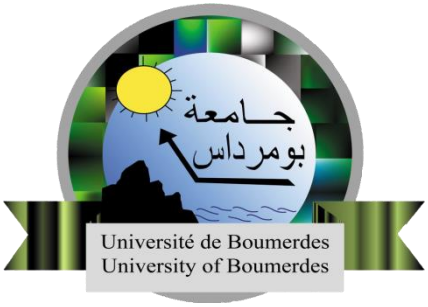
République Algérienne Démocratique et Populaire

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

UNIVERSITE M'HAMED BOUGARA-BOUMERDES



Faculté des **S**ciences de l'**I**ngénieur

Mémoire de Master

Présenté par :

Mr. LEMDANI Rafik

Mr. MALOUADJMI Nabil

En vue de l'obtention du diplôme de **Master** en

Génie Electrique

Option : Automatique

Thème :

Etude, conception et réalisation d'une plateforme pour l'automatisation et le contrôle à distance des serres agricoles.

Président	K. HARRAR	M.C.B	UMBB
Rapporteurs	H. AKROUM	M.C.B	UMBB
Examinateurs	F. NAFA	M.C.B	UMBB
	K. HAMDAOUI	M.A.A	UMBB

- Promotion Juin 2017 -

ملخص

يُعتبر هذا العمل دراسة، تصميم وتنفيذ منصة من اجل التحكم الذاتي والتحكم عن بعد في البيوت الزراعية. ويتكون هذا المشروع من ثلاثة أجزاء:

الجزء الأول هو تصميم وانشاء لوحة التحكم بالاعتماد على المتحكم المصغرESP8266، فإنه يسمح بالتحكم الآلي على المعلومات الخاصة بالمناخ من خلال خوارزمية تحكم الضبابي، وكذالك يرسل هذه البيانات من خلال شبكة الإنترنت.

في الجزء الثاني، قمنا بتطوير واجهة الإنسان والآلة التي تمكن عبر الإنترنت من: عرض المعلومات المناخ في الوقت الحقيقي على الواجهة الرئسية، تخزينها في قاعدة بيانات، امكانية اعادة ضبط المضبوطة الخاصة بالمتحكم والتدخل اليدوي على المنفذات المتحكمات بلبيت الزراعي.

وأخيرا قمنى بانشاء نموذج لبيت زراعي الذي سمح لنا باختبار الأداء وحسن سير المشرع الذي قمنا به.

الكلمات المفتاحية: البيوت الزراعية، التحكم في المعلومات المناخية، المناخ المحلي، المتحكم المصغرESP8266، المتحكم لضبابي، تطبيق ويب، المتحكم كل اولا شيء، التحكم عن بعد،WiFi .

RESUME

Ce travail consiste à faire l'étude, la conception ainsi que la réalisation d'une plateforme pour l'automatisation et le contrôle à distance des serres agricoles. Ce projet est constitué en trois parties :
La première partie consiste à concevoir une carte de commande à base du microcontrôleur ESP8266. Elle permet de faire le contrôle automatique des paramètres climatiques par un algorithme de régulation flou, et d'envoyer ces données par internet.
Dans la deuxième partie, nous avons développé une interface homme-machine qui permet via internet de : visualiser les paramètres climatiques en temps réel et de les enregistrer dans une base de données, configurer les consignes du régulateur et intervenir manuellement sur les actionneurs de la serre.
Enfin nous avons réalisé une serre prototype qui nous a permis de tester les performances et le bon fonctionnement de notre système.

Mots clés : serres agricoles - contrôle des paramètres climatiques - microclimat - microcontrôleur ESP8266 - régulateur flou - régulateur TOR - application web - contrôle à distance - WiFi.

ABSTRACT

This work consist the study, design and realization of a platform for the automation and remote control of agricultural greenhouses. This project consists of three parts:
The first part consists of designing a control board based on the ESP8266 microcontroller, it allows the automatic control of climatic parameters by a fuzzy logic algorithm, and sends this data via Internet.
In the second part, we have developed an online human-machine interface which allows: to visualize the climate parameters in real time and to record them in a database, to configure the setpoints of the regulator and to manually control the actuators of the greenhouse.
Finally, we realized a prototype greenhouse that allowed us to test the performance and the good functioning of our system.

Key words: greenhouses - control of climate parameters – microclimate – ESP8266 microcontroller - Fuzzy controller - on–off controller – Web application - remote control – WiFi.

Remerciements

Nous remercions le bon Dieu de nous avoir accordé toute la patience, le courage, la volonté et la motivation qui nous ont permis de mener à bien ce travail jusqu'à sa fin.

Nous tenons à présenter nos remerciements à notre promoteur Dr. AKROUM Hamza pour avoir accepté de diriger ce travail, pour l'aide, pour tous les conseils qu'il nous a apportés et l'encouragement constant qu'il n'a cessé de nous prodiguer tout au long de cette étude, qu'il trouve ici l'expression de toute notre gratitude.

Nous tenons à exprimer notre profonde gratitude et nos sincères remerciements à notre encadreur Mr. ADJERID Hussem-Eddine, doctorant à l'USTHB, pour sa précieuse aide, ces encouragements constants, ces bons conseils et pour nous avoir guidés durant toute la période d'exécution de ce travail malgré ses nombreuses préoccupations.

Nous remercions les membres du jury de bien vouloir évaluer notre mémoire.

Nous souhaitons remercier tous nos enseignants du département de génie électrique.

Nos vifs remercîments s'étendent à tous les membres du club scientifique ELECTRO pour leur soutien moral et matériel.

En fin nous remercions vivement tous ceux qui de près ou de loin, ont contribué d'une manière ou d'une autre à la réalisation de ce mémoire.

Dédicace

C'est avec profonde gratitude et sincères mots, que je dédie ce modeste travail de fin d'étude à mes chers parents, pour leur patience, leur encouragement et leur soutien. J'espère qu'un jour, je puisse leurs rendre un peu de ce qu'ils ont fait pour moi. Que dieu leur prête bonheur et longue vie.

Je dédie aussi ce travail à ma sœur, ma grand-mère, mon oncle et mes tantes,

A mon binôme Nabil,

A mes amis : Aïssa, Youcef, Hussem-eddine, Ihab, Oulhadj, Ibtissam, Kenza, Schehrazed, Rosa, Karim et Chamseddine,

A tous mes amis du club Electro,

Et à tous ceux qui me sont chers.

Rafik

Dédicace

Je dédie ce modeste travail de fin d'étude à mes chers parents, que tous les mots du monde ne sauraient exprimer ma profonde gratitude pour leur patience, leurs encouragements et leur soutien. Que dieu leur prête bonheur et longue vie.

A mes chères sœurs.

A mes chers frères.

Aux enfants de mes frères.

Mes oncles et mes tantes, tous mes cousins et tous mes cousins et cousines ainsi que tout ma famille.

A mon binôme Rafik.

A tous les membres du club scientifique Electro.

A mes chers amis, Aïssa, Youcef, Kousseila, Karim, Kenza, Rosa, Ibtissam, Schehrazed, Oulhadj, Ihab, Chamseddine et Hussem-eddine.

A tous mes amis.

A tous mes professeurs qui m'ont enseigné.

Et tous ceux qui me sont chers.

Nabil

Table des matières

Introduction générale……..……...1

CHAPITRE 1

La serre et son climat

I.1. Introduction 3

I.2. Les serres agricoles 3

I.2.1. Intérêt de la serre 3

I.2.2. Choix de la serre et de sa couverture 4

I.2.3. Classification des serres 4

I.3. Type de ventilation de la serre 7

I.3.1. Ventilation naturelle 7

I.3.2. Ventilation forcée 7

I.4. Les équipements d'une serre 7

I.4.1. Les actionneurs 7

I.4.1.1. Chauffage d'air 7

I.4.1.2. Extracteurs 8

I.4.1.3. Double paroi gonflable 8

I.4.1.4. Le système de réfrigération (Cooling) 9

I.4.1.5. Brasseur d'air 9

I.4.1.6. Le fog system 10

I.4.1.7. L'éclairage 10

I.4.2. Les Capteurs 12

I.4.2.1. L'anémomètre 12

I.4.2.2. Le pluviomètre 13

I.4.2.3. Girouette 14

I.4.2.4. Capteurs d'humidité volumétrique 15

I.4.2.5. Température et humidité combinées (AFTF-35) 15

I.4.2.6. Sonde de dioxyde de carbone GMP252 16

I.5. Le Micro climat de la serre 17

I.5.1. Intérêts de la détermination du microclimat....17
I.5.2. La température....18
I.5.2.1. L'effet de serre....18
I.5.2.2. Consigne pratique de température ambiante....18
I.5.3. L'humidité....19
I.5.3.1. Consigne pratique d'hygrométrie....19
I.5.3.2. La déshumidification....19
I.5.4. Le Rayonnement....20
I.5.5. Le gaz carbonique....20
I.6. Conclusion....20

CHAPITRE 2

Description du systeme à réaliser

II.1. Introduction....21
II.2. Schéma synoptique....21
II.3. Choix de la solution technologique....22
II.4. Partie commande....22
II.4.1. Etude des microcontrôleurs....22
II.4.1.1 Avantages des microcontrôleurs....22
II.4.1.2 Choix du microcontrôleur....23
II.4.1.3 Etude comparative....23
II.4.1.4 Le microcontrôleur ESP8266....24
II.4.1.5. Brochage de l'ESP8266 12E....24
II.4.1.6. Architecture interne d'un ESP8266....24
II.4.2. Les circuits relatifs au microcontrôleur....26
II.4.3. Le bus I²C....26
II.4.4. Limites de l'ESP8266 et solutions....27
II.4.5. L'affichage....28
II.4.6. Utilisation du bus I²C....29

