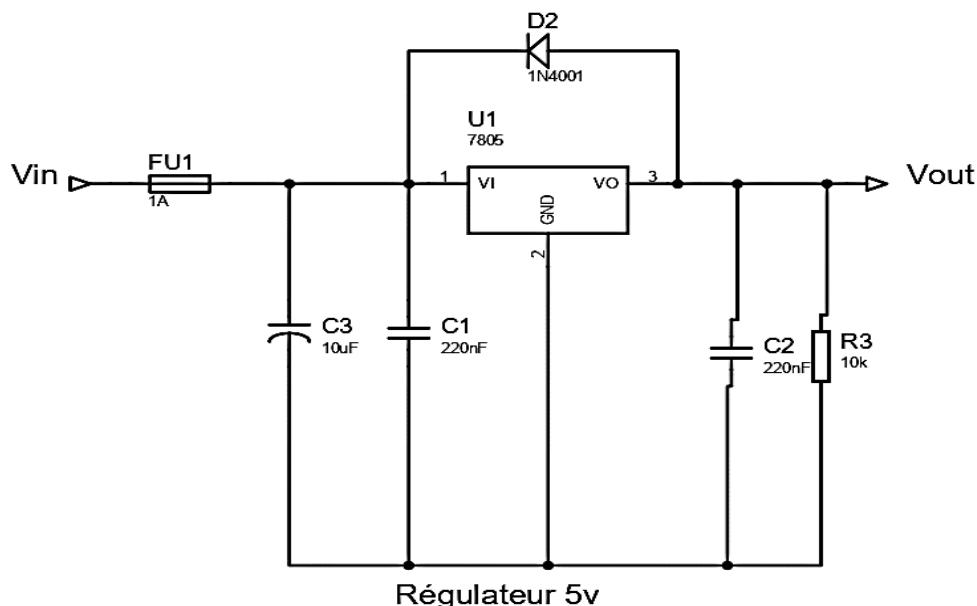


Figure 15 : schéma du régulateur 7805 avec ajout de diode de protection

2.3. Protection contre les courts-circuits

Il existe une protection contre les courts-circuits intégrée dans les régulateurs LM338T et 7805 leurs permettant un fonctionnement limité avec une sortie en court-circuit.

Mais, si par malheur, le régulateur devient toujours passant (court-circuit entrée-sortie, **C-C-E/S**), la sortie 20V des PC, deviendra 24V ou plus, et si c'est le cas pour le chargeur des téléphones, elle deviendra en sortie 12V au lieu de 5V, fortement néfaste pour le pc ou le téléphone portable.

Nous avons introduit une protection **C-C-E/S** comme suit : Le circuit intégré LM324 compare entre la tension de sortie du régulateur et la tension de référence appliquée à la borne inverseuse du comparateur. La tension de référence est réglée à $V_{ref} = 20.7V$ pour le régulateur 20V et $V_{ref} = 5.3V$ pour le régulateur 5V. Elles sont créées à partir de l'association d'une diode 1N4148 de tension seuil de 0.7V et d'une diode zener 20V (1N4747A) pour le circuit du LM338T et une diode zener 4.7V (1N4732A) pour le circuit du 7805.

Si la tension de sortie du régulateur est inférieure à la tension de référence, le circuit intégré LM324 active l'optocoupleur MOC3022 qui à son tour provoque une impulsion au niveau de la gâchette du thyristor TYN690.

L'impulsion au niveau de la gâchette provoque un court-circuit qui fusionne un fusible coupant l'alimentation du circuit.

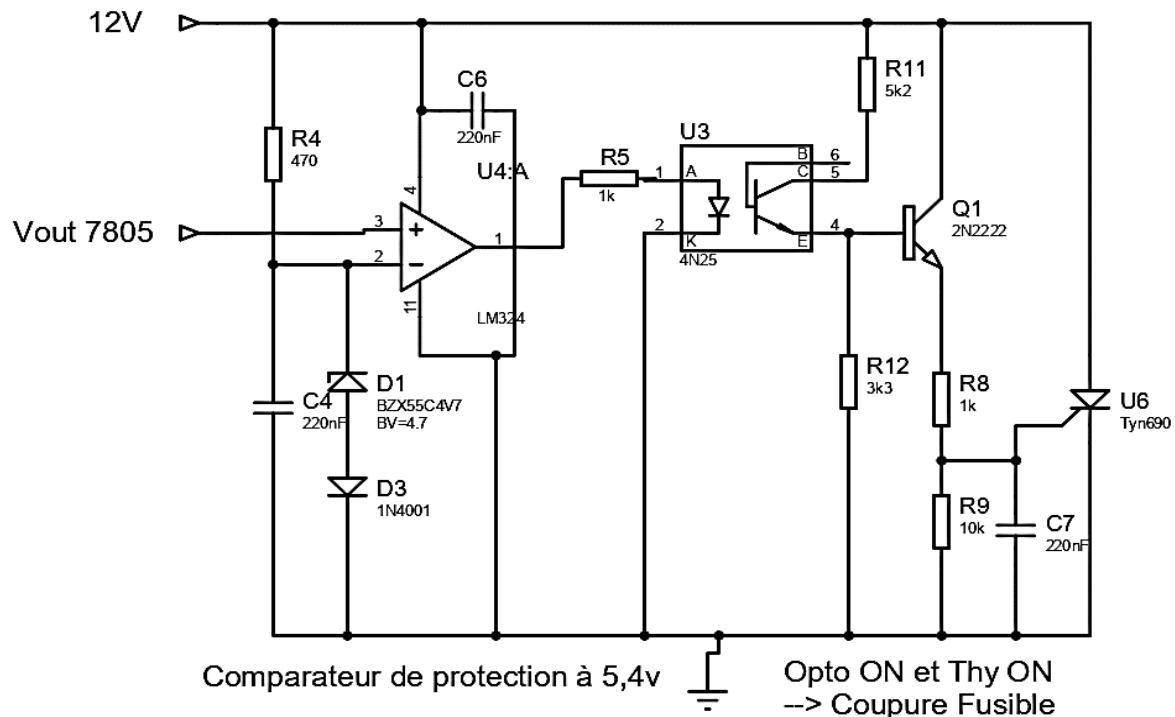


Figure 16 : circuit de protection contre les courts-circuits pour le régulateur 5V

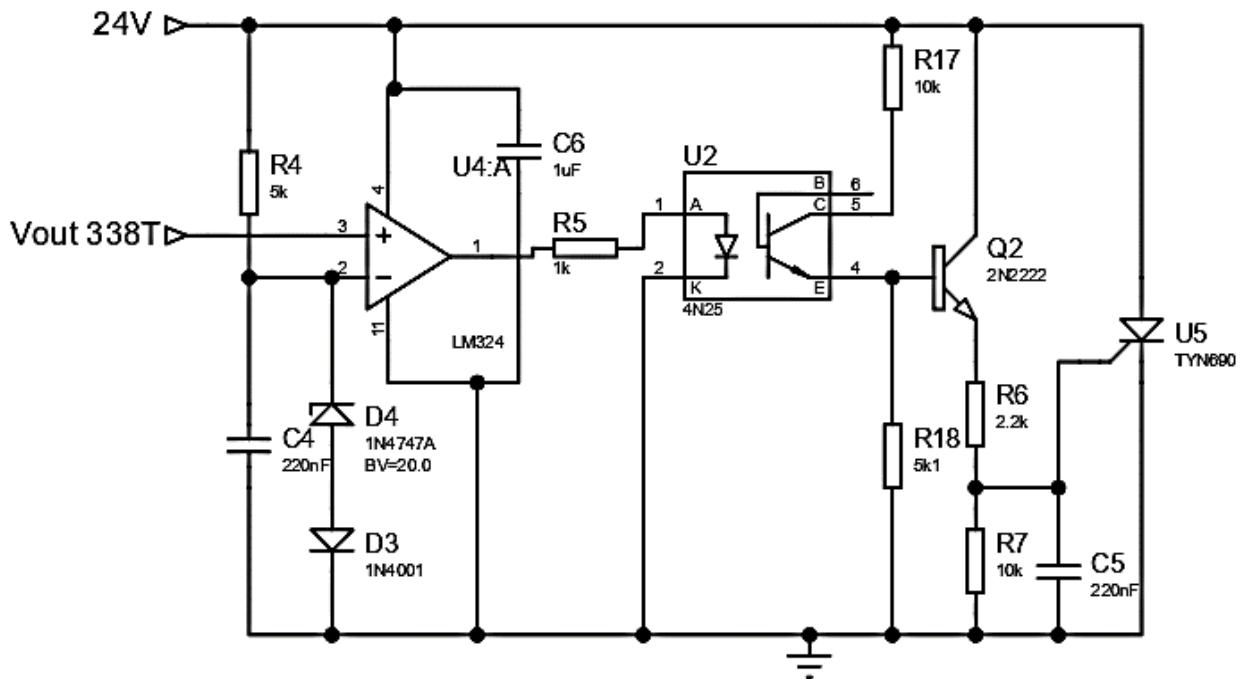


Figure 17 : circuit de protection contre les courts-circuits pour le régulateur 20V

2.4. Circuit de la protection contre la haute température

Le circuit intégré LM338T dissipe une puissance qui peut atteindre 24W lorsque le courant absorbé par les ordinateurs portables est égal à 7A. Pour cela, nous envisageons de placer un ventilateur sur le radiateur pour refroidir le LM338T.

Faire fonctionner la carte sous une température inférieure à 50°C permet de garder le rendement maximal et d'obtenir le courant nominal à la sortie sans activer la protection interne contre la température du LM338T.

Le capteur de température LM35 permet de délivrer une tension égale à 0.5V pour 50°C. Cette tension est amplifiée par l'amplificateur LM324 de gain $K = 10$ puis liée à la borne non inverseuse du comparateur LM324 qui est configuré en mode comparateur de Trigger Schmitt non inverseur de tension de basculement $V_H = 5V$ et $V_B = 4V$. C'est à dire :

- Si la température dépasse 50°C, la tension de sortie de l'amplificateur dépasse 5V, alors, la tension de sortie du comparateur $V_s = + V_{sat} = 24V$ sature le transistor 2N2222 qui met le ventilateur sous tension pour refroidir le circuit.
- Pour arrêter le ventilateur, la température doit diminuer sous 40°C, la tension diminue sous 4V, alors la tension de sortie du comparateur $V_{sat} = - V_{sat} = 0V$ et le transistor sera bloqué.

Nous prévoyons une diode de roue libre 1N4001 pour protéger le transistor contre l'interruption brutale du courant inductif afin de prévenir la surtension au moment du blocage du transistor. La capacité montée au niveau du ventilateur permet de filtrer la tension provenant de la batterie.

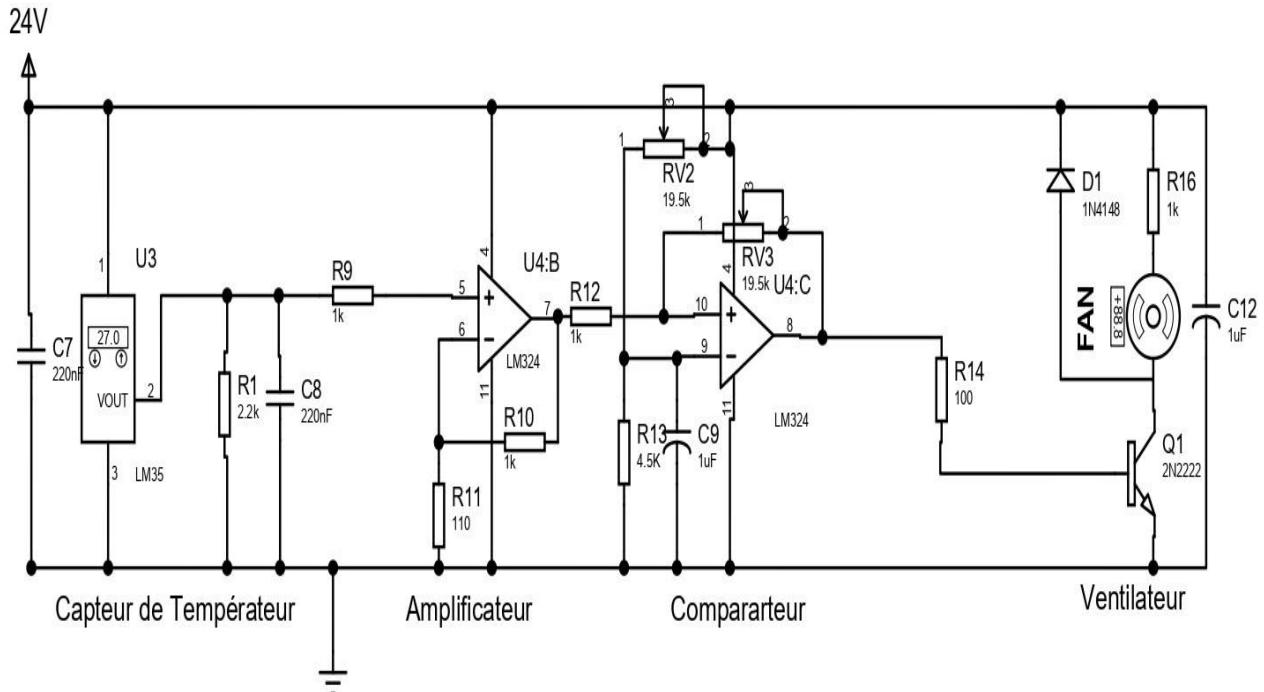


Figure 18 : circuit de protection contre une haute température

2.5. Carte Relais

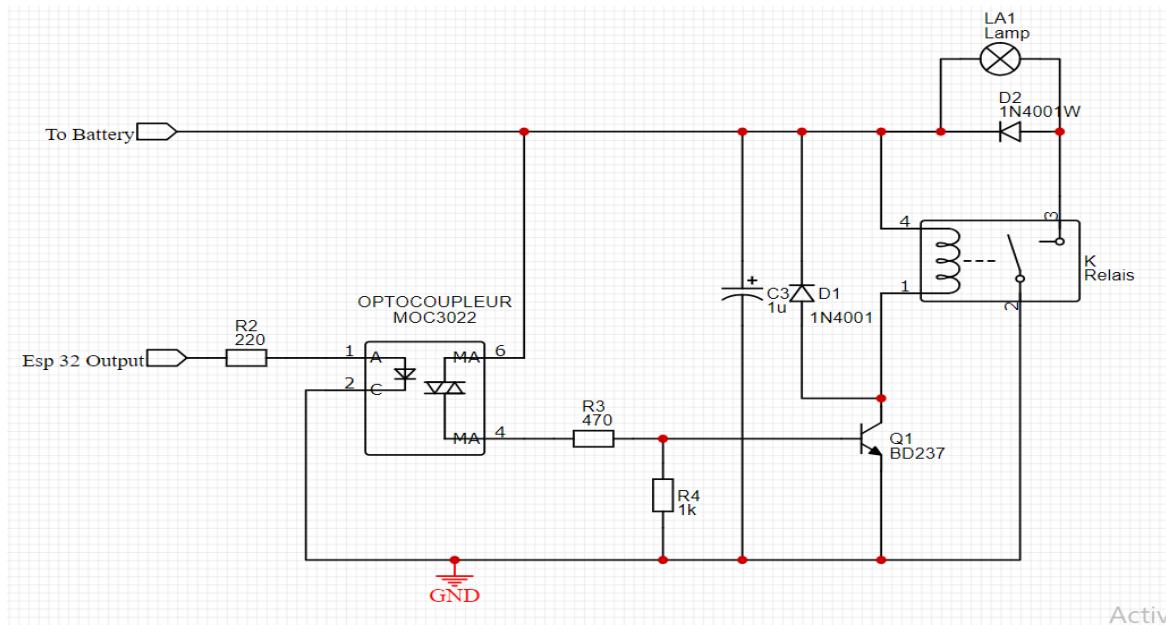


Figure 19 : circuit de la carte de Relais

Dans le but d'automatiser et de commander les différents actionneurs du projet à distance, nous avons opter à leur connecter un relais pour la plupart d'où vient le besoin d'une carte Relais.

L'optocoupleur est commandé par un signal de sortie provenant du microcontrôleur qui a pour rôle principale l'isolation de chaque circuit. Nous prévoyons une diode de protection monté en parallèle de l'actionneur et une capacité de découplage afin de réduire les surtensions.

Après avoir réalisé une carte relais pour commander les actionneurs, nous avons constaté un module relais disponible pour le projet donc pour faire gagner l'espace dans l'armoire, le temps et l'argent nous avons choisis de l'utiliser.

2.6. Carte Commande

Notre carte de commande se compose de différents blocs :

- Emplacement des pins d'entrées et des sorties du module Relais qui permet de commander l'éclairage et la pompe et de couper l'alimentation en cas de surcharge (Niveau de batterie inférieur 11 V).
- La zone d'emplacement du microcontrôleur ESP32.
- Les diviseurs de tension qui permettent d'adapter les tensions des batteries à l'entrée analogique de microcontrôleur.
- Emplacement des entrées de capteur de détection du niveau d'eau dans le réservoir 1 (Flotteur).
- Emplacement des entrées de capteur de mesure du niveau d'eau dans le bidon 1 (Capteur à ultrason)
- Emplacement des entrées de capteur de mesure du température et l'humidité (Capteur DHT11)
- Emplacement des entrées des servomoteurs qui permet d'ouvrir les vannes.
- Un module d'alimentation 3,3V/5V pour alimenter la pompe, les servos et le microcontrôleur. (Voir figure 21)
- Des sorties de signalisations en cas de problème (Buzzer et voyant)

En vue de respecter les recommandations liées à la CEM (Compatibilité Electromagnétique), toutes les entrées doivent être capotisées (capacité de découplage) proche des pins d'entrée pour filtrer les tensions parasites. **L'annexe 1** illustre la carte de commande.



Figure 20 : Arduino Module d'alimentation 3,3V/5V

Le tableau mentionné dans l'**annexe 4** illustre la liste du composants électroniques nécessaire pour les cartes.

3. Choix des capteurs

3.1. Capteur de température et humidité ambiante

Le DHT11 est le capteur choisi pour la mesure de la température et de l'humidité de l'air. Il est précis et son signal de sortie est numérique. Il est relativement peu coûteux pour la performance donnée.

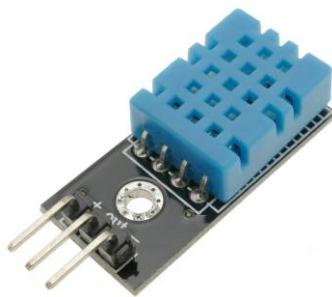


Figure 21 : Capteur DHT11

3.2. Capteur de niveau (capteur à ultrasons)

Ce capteur à ultrasons HC-SR04 est capable de mesurer la distance des objets situés de 2 cm à 400 cm du capteur avec une précision de 3mm. Le capteur est composé d'un émetteur d'ultrasons, d'un récepteur et du circuit de commande. [9]



Figure 22 : Capteur ultrason HC-SR04

3.3. DéTECTEUR DE NIVEAU (FlotTeur)

C'est un capteur magnétique qui contient un relais reed (contact à bilame activé par un champ magnétique). C'est un détecteur de niveau qui renvoie deux valeurs logiques 1 ou 0. Il servira à déterminer le remplissage des réservoirs d'eau.



Figure 22 : Capteur Flotteur

3.4. Humidité du sol

La distribution d'eau dans le sol n'est pas homogène ce qui influence la précision de la lecture. Nous avons donc décidé de ne pas installer le capteur sous-sol mais plutôt de lire l'humidité ambiante sans tenir compte pour la planification de l'arrosage.

4. Les pré-actionneurs et Les actionneurs

L'action principale souhaitée dans notre projet, c'est la commande des électrovannes, des pompes et des lampes. Donc dans cette partie nous avons expliqué comment réaliser cette action, en définissant le bon actionneur et pré actionneur.

4.1. Les pré-actionneurs

Parmi les pré-actionneurs électriques les plus utilisés on trouve les relais. Ces dispositifs permettent de commander un circuit de puissance à partir d'un circuit de commande. Les relais sont utilisés avec des circuits intégrés et un petit circuit de commutation (transistor), ils permettent de commander un circuit de puissance (contacteurs, lampes...)