II.4.7. Programmation du microcontrôleur 29
II.5. Capteurs utilisés 30
II.5.1. La photorésistance 30
II.5.3. Capteur de température LM35 32
II.5.4. Capteur d'humidité DHT11 33
II.5.5. Le potentiomètre 34
II.6. Partie puissance 35
II.6.1. Le moteur DC 35
II.5.2. L'extracteur 35
II.6.3. La pompe d'arrosage 36
II.6.4. Le servomoteur 36
II.6.5. Circuit L293D 37
II.6.6. L'humidificateur 38
II.6.6.1. Description du Shengmeiyu Mini-humidifier 39
II.6.7. Le relais 39
II.6.8. L'alimentation 40
II.7. Conclusion 41

CHAPITRE 3

Commande et régulation automatique

III.1. Introduction 42
III.2. La régulation automatique 42
III.2.1. Notion de système 42
III.2.2. La régulation numérique 43
III.2.3. Régulation Tout Ou Rien 43
III.2.4. Régulation PID 44
III.2.5. Régulation par logique floue 45
III.2.5.1. Variables floues 45

III.2.5.2. Règles d'inférence 46
III.2.5.3. La défuzzification 46
III.2.5.4. Les avantages et inconvénients de la logique floue 47
III.2.6. Choix du régulateur 47
III.2.6.1. Etude comparative des régulateurs PID et flou 48
III.2.7. Présentation du Toolbox fuzzy logic 48
III.2.8. Fonctionnement du régulateur TOR 53
III.3. Simulation sous Simulink 53
III.3.1. Simulation du Tout ou Rien 53
III.3.2. Simulation de la commande floue 55
III.4. Implémentation de la logique floue 58
III.5. Conclusion 58

CHAPITRE 4

Réseau sans fil et logiciels

IV.1. Introduction 59
IV.2. Les réseaux sans fils 59
IV.2.1. Avantages des réseaux sans Fil 59
IV.2.2. Etude comparative 59
IV.2.3. Réseau local sans fil Wi-Fi 60
IV.2.3.1. Les normes Wi-Fi 60
IV.2.4. Le modèle OSI 60
IV.2.5. Les protocoles de communication 61
IV.2.6. L'environnement client-serveur 62
IV.2.6.1. Le Serveur HTTP 62
IV.2.6.2. Utilisation d'un Raspberry PI comme serveur Web 62
IV.2.6.3. Client HTTP 63
IV.2.6.4. Navigateur Web 63
IV.3. Les applications WEB 63

IV.3.1 Avantages des applications WEB.....63
IV.3.2. Utilisation de l'application Web.....64
IV.4. Présentation de l'interface Homme-Machine.....65
IV.5. Conclusion.....68

CHAPITRE 5

Réalisation et tests

V.1. Introduction.....69
V.2. Réalisation de la serre.....69
V.2.1. Construction de la structure en bois.....69
V.2.2. Montage des parois en plexiglass.....69
V.2.3. Montage des capteurs et actionneurs.....70
V.2.4. Montage de la partie électrique.....71
V.3. Organigramme générale du système.....72
V.3.1. Organigramme de la carte de commande.....72
V.3.2. Organigramme de la partie serveur.....75
V.4. Conception de la carte PCB de commande.....76
V.6. Evaluation des performances du système.....78
V.6.1. Evolution des paramètres climatique sans contrôle.....78
V.6.2. Evolution des paramètres climatique avec contrôle.....80
V.7. Etude socioéconomique.....81
V.8. Conclusion.....82

Conclusion générale……...……..……...84

Annexe I : Schéma électrique de la carte de commande réalisé avec le logiciel EasyEDA.

Annexe II : Implémentation de la logique floue sur la carte de commande en langage C.

Liste des figures

Figure I.1.Une serre tunnel. 4
Figure I.2. Une serre multi-chapelle 5
Figure I.3. Une aération uniforme située sur le faîtage 5
Figure I.4. Ouverture par déplacement vertical 6
Figure I.5. Une serre jardin 6
Figure I.6. Un chauffage d'air 8
Figure I.7. Un Extracteur 8
Figure I.8. Une double paroi gonflable 9
Figure I.9. Un système de réfrigération (Cooling) 9
Figure I.10. Un brasseur d'air 10
Figure I.11. Un fog systeme 10
Figure I.12 Eclairage LED dans une serre agricole 11
Figure I.13. Une lampe de sodium haute pression (SHP) 11
Figure I.14. Une lampe fluorescente 12
Figure I.15. Un anémomètre et son schéma de fonctionnement 13
Figure I.16. Un pluviomètre à augets électronique 14
Figure I.17. Une girouette 14
Figure I.18. Un capteur d'humidité volumique d'un substrat 15
Figure I.19. Capteur de température et d'humidité combinés (AFTF-35) 16
Figure I.20. Une sonde de dioxyde de carbone GMP252 16
Figure I.21. Principe de fonctionnement d'un capteur de CO2 17

Figure II.1. Schéma synoptique du dispositif à réaliser. 21
Figure II.2. Schéma de brochage de l'ESP8266 24
Figure II.3. Schéma bloc représentant l'architecture interne de l'ESP8266 25
Figure II.4. Architecture I²C avec plusieurs maîtres et plusieurs esclaves. 27
Figure II.5. Exemple d'utilisation du circuit PCF8574 28
Figure II.6. Utilisation du bus I²C dans notre carte de commande. 29
Figure II.7. Interface de l'environnement de développement Arduino 30
Figure II.8. Une Photorésistance avec son symbole 31
Figure II.9. Capteur de CO2 SEN0159 32
Figure II.10. Précision des différentes versions de LM35 33
Figure II.11. Capteur DHT11 et son schéma de brochage 33
Figure II.12. Un potentiomètre et son symbole. 34
Figure II.13. Le montage d'un potentiomètre. 34

Figure II.14. Un moteur à courant continue. 35
Figure II.15. Type d'extracteur utilisé dans notre prototype. 36
Figure II.16. Pompe d'arrosage utilisée. 36
Figure II.17. Un servomoteur. 37
Figure II.18. Le schéma de brochage du L293D 38
Figure II.19. L'humidificateur Shengmeiyu Mini-humidifier. 39
Figure II.20. Le relai utilisé dans notre projet. 39
Figure II.21. Alimentation de PC utilisée.. 40

Figure III.1. Présentation générale d'une régulation en boucle fermée 42
Figure III.2. Présentation générale d'un système en boucle ouverte 42
Figure III.3. Schéma fonctionnel du mode de fonctionnement d'une régulation numérique. 43
Figure III.4. Organigramme d'un exemple de régulation TOR.. 44
Figure III.5. Schéma fonctionnel d'un processus réglé par un régulateur PID classique où on y trouve les perturbations V(t) et W(t). 45
Figure III.6. Exemple de partition floue d'un univers de discours. 46
Figure III.7. Structure d'une commande floue. 46
Figure III.8. Représentation de la méthode de défuzzification Centre de gravité (COG). 47
Figure III.9. Les différentes fenêtres du Toolbox fuzzy logic de Matlab 49
Figure III.10. Les entrées/sorties de notre système sur la fenêtre FIS Editor. 50
Figure III.11. La fonction d'appartenance de la température intérieure. 50
Figure III.12. Le régulateur flou avec ses 5 entrées et ses 2 sorties. 51
Figure III.13. Implémentation des règles d'inférence. 52
Figure III.14. Simulation des sorties du régulateur par le Rule viewer. 52
Figure III.15. Le régulateur TOR avec ses 4 entrées et ses 4 sorties. 53
Figure III.16. Modèle Simulink utilisé pour la simulation de la commande TOR. 54
Figure III.17. Résultat de la simulation TOR. 54
Figure III.18. Evolution de la commande TOR. 55
Figure III.19. Résultat de la simulation floue. 56
Figure III.20. Model Simulink avec régulateur flou. 57
Figure III.21. Résultat de la simulation TOR et floue. 57
Figure III.22. Evolution de la commande floue du système de climatisation. 58

Figure IV.1. Diagramme du modèle OSI 61
Figure IV.2. Utilisation d'une application web sur plusieurs types de terminaux sur le réseau. 63

Figure IV.3. Page principale de notre interface Homme-Machine. 65
Figure IV.4. Graphe qui montre l'évolution des paramètres de température intérieure et extérieure sans régulation. 66
Figure IV.5. Partie Commande de la serre qui nous montre l'état de chaque actionneur. 67
Figure IV.6. Page de Configuration de la serre. 67

Figure V.1. La structure de notre serre prototype. 69
Figure V.2. Montage des parois en plexiglass. 70
Figure V.3. Montage des capteurs et actionneurs. 70
Figure V.4. Description de la partie électrique de la serre. 71
Figure V.5. Plantation des tomates 71
Figure V.6. Diagramme global du programme de la carte de commande. 74
Figure V.7. Diagramme du programme de la partie serveur. 75
Figure V.8. Carte de commande face A. 76
Figure V.9. Carte de commande face B. 76
Figure V.10. Description des différentes parties de la serre prototype. 77
Figure V.11. La serre agricole finale exposée au salon national des produits de la recherche 78
Figure V.12. Enregistrement de l'évolution de l'humidité relative par rapport au taux d'ensoleillement sans contrôle. 79
Figure V.13. Enregistrement de l'évolution des températures intérieur et extérieur par rapport au taux d'ensoleillement sans contrôle. 79
Figure V.14. Enregistrement de l'évolution de l'humidité relative par rapport au taux d'ensoleillement avec contrôle. 80
Figure V.15. Enregistrement de l'évolution des températures intérieur et extérieur par rapport au taux d'ensoleillement avec contrôle. 81

Liste des tableaux

Tableau II.1. Etude comparative de quelques microcontrôleurs disponibles sur le marché 23
Tableau II.2. Caractéristiques du capteur LM35.....32
Tableau II.3. Caractéristique du L293D.....38
Tableau II.4. Brochage du L293D.....38
Tableau III.1. Comparatif des principales familles de régulateurs 48
Tableau IV.1. Comparatif des trois technologies de communication: Wifi, Bluetooth et Xbee ..59
Tableau IV.2. Les principales révisions de la norme 802.11 et leurs signification.....60
Tableau V.1. Estimation du prix de revient de notre projet. 81

Abréviations

µc : Microcontrôleur

ADC : Analog to Digital Converter.

API : Application Programming Interface.

API : Automate programmable industriel.

A/D : Convertisseur analogique-numérique.

E/S : Entrée/sortie.

EPROM : Erasable Programmable Read-Only Memory

GPIO : General Purpose Input/Output.

HTTP : HyperText Transfer Protocol.

I²C : Inter-Integrated Circuit.

I²S : Integrated Interchip Sound.

ISO : International Organization for Standardization.

IDE : Integrated Development Environment.

IEEE : Institute of Electrical and Electronics Engineers.

LCD : Liquid Crystal Display.

LED : light emiter diode.

MLI : Modulation par largeur d'impulsion.

CTN : Coefficient de Température Négatif.

OSI : Open Systems Interconnection.

PCB : Printed Circuit Board.

PIC : Peripheral Interface Controller.

PMU : Power Management Unit.

PLL : Phase-locked loop.

PWM : Pulse width modulation.

RAM : random access memory.

ROM : read only memory.

RF : Radio frequency.

RSMA : Reverse SubMiniature version A.

RISC : Reduced instruction set computing.

SDK : Software Development Kit.

SDIO : Secure Digital Input Output.

SHP : Sodium haute pression.

SPI : Serial Peripheral Interface.

TCP/IP : Transmission Control Protocol/Internet Protocol.

QSPI : queued serial peripheral interface

UART : Universal Asyncronous Receiver Transmitter.

WEP : Wired Equivalent Privacy.

WPA : Wi-Fi Protected Access.

WiFi : Wireless Fidelity.

Introduction générale

La culture sous serre connait depuis plusieurs années, un développement important dans tous les pays concernés par la volonté d'augmenter la qualité et la quantité des produits agricoles, destinés à la consommation alimentaire des populations et à leurs cadres de vie [1].

Afin de faire face à un marché de plus en plus concurrentiel, les systèmes de production sous serre deviennent considérablement sophistiqués. C'est grâce à la mécanisation des outils, et l'augmentation des surfaces irriguées, mais surtout grâce à l'intervention de l'automatique que cet objectif a pu être réalisé.

Il est important de choisir un contrôle des paramètres climatiques qui correspond aux besoins visés. Pour améliorer la rentabilité, on doit faire croître les cultures dans des environnements optimaux [3]. Il est donc important de bien contrôler les paramètres suivants [2] :

- La température et l'humidité de l'air : ce sont les variables les plus importantes à contrôler du point de vue de la survie et de la croissance des plantes.
- L'humidité du sol est moins cruciale pour des durées de quelques heures, mais c'est une variable importante pour que la plante puisse se nourrir.
- Le taux de CO2 doit être régulé dans des serres très hermétiques qui consomment rapidement le taux de gaz carbonique lors de la photosynthèse.
- L'éclairage peut également être régulé au moyen d'un éclairage artificiel pour accélérer le développement des plantes.

L'automaticien est très sollicité pour réaliser cette gestion bioclimatique, il est confronté aux difficultés les plus grandes qu'il puisse rencontrer dans le cadre de sa science [4]. En effet, il lui est demandé de mettre en place des régulateurs pour une classe de systèmes parmi les plus difficiles à contrôler, car ils sont [5]:

- Multi entrées: ouvrants, humidificateurs, chauffages, arroseurs, ombrage, injecteurs de CO2.
- Multi sorties: température interne, degrés d'humidité interne, taux de CO2 interne
- Non linéaires: systèmes biologiques, changements de phase (phénomène de condensation),
- Non stationnaires: croissance des plantes, vieillissement des composants de la structure de la serre, etc.,
- Ouverts: des perturbations (condition des météorologiques externes: rayonnement global solaire, température externe, degrés d'humidité externe, vitesse et direction de vent) qui mettent en jeu des puissances souvent beaucoup plus élevées que celle des actionneurs dont on dispose.

L'automaticien sera aussi amené à traiter des problèmes de commande difficiles compte tenu des performances souhaitées. Il devra essayer plusieurs types de commande et après comparaison, choisir la commande la plus adaptée [4]. Ainsi, après avoir envisagé des commandes classiques (commande optimale, etc...), il s'orientera vers des commandes dites «avancées», qui s'appuient sur de nouveaux concepts tel que: la logique floue, les réseaux de neurones et les algorithmes génétiques.

Dans le cadre de ce travail, nous souhaitons développer une plateforme composée d'une application web et d'une carte de commande à base de microcontrôleur pour la mesure et le contrôle des paramètres bioclimatiques par un algorithme de régulation floue sur une serre prototype que nous allons réaliser.

Via internet, cette plateforme permettra de :

- Controller l'état des paramètres climatiques en temps réel.
- Configurer les consignes du régulateur selon le type de plantes cultivées.

- Possibilité de l'intervention manuelle sur les actionneurs et choisir le type de commande automatique.
- Enregistrement des paramètres climatiques sur une base de données et la visualisation de ces données sur des graphes.

Ce mémoire est organisé en cinq chapitres :

- Le premier chapitre portera sur une étude générale de la serre, des facteurs climatiques essentiels pour le développement de la plante et les composants importants pour la gestion de ces paramètres climatiques.
- Le deuxième chapitre portera sur le choix du microcontrôleur le plus adapté et le choix des capteurs et actionneurs que nous allons utiliser pour la réalisation de notre projet.
- Le troisième chapitre sera consacré à l'étude des commandes automatiques utilisées dans ce domaine, une comparaison théorique entre les différentes méthodes de commande et la simulation du comportement des régulateurs flou et TOR sur notre système.
- Le quatrième chapitre présentera en premier lieu les différentes technologies de réseau sans fil et en second lieu les différents outils informatiques qui nous permettront de développer l'interface homme-machine.
- Le cinquième chapitre consistera à présenter les étapes de la construction du prototype de serre, le montage des différents capteurs et actionneurs, la réalisation de la carte de commande PCB et l'interprétation des résultats des tests effectués.

Enfin, nous terminerons ce mémoire par une conclusion générale et les perspectives.

CHAPITRE 1

La serre et son climat

I.1. Introduction

Dans ce chapitre, nous présenterons une vue globale sur les différents types de serres agricoles, ainsi que les équipements et outils qui nous permettent de contrôler et de gérer les paramètres climatiques, en les maintenant aux voisinages des consignes tout en s`affranchissant des paramètres climatiques externes.

Ces consignes sont définies pour atteindre les deux objectifs suivants : maximiser la production sous serre et protéger la culture sous abri contre les maladies fortuites lors de sa croissance [1] [2].

La serre est conçue comme une enceinte limitée par une paroi transparente, protégée de certains effets indésirables du climat extérieur, créant alors des conditions peu favorables aux cultures. Le recours aux techniques de régulation automatique a permis de réaliser les objectifs techniques et économiques par le producteur tels que :

- Obtenir les meilleurs rendements.
- Optimiser la qualité de production.
- Rationaliser la consommation d'énergie.
- Gérer le planning de production.

On se tourne alors vers la micro-informatique, pour maîtriser aisément la conduite climatique sous serre.

I.2. Les serres agricoles

❖ Définition

Une serre est une structure qui peut être parfaitement close, Elle est destinée en général à la production agricole. L'objectif étant de créer un environnement propice à leur développement en tirant parti de l'influence du climat , en créant un micro climat, pour une meilleure gestion des besoins des plantes , pour en accélérer la croissance ou pour les produire indépendamment des saisons [6] [7].

I.2.1. Intérêt de la serre

La serre de culture offre la possibilité de s'affranchir des contraintes climatiques extérieures (pluie, vent, froid). Elle est conçue pour recréer un environnement donné (microclimat). Elle permet le chauffage de l'air et des racines, le contrôle de l'irrigation et de la fertilisation, l'enrichissement en CO_2 et le contrôle de l'humidité. Elle joue un rôle économique en présentant des produits sur le marché en hors saison.

La serre permet ainsi d'obtenir une production végétale dans des conditions meilleures que celles existantes naturellement et ce par une meilleure qualité du produit. Pour aboutir à ces résultats, il faut répondre minutieusement aux exigences de la culture pour les différents facteurs intervenant dans sa croissance et son, développement, ceci suppose la connaissance des interactions entre ces divers paramètres [8].

I.2.2. Choix de la serre et de sa couverture

Les principaux critères de choix d'une serre et sa couverture sont [9]:

- La transmission au rayonnement utile à la photosynthèse (pour améliorer le potentiel de production).
- La solidité et la durabilité.
- Les économies d'énergie et le cout.

I.2.3. Classification des serres

La classification des serres est complexe et difficile ; Elle est souvent faite selon les formes données par les cadres porteurs qui constituent l'assemblage, on distingue deux principaux types appartenant à deux grandes familles de serres : Les serres tunnels et les serres chapelles [9].

a) Serres tunnel

Une serre tunnel (Figure I.1) est formée de plusieurs grands arceaux métalliques, recouverts d'un film souple en plastique transparent, qui lui donne la forme d'un tunnel ! Les dimensions de la serre dépendent des besoins du jardinier [10].

Figure I.1.Une serre tunnel.