Module relais

Un relais est un interrupteur à commande électrique et comme tout autre interrupteur, il peut être allumé ou éteint, laissant passer le courant ou non. Il peut contrôler des tensions élevées

comme le 3,3V fournit par le GPIO ESP32 et nous permet de contrôler des tensions élevées comme 12V, 24V ou la tension secteur 230V.

Caractéristiques :

- 4 sorties relais RTC (NO-COM-NF)
- Raccordement sur les relais via bornes à vis
- Pilotable direct depuis microcontrôleur
- Entrées isolées par des optocoupleurs
- Pouvoir de coupure : 30V / 10A max.
- Dimensions : 75 x 55 x 19,3 mm



Figure 23 : Module relais 4 canaux

4.2. Les actionneurs

4.2.1. Lampes

Pour produire un phénomène physique (lumière, mouvement, chaleur...), on utilise un actionneur (par exemple : lampe, moteur, résistance chauffante...) donc afin d'éclairer l'espace nous avons utilisé 4 lampes commandées par deux interrupteurs à minuterie électronique pour économiser l'énergie disponible.



Caractéristiques :

V= 12v

I= 642 mA

P=7w

Figure 24 : Lampes

4.2.2. Mini pompe d'eau

La notion d'irrigation consiste à fournir de l'eau à des plantes durant un certain temps sans intervention humaine. Il peut s'agir également de système de type "Goûte à goûte". Une pompe à eau est une pompe qui permet la mise en mouvement de l'eau.

Nous utilisons une mini pompe d'eau. Elle est construite autour d'un moteur sans balais offrant un fonctionnement silencieux. Aussi en a utilisé des électrovannes comme actionneurs.



Caractéristiques :

$V=12V$

$I=0,37A$

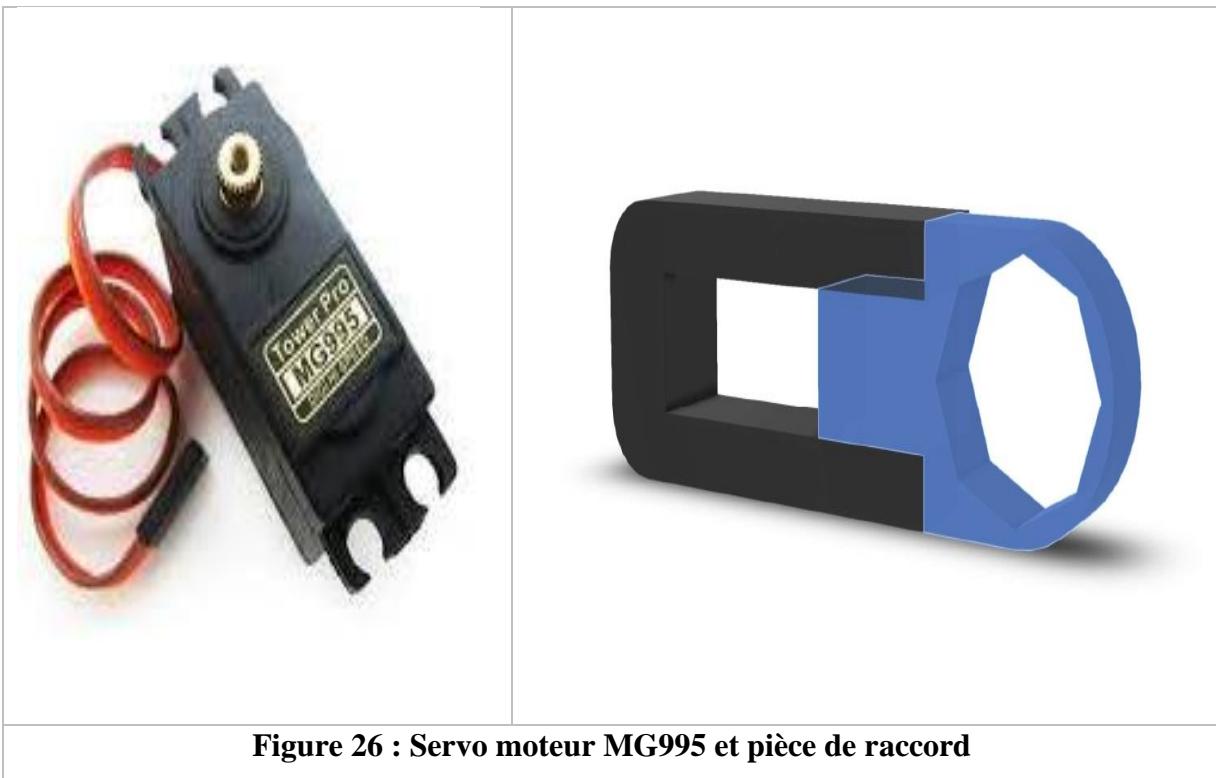
Débit : 1L /min

Hauteur max : 3m

Figure 25 : Mini pompe d'eau

4.2.3. Électrovanne motorisée

Nous avons rencontré des problèmes au niveau de la disponibilité des vannes, alors nous avons décidé de développer notre solution. On a intégré un servomoteur sur une vanne manuelle pour la contrôler.



Caractéristiques techniques de servomoteur choisi :

- Type standard "TowerPro MG995"
- Couple : 11Kg/cm sous 4.8V - 13Kg/cm sous 6.0V
- Tension : 4.8V - 6V
- Consommation : 120 mA sans charge 1 450 mA pour 11Kg/cm environ

5. L'unité de commande

5.1. Choix du microcontrôleur

Le choix du microcontrôleur avec lequel on va gérer le fonctionnement du système se fera en répondant aux questions suivantes :

Quel est le nombre de broches d'E/S nécessaires ?	16 Minimum
Est-ce que on a besoin d'une communication WIFI ?	Oui
Haut ou faible débit de transmission de données ?	Haut
Quelles sont les dimensions du microcontrôleur ?	Le plus petit possible
Quel est le prix qu'on ne doit pas dépasser ?	50DT

Pour choisir le microcontrôleur le plus adapté à nos besoins, on doit faire une comparaison entre des différents cartes à bases de MCU.

5.2.Comparaison des différentes cartes à bases de microcontrôleurs

Cartes	STM 32	ESP-32	ARDUINO
Référence MCU	STM32F411	ESP WROOM 32D	ATmega328
Nombre de cœur	1	2	1
Architecture	32 bit	32 bit	8 bit
Fréquence MCU	180 Mhz	160 Mhz	16 Mhz
RAM	192 KO	512 KB	2 KB
Flash	1 MO	16 MO	32 KB
Wifi intégré	NON	OUI	NON
Bluetooth intégré	NON	OUI	NON
Pins GPIO	100	36	14
Interfaces de communication	SPI-I2C-URAT- USART-I2S-CAN	SPI-I2C-URAT-I2S- CAN	SPI-I2C-URAT
ADC pins	16	18	6
DAC pins	2	2	0
Prix	140DT	45DT	60DT

Tableau 2 : Comparaison de différentes cartes à bases de microcontrôleurs

Le microcontrôleur le plus adéquat à notre besoin est **ESP WROOM-32D**.

5.3. Présentation du ESP WROOM-32D

L'ESP32 développé par la société Expressif, est une carte de développement à faible coût dédié à l'internet des objets (IoT) et les applications embarquées. C'est un (Soc) system on a chip doté de communications sans fil Wifi et Bluetooth.



Figure 27 : Le microcontrôleur ESP 32

5.4. Caractéristique du ESP WROOM-32D

Microprocesseur	Tensilica Xtensa LX6
Courant pour 3.3V	50 mA
Courant DC pour E/S	40 mA
Fréquence maximale optimal	240 Mhz
Sortie PWM	25 canaux

Tableau 3 : Caractéristique du ESP32

5.5. Réseau de communication

Afin d'assurer une communication entre les différentes parties de ce projet, on a choisi le réseau [Wifi](#) qui est convenable au notre application avec une qualité du signal assurée et connexion simple et rapide à la passerelle en plus de sa disponibilité dans notre espace.

6. Bilan du projet

6.1. Calcul énergétique

Notre Projet Smart Garden est totalement alimenté par un seul panneau photovoltaïque, donc on s'intéresse aux calculs qui seront nécessaires pour le choix des panneaux et des batteries pour garantir le bon fonctionnement de système.

Calcul de la consommation journalière :

Dans cette partie il s'agit d'estimer la consommation d'équipements afin de d'obtenir la consommation totale moyenne par jour.

L'énergie totale moyenne nécessaire par jour E (Wh/j) est la somme des consommations énergétiques des divers équipements constituant le système Smart Garden (micro-Contrôleur ESP32, les capteurs, les actionneurs, charge des PC et téléphones portables, les lampes d'éclairage ...)

L'énergie est donnée par la formule suivante : $E_j = P \times t$

Avec :

E_j : Energie journalière consommée de l'équipement (Wh/j)

P : Puissance de l'équipement (W)

t : Durée d'utilisation (h).

Le tableau suivant définit la puissance totale de consommation journalier (en watt) qui dépend des appareils qui devrait alimenter.

Composant	Référence	Quantité	I(mA)	V(v)	P(W)
Microcontrôleur	ESP32	1	260	3,3	0.858
Capteur de température	DHT11	1	2,5	3,3 - 5	1.25
Capteur Ultrason	HOSR04	1	2	3,3 – 5	0,01
Capteur Flotteur	X	1	X	3.3	0.001
Pompe	X	1	370	12	4,44
Module de relais 4 canaux	Relais	4	320	5	1 ,85
Servo moteur	MG995	2	400	4.8 - 6	2
Lampes 1	X	2	1200	12	14,4
Lampes 2	X	2	1200	12	14,4

Tableau 4 : Estimation des besoins énergétique journaliers pour Smart Garden

Les ordinateurs portables consomment classiquement des puissances électriques de 50 à 100 W et le chargeur du téléphone consomme une puissance de 10 W, faible en comparaison avec le PC. Donc énergie Total est : $E_j = 1143$ w/j

6.2. Cout approximatif

Dans cette partie, Nous allons déterminer le cout approximatif de notre projet :

- Les cartes réalisées ne sont pas disponibles dans les marchés tunisiens mais le plus ajouté par rapport les cartes prêtées en ligne dans les sites internationaux est la sécurité de circuit optimale. L'**Annexe 4** pour illustre la liste des composants.
- La carte de commande contient le câblage de l'ESP32.
- Notre solution d'électrovanne motorisée du servomoteur est plus fiable que l'électrovanne prêté et coute moins cher.
- Le cout total de projet est : **1400D**

	Les objets	Cout	Quantité
Matérielles			
Carte Electronique	Chargeur 5V	13D	2
	Chargeur 20V + Fan	24D	2
	Carte commande	X	1
Matérielles électroniques	Panneau	300D	2
	Batteries	70D	4
	Lampes	15D	4
	Relais	18D	1
	Microcontrôleur	45D	1
	Capteur ultrason	10D	1
	Capteur température	9D	1
	Capteur flotteur	20D	1
	Pompe	80D	1
	Servomoteur	40D	2
Matérielles	Les tuyaux, sert-câble	80D	X
	Tube PVC, Vannes		

Tableau 5 : Bilan économique

II. Logiciels

1. Proteus ISIS

Proteus est une suite logicielle, éditée par la société Labcenter Electronics et revendue en France exclusivement par Multi power. Proteus est actuellement (2020) la seule CAO électronique qui permet la conception d'un système électronique complet et de le simuler, y compris avec le code des microcontrôleurs. Pour ce faire, elle inclut un éditeur de schéma (ISIS), un outil de placement-routage (ARES), un simulateur analogique-numérique, un environnement de développement intégré pour microcontrôleurs, un module de programmation par algorigrammes ainsi qu'un éditeur d'interface pour smartphone afin de piloter à distance des cartes Arduino ou Raspberry Pi.



Figure 28 : Logo Proteus ISIS

2. Arduino IDE

Présentation :

C'est une plate-forme de développement et de prototypage Open Source.

IDE (environnement de développement intégré) multi OS :

- Édition du programme.
- Compilation du programme.
- Transfert du programme vers le microcontrôleur via le port USB.



Figure 29 : Logo IDE Arduino

Programmation :

Le langage d'Arduino basée sur le langage C embarquée.

Le programme se compose de deux sections principales :

- Une section « setup » : 1 seule exécution après RAZ.
- Une section « Loop » : exécutée indéfiniment en boucle.

3. App Inventor

App Inventor est un outil de développement en ligne pour les applications mobiles et les tablettes Android développé par Google et entretenu par le Massachusetts Institute of Technology (MIT).

App Inventor est l'un des outils les plus simples permettant de créer des applications sur Android.



Figure 30 : Logo MIT APP INVENTOR

4. SolidWorks

SOLIDWORKS est un logiciel propriétaire de conception assistée par ordinateur 3D fonctionnant sous Windows.



Figure 31 : Logo SolidWorks

5. Firebase

Firebase est un ensemble de services d'hébergement pour n'importe quel type d'application. Il propose d'héberger en NoSQL et en temps réel des bases de données, du contenu, de l'authentification sociale, et des notifications, ou encore des services, tel que par exemple un serveur de communication temps réel.



Figure 32 : Logo Firebase

Conclusion

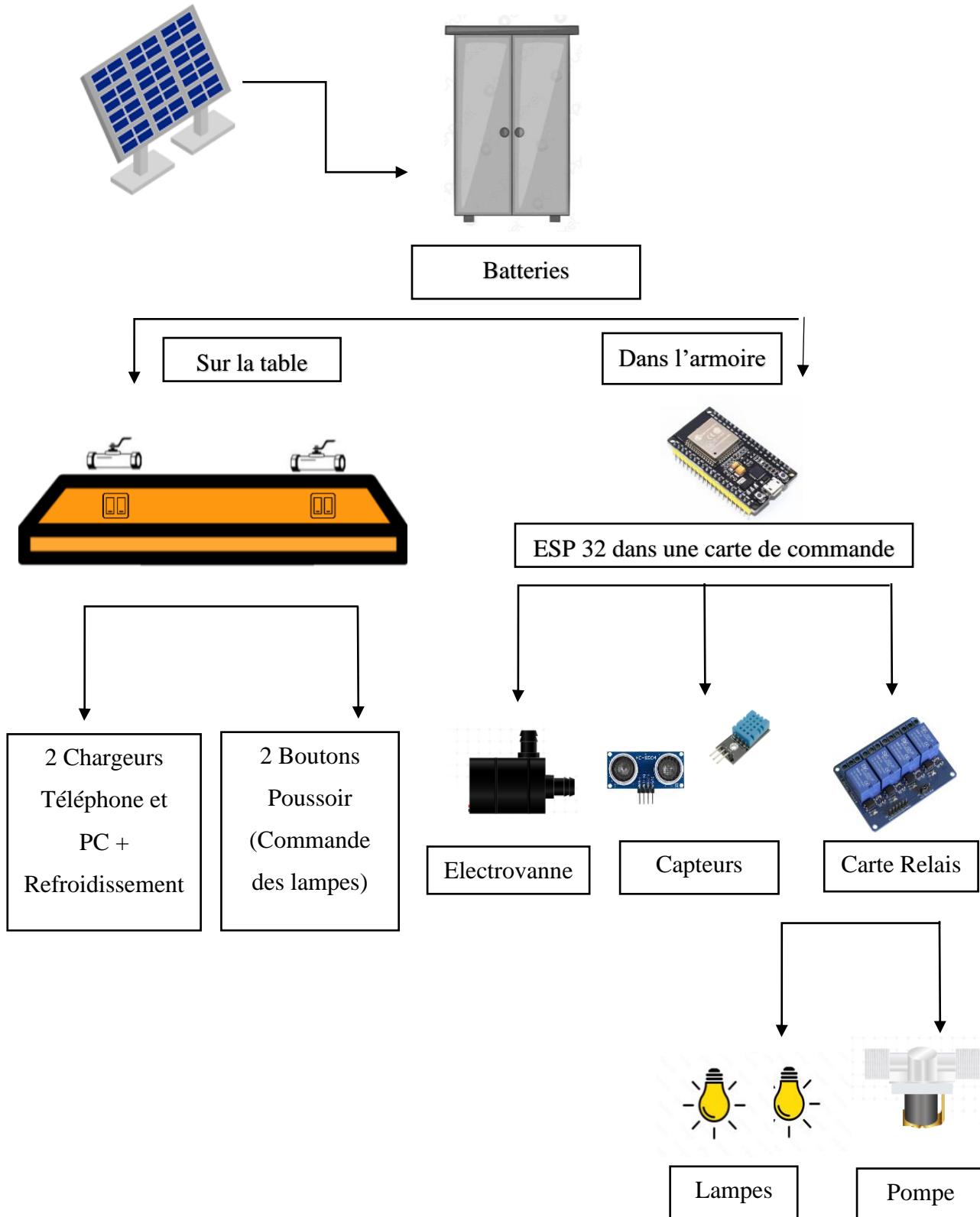
Dans ce chapitre, nous avons présenté une analyse fonctionnelle de la solution, suivi d'une spécification technique dans laquelle nous avons étudié les différents composants choisi et présenté la plateforme lot qui sera utilisée dans le déploiement de la solution

Chapitre 3 : Présentation de la solution proposée

Introduction

Ce chapitre sera consacré à la description détaillée de la solution adaptée et la réalisation du projet ainsi que le test et la simulation du système.

I. Présentation générale de notre solution



Notre solution se compose de deux parties principales :

Une partie Hardware : On va réaliser une carte de commande qui permet de gérer notre système (l'irrigation, l'éclairage et l'alimentation) et qui va assurer la sécurité et le bon fonctionnement de ce jardin intelligent.

De plus, on va faire la conception des vannes motorisées qui assurent l'irrigation.

Une partie Software : C'est la partie de la programmation du microcontrôleur afin d'automatiser le système.

II. La partie hardware

1. Conception des cartes électroniques

Pour réaliser les cartes qui gère le système on a utilisé Proteus ISIS comme outil de développement et conception.

Toute la figure suivante représente la réalisation du schématique, le routage et la vue en 3D de nos cartes électroniques à l'aide du Proteus.

1.1. Conception de la carte chargeur 5V

D'après le cahier des charges, on dispose d'une source 12V à partir des batteries rechargeables. Nous devons proposer une solution pour obtenir une tension 5V pour charger deux téléphones portables sachant qu'il faut intégrer une protection par fusibles si la tension pour les chargeurs portables dépasse 5.7v/3A.

- L'arrivée des tensions disponible est : 12V DC et 0V à partir des batteries rechargeables par énergie solaire.

Les figures mentionnées dans **l'annexe 5** illustrent le schéma de la carte chargeur 5V et le PCB de la carte chargeur 5V.

1.2. Conception de la carte chargeur 20V

D'après le cahier des charges, on se dispose d'une source 24V à partir des batteries rechargeables. Nous devons proposer une solution pour obtenir une tension 20V pour charger deux pcs en respectant les critères ci-dessous :

- Les cartes des chargeurs d'ordinateurs portables doivent avoir une dissipation de chaleur active déclenchée par un capteur de température (circuit contre les hautes températures)
- Il faut intégrer une protection par fusibles si la tension pour les chargeurs d'ordinateurs portables dépasse 20.7V/7 A.

L'arrivée des tensions disponibles cst : 24 V DC et OV est à partir des batteries rechargeables par énergie solaire.

- Les figures mentionnées dans **l'annexe 6** illustrent le schéma de la carte chargeur 20V et Le PCB de la carte chargeur 20V.
- Les cartes électroniques (les chargeurs ordinateurs portables et téléphones) sont assemblées dans un tube cylindrique en PVC (le diamètre 16cm et de longueur 30cm)

1.3. Conception de la carte de commande

L'annexe 3 présente le schéma et le PCB de la carte commande réalisé.

Le tableau ci-dessous illustre la connexion entre le microcontrôleur et les différents composants

ESP 32	Pins
GPIO 34	Pont diviseur 24V
GPIO 35	Pont diviseur 12V
GPIO 17	Entrée relais 1 vers Lampe 2
GPIO 16	Entrée relais 2 vers Lampe 1
GPIO 4	Entrée relais 3 vers Pompe
GPIO 0	Entrée relais 4 vers batteries
GPIO 26	Bouton Lampe 1
GPIO 27	Bouton Lampe 2
GPIO 32	Servo 1
GPIO 33	Servo 2
GPIO 14	DHT11
GPIO 5	Ultrason TRIG
GPIO 18	Ultrason ECHO
GPIO 19	Led
GPIO 21	Buzzer
GPIO 12	Détecteur de niveau

Tableau 6 : Câblage des entrées et sorties du microcontrôleur ESP32

Pour protéger le circuit nous avons ajouté un Sectionneur qui est un appareil électromécanique permettant de séparer, de façon mécanique, un circuit électrique et son alimentation, tout en assurant physiquement une distance de sectionnement satisfaisante électriquement.