❖ Caractéristique des serres tunnel

- La hauteur varie généralement entre 2 et 2,50 mètres.
- La largeur se situe entre 3 et 5 mètres.
- Une surface au sol minimale de 6 m2.
- La deuxième porte est nécessaire dès que la serre tunnel dépasse les 6 mètres de long.

b) Serres multi-chapelle

Les différentes pièces qui composent la structure de la gamme à serres Multi chapelle (Figure I.2) sont conçues pour être parfaitement assemblées grâce à des joints vissés qui simplifient le montage tout en absorbant idéalement les différentes forces qui ont un impact sur la structure. Elles sont très demandées pour leur robustesse et leur grande capacité d'adaptation aux dimensions et aux caractéristiques du terrain [11].

Figure I.2. Une serre multi-chapelle [11].

❖ Caractéristique des serres multi-chapelle

1. Matériaux de revêtement :

- Film plastique.
- Matériaux semi-rigides
- Plaque rigide

2. Les éléments en aluminium et les profils d'ancrage.

3. Un volume intérieur important.

4. L'inclinaison des arceaux permettent de tirer le meilleur parti des rayons solaires.

- Aérations au faîtage (Figure I.3).

Figure I.3. Une aération uniforme située sur le faîtage [11].

- Aérations latérales (Figure I.4).

Figure I.4. Ouverture par déplacement vertical [11].

c) Serre de jardin

C’est un petit modèle, parfait pour faire pousser des plantes, idéale pour protéger ses plantes en hiver, récolter des fruits et légumes toute l'année ou réaliser des semis sous abri, Une serre peut être construite avec divers matériaux et selon différentes formes, même avec des matériaux de récupération [12] (figure I.5).

Figure I.5. Une serre jardin [12].

❖ Caractéristique d’une serre de jardin

1. Les matériaux fréquemment utilisés pour l’armature de la serre :

- Les structures en aluminium.
- Les structures en acier.
- Les structures en bois.

2. Les matériaux fréquemment utilisés pour les panneaux de la serre :

- Le verre horticole.
- Le verre trempé.
- Le polycarbonate alvéolaire.

- Le polycarbonate transparent.

I.3. Type de ventilation de la serre

La température et le taux d'humidité à l'intérieur d'une serre dépendent de son aération. Il s'agit d'un processus complexe qui participe à l'essentiel des échanges de chaleur avec l'extérieur, sa maîtrise permet donc de contrôler certains paramètres essentiels au bon fonctionnement de la serre tels que la température, l'humidité, ou les concentrations de gaz comme le CO_2 [13].

I.3.1. Ventilation naturelle

La ventilation naturelle est le système le plus économique pour réguler le microclimat interne de la serre. Il s'agit des ouvrants disposés sur la structure. Les ouvertures doivent représenter environs 20% de la surface au sol et avoir un angle d'ouverture assez important pour permettre à l'air de bien circuler. Par le toit, une installation de l'ouvrant côté Nord est préconisée, les ouvertures au plafond permettent à l'air de se mélanger correctement dans la serre.

Ne négligez pas les ouvertures sur les côtés au niveau des plantes qui augmenteront l'efficacité de l'ouvrant du toit et qui moduleront la ventilation. La circulation de l'air sera alors optimale, créant un circuit interne autorégulé.

Au printemps et en été on peut bien évidemment, ajouter l'ouverture totale des portes pour faire encore baisser la température interne [13].

I.3.2. Ventilation forcée

La ventilation forcée nécessitera une installation électrique spécifique dans la serre : boîtiers électriques avec protection des circuits, démarreurs, lignes, canaux électriques … Par la suite vous pourrez installer votre ventilateur : Il s'agit habituellement de ventilateurs de type hélicoïdal avec des persiennes, fonctionnant en mode extracteur d'air.

Par son action ces types de ventilateurs fournissent le renouvellement du volume d'air logé dans la serre et permettent l'extraction du CO2 et le rééquilibrage du climat général.

Ces ventilateurs doivent de préférence être installés à l'opposé des vents dominants.

Ce type d'installation est adéquat pour suppléer le fonctionnement de l'aération naturelle dans les serres, garantissant l'aération à des moments déterminés pendant lesquels les conditions environnementales ne conviennent pas aux cultures (fort enneigement empêchant l'ouverture, vent violent) [13].

I.4. Les équipements d'une serre

I.4.1. Les actionneurs

I.4.1.1. Générateur d'air

Les générateurs d'air chaud (Figure I.6) sont spécialement indiqués lorsqu'il n'existe pas de demande de chauffage importante de façon continue et comme défense ponctuelle face aux gelées. Avec ce système, il est possible d'obtenir une précocité et productivité majeures dans les saisons froides, avec un degré de technicité moyen [14].

Figure I.6. Un chauffage d'air [14].

I.4.1.2. Extracteurs

Les ventilateurs extracteurs (Figure I.7) permettent de forcer l'aération dans la serre lorsque la ventilation naturelle à travers les ouvrants du toit et/ou du périmètre ne permet pas d'atteindre le taux de renouvellement d'air désiré.

Il s'agit d'un besoin essentiel dans la production de cultures et dans les fermes d'élevage.

Ils sont indispensables pour un système de refroidissement avec des panneaux évaporateurs et souvent aussi avec des systèmes de brumisation d'eau afin d'obtenir un certain niveau de réfrigération adéquat [14].

Figure I.7. Un Extracteur [14].

I.4.1.3. Double paroi gonflable

Le système de double paroi gonflable (Figure I.8) se base dans la création d'une chambre d'air entre deux couches de film plastique. La chambre d'air est gonflée avec de petits ventilateurs qui insufflent l'air dans la chambre à travers des tuyaux en PVC.

Cela permet de créer une chambre qui réduit le coefficient de transmission de chaleur vers l'extérieur pour obtenir une économie considérable d'énergie et un contrôle de la température [14].

Figure I.8. Une double paroi gonflable [14].

I.4.1.4. Le système de réfrigération (Cooling)

Ce système de réfrigération par évaporation d'eau (Figure I.9) est composé d'extracteurs et de panneaux de refroidissement placés sur les parois opposées de la serre pour créer une zone de pression négative dans la serre. Cela permet que l'air extérieur qui traverse les panneaux humides en se chargeant avec des molécules d'eau, afin de se refroidir et de réduire ainsi la température à l'intérieur de la serre [14].

Figure I.9. Un système de réfrigération (Cooling) [14].

I.4.1.5. Brasseur d'air

Les brasseurs d'air ou ventilateurs (Figure I.10) permettent d'obtenir un mouvement de l'air pour aider à maintenir un climat intérieur homogène, de façon à éviter l'accumulation d'air chaud dans la partie supérieure de la serre, ce qui réduit considérablement l'effet de condensation de l'eau et favorise la transpiration et l'absorption de CO_2 par les cultures.

Ils peuvent aussi être employés comme soutien des extracteurs ou bien comme systèmes d'humidification ou d'application de traitements [14].

Figure I.10. Un brasseur d'air [14].

I.4.1.6. Le fog system

Il s'agit d'incorporer un grand nombre de micro particules d'eau qui restent suspendues dans l'air de la serre pendant assez de temps pour s'évaporer sans mouiller les cultures. L'eau est incorporée sous forme de brouillard par des buses spéciales distribuées uniformément tout le long de la surface de la serre.

Le Fog System (Figure I.11) est très pratique pour humidifier et refroidir de façon contrôlée la serre en ayant aussi la possibilité d'effectuer des traitements de désinfection avec des produits phytosanitaires solubles [14].

Figure I.11. Un fog systeme [14].

I.4.1.7. Système d'éclairage

La lumière du soleil constitue la source de lumière la moins chère pour la croissance des plantes mais elle n'est pas toujours disponible. Grâce à une lumière d'intérieur, il est possible de mieux maîtriser la croissance des plantes et d'assurer une production effective tout au long de l'année.

a) L'éclairage LED

La technologie LED (Figure I.12) a de plus en plus de succès comme source d'éclairage innovante dans le secteur mondial de l'horticulture sous serre. L'avenir des lampes LED est prometteur. L'éclairage par LED permet déjà actuellement une efficacité supérieure de 30 à 46%. Ceci s'explique par le fait que les plantes utilisent l'énergie émise par la technologie LED de manière plus efficace que l'énergie émise par les lampes SON-T classiques. Les chiffres sont encore plus impressionnants en matière d'économie d'énergie : la différence avec les lampes classiques peut dans certains cas s'élever à 80% [15].

Figure I.12 Eclairage LED dans une serre agricole [15].

❖ Avantages de l'éclairage LED

- Les lampes LED ont par ailleurs une durée de vie prolongée par rapport aux autres lampes.
- L'installation d'éclairage de croissance avec des LED peut très bien être montée entre les plantations.
- Les lampes LED sont très efficaces et économes en énergie.

b) Éclairage par lampe de sodium haute pression (SHP)

Les lampes à sodium haute pression (SHP) (Figure I.13) produisent une lumière plus rouge, qui peut être comparée à la lumière naturelle d'un coucher de soleil en automne. Une couleur plus jaune/rouge du spectre et moins bleue favorise une fleur haute par rapport aux feuilles.

Les lampes à sodium haute pression sont utilisées pendant les heures de faible luminosité ou de totale obscurité [16].