2. Câblage des composants

2.1. Organigramme de fonctionnement du système

L'organigramme mentionné dans la figure suivante présente la stratégie qui permet de gérer l'irrigation des plantes et le stockage d'eau, aussi la surveillance de système à distance.

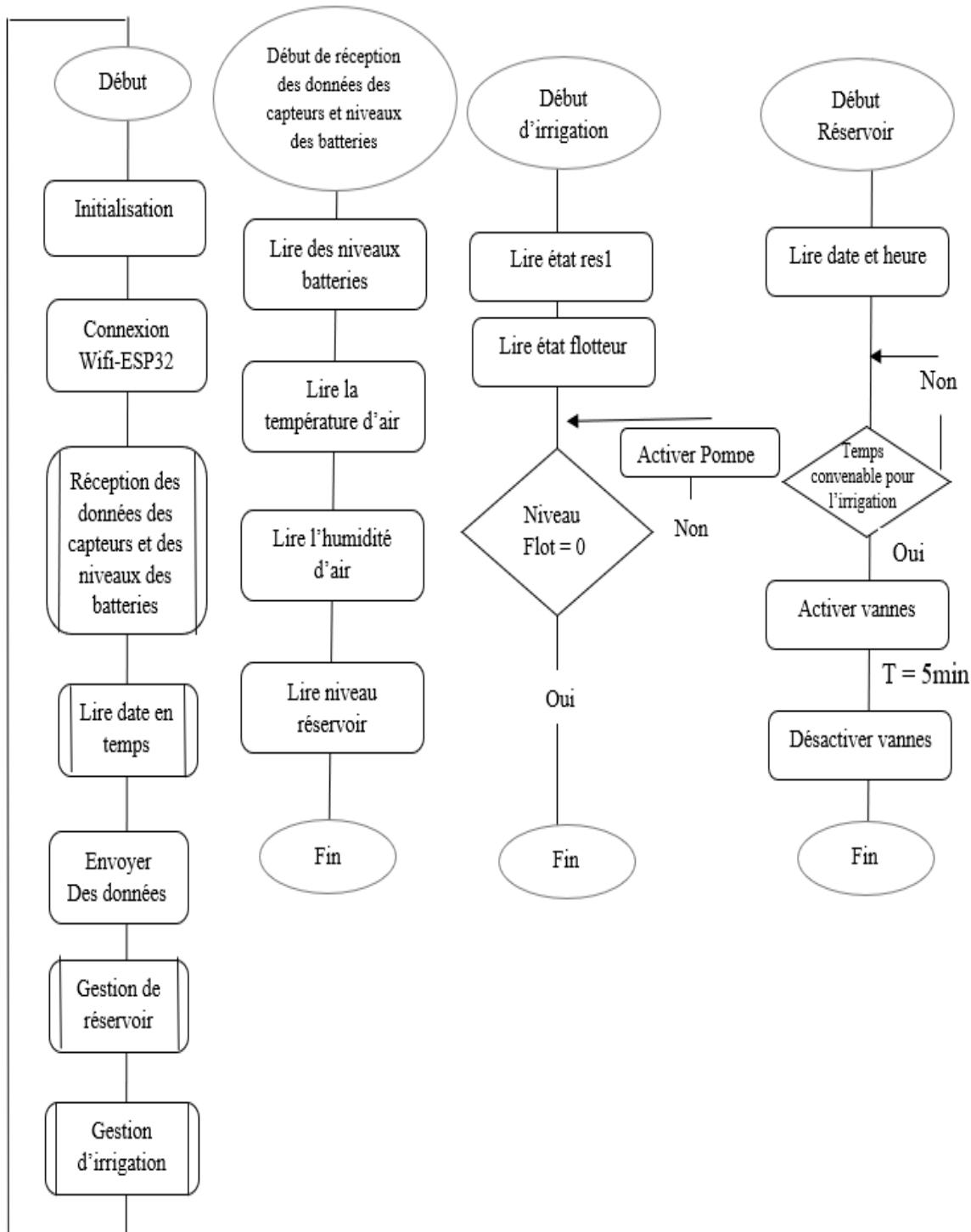


Figure 33 : Organigramme de fonctionnement du système

2.2. Câblage des batteries

Nous avons quatre batteries série parallèle deux à deux, La mesure les niveaux des batteries 1 et 2 est à travers les ponts diviseurs sur la carte de commande.

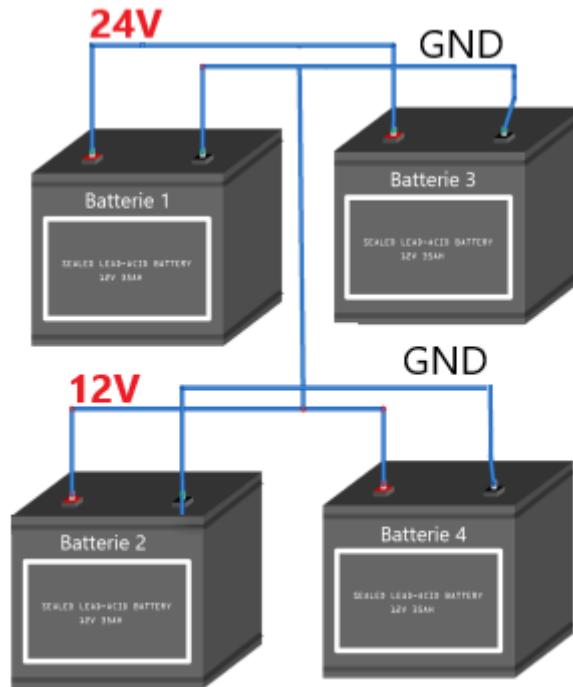


Figure 34 : Câblage des batteries

Nous pouvons suivre l'utilisation des chargeurs au cours du temps avec l'application pour suivre le taux d'utilisation et juger des modifications ou bien des améliorations si c'est nécessaire.

Battery E SP 32	
Sortie pont diviseur 24V	GPIO 34
Sortie pont diviseur 12V	GPIO 35

Tableau 7 : Tableau de connexion des batteries avec ESP32

Afin d'assurer le bon fonctionnement des batteries nous avons utilisé un système BMS expliqué ci-dessous :

L'organigramme mentionné dans la figure suivante présente la stratégie qui permet de gérer l'irrigation des plantes et le stockage d'eau, aussi la surveillance de système à distance.

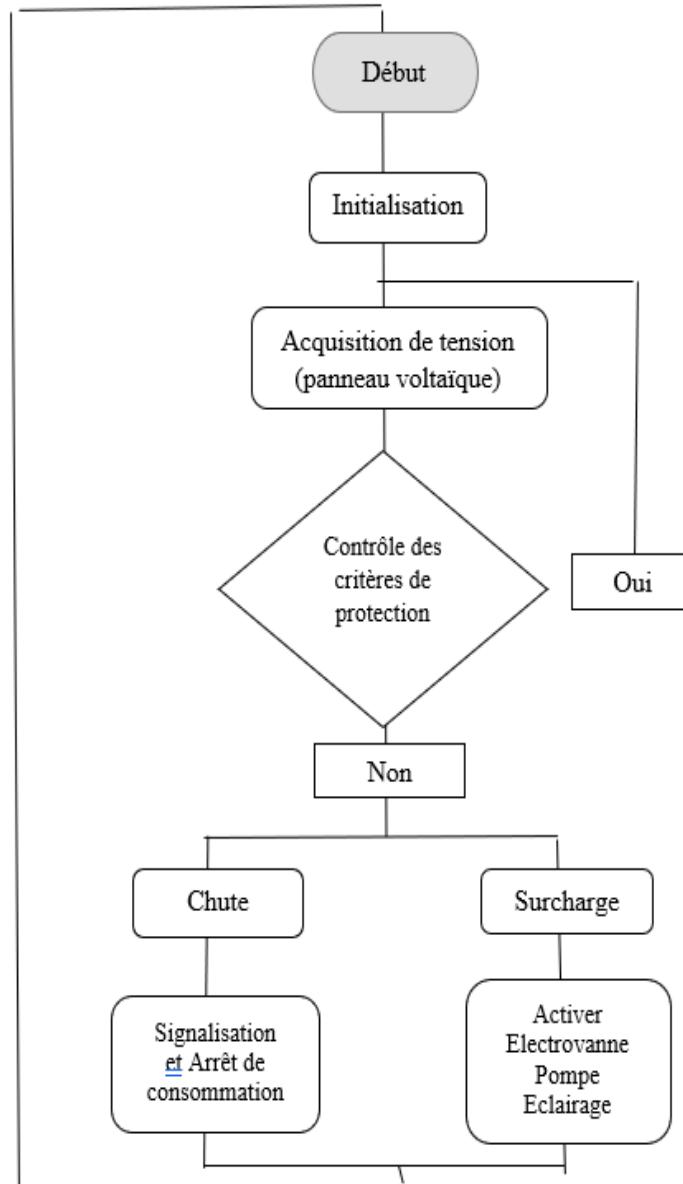


Figure 35 : Organigramme de fonctionnement du système BMS

2.3. Câblage de l'ESP 32

Comme mentionné précédemment, l'ESP32 est une carte électronique permettant de réaliser des projets « IoT » assez facilement. L'objet « IoT » est très intéressant, cependant il nécessite un accès à Internet afin de pouvoir assurer une communication permanente avec l'utilisateur. L'ESP32 est en ce sens une solution intéressante car elle contient au sein de son architecture un module Wifi intégré. Dans cette mesure, et donc sans avoir besoin de monopoliser des broches en ajoutant un module Wifi externe, l'utilisateur va être capable de connecter son objet IoT à Internet grâce à trois lignes de codes. [5]

Le schéma mentionné dans l'**annexe 2** présente le câblage entre la carte de commande et les différents capteurs et actionneurs.

ESP32-WROOM-32D

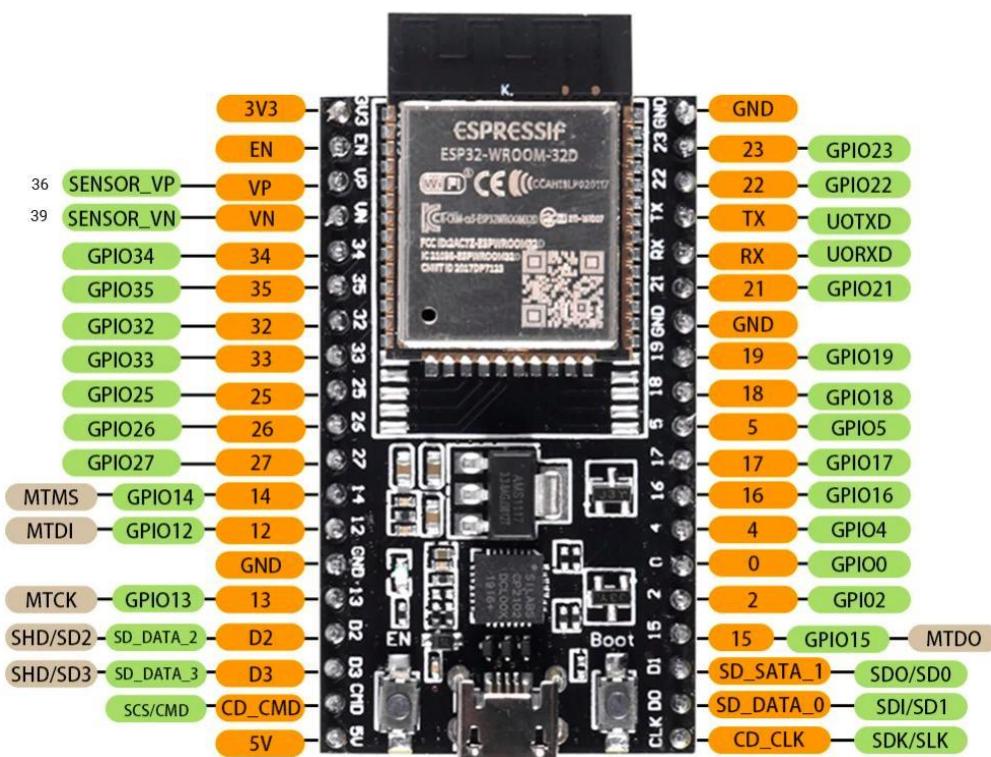


Figure 36 : Architecture de ESP 32D

Lorsqu'un système comporte des actionneurs demandant un certain niveau de puissance que le microcontrôleur ne peut pas fournir, il faut avoir recours à deux circuits distincts interconnectés, : « circuit de commande » et « circuit de puissance ».

Donc un **Circuit de commande** dans ce circuit que sont rassemblés tous les éléments de contrôle comme les boutons, les interfaces et le microcontrôleur.

La manière la plus simple de transmettre les commandes de ce circuit au circuit de puissance est d'utiliser des transistors ou encore des relais. Lorsque les tensions d'alimentation des deux circuits sont importantes ou si l'on veut protéger la commande, des optocoupleurs (plutôt que des transistors) assurent une isolation galvanique : l'information est transmise sous forme de lumière.

2.4. Câblage du module Relais

Le relai sera défini dans les schémas de la carte de commande comme suit : Relai 1

- Entrée : I1
- Contact de sortie : a1 et b1

Notons qu'il faut enlever le Jumper, alimenter le VCC par 3.3V, et le relai par JD-VCC=5V, et ceci car l'entrée I1 varie de 0 à 3.3V.

Comme illustré ci-dessous pour le relai 1 .

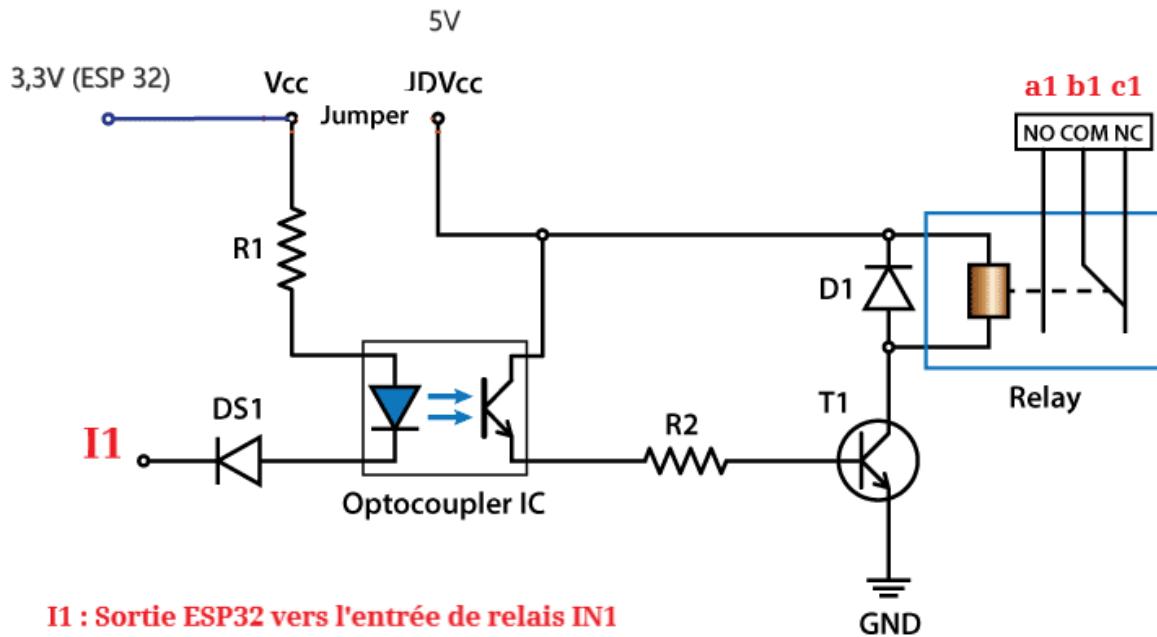


Figure 37 : Relai 1

Module Relais	ESP 32	Signification
VCC	3V3	Alimentation
GND	GND	Alimentation
Broche	VIN	Alimentation
IN1	GPIO 17	Entrée relais 1 vers Lampe 2
IN2	GPIO 16	Entrée relais 2 vers Lampe 1
IN3	GPIO 4	Entrée relais 3 vers Lampe Pompe
IN4	GPIO 0	Entrée relais 4 vers batteries

Tableau 8 : Tableau connexion carte relais avec ESP32*

2.5. Capteur température DHT11

Le capteur DHT11 est utilisé pour mesurer la température et l'humidité relative.

DHT11	ESP32
PIN 1 (+)	3,3V
PIN 2 (OUT)	GPIO 14
PIN 3 (-)	GND

Tableau 9 : Tableau de connexion de DHT11 avec ESP32

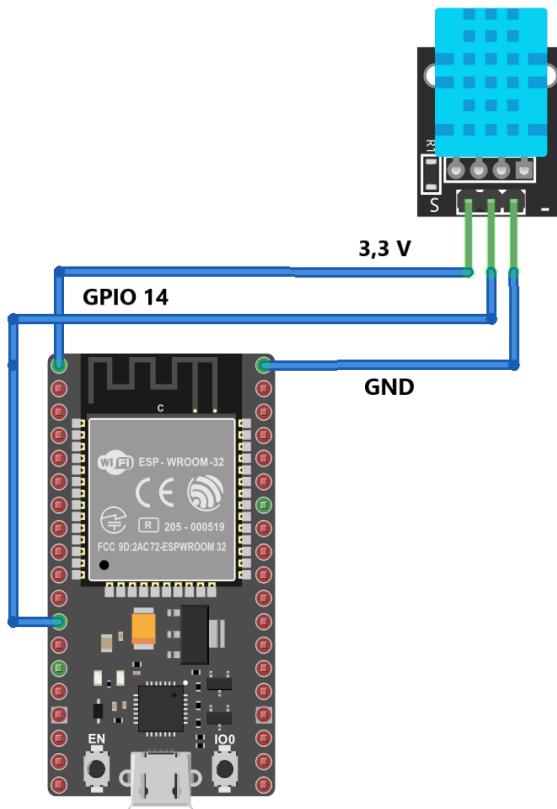


Figure 38 : Câblage de capteur température DHT11

2.6. Capteur de niveau (capteur ultrason)

Le capteur HC-SR04 est choisi pour mesurer le niveau d'eau dans le réservoir de récupération d'eau de pluie. Il peut mesurer la distance d'objets de 2cm à 400cm du capteur avec une précision de 3 mm. Ce capteur se compose d'un émetteur à ultrasons, d'un récepteur et d'un circuit de commande. [10]

HC-SR04	ESP32
VCC	3.3V
TRIG	GPIO 5
ECHO	GPIO 18
GND	GND

Tableau 10 : Tableau de connexion de HC-SR04 avec ESP32

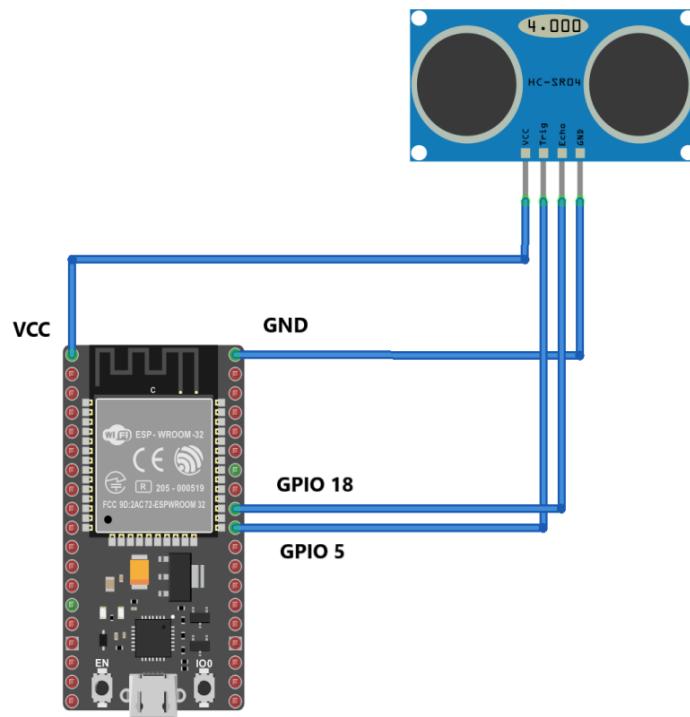


Figure 39 : Câblage de capteur de niveau ultrason

2.7. Détecteur de niveau (capteur flotteur)

C'est un capteur magnétique qui contient un relais reed (contact à bilame activé par un champ magnétique). C'est un détecteur de niveau qui renvoie deux valeurs logiques 1 ou 0. Il servira à déterminer le remplissage des réservoirs d'eau. [11]

Flotteur	ESP32
Signal	GPIO 12
GND	GND

Tableau 11 : Tableau de connexion de Flotteur avec ESP32

2.8. Pompe à eau

Nous avons extrait ces caractéristiques par un essai, on a mis 2 bouteilles d'eau puis on a activé la pompe pour conclure la distance et la quantité d'eau par rapport au temps et au courant.

Nous avons constaté que elle permet de remplir 1L dans 1min avec un courant 370mA et une distance 2.5m.



Les trois bidons seront remplis parallèlement, l'un contient un capteur ultrason qui permet de mesurer le niveau d'eau, afin d'assurer la sécurité de capteur on a attaché deux tuyaux au-dessus de bouteille en cas d'excès d'eau.

Pour conserver l'eau remplis on a une pompe pour descendre l'eau au bidon en haut.

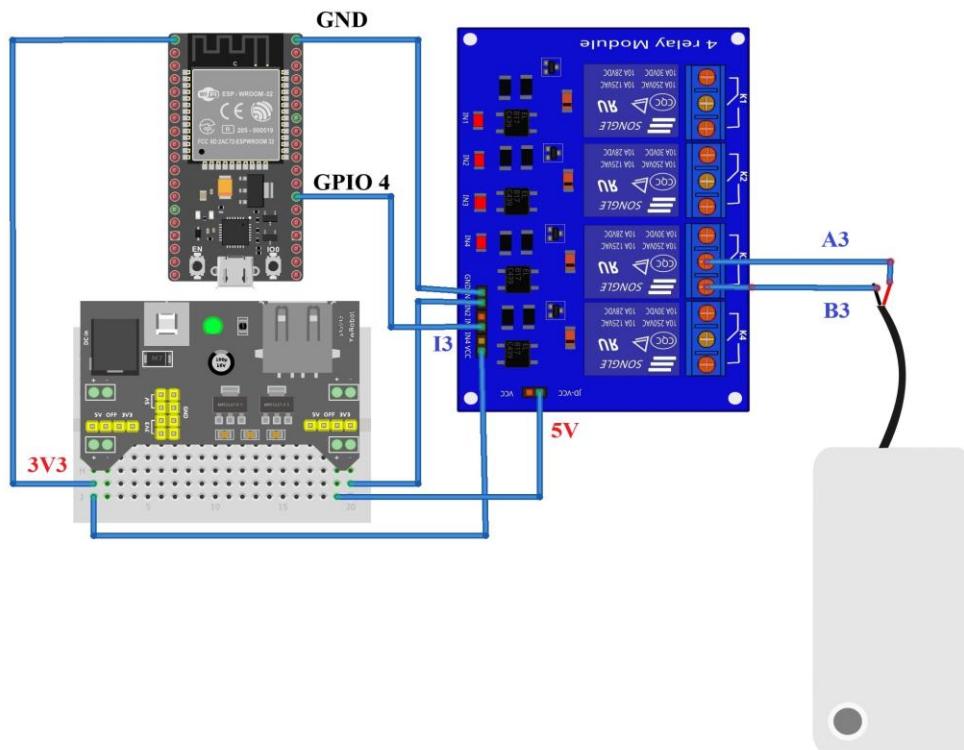




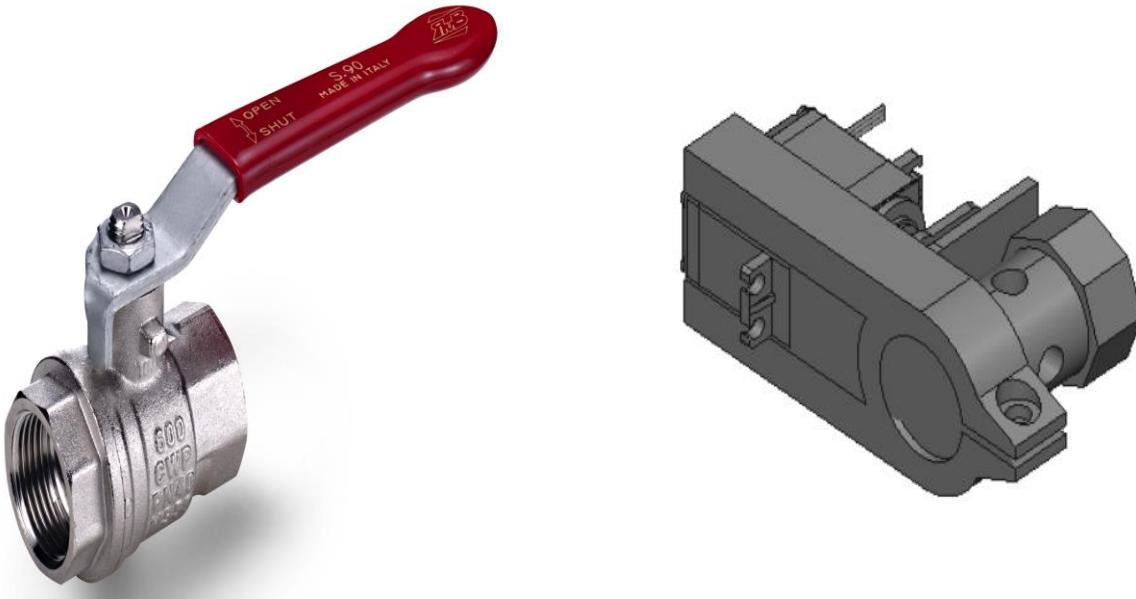
Figure 40 : Câblage de la pompe

2.9. Conception de la vanne motorisée

La conception de notre solution a été réalisée sur un logiciel de conception 3D SolidWorks et la fabrication des pièces nous avons utilisé la technologie fabrication additive avec une imprimante 3D.

Modèle du support : le support modélisé est composé de deux pièces l'un support pour le servo et l'autre une tige que on peut fixer la vanne dont son axe de manœuvre coaxial avec celui du servomoteur pour augmenter le couple. La totalité de la pièce va être fixe dans le poteau du jardin à travers des vis. Cette solution assure aussi l'étanchéité du servomoteur.

La conception est mentionnée dans la figure ci-dessous :



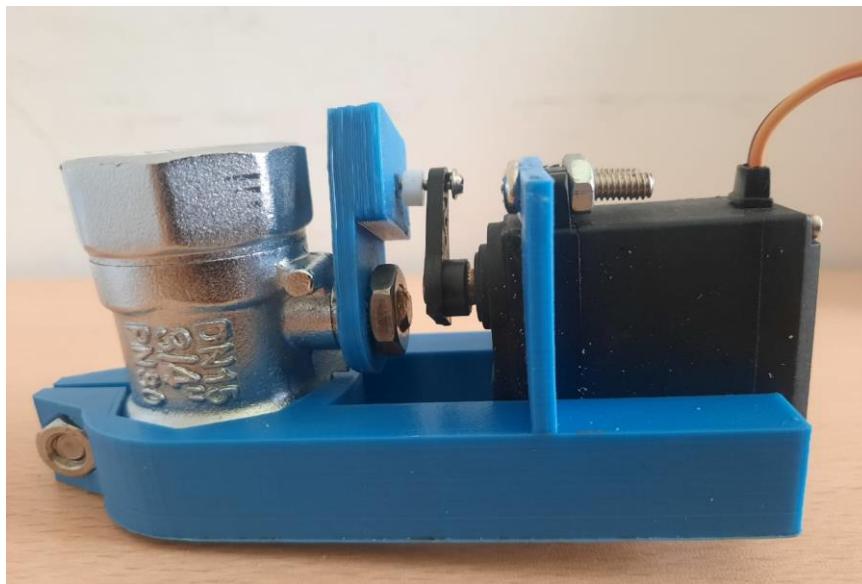


Figure 41 : Conception de la vanne motorisée

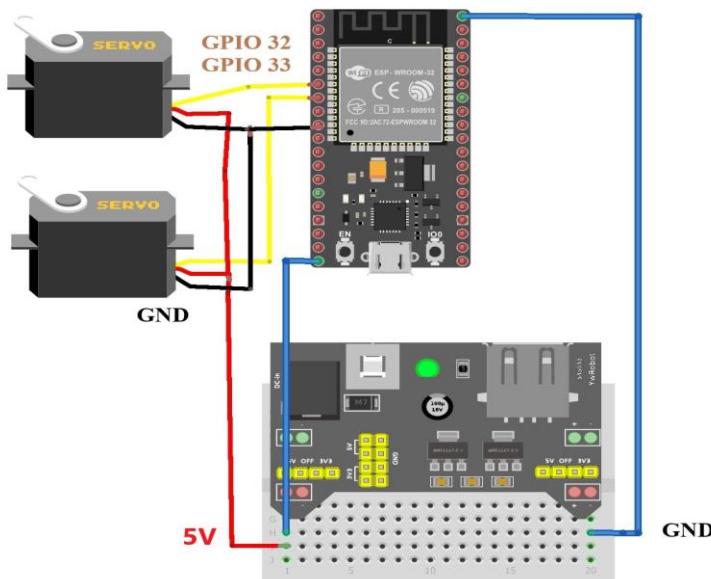
Test de La vanne motorisé :

Nous avons rempli une bouteille d'eau (équivalente à 1kg) puis nous avons ajouté l'eau lentement jusqu'au l'ouverture de la vanne afin de conclure le couple de la vanne à travers la force appliquée

D'après la datasheet du servomoteur : 11kg → 1450mA Donc pour 1.5kg → 200 mA

Servo	ESP 32
Servo 1 (pin signal)	GPIO 32
Servo 2 (pin signal)	GPIO 33

Tableau 12 : Tableau de connexion de Servo avec ESP32



Jaune => Signal PWM TTL standard de modélisme, de période 20ms avec un temps haut compris entre 0,7 et 2,3ms environ

Rouge => Alimentation positive entre 4,8V et 6V

Noir => Masse de l'alimentation et du signal

Figure 42 : câblage des servos

2.10. Câblage des lampes

Les lampes sont commandées par des boutons poussoirs à travers ESP32 :

Boutons poussoirs	ESP 32
Bouton lampe 1	GPIO 26
Bouton lampe 2	GPIO 27

Tableau 13 : Tableau de connexion des lampes avec ESP32

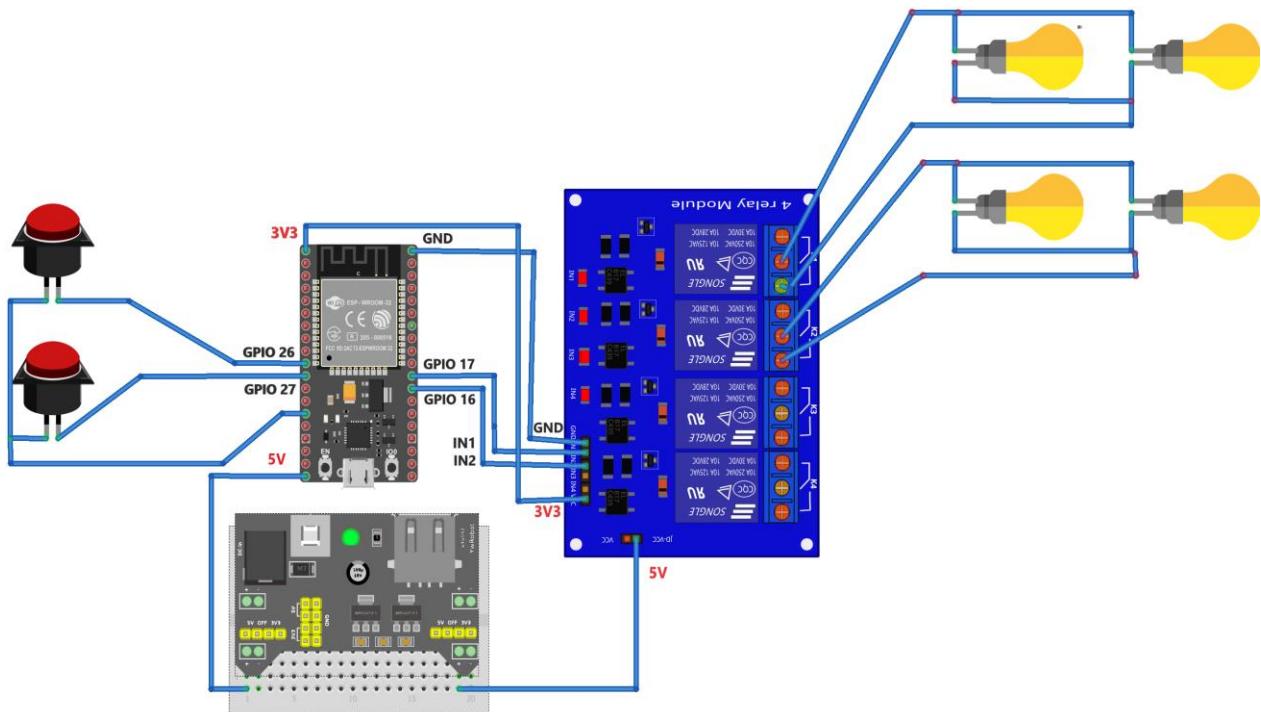


Figure 43 : Câblage des lampes

III. La partie Software

Reste à voir comment fonctionnent le code Arduino et l'application Android. Comme le code est un peu plus long, pour une meilleure compréhension, nous posterons le code source du programme en sections avec une description pour chaque section.

1. Architecture de la plateforme

Le microcontrôleur ESP 32 traite plusieurs grandeurs physiques mesurés à partir des différents capteurs et les transmette à une base de données via une connexion sans fil Wifi où ils seront stockés.

L'application mobile récupère ces données pour les afficher et permettre aux utilisateurs de superviser l'état de chaque capteur et actionneur ainsi à leur commande à distance.



Figure 44 : Architecture de la plateforme

2. La Configuration de Firebase

Une Firebase est une plate-forme intégrée à l'application pour stocker les données en ligne. Il est également utilisé comme système d'authentification. Un utilisateur peut envoyer et récupérer les données requises à tout moment.

La première étape pour nous est de mettre en place la base de données cheminée dans le cloud par la création d'un compte puis un nouveau projet. Les figures ci-dessous illustrent des détails sur notre base de données. [6]

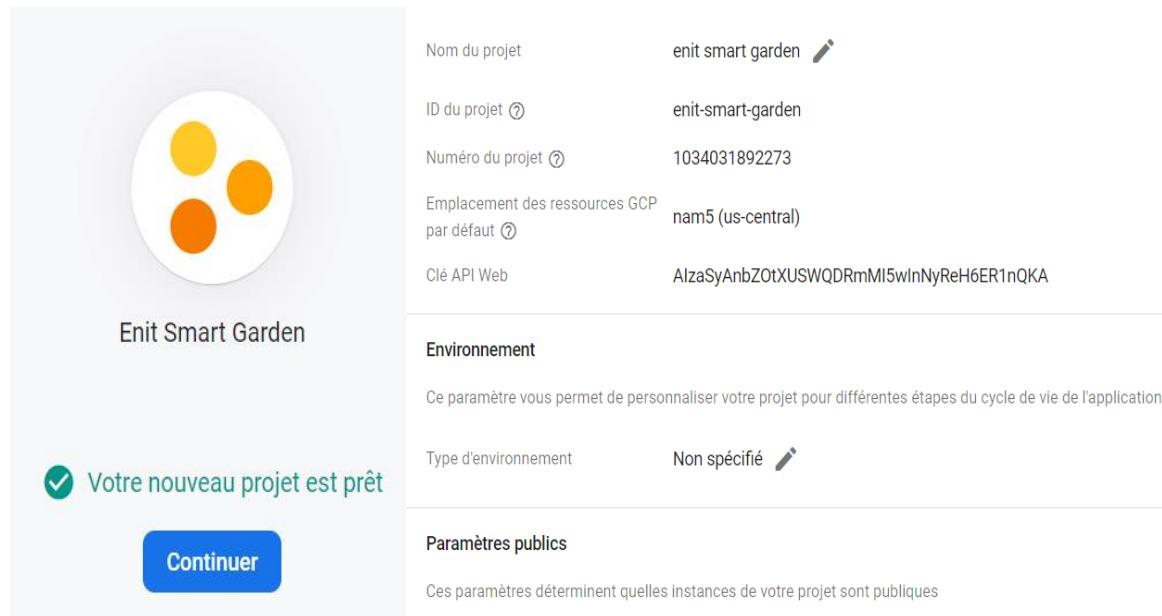


Figure 45 : Firebase d'ENIT SMART GARDEN

Cela va associer les règles avec la base de données. Il indique également clairement que nous avons défini notre base de données pour être accessible au public (lire et écrire).

```

1: {
2:   "rules": {
3:     ".read": false,
4:     ".write": false
5:   }
6: }

```

Maintenant, tout ce dont nous avons besoin est d'écrire ceci à notre application Android.

3. L'application Android « ENIT SMART GARDEN »

3.1. Développement l'application Android ENIT Smart Garden

L'application Android est créée à l'aide de l'application en ligne MIT App Inventer. Il se compose de boutons simples qui ont des images appropriées en arrière-plan.