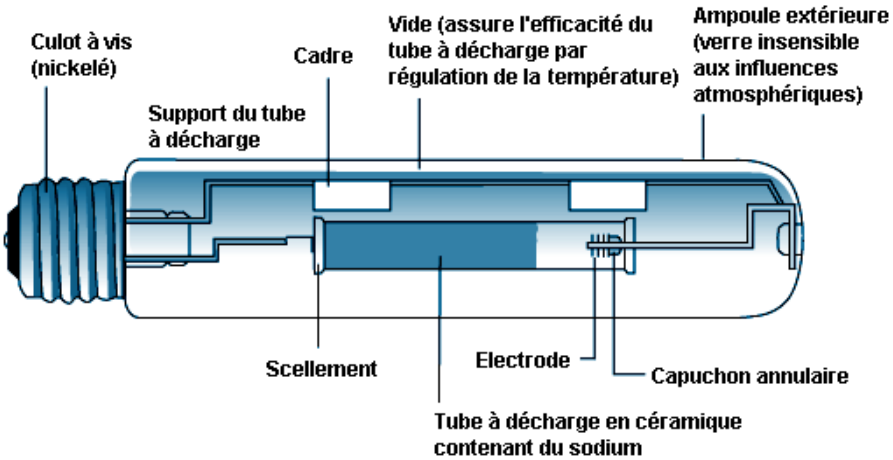


Figure I.13. Une lampe de sodium haute pression (SHP) [16].

c) Eclairage par lampes fluorescentes

Les lampes fluorescentes (Figure I.14) présentent une intensité plus faible que les lampes au sodium. Comme elles ne dégagent pas beaucoup de chaleur, elles n'assèchent pas le support de croissance lorsqu'elles sont placées à proximité de la plante. Elles sont donc idéales pour la propagation et la croissance végétative précoce ; Ainsi ne dégageant pas beaucoup de chaleur, elles peuvent être placées à environ 2,5 cm des plantes, sans nécessiter la présence d'un système d'aération pour éliminer la chaleur excessive [15].

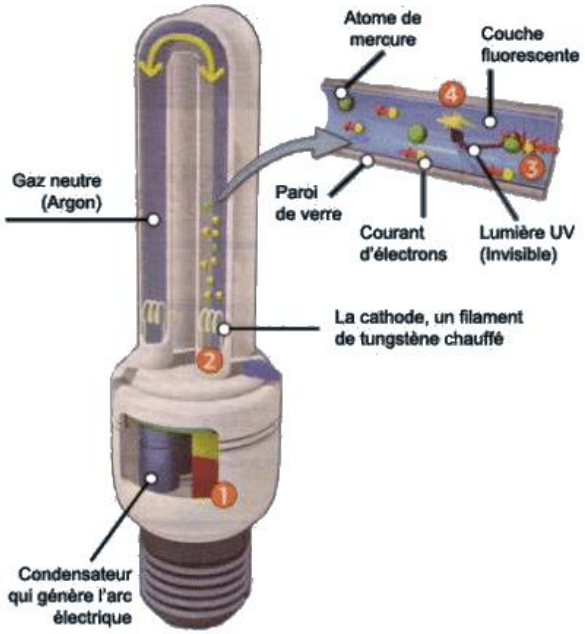


Figure I.14. Une lampe fluorescente [15].

I.4.2. Les Capteurs

Un capteur transforme une grandeur physique en une grandeur normée, généralement électrique, qui peut être interprétée par un dispositif de contrôle commande.

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

Voici quelques capteurs industriels utilisés dans le domaine des serres agricoles :

I.4.2.1. L'anémomètre

Ce capteur (Figure I.15) mesure la vitesse de l'air et le débit volumétrique, Il permet de mesurer la force éolienne et la vitesse de circulation de l'air. Il est muni d'un capteur mécanique de type éolien qui tourne en fonction de la puissance du vent. La vitesse de rotation de l'hélice est proportionnelle à la vitesse du vent [17].

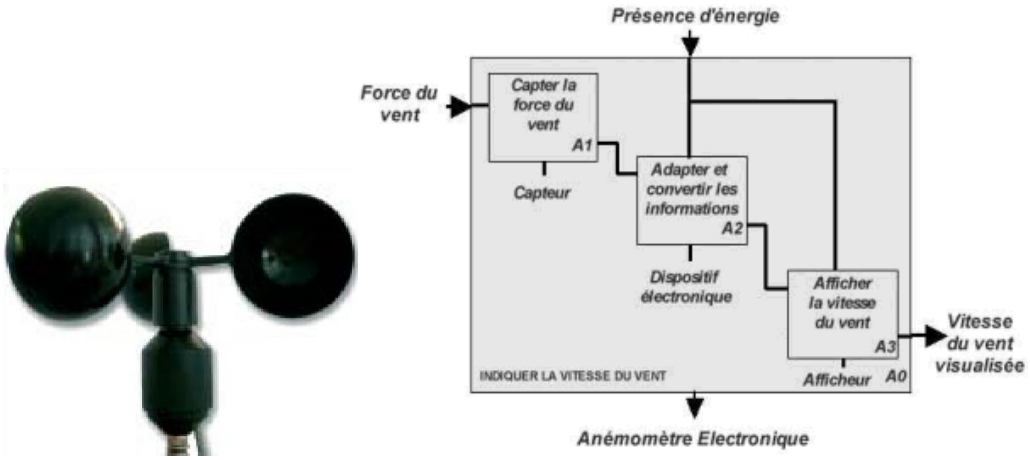


Figure I.15. Un anémomètre et son schéma de fonctionnement [17].

❖ Principe de fonctionnement

L'anémomètre est un appareil doté d'un chapeau à 3 coupelles ayant une prise au vent. Il tourne grâce à une liaison pivot sur un axe. Ce système se compose de deux composants :

- D'un moulinet se caractérisant d'un support à 4 aimants, fixé au bâti du chapeau de l'anémomètre en suivant ainsi la rotation des coupelles et donc du vent.

- Un interrupteur à lame souple (I.L.S) qui se compose de deux lames d'aciers se situant dans les champs magnétiques d'un tube de verre. Cet interrupteur est intérieurement fixe par rapport au bâti du chapeau de l'anémomètre. Cet I.L.S est directement relié au boîtier de commande de l'anémomètre.

Le moulinet agit donc sur l'interrupteur grâce à un phénomène physique de polarisation et de dépolarisation (impulsions électriques), provoqué par les aimants, et du au champ magnétique présent dans le tube de l'I.L.S.

Le signal électrique T.O.R (Tout ou Rien) de l'anémomètre est caractérisé par la superposition des lames d'acier dc l'I.L.S. Lorsque les lames se polarisent, elles se touchent et envoient ce signal [17].

I.4.2.2. Le pluviomètre

Le pluviomètre (Figure I.16) est un instrument météorologique servant à mesurer la quantité de précipitation tombée à un endroit donné pendant une période précise. La hauteur du niveau d'eau est habituellement indiquée en millimètres. Elle donne la quantité de précipitations pour un mètre carré [18].

Figure I.16. Un pluviomètre à augets électronique [18].

❖ Principe de fonctionnement

L'appareil de mesure avec les augets recueille l'eau issue de l'entonnoir. Sur le centre des augets un petit aimant est fixé, il actionne un ILS (interrupteur lame souple) qui se trouve derrière. Un condensateur est branché en parallèle pour temporiser la durée du contact. L'auget se vide dans le fond et l'eau est évacuée à l'extérieur. Un niveau à bulle permet de régler une parfaite horizontalité. A chaque basculement de l'auget une impulsion est générée, elle correspond à 0,2mm. Il est possible de procéder à un étalonnage à l'aide de vis de réglage, elles permettent de remplir plus ou moins les augets [18].

I.4.2.3. Girouette

Une girouette (Figure I.17) est un dispositif généralement métallique, la plupart du temps installé sur un toit, constitué d'un élément rotatif monté sur un axe vertical fixe. Sa fonction est de montrer la provenance du vent ainsi que son origine cardinale [19].

Ceci est permis par sa structure asymétrique, généralement matérialisée par une flèche ou un coq.

Figure I.17. Une girouette [19].

❖ Principe de fonctionnement

Lorsque le vent change de direction, il pousse sur la grosse partie de la flèche (l'arrière) jusqu'à ce qu'elle soit alignée avec le vent (parallèle au vent). Cela a pour conséquence de faire pointer la flèche dans la direction d'où provient le vent.

On se réfère alors aux quatre principaux points cardinaux pour juger de la direction d'où vient le vent. On place habituellement la girouette à 10 m du sol [19].

I.4.2.4. Capteurs d'humidité volumétrique

Les capteurs d'humidité volumétrique (Figure I.18) sont insérés dans le substrat et mesurent l'humidité volumétrique du substrat en contact avec les capteurs. La zone de détection des capteurs varie entre 1 et 4 pouces de longueur (selon le modèle et le type de capteur) [20].

Figure I.18. Un capteur d'humidité volumique d'un substrat [20].

❖ Principe de fonctionnement

Il existe deux principaux types de capteurs qui mesurent l'humidité volumétrique dans les substrats : les capteurs TDR (réflectométrie de domaine temporel) et les capteurs de capacitance.

Les deux types de capteurs mesurent la constante diélectrique du substrat. La constante diélectrique est de 1 pour l'air, de 3-5 pour les sols minéraux et de 80 pour l'eau. Ainsi, tout changement de l'un ou l'autre de ces trois paramètres changera la constante diélectrique du substrat [20].

I.4.2.5. Température et humidité combinées (AFTF-35)

Les sondes d'humidité de la série AFTF-35 (système de mesure d'humidité chauffée) (Figure I.19) servent à la mesure de l'humidité relative et de la température dans l'air ambiant. Ce Capteur / Sonde combine humidité et température pour montage mural ou en saillie, avec chauffage intégré (résistante à la condensation) pour haute humidité. Avec deux sorties analogiques : 0-10V, une précision jusqu'à 2% H.R, sur une plage de mesure de 0...100% H.R. et une plage de mesure pour la température de 0...+50°C [21].

Figure I.19. Capteur de température et d'humidité combinés (AFTF-35) [21].

❖ Principe de fonctionnement

Le principe de fonctionnement est très simple, le capteur est chauffé par un " chauffage " à une température se situant à environ 2K au-dessus de la température ambiante, Le chauffage intégré en fait des sondes résistantes à la condensation ! Cette résistance à la condensation en fait un outil indispensable dans les milieux humides tel que les piscines etc...Elles convertissent ces valeurs mesurées dans un signal normalisé de 0 à 10 V [21].

I.4.2.6. Sonde de dioxyde de carbone GMP252

La sonde de dioxyde de carbone GMP252 (Figure I.20) est une sonde intelligente et autonome pour la mesure du CO2 dans l'agriculture, les installations de réfrigération, les serres et les applications CVC exigeantes. Elle convient à des milieux de mesure de CO2 difficiles et humides, pour lesquels il est nécessaire d'avoir des mesures stables et précises du niveau de CO2 en ppm [22].

Figure I.20. Une sonde de dioxyde de carbone GMP252 [22].

- Plage de mesure de 0… à 10 000 ppm de taux de CO2
- Peut être utilisée pour des mesures allant jusqu'à 30 000 ppm de taux de CO2 avec une précision réduite
- Sonde intelligente et autonome avec sorties analogiques (V, mA) et numériques (RS485 avec le protocole Vaisala ou Modbus)
- Large plage de températures de fonctionnement de -40… à +60 °C

- Compensation de pression et de température
- Tête du capteur chauffée pour éviter la condensation

❖ Principe de fonctionnement

La concentration du CO2 dans l'air est mesurée avec des capteurs optiques, cette technologie de mesure se base sur la quantité de lumière absorbée par le gaz. Les molécules de CO2 absorbent la lumière dans une longueur d'onde spécifique (4.26µm), en mesurant cette absorption nous pouvons déterminer la concentration du CO2 [22].

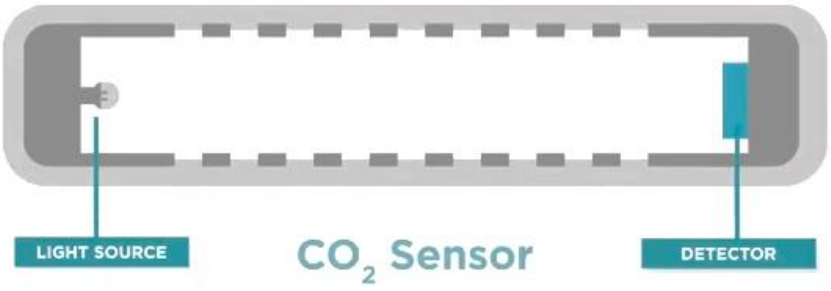


Figure I.21. Principe de fonctionnement d'un capteur de CO2 [22].

I.5. Le Micro climat de la serre

Les facteurs climatiques qui influent le plus sur le climat à l'intérieur de la serre sont la température, l'humidité de l'air, le rayonnement solaire et le vent extérieur. En réalité, chacun de ces facteurs engendre une combinaison d'effets qui peuvent être favorables ou non au fonctionnement de la serre selon les conditions locales qui prévalent. En Algérie, les aires climatiques sont très diversifiées et le climat varie de type méditerranéen au type saharien. Au nord, les hivers pluvieux et froids, les étés chauds et secs tandis qu'au sud, les températures sont très élevées les jours (35°) et très bas la nuit (0°) et l'aridité des sols est extrême [23].

La température et l'humidité interviennent de façon prépondérante dans la croissance et le développement de la végétation puisque les espèces pour lesquelles sont mises en œuvre les techniques de cultures protégées sont essentiellement des espèces de saison chaude adaptées à des températures moyennes mensuelles de l'air situant entre 17°C et 28°C [9], ce qui correspond aux conditions limites ci-après :

- Moyenne mensuelle des températures minimales : 12°C
- Moyenne mensuelle des températures maximales : 32°C

I.5.1. Intérêts de la détermination du microclimat

Pour répondre à ces besoins, de nouveaux outils et techniques de production ont été développés. L`outil serre et ses équipements de chauffage permettent de mieux gérer la croissance des plantes car la gestion du climat est maîtrisée.

Pour une croissance optimale, il y a la nécessité d'un climat maîtrisé. Plusieurs paramètres du climat influencent fortement la croissance de la plante.

I.5.2. La température

Le contrôle adéquat de la température est le facteur le plus important pour garantir une bonne croissance des plantes. En effet, la température et l'un des paramètres le plus important dans la gestion du climat, et aussi le plus difficiles à gérer.

La température optimale diffère d'une culture à une autre et selon le stade de la culture elle influe sur plusieurs phénomènes bioénergétiques (respiration, photosynthèse,…)

Toute grande déviation de la température par rapport aux températures typiques d'une plante provoque un ralentissement de la croissance et la détérioration dans certains cas, de ce fait on note l'importance de ce paramètre dans les systèmes de gestions des serres agricoles [23].

I.5.2.1. L'effet de serre

L'effet de serre est le résultat d'interposition entre l'atmosphère et le sol d'un matériau transparent pour les radiations courtes provenant du soleil et absorbant au maximum les radiations longues en provenance du sol.

L'effet de serre se manifeste essentiellement par l'élévation de la température. Sa détermination peut être ramenée à la différence de température entre l'intérieur et l'extérieur de la serre [DT] multipliée par le nombre d'heures pendant lesquelles cette différence existe (H).

Effet de serre = DT.H (1.1)

Le phénomène d'effet de serre permet une augmentation conséquente de la température, mais dans d'autres circonstances, la serre devient un émetteur de chaleur et les températures peuvent fortement diminuer [8].

Ce phénomène s'explique par plusieurs raisons :

- la nuit, les parois et la toiture échangent de la chaleur avec l'extérieur par conduction, convection et rayonnement ;
- les fuites potentielles provoquent un renouvellement d'air qui substitue à l'air chaud intérieur l'air froid de l'extérieur ;
- d'autres pertes ont également lieu par le sol par conduction et évaporation ;
- de plus, la serre ne possédant pas des parois épaisses n'est pas capable d'emmagasiner beaucoup de chaleur, on dit qu'elle possède une faible inertie thermique. Elle correspond alors à un volume dans lequel la température s'éleve facilement mais en contrepartie peut rapidement diminuer.

I.5.2.2. Consigne pratique de température ambiante

Nous avons vu que la détermination des consignes de température était très délicate. En attendant de disposer de modèles élaborés prenant en considération la physiologie des plantes, les échanges de chaleur el les données économiques, on peut utiliser les valeurs consacrées par 1`usage ou fournies dans la littérature. Mais ce sont en général les fournisseurs de matériel végétal qui donnent les meilleurs conseils en matière de consignes de températures [23].

I.5.3. L'humidité

L'humidité est la présence d'eau ou de vapeur d'eau dans l'air ambiant, elle influe sur la croissance de la plante et sur le développement des maladies.

En pratique quand on parle de la mesure de l'humidité on fait allusion au taux d'humidité exprimé en % ce qui représente l'humidité relative [23].

L'humidité de l'air et la température présentent une grande relation. En effet la capacité de l'air à retenir la vapeur d'eau double chaque fois que la température augmente d'une dizaine de degrés.

Maintenir un niveau d'humidité et de température optimal dans une serre permet d'avoir un bon rendement de croissance.

Le contrôle de tous ces paramètres se fait via de nombreux dispositifs des plus simples au plus complexes tels que : l'ouverture des toits ouvrant, les ventilateurs extracteurs, les chauffages, les humidificateurs… [8].

I.5.3.1. Consigne pratique d'hygrométrie

Il n'existe pas des recommandations espèce par espèce. Tout au plus peut-on dire que [23]:

- Pour la phase de multiplication, humilité relative doit être supérieure à 80%.
- Pour les phases de croissance, de floraison et de fructification, la plupart des plantes s'accommodent d'une humidité relative comprise entre 60 et 80%, à l'exception des plantes vertes et du concombre qui préfèrent une hygrométrie supérieure à 70%, et des plantes succulentes (Cactées) qui préfèrent une humidité relative comprise entre 30 et 60%. La pollinisation chez la tomate requiert une humidité relative comprise entre 50 et 70%.

On peut aussi donner les conseils généraux suivants :

- Eviter les condensations.
- Eviter les hygrométries proches de la saturation (l00%).
- Eviter les hygrométries inferieures à 40% chez les plantes arrosées de manière incorrecte ainsi que chez les jeunes plants qui n'ont pas encore été durcis.
- Eviter absolument les hygrométries inferieures à 20%.

I.5.3.2. La déshumidification

L'humidité de l'air est généralement exprimée en humidité relative ou déficit hydrique.

On constate que la technique de déshumidification traditionnellement employée par les producteurs combine l'aération et le chauffage. Le chauffage permet d`une part de diminuer 1`humidité relative de l'air en augmentant la pression de vapeur saturante, d`autre part de faciliter 1`évaluation de I 'air chaud chargé humidité grâce à l'aération.

Cette technique entraîne une perte d`énergie puisqu`une partie de l'énergie dégagée par le chauffage est évaluée par 1`aeration (celle pratique de la déshumidification représente environ 20 à 30% des dépenses énergétiques) [5].

I.5.4. Le Rayonnement

Le rayonnement : échange de chaleur entre deux parois séparées par un milieu transparent. Tous les corps solides, liquides ou gazeux émettent un rayonnement de nature électromagnétique. Cette énergie est échangée directement des parois à la surface des plantes et non à l'air ambiant.

II transmet l'énergie sans support matériel par le biais d'une onde électromagnétique solaire. Ce transfert est défini à travers 1`exemple suivant d'un corps noir porté' a la température T émet un rayonnement dont l`énergie rayonnée par unité' de temps et de surface est 1 (loi de Stéphane)

$$Q = \delta T^{\wedge}4$$

La température apportée par le rayonnement et l`effet de serre mais diminuée par les pertes thermiques ne permet pas d`assurer tout au long de l`année des températures optimales pour la croissance des plantes.