La prochaine étape est de concevoir notre application. Nous allons garder les choses simples et la capture d'écran pour l'application est donnée ci-dessous :

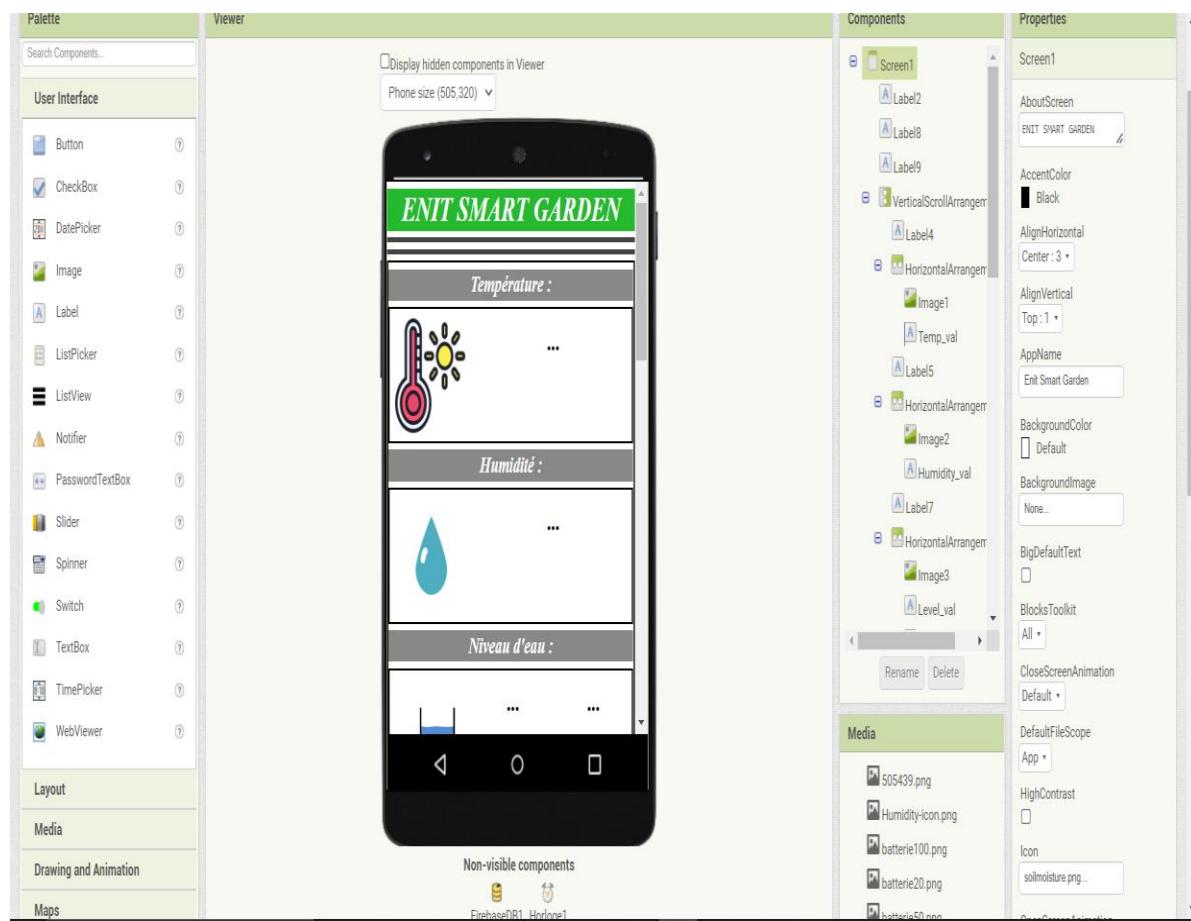
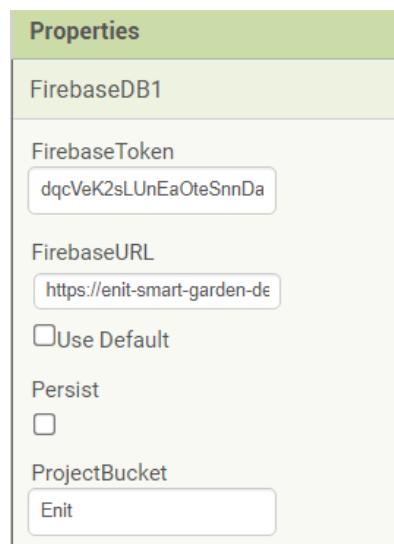


Figure 46 : l'application Android ENIT Smart Garden

La chose importante à configurer ensuite est notre composant FirebaseDatabase que nous avons ajouté à l'écran comme composant non visible. Cliquez sur le composant FirebaseDatabase et voyez les propriétés. App Inventor héberge un service dans le backend où vous pouvez utiliser leur version hébergée (par défaut) de Firebase, mais nous utiliserons notre base de données que nous avons créée dans la section précédente. Pour ce faire, nous devrons donner la valeur de la propriété DatabaseReference comme celle que nous avons notée plus tôt. [7]



Passez maintenant à la section Blocs de l'application : La méthode StoreValue du composant Firebase prend deux paramètres, la balise et la valeurToStore. [8]

Notez que nous déclarons des variables pour chacune des valeurs que nous voulons collecter et les initialisons à des valeurs vides. Nous stockons ensuite les valeurs des composants de l'interface utilisateur dans ces valeurs, puis nous invoquons deux fois la méthode StoreValue des composants FirebaseDatabase. Une pour le nom et l'autre pour les commentaires. [10]

Un extrait d'ensemble des blocs est donné ci-dessous :

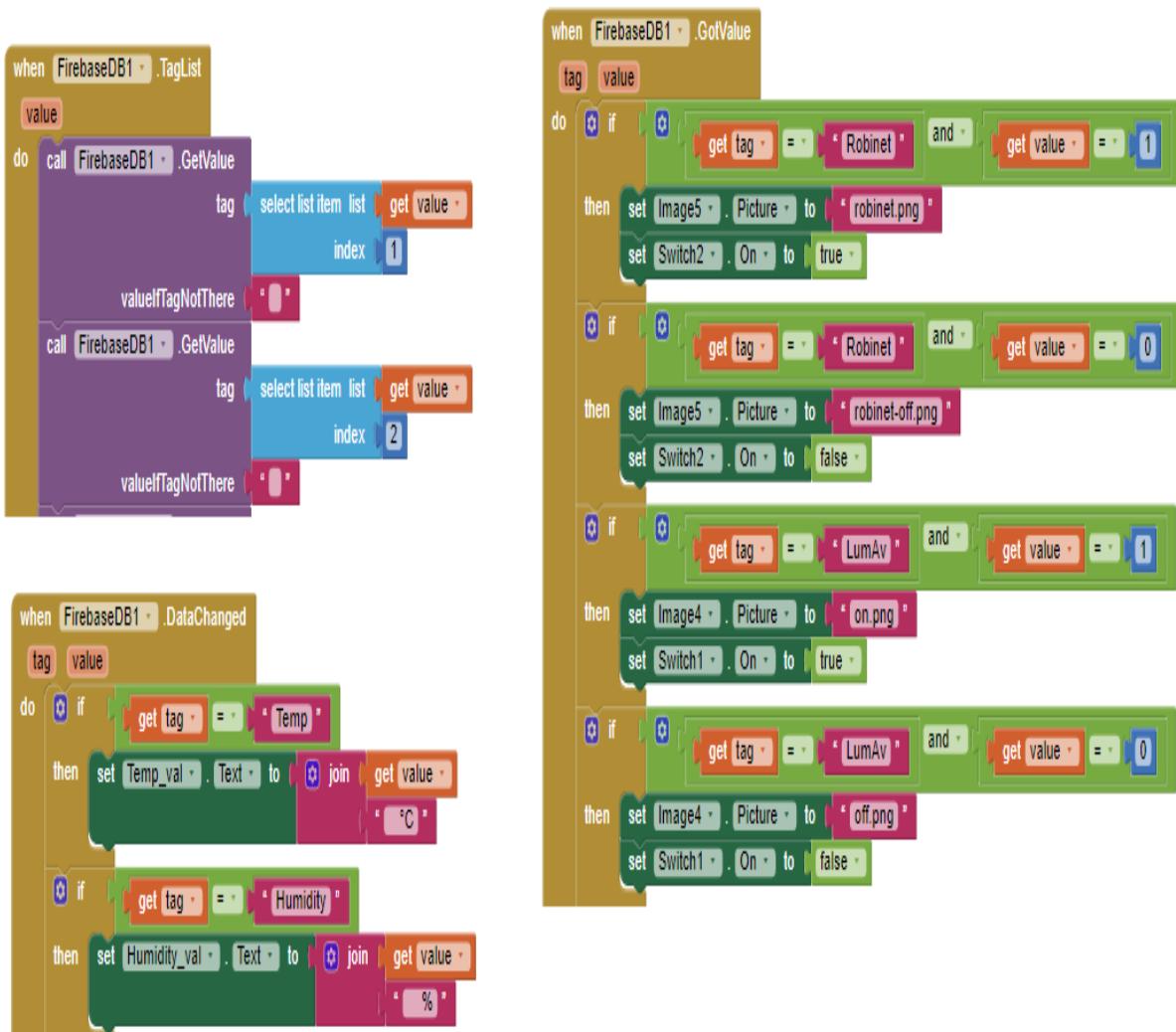


Figure 47 : blocs App Inventor

Les figures ci-dessous illustre une simulation entre l'application et la base de données, il s'agit de contrôler les lampes avant et l'électrovanne. Les données seront envoyées de l'application vers Firebase et de Firebase vers l'ESP 32 afin d'envoyer un signal aux pré actionneur.

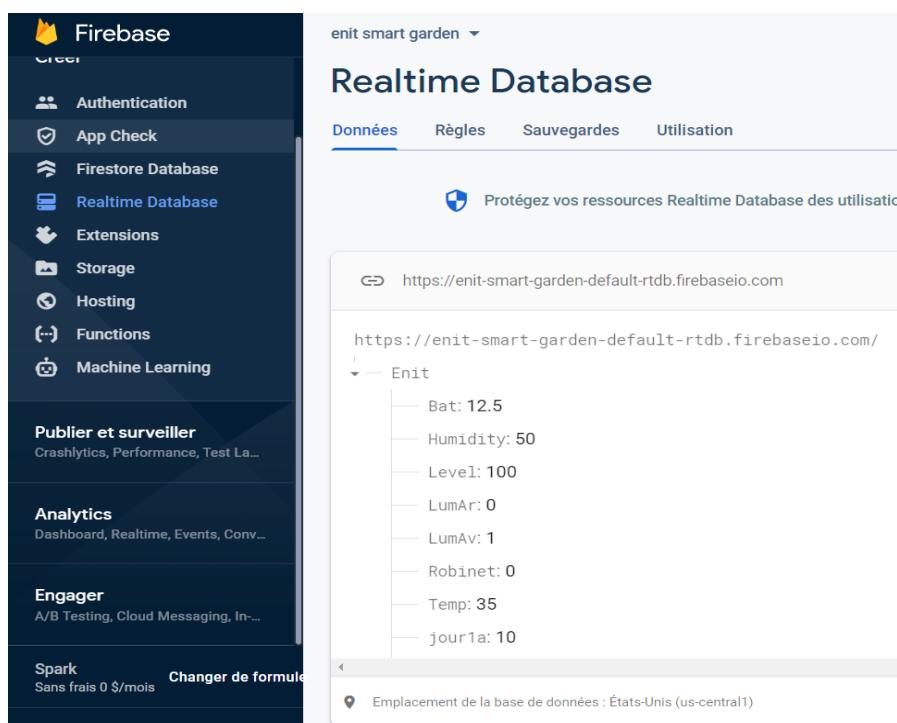
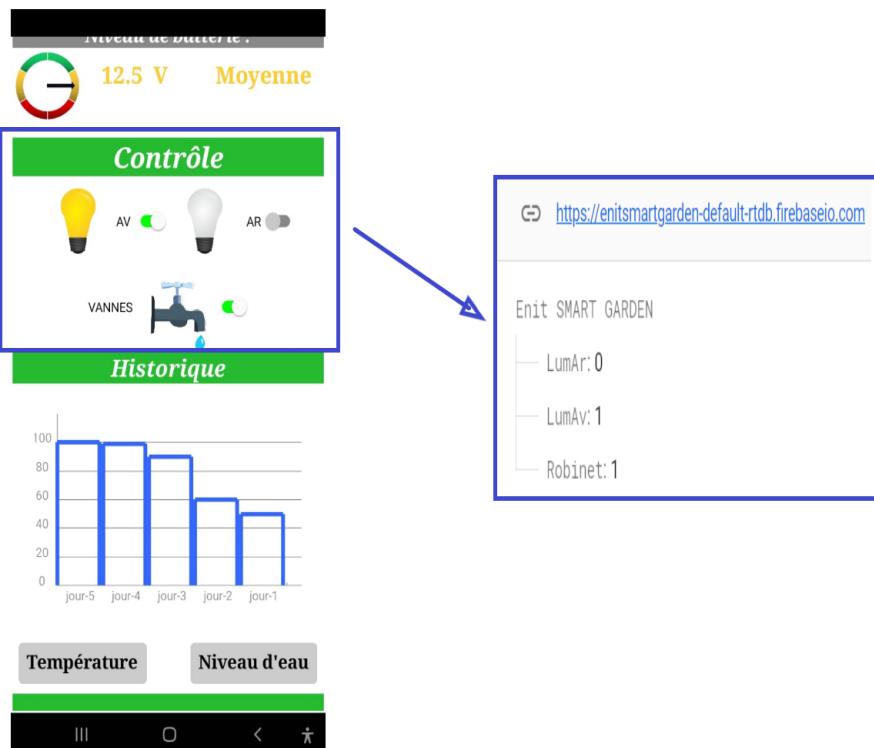


Figure 48 : base donnée de SMART GARDEN

3.2. Guide d'utilisateur

Notre solution basée sur la plateforme app inventer. Elle peut contrôler les matériels à distance, afficher les données des capteurs, stocker des données et les visualiser. (Voir **Annexe 7**)

Notre application est composée de 3 parties comme mentionnée ci-dessous :

- Un espace qui contient des valeurs à afficher
- Un espace qui contient des boutons de commande
- Une historique qui contient l'évolution de température et niveau d'eau

4. Programmation de l'ESP-32

Pour injecter un programme dans la mémoire de l'ESP depuis notre ordinateur on utilise l'environnement de développement Arduino, Il y a une condition à remplir au début.

Activer le mode "flash" : Ce mode est nécessaire pour injecter un programme dans la mémoire de l'ESP. Test de bon fonctionnement de l'ESP 32

4.1. Connexion a un point d'accès WIFI et Firebase

```
//Connexion a un point d'accès WIFI et Firebase
#include <WiFi.h>
#include <FirebaseESP32.h>
#define FIREBASE_HOST "https://enit-smart-garden-default-rtdb.firebaseio.com"
#define WIFI_SSID "ENITWireless" // Change the name of your WIFI
#define WIFI_PASSWORD "ENITSTUDENT" // Change the password of your WIFI
#define FIREBASE_Authorization_key "dqcVeK2sLUnEaOteSnnDa1rl8eBwYfK7RWbcfUvi"
FirebaseData firebaseData;
FirebaseJson json;

Serial.begin(115200);
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
Serial.print("Connecting...");
while (WiFi.status() != WL_CONNECTED)
{
    Serial.print(".");
    delay(300);
}
Serial.println();
Serial.print("IP Address: ");
Serial.println(WiFi.localIP());
Serial.println();

Firebase.begin(FIREBASE_HOST, FIREBASE_Authorization_key);
```

4.2. Déclarations des variables

```
#define DHTPIN 14
const int trigPin = 5;
const int echoPin = 18;
#define RELAIS_POMPE 4
#define RELAIS_LAMPES_AV 17
#define RELAIS_LAMPES_AR 16
#define RELAIS_GENERAL 0
#define BUTTON_AV 26
#define BUTTON_AR 27
#define BAT_AV 35
#define BAT_GEN 34
#define Led 2
#define Buzzer 15
#define flotteur 12
```

Void Setup :

```
myservo1.attach(32);
myservo2.attach(33);
temps.begin();
pinMode(BUTTON_AV, INPUT);
pinMode(BUTTON_AR, INPUT);
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
pinMode(flotteur, INPUT);
pinMode(BAT_GEN, INPUT);
pinMode(BAT_AV, INPUT);
pinMode(RELAIS_POMPE, OUTPUT);
pinMode(RELAIS_LAMPES_AV, OUTPUT);
pinMode(RELAIS_LAMPES_AR, OUTPUT);
pinMode(RELAIS_GENERAL, OUTPUT);
pinMode(BUZZER, OUTPUT);
dht.begin();
```

4.3. Obtention du date et heure

```
//Obtention du date et heure
#include <NTPClient.h>
#include <WiFiMulti.h>
#include <WiFiUdp.h>
WiFiUDP ntpUDP;
NTPClient temps(ntpUDP, "fr.pool.ntp.org", 3600, 60000);

//Temps et date
temps.update();
epochTime = temps.getEpochTime();
heure = temps.getHours();
minutes = temps.getMinutes();
seconde = temps.getSeconds();
struct tm *ptm = gmtime ((time_t *)&epochTime);
day = ptm->tm_mday;
Serial.print("Month day: ");
Serial.println(day);

mois = ptm->tm_mon+1;
Serial.print("Month: ");
Serial.println(mois);

mois_chaine = months[mois-1];
Serial.print("Month name: ");
Serial.println(mois_chaine);

annee = ptm->tm_year+1900;
Serial.print("Year: ");
Serial.println(annee);
}
```

Pour commander le système d'irrigation, on a programmé le microcontrôleur pour suivre la courbe pluviométrique, Le cycle d'irrigation se répartit en :

Mois	Nombre d'irrigation
Janvier	Une fois par semaine
Février	Une fois par semaine
Mars	Deux fois par semaine
Avril	Deux fois par semaine
Mai	Trois fois par semaine
Juin	Trois fois par semaine

Juillet	Quatre fois par semaine
Août	Trois fois par semaine
Septembre	Une fois par semaine
Octobre	Une fois par semaine
Novembre	Une fois par semaine
Décembre	Une fois par semaine

Tableau 14 : Le cycle d'irrigation

4.4. Système d'irrigation

Lorsque la date est convenable à l'irrigation, on commande les servomoteurs pour ouvrir la vanne. Le mouvement du servo est une rotation de 90 degrés vers la gauche pour une durée de 15 minutes (temps d'irrigation) et puis fermeture des vannes.

Ci-dessous un extrait de code d'arrosage :

```
void Arrosage()
{
    Temps();
    //Arrosage automatique
    if ((mois == 1 && day == 2) || (mois == 1 && day == 9) || (mois == 1 && day == 16)
    || (mois == 1 && day == 23) || (mois == 1 && day == 30) )
    {
        if(heure == 0 && minutes ==0 && (seconde == 0 || seconde == 1 || seconde == 2 || seconde == 3))
        {
            for (pos = 0; pos <= 90; pos += 1)
            {
                myservo1.write(pos);
                myservo2.write(pos);
                delay(15);
            }
        }
        if(heure == 0 && minutes ==15 && (seconde == 0 || seconde == 1 || seconde == 2 || seconde == 3))
        {
            for (pos = 90; pos >= 0; pos --)
            {
                myservo1.write(pos);
                myservo2.write(pos);
                delay(15);
            }
        }
    }
}
```

4.5. Système de gestion des batteries

Si le niveau des deux batteries est inférieur à 12.5 donc on doit activer un buzzer pour alerter les personnes qui se trouvent dans le jardin que tous les processus seront arrêtés (éclairage, irrigation, alimentation PC et téléphones ...) et on attend le chargement des batteries.

Si les batteries subissent une surcharge et leurs niveaux dépassent 25.5 V donc on coupe la liaison entre le PV et les batteries.

```

void Battery()
{
    Bat_Av = (digitalRead(BAT_AV)/4095)*37.5;
    Bat_Gen = (digitalRead(BAT_GEN)/4095)*37.5;
    Bat_Ar = Bat_Gen - Bat_Av ;
    Firebase.setFloat(firebaseData, "/Enit/Bat_Av", Bat_Av);
    Firebase.setFloat(firebaseData, "/Enit/Bat_Ar", Bat_Ar);
    Firebase.setFloat(firebaseData, "/Enit/Bat_Gen", Bat_Gen);

    Serial.println(Bat_Av);

    // Alarme et coupure recharge par panneau
    while (Bat_Av > 12.5)
    {
        Serial.println("programme lumiere");
        digitalWrite(RELAIS_GENERAL, LOW);
        digitalWrite(BUZZER, HIGH);
        digitalWrite(RELAIS_LAMPES_AV, HIGH);
        digitalWrite(RELAIS_LAMPES_AR, HIGH);
        Temperature();
        Flotteur();
        Niveau();
        Acquisition();
        Vanne();
        Arrosage();
        Allumer1();
        Allumer2();
    }
}

```

4.6. Système de Gestion du réservoir 1

Le réservoir I est géré par un capteur ultrasons qui mesure la hauteur de l'eau et envoie l'information au Firebase puis vers l'application.

```
void Niveau()
{
    //Niveau eau ultrason
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    // Sets the trigPin on HIGH state for 10 micro seconds
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);

    // Reads the echoPin, returns the sound wave travel time in microseconds
    duration = pulseIn(echoPin, HIGH);

    // Calculate the distance
    distanceCm = duration * SOUND_SPEED/2;
    Serial.print("distance: ");
    Serial.println(distanceCm);
    distancePercent = 100 - distanceCm/70*100;
    Serial.println(distancePercent);
    Firebase.setFloat(firebaseData, "/Enit/Level", (int)distancePercent);
}
```

4.7. Commande de la pompe

Le flotteur fixé au niveau du réservoir 1 délivre toujours un signal numérique. Si l'état du capteur est égal à 1 on actionne la pompe pour remplir l'eau du réservoir 1 vers 2. Lorsque l'état du flotteur change à 0 donc le réservoir est rempli et on arrête la pompe.

```
void Flotteur ()
{
    etat_FLOT = digitalRead(flotteur);
    if (etat_FLOT == 1)
    {
        digitalWrite(RELAIS_POMPE, HIGH);
    }
    else
    {
        digitalWrite(RELAIS_POMPE, LOW);
    }
}
```

4.8. Système d'éclairage du jardin

La commande d'éclairage du jardin se fait à travers les boutons poussoirs muni d'une minuterie réalisé par une fonction sur le code ci-dessous

L'appui une fois active les lampes qui seront automatiquement d'activées après 10 minutes.

```
void Allumer1()
{
    if (Firebase.RTDB.getInt(&firebaseData, F("/Enit/LumAv")))
    {
        String(firebaseData.toInt<int>()).c_str();
        LumAv = firebaseData.intData();
        Serial.print("lumAv = ");
        Serial.println(LumAv);

    }
    //Lecture des boutons posés sur la table
    ButtonAv = digitalRead(BUTTON_AV);
    //Allumer les lampes
    if (( ButtonAv == HIGH) || (LumAv == 1))
    {
        Serial.println("programme lumiere av");
        Serial.println("button 1 activé");
        temps_depart = millis();
        digitalWrite(RELAIS_LAMPES_AV, HIGH);
        while ((millis() - temps_depart) <= 600)
        {
            Temperature();
            Flotteur();
            Niveau();
            Acquisition();
            Vanne();
            Arrosage();
            Allumer2();
            Battery();
        }
        digitalWrite(RELAIS_LAMPES_AV, LOW);
        Firebase.setInt(firebaseData, "/Enit/LumAv", 0);
    }
}
```

4.9. Supervision de la température et d'humidité

Le capteur DHT Il permet de superviser la température et l'humidité à temps réel à partir de l'application Android.

```
void Temperature()
{
    // temperature
    float hum = dht.readHumidity();
    float temp = dht.readTemperature();
    if (isnan(hum) || isnan(temp))
    {
        Serial.println(F("Failed to read from DHT sensor!"));
        return;
    }

    Serial.print("Temperature: ");
    Serial.print(temp);
    Serial.print("°C");
    Serial.print(" Humidity: ");
    Serial.print(hum);
    Serial.print("%");
    Serial.println();

    Firebase.setFloat(firebaseData, "/Enit/Temp", temp);
    Firebase.setFloat(firebaseData, "/Enit/Humidity", hum);
}
```

Conclusion

Ce chapitre était pour le but d'exposer les différentes parties réalisées durant ce projet et d'exhiber les différents logiciels utilisés.