De plus, l`humidité à l`intérieur de la serre, dépendante des conditions climatiques de la transpiration de la culture en fonction de l`espèce considérée doit être bien maîtrisée pour ne pas limiter la croissance de la plante et éviter les risques phytosanitaires (Développement de maladies) [5].

I.5.5. Le gaz carbonique

Les producteurs cherchent à obtenir une teneur dans la serre en CO2 supérieur à la normale pour maximiser l'activité de la photosynthèse, car le gaz carbonique est un facteur limitant de la photosynthèse. Lorsque la teneur de CO2 augmente, la photosynthèse augmente d'abord de façon linéaire, puis moins rapidement jusqu'à 1000 ou 1500 ppm.

Quel que soit le type des plantes, le CO2 est un élément déterminant de la croissance. En stimulant la photosynthèse, il accroit les rendements et augmente de façon significative la qualité de la production. L'enrichissement de l'atmosphère en CO2 peut être effectue par le rejet direct des produits de combustion de propane.

Le CO2 peut être obtenu par la combustion de carburants tels que le gaz naturel, le propane, le kérosène ou encore directement à partir de réservoirs contenant du CO2 à l'état pur. Chaque source a ses avantages et ses inconvénients [5].

I.6. Conclusion

Le choix d'une structure de serre, des équipements et de la conduite climatique est le maillon clé de la qualité de la serre-culture. Dans ce chapitre nous avons vu les types de serres agricoles les plus populaires, ensuite nous avons défini les différents composants nécessaires à la gestion efficace des paramètres climatiques sous serre. Enfin nous avons déterminé les paramètres les plus agissants sur la croissance de la culture sous serre.

L'amélioration de la gestion du climat repose sur une meilleure connaissance de la physiologie des plantes afin de déterminer ses besoins et les répercussions de l'environnement sur le fonctionnement de la culture.

CHAPITRE 2

Description du systeme à réaliser

II.1. Introduction

L'étude conceptuelle est une étape très importante avant d'entamer une quelconque réalisation. Pour bien concevoir un dispositif de contrôle des paramètres climatiques d'une serre agricole, il est nécessaire de bien choisir la solution technologique adéquate. Cette solution est inspirée des solutions techniques existantes utilisées dans ce domaine et vise à répondre aux besoins et aux exigences du cahier des charges fonctionnel dont les fonctions de services à assurer, se résument en :

- Contrôle et régulation des paramètres climatiques de la serre.
- Envoi des données via un réseau sans fil.
- Enregistrement et visualisation des paramètres à distance.
- Configuration à distance sur site web des consignes de régulation (selon le type de plantes cultivées).

II.2. Schéma synoptique

La figure II.1 présente le schéma synoptique englobant les différents modules de base du dispositif à réaliser.

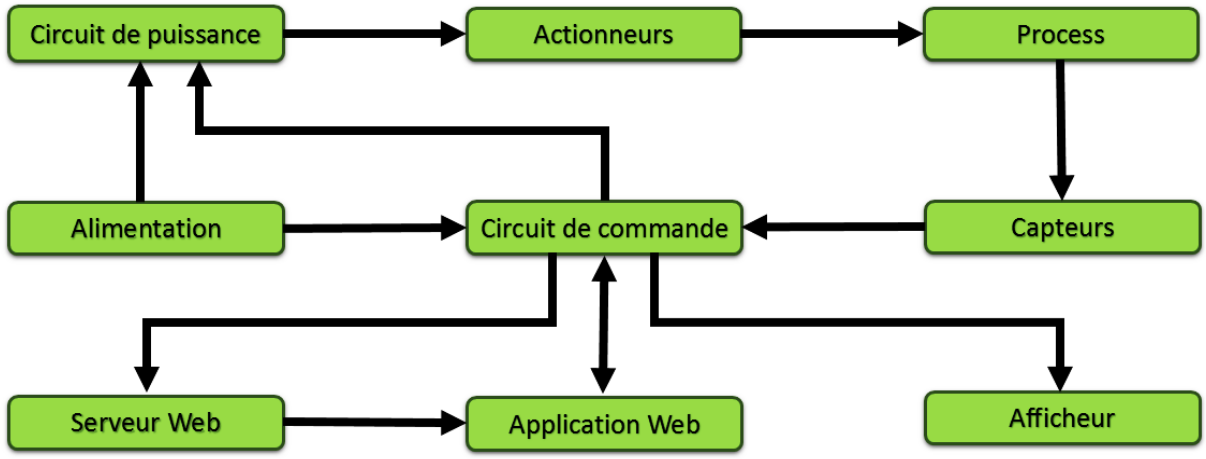


Figure II.1. Schéma synoptique du dispositif à réaliser.

Le schéma synoptique est constitué de :

1. Un bloc d'alimentation.
2. Des actionneurs (chauffage, humidificateur, pompe d'arrosage, etc.) agissent sur les paramètres climatiques de la serre.
3. Le process qui représente notre serre prototype.
4. Un circuit de puissance assurant la connexion entre l'unité de commande et les actionneurs.
5. Un afficheur pour la présentation des paramètres climatiques.
6. Des capteurs (humidité, température, photorésistance, etc.) pour la mesure des paramètres.
7. Un serveur web distant pour la réception et l'enregistrement des données climatiques, ainsi que l'hébergement de l'application web.

8. Une application web pour l'affichage des données climatiques à distance, la configuration des consignes de régulation selon le type de plantes cultivées et la commande manuelle des actionneurs.
9. L'unité programmable assurant la synchronisation des différentes fonctions de service : régulation des paramètres climatiques, acquisition des données, envoie des données vers le serveur distant, gestion de l'affichage, réception des commandes manuelles depuis l'application web.

II.3. Choix de la solution technologique

Le contrôle des paramètres climatiques d'une serre agricole nécessite des capteurs appropriés, son état doit aussi être traité par une carte de commande, qui transmet les signaux de commande aux actionneurs via une carte de puissance.

Pour le choix de la solution technologique, en premier lieu, nous entamons la partie commande puis nous passeront vers la partie puissance.

II.4. Partie commande

II.4.1. Etude des microcontrôleurs

Les microcontrôleurs sont largement utilisés pour les applications de régulation et de commande de processus [9]. Un microcontrôleur comporte entre autre :

- Un microprocesseur.
- Une mémoire vive (RAM).
- Une mémoire permanente (ROM).
- Interfaces d'E/S parallèles et séries (RS232, I2C, SPI…).
- Interfaces d'E/S analogiques.
- Registres « Timers » pour la gestion du temps et d'évènements.

Pratiquement tous les fabricants de microprocesseurs (Motorola, Intel. Hitachi, Texas lnstrument, Espressif, etc.) proposent une ou plusieurs gammes de microcontrôleurs :

- Les microcontrôleurs 4 bits servent essentiellement à des tâches simples. De tels microcontrôleurs sont par exemple utilisés au sein des appareils ménagers grand public (cuisinières, machines à laver ou aspirateurs …).

- Les microcontrôleurs 8 bits sont capables de répondre à des exigences plus élevées.

- Les microcontrôleurs 32 bits sont utilisés pour la commande de machines ou le contrôle de processus lorsque les contraintes temps réel sont sévères ou lorsque les algorithmes de régulation nécessitent une puissance de calcul importante [24].

II.4.1.1 Avantages des microcontrôleurs

Par rapport au microprocesseur, l'utilisation des microcontrôleurs a plusieurs points forts parmi lesquels on peut citer:

- Une diminution évidente de l'encombrement matériel et de circuit imprimé.
- Une plus grande fiabilité du système, car le nombre des composants sera plus réduit, donc un nombre de connexion composants/supports ou composants/circuits plus réduit.

- Le microcontrôleur contribue à réduire les coûts à plusieurs niveaux, car il est moins cher que les autres composants qu`il remplace.
- Les outils de développement sont, en général, téléchargeable gratuitement sur le WEB.
- Le jeu d'instruction réduit est souple, puissant et facile à maîtriser.
- Les versions avec mémoire flash présentent une souplesse d'utilisation et des avantages pratiques indéniables.

II.4.1.2 Choix du microcontrôleur

L'une des étapes les plus importantes à la réalisation de notre projet est le choix du microcontrôleur, il est nécessaire de choisir le plus adapté à nos objectifs, Il existe beaucoup de critères de sélection dont nous devons tenir compte comme :

- Le nombre d'entrées/sorties analogiques.
- Le nombre d'entrées/sorties numériques.
- Puissance de calcul suffisamment élevée pour gérer des algorithmes en temps réel.
- Taille de la mémoire programmée (pour contenir l'ensemble du programme).
- La taille de la mémoire RAM (pour les calculs que le microcontrôleur doit effectuer).
- La mémoire EPROM (si on a besoin que certaines données soient sauvegardées si l'alimentation se coupe).
- Le prix et la disponibilité du microcontrôleur sur le marché.
- Idéalement : la possibilité de se connecter à internet pour envoyer et recevoir des données depuis un serveur distant.

II.4.1.3 Etude comparative

Le tableau suivant représente une étude comparative de quelques microcontrôleurs disponible sur le marché :

Tableau II.1. Etude comparative de quelques microcontrôleurs disponibles sur le marché [26][27][28].

Type du microcontrôleur	FlashROM (en octet)	RAM (en octet)	EEPROM (en octet)	Fmax (en Mhz)	Bus de données	E/S	Boitier
PIC16F877A	8198x14	368	256	20	8bits	33	40 broches
PIC18F4550	32K	2K	256	48	8bits	35	40 broches
ATMEGA 328 (Arduino UNO)	32K	2K	1024	20	8bits	23	32 broches
ATMEGA 2560 (Arduino Méga)	256K	8K	4098	20	8bits	54	100 broches
ESP8266 12	**4M**	**64K+96K**	**1024**	**160**	**32bits**	**16**	**22 broches**

Le microcontrôleur ESP8266 12E a la puissance de calcul la plus élevée avec un bus de données 32 bits, une fréquence pouvant atteindre jusqu'à 160 Mhz et une grande capacité mémoire RAM et ROM qu'ils lui permettront de gérer des algorithmes de régulation complexes avec un grand nombre de variables.