Conclusion Générale

Ce projet de fin d'année nous a permis de mettre en pratique les connaissances théoriques et pratiques acquises durant notre formation à ISET RADES.

L'environnement de l'institut était très intéressant. En effet, nous avons pu observer le fonctionnement d'un laboratoire de recherche et l'articulation des différents départements. Ainsi le directeur et tous les membres de l'équipe se côtoient et sont aisément accessibles.

De ce fait, la communication est simple et aisée entre les différents membres de l'équipe. Par ailleurs, les relations humaines entre les différents employés, indépendamment de l'activité exercée par chacun d'eux, nous ont appris sur le comportement à avoir en toutes circonstances.

Dans ce projet nous avons fait **la conception** et **la réalisation** des cartes chargeur téléphones et pcs et d'une carte de commande qui sert à :

- **Gérer** le système d'irrigation.
- **Commander** l'éclairage, la pompe et les électrovannes.
- **Assurer** le processus de chargement des PCS et des téléphones dans les mesures de sécurité.

Nous avons développé la partie programmation du microcontrôleur pour gérer les tâches de notre système.

Enfin nous avons réalisé **une application Android** liée à **une base de données** en temps réel.

En outre, les perspectives d'évolution de notre projet sont :

- Introduire la notion de l'intelligence artificielle dans notre système pour permettre d'effectuer des prédictions météorologiques plus précises et transformer le système de supervision de la température et d'humidité en une station météorologique complète.
- Travailler avec un microcontrôleur plus puissant (Raspberry Pi, STM32 ..) qui peut nous assurer une meilleure performance et plus d'outils efficaces et révolutionnaires

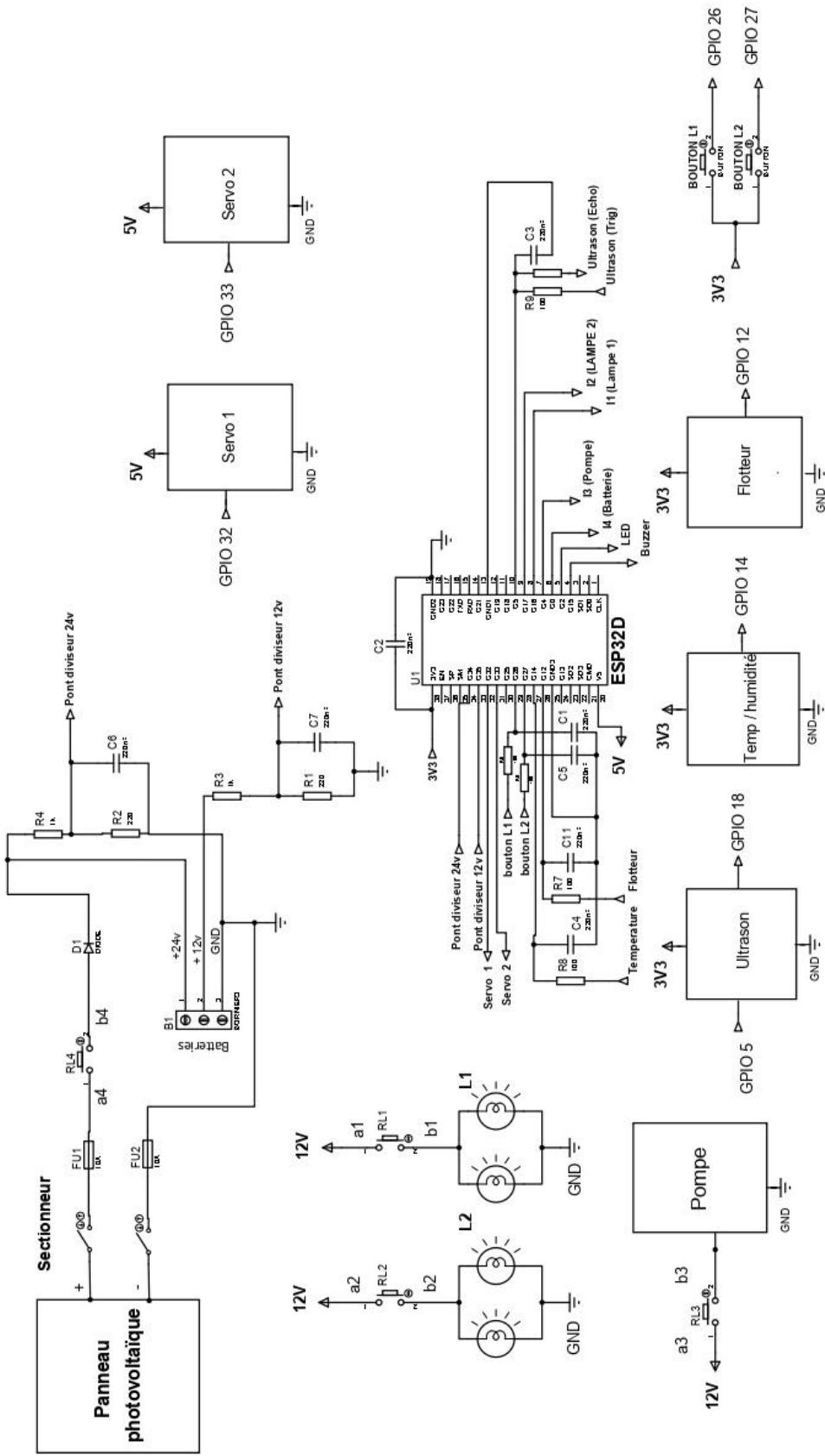
Nétographie

- [1] La réalisation d'une carte électronique : les logiciels, les différentes technologies, les étapes de fabrication (electronique-mixte.fr)
- [2] Batterie : memoire syphax.pdf (univ-bejaia.dz)
- [3] <https://www.meteo.tn/fr/pluviometries>
- [4] Chargeur 5V : Electronique - Realisations - Chargeur 003 (sonelec-musique.com)
- [5] ESP32 : Tutoriel ESP32.pdf (fablabutc.fr)
- [6] Support_formation_Arduino.pdf (myhumankit.org)
- [7] Tutorial: MIT App Inventor + Firebase: Tutorial: MIT App Inventor + Firebase | by Romin Irani | Romin Irani's Blog
- [8] Firebase: sfirebase (onvaessayer.org)
- [9] Datasheet HC-SR04 (robot-maker.com)
- [10] How to link Firebase to MIT App Inventor and Send or Retrieve data from Firebase database using MIT App Inventor 2 — Steemit
- [11] Capteur de niveau d'eau à flotteur - MCHobby - Vente de Raspberry Pi, Arduino, ODROID, Adafruit

Annexes

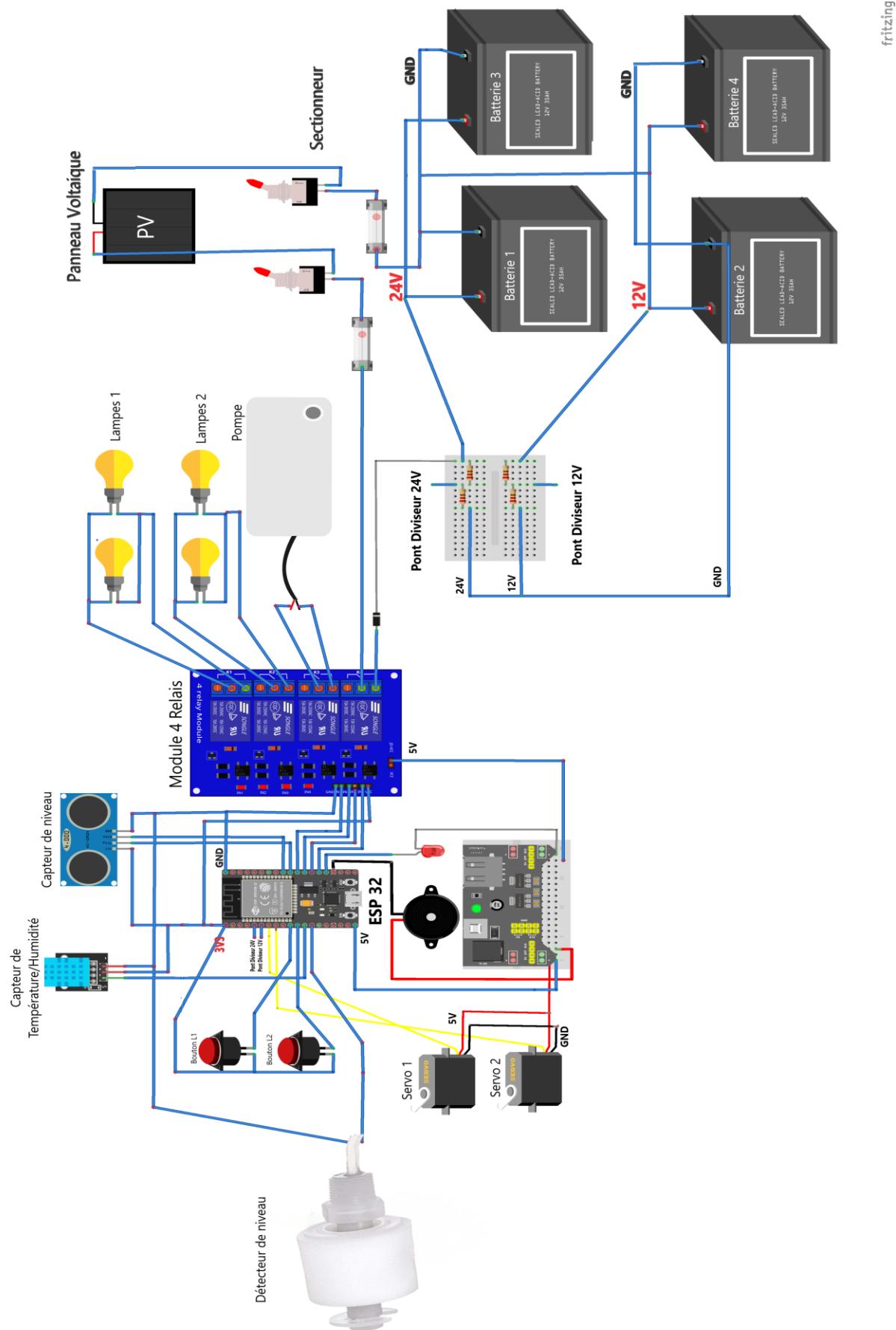
Annexe 1 : Synoptique de la carte commande

Synoptique Carte de Commande



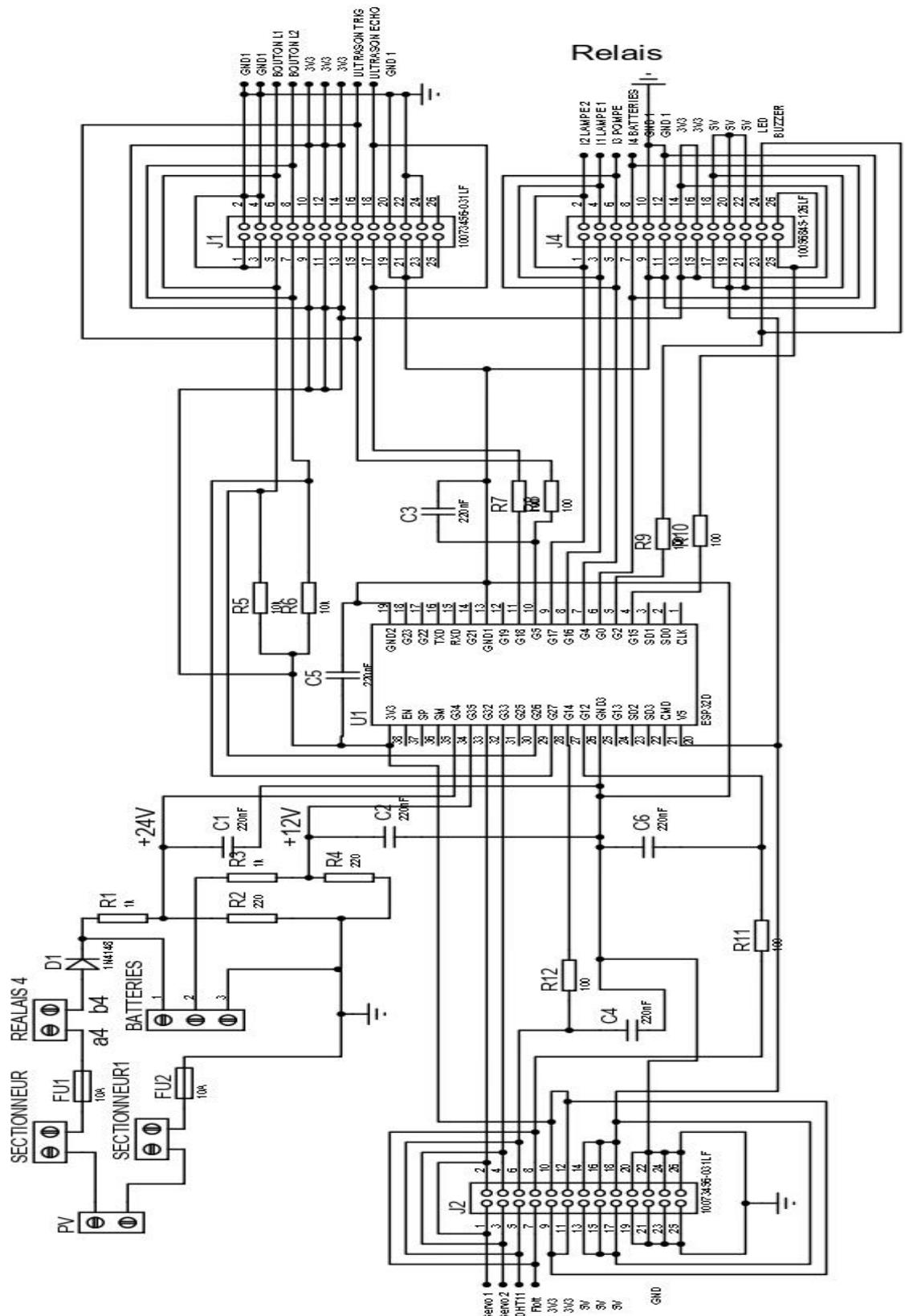
Tous les entrées doivent être capatisées (capacité de découplage = 220nF)

Annexe 2 : Câblage de l'ESP 32, Relais, les actionneurs, les capteurs et l'alimentation

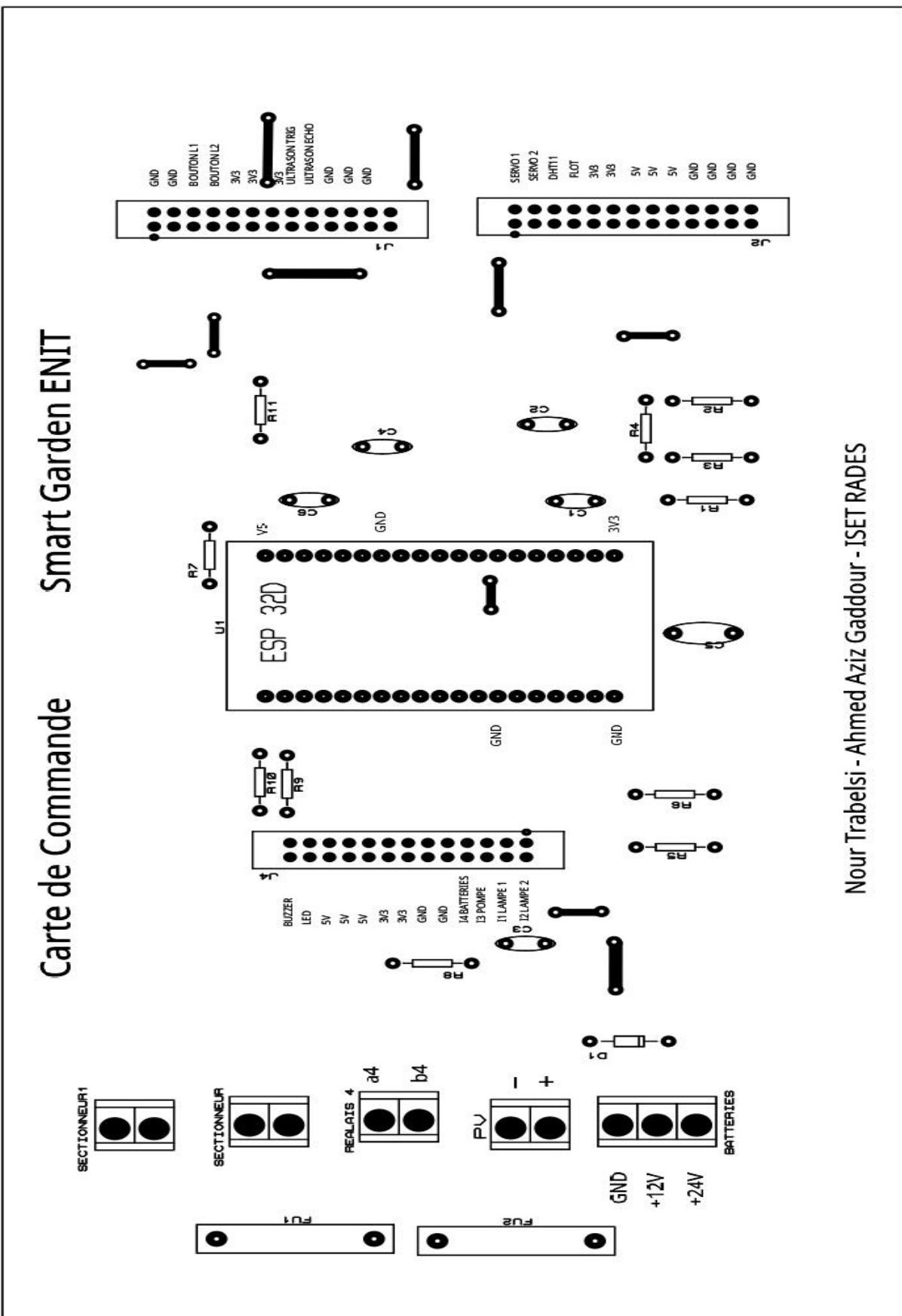


Carte de commande - ENIT SMART GARDEN

Annexe 3 : Schéma de la carte de commande



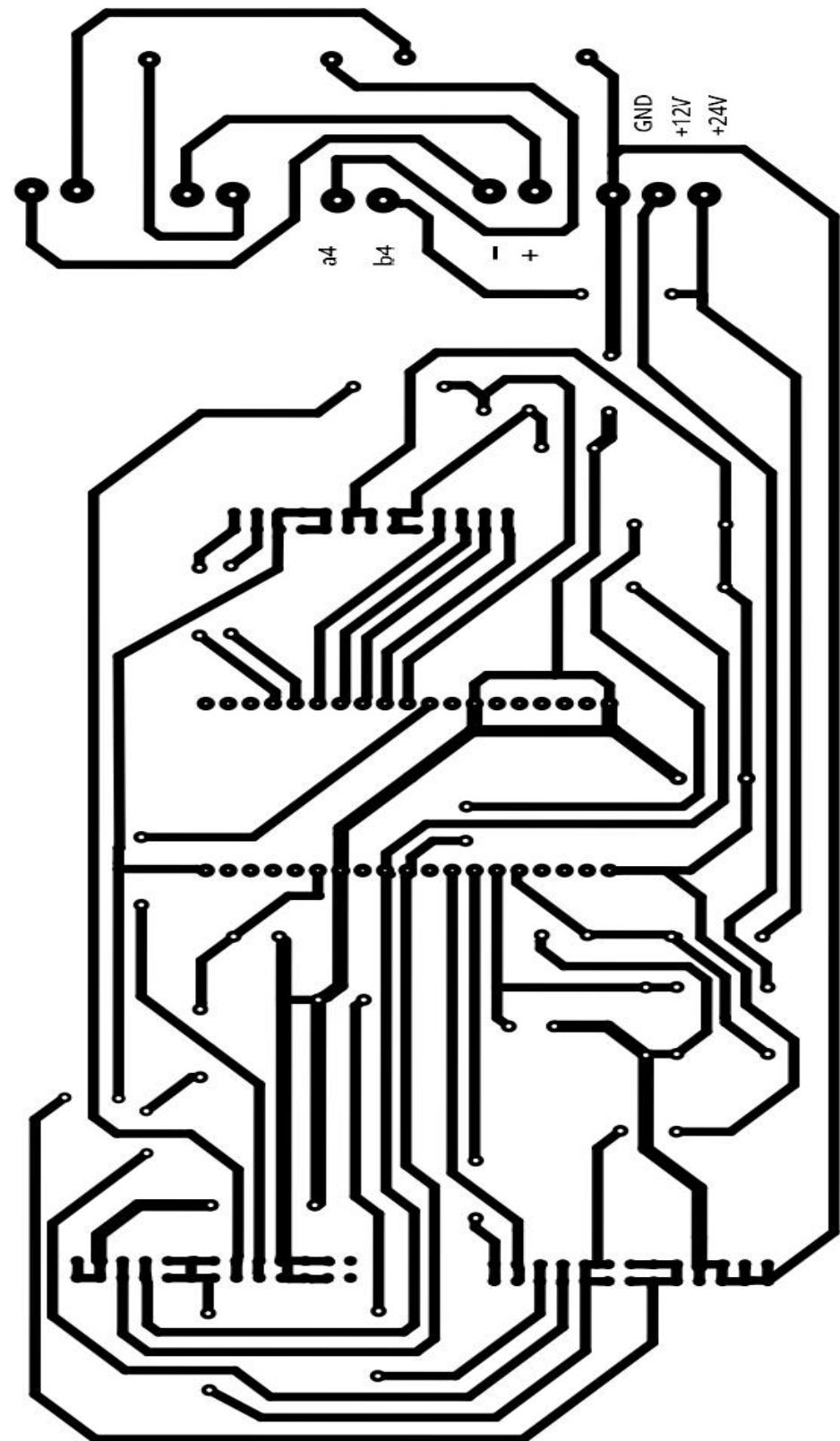
- Face composant de la carte de commande



Nour Trabelsi - Ahmed Aziz Gaddour - ISET RADES

- Face cuivre de la carte de commande

Smart Garden ENIT
Carte de Commande



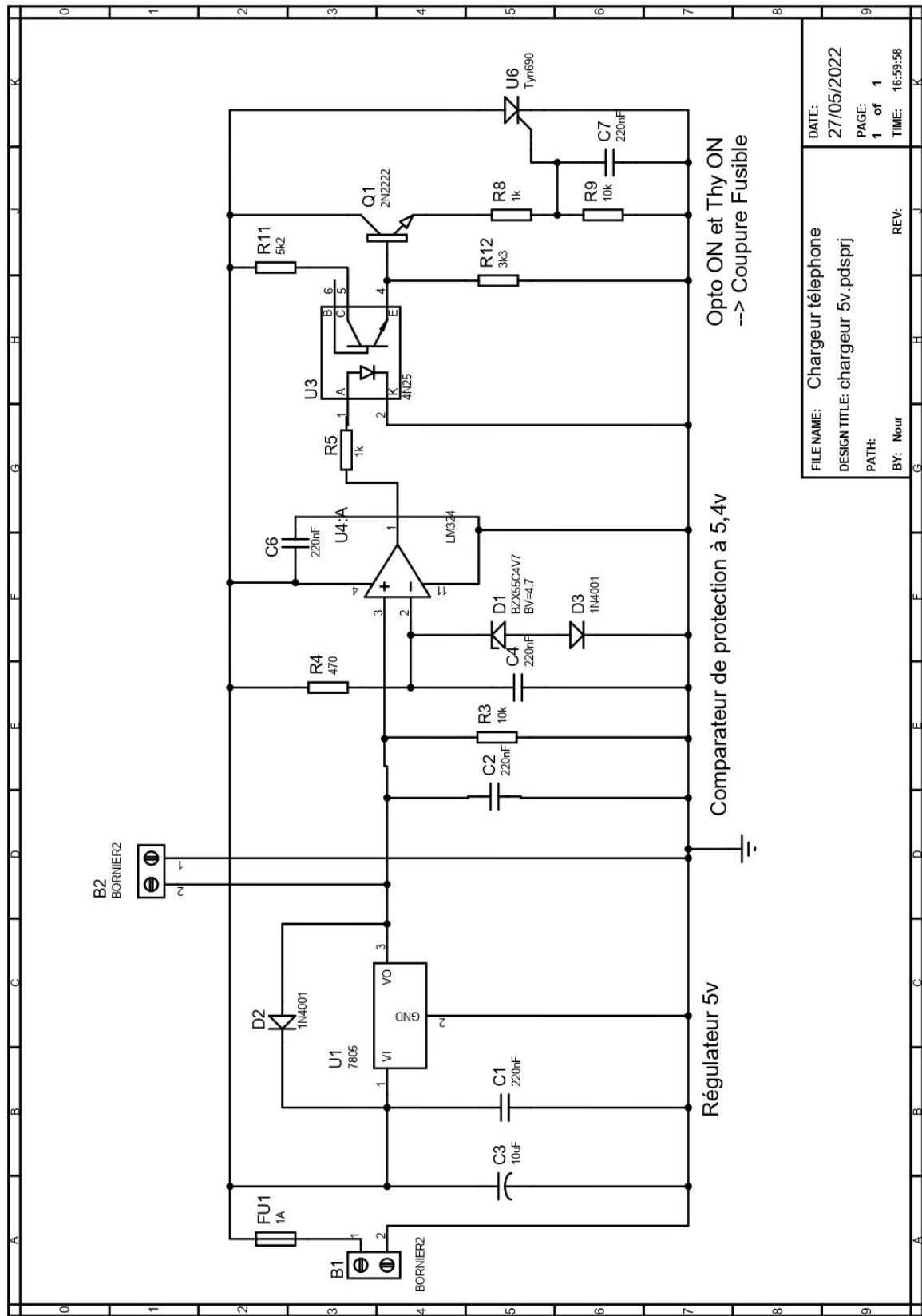
Nour Trabelsi - Ahmed Aziz Gaddour - ISET RADES

Annexe 4 : Liste des composants électroniques

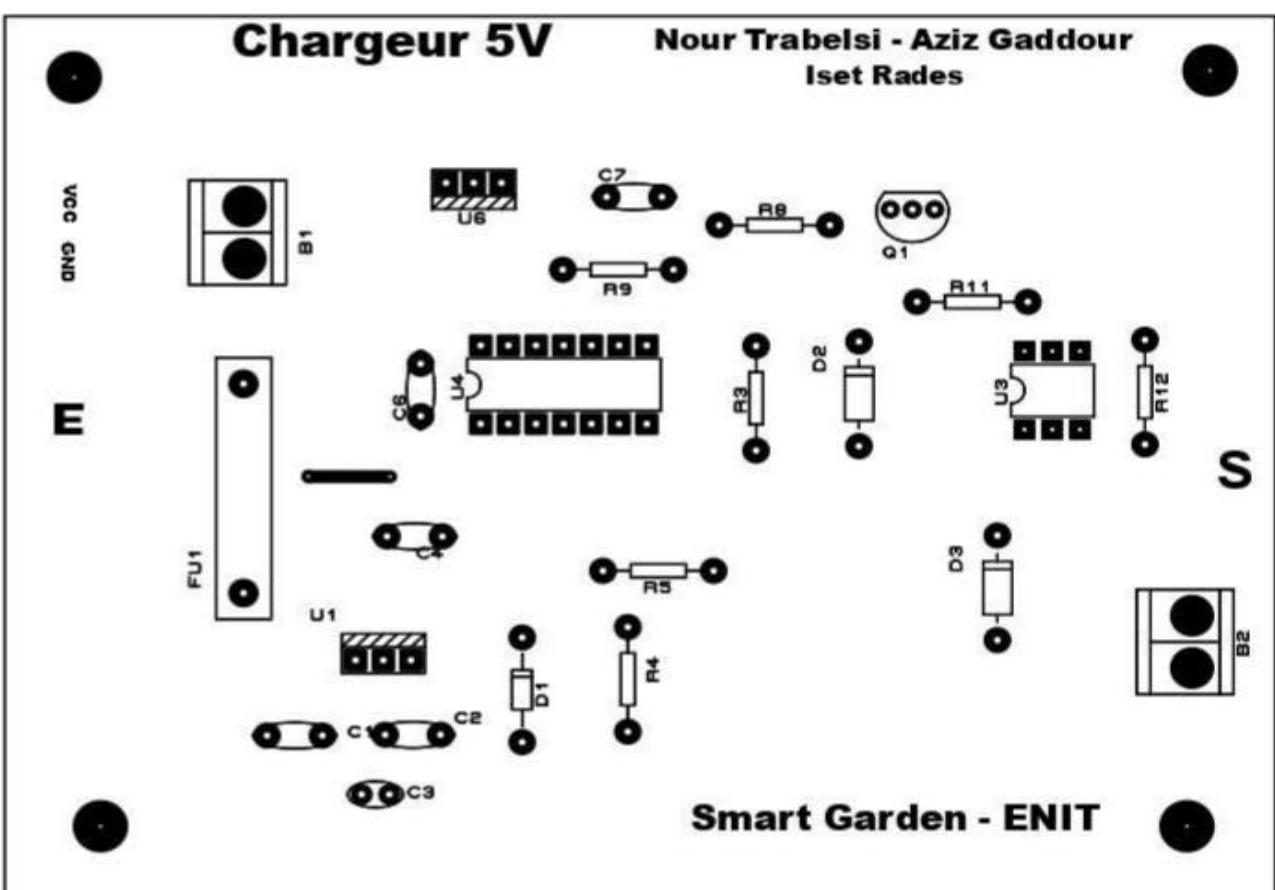
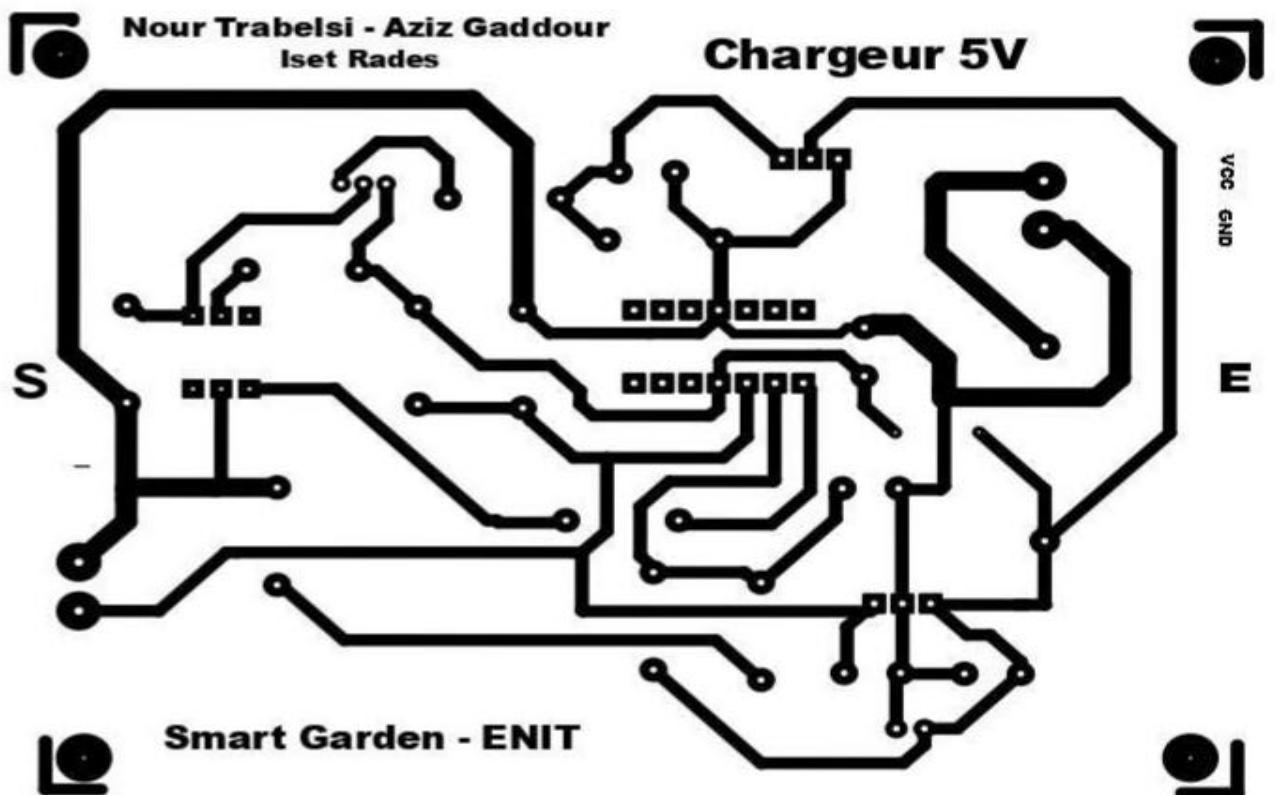
Nom Composant	Qte	Valeur	Description
Capacité céramique	8	220nF	
Capacité chimique	4	10uF	
Capacité chimique	8	1uF	
Capacité céramique	4	100nF	
Capacité céramique	8	10nF	
Capacité céramique	6	1uF	
Diode	22	1A/50V	1N4001
Diode Zener		20V	1N4747
Diode Zener		4.7V	1N4732A
Transistor	6		BD237
Transistor	2		2N2222
Thyristor	4		TM8050H
Optocoupleur	10		MOC3022
Résistance variable	4	20 KΩ	
Résistance	4	110Ω	
Résistance	11	220Ω	
Résistance	6	470Ω	
Résistance	20	1KΩ	
Résistance	8	2.2KΩ	
Résistance	2	4.5KΩ	
Résistance	10	10KΩ	
Fusible	2	5A	
Fusible	2	10A	
Amplificateur	4		LM324
Régulateur de Tension	2	20V	LM338T
Régulateur de Tension	2	5V	7805
Capteur température	2		LM35
Connecteur	16		Bornier 2P
Connecteur	1		Bornier 3P
Connecteur	6		Connecteur Dupont 1P

Annexe 5 : La carte chargeur 5V

- Schéma de la carte 5V

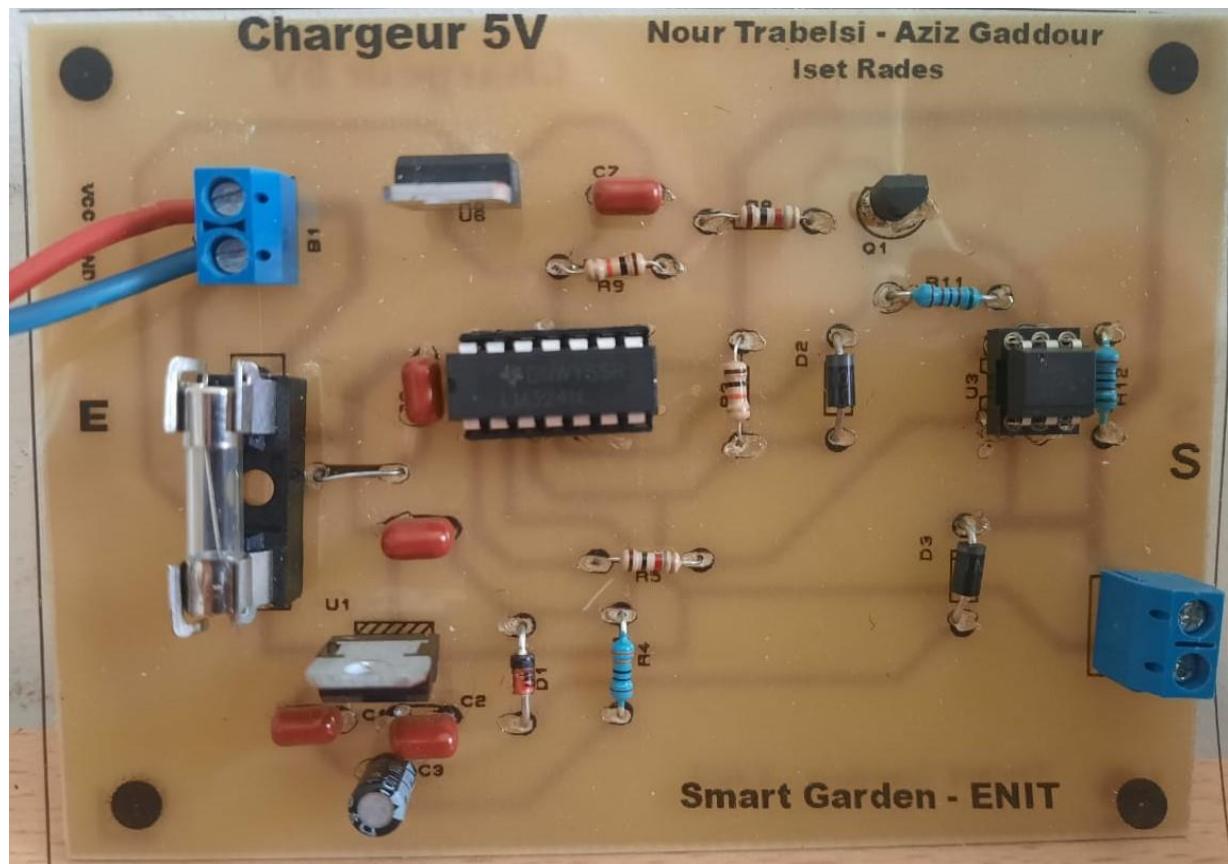


- PCB Face cuivre et face composant de la carte 5V



Chargeur 5V

Nour Trabelsi - Aziz Gaddour
Iset Rades



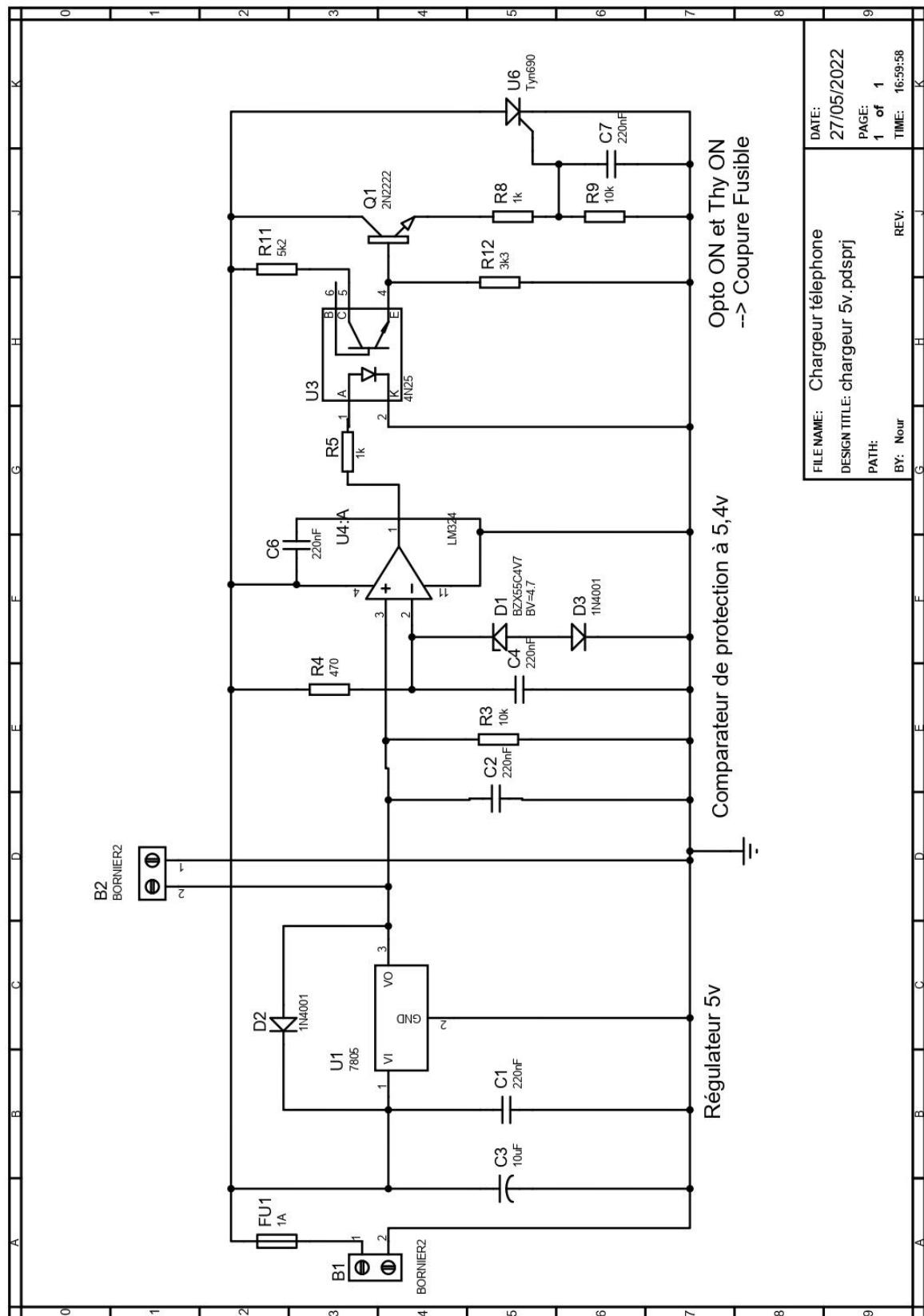
Nour Trabelsi - Aziz Gaddour
Iset Rades

Chargeur 5V

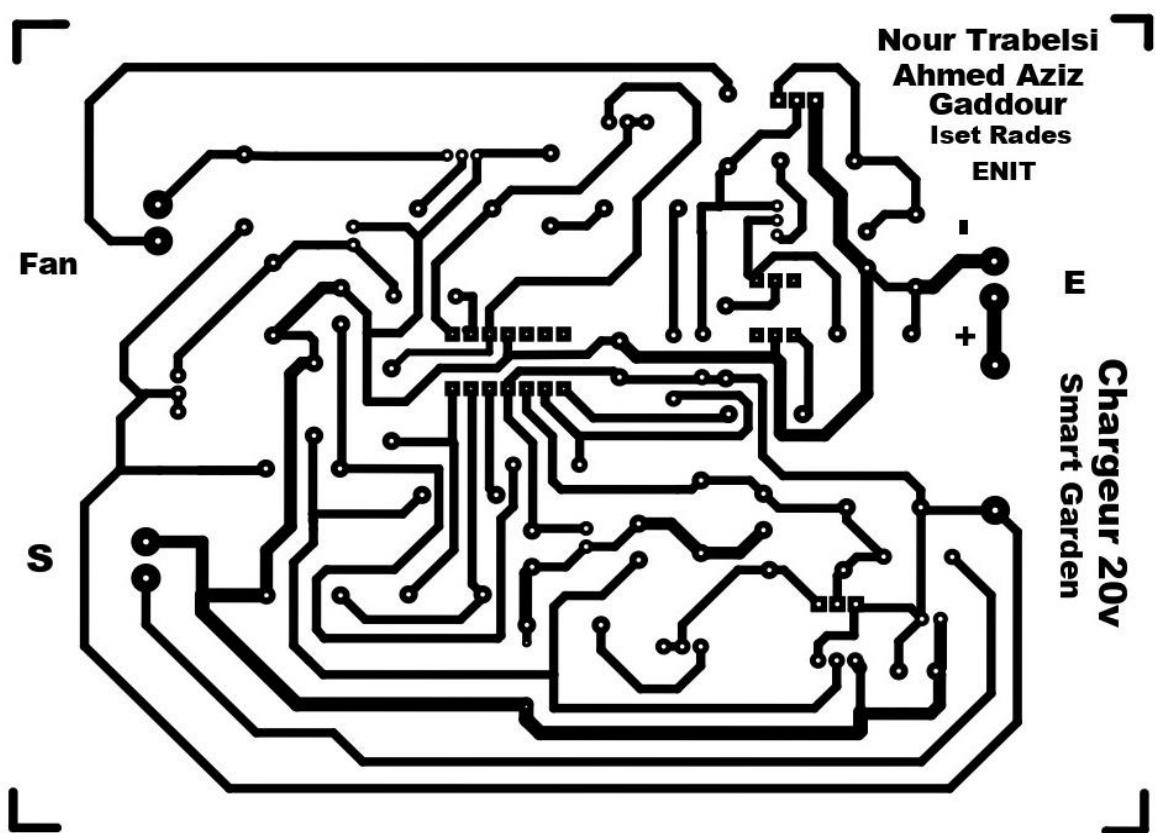
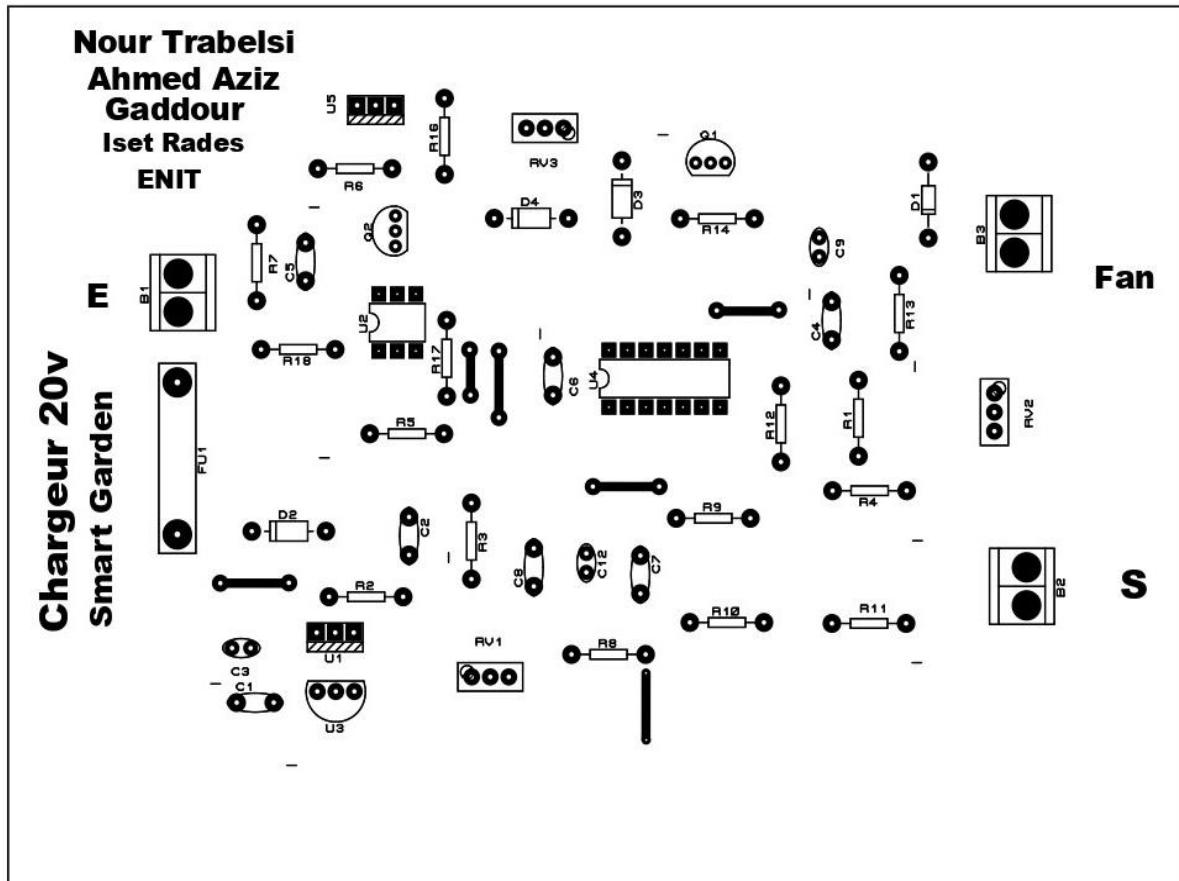
Smart Garden - ENIT

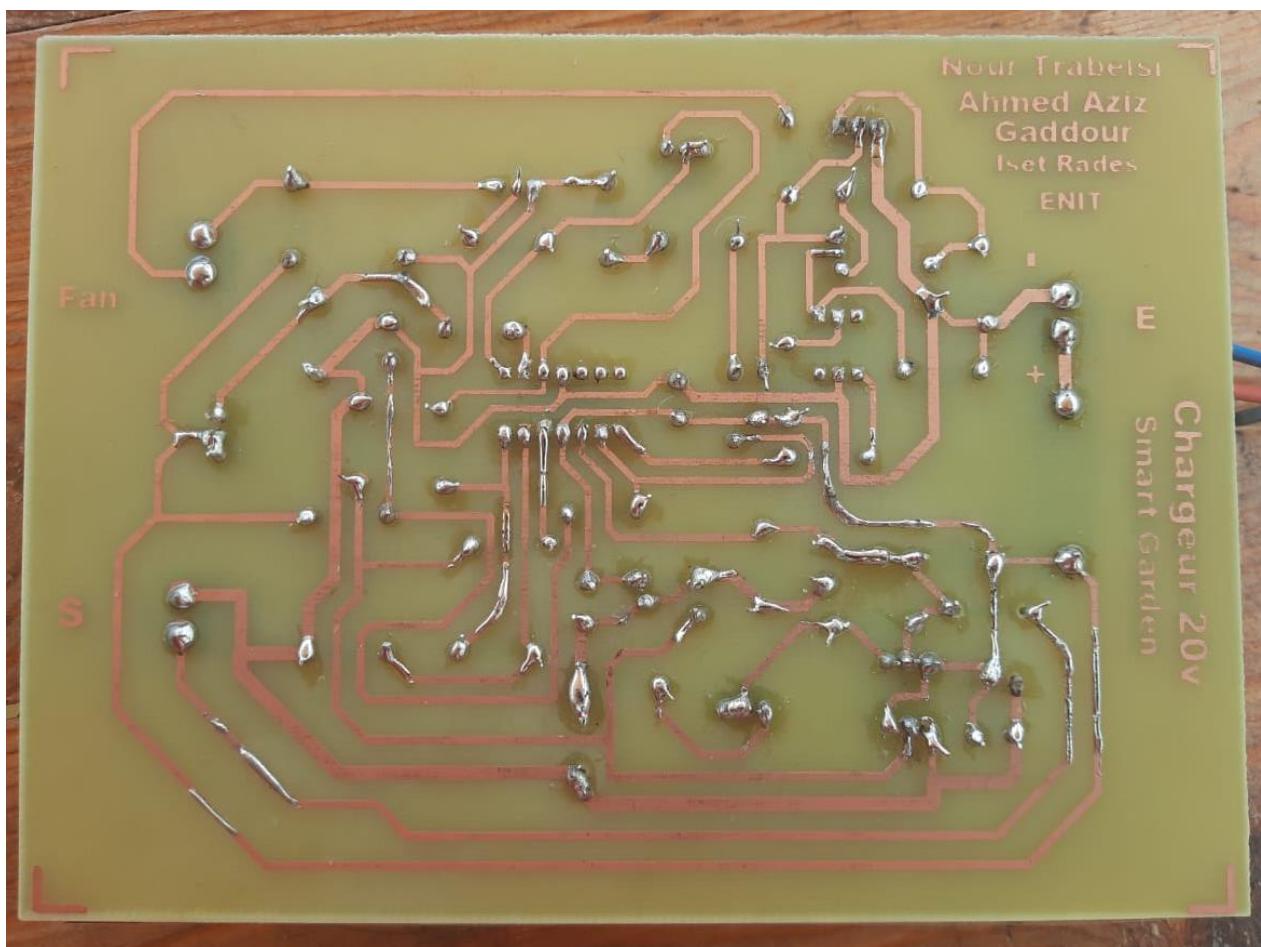
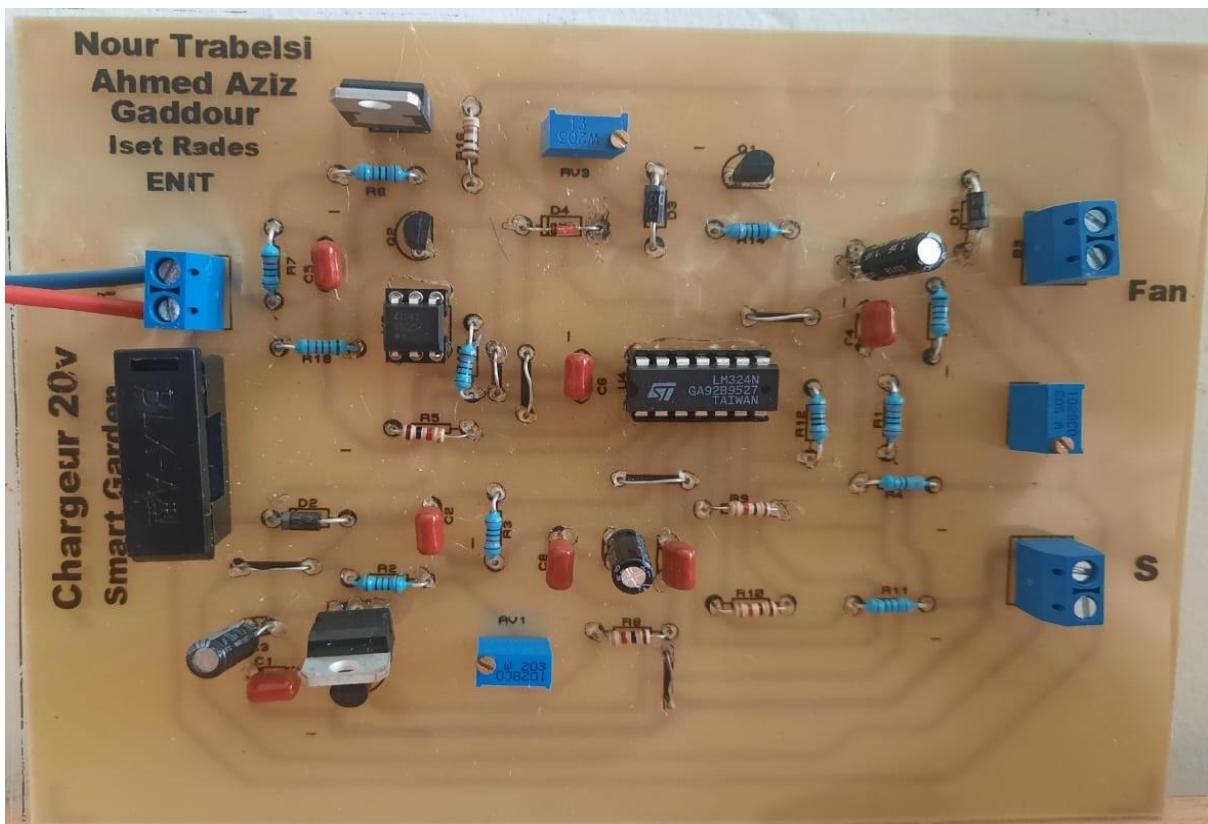
Annexe 6 : La carte de Chargeur 20v

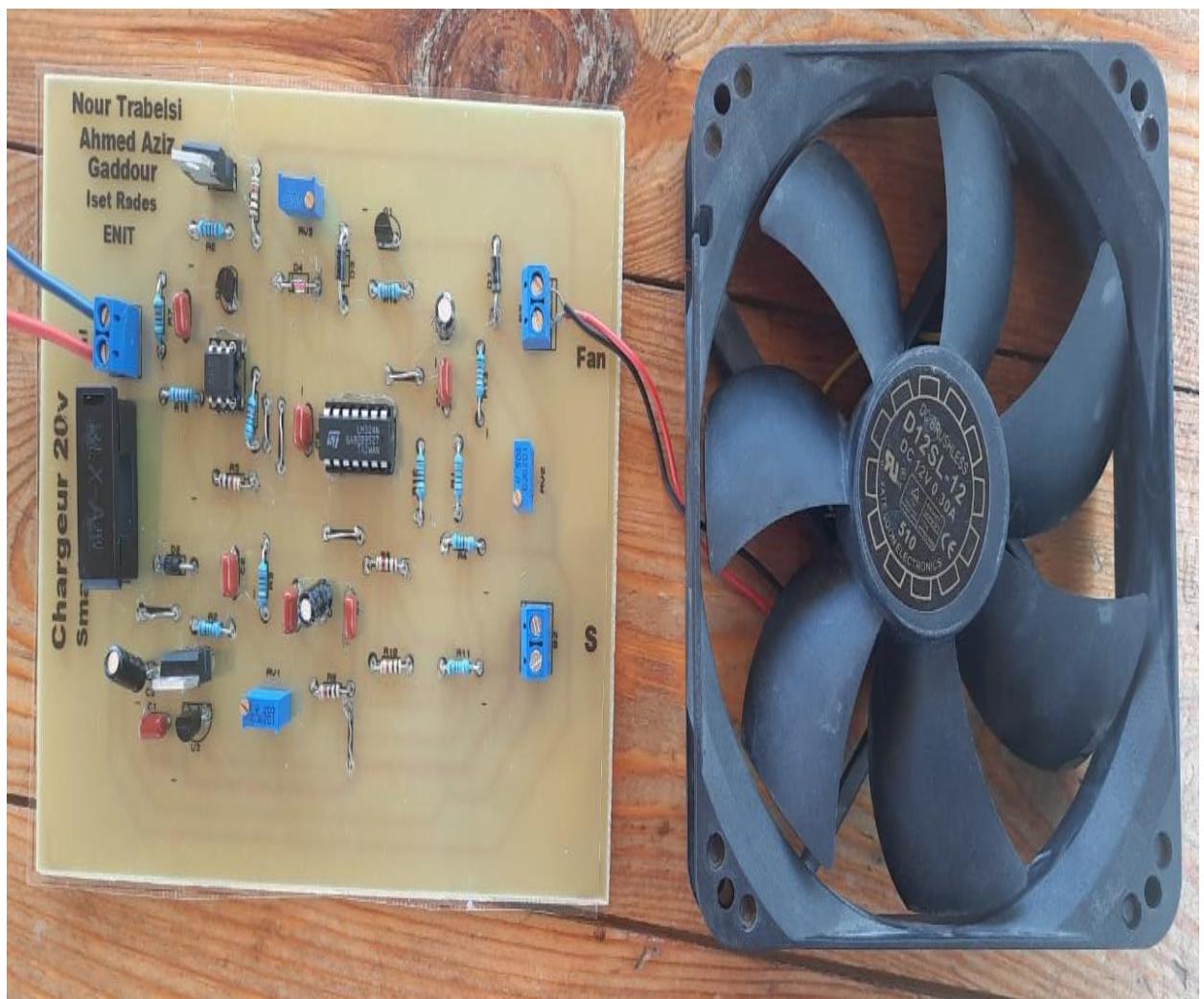
• Schéma de la carte 20V



- PCB Face cuivre et face composant de la carte 20V







Annexe 7 : Application Mobile ENIT SMART GARDEN