Il est aussi capable de se connecter à un réseau Wifi pour envoyer et recevoir des données par internet, ce qui le rend un microcontrôleur idéal pour la réalisation de notre projet.

II.4.1.4 Le microcontrôleur ESP8266

L'ESP8266 est un microcontrôleur 32bits produit en 2014 par la société chinoise Espressif, comme déjà mentionné, il intègre un module Wifi 802.11b/g/n permettant d'établir des connexions TCP/IP, en mode client/serveur HTTP [25].

❖ **Principales caractéristiques de l'ESP8266 [26] :**

- Processeur RISC 32bits cadencé à 80Mhz (par défaut) ou à 160 Mhz.
- 64 Ko de RAM pour les instructions et 96 Ko pour les données.
- Mémoire flash externe QSPI entre 512 KB et 4MB selon les modèles.
- Puce Wifi 2.4 GHz (802.11 b/g/n) avec antenne intégré.
- WEP or WPA/WPA2 authentication, or open networks.
- 16 Entrées/sorties numériques GPIO.
- PWM / ADC 10bits (variante 12E).
- UART / I2C / I2S / SPI.
- Alimentation en 3,3V.
- Consommation : entre 60mA et 215mA en fonctionnement normal, quelques dizaines de µA en veille.

II.4.1.5. Brochage de l'ESP8266 12E

La figure II.2 représente le schéma de brochage de l'ESP8266 variante 12E qui se présente sous la forme d'un boitier à 22 broches avec une antenne Wifi intégrée (antenne PCB) [25].

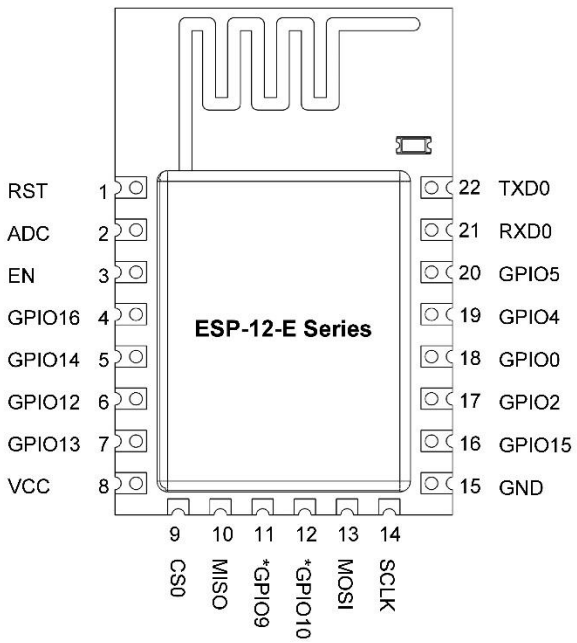


Figure II.2. Schéma de brochage de l'ESP8266 [26].

II.4.1.6. Architecture interne d'un ESP8266

La Figure II.3 ci-dessous présente les principaux blocs fonctionnels présents à l'intérieur d'un ESP8266.

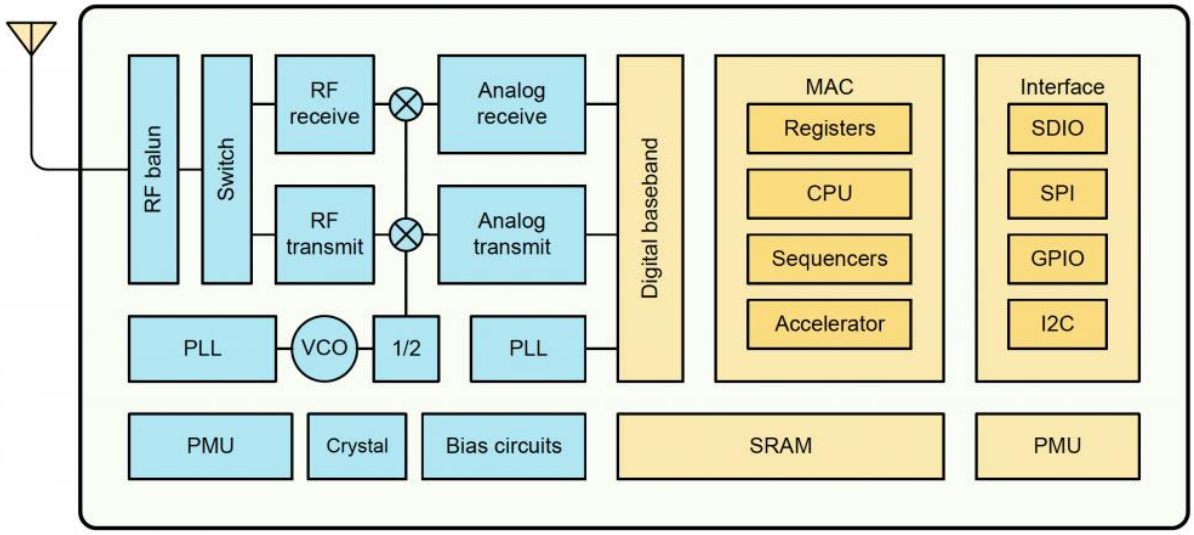


Figure II.3. Schéma bloc représentant l'architecture interne de l'ESP8266 [26].

L'architecture de l'ESP8266 est composée de deux parties [26]:

- **Partie RF ou Radio fréquence (couleur bleue dans la figure II.3)**

Cette partie permet de gérer les communications par ondes radio entre deux appareils, elle se compose principalement d'un émetteur/récepteur radio, de deux synthétiseurs de fréquence (PLL) et d'une unité de gestion de l'énergie (PMU).

- **Partie logique (couleur jaune dans la figure II.3)**

Elle contient tous les éléments d'une structure à base de microprocesseur :

1. **L'unité de Calcul:** interprète les instructions et traite les données d'un programme à une vitesse définie par la fréquence d'horloge (en général à quartz).
2. **L'unité de Contrôle:** commande et contrôle le fonctionnement du système, elle permet de gérer de manière optimale les enchaînements de transferts élémentaires pendant et entre le traitement des instructions.
3. **La mémoire vive ou RAM :** est une mémoire volatile. Elle est utilisée pour stocker des données temporaires. L'ESP8266 utilise un type de mémoire vive appelé « statique » ou SRAM qui utilise des bascules pour mémoriser les données, contrairement à la mémoire dynamique, elle n'a pas besoin de rafraîchir périodiquement son contenu.
4. **La mémoire FlashROM :** assure le stockage ou la mémorisation des programmes à exécuter. Elle peut conserver ses données même si le microcontrôleur n'est pas alimenté.
5. **Les registres temporaires :** ils assurent le stockage de la valeur nécessaire pour une opération de l'unité de calcul.
6. **Les ports ou GPIO :** assurent la connexion avec l'environnement extérieur.
7. **Les interfaces séries :** permet au microcontrôleur de communiquer avec d'autres systèmes à base de microprocesseur. Les données envoyées ou reçues se présentes sous la forme

d'une succession temporelle (sur un seul bit) de valeurs binaires images d'un mot. Il y a 2 types de liaison série : synchrone et asynchrone.

- **Liaison série synchrone** : Dans ce dispositif la transmission est synchronisée par un signal d'horloge émis par l'unité maître.
- **Liaison série asynchrone :** Ce dispositif ne possède pas de signal d'horloge de synchronisation. Les unités en liaison possèdent chacune une horloge interne cadencée à la même fréquence.

8. **Modules annexes :** ils représentent toutes les fonctions annexes (timers, comparateurs, convertisseurs analogiques/numériques…).

II.4.2. Les circuits relatifs au microcontrôleur

1- L'oscillateur (Horloge)

L'horloge peut être réalisée soit avec un quartz, soit une horloge extérieure, soit avec un circuit RC, Dans ce dernier cas, la stabilité du montage est limitée.

Une horloge à quartz utilise un oscillateur à quartz pour définir le temps. Le quartz est un composant électronique passif, qui a la particularité d'osciller à une fréquence précise lorsqu'il est stimulé électriquement.

L'ESP8266 intègre un oscillateur à quartz d'une fréquence de 26MHz, cette fréquence sera multipliée par le circuit PPL selon la configuration du microcontrôleur [26].

2- Le synthétiseur de fréquence (ou PLL)

La synthèse de fréquence consiste à générer à partir d'un signal fourni par un oscillateur de référence, un signal de fréquence différente. La stabilité de fréquence de l'oscillateur de référence est alors transférée au signal généré dont la fréquence correspond aux besoins du microcontrôleur [27].

Un synthétiseur de fréquence est par extension un instrument qui permet de générer, dans une gamme de fréquences donnée, un signal dont on peut ajuster la fréquence et l'amplitude et auquel il peut être imposé une modulation de fréquence, de phase ou d'amplitude.

3- RST (reset)

Pour la remise à zéro (reset), nous avons opté pour la mise en place d'une commande manuelle. La remise à zéro s'effectue par un simple appui sur le bouton poussoir (Pin N°1) correspond à RST (mise à la masse).

II.4.3. Le bus I²C

L'I²C est un bus série permettant de transmettre des informations entre divers circuits connectés sur le bus grâce à seulement trois fils :

- SDA (Serial Data Line) : ligne de données bidirectionnelle,
- SCL (Serial Clock Line) : ligne d'horloge de synchronisation bidirectionnelle.
- La masse qui doit être commune aux équipements.

Ce qui permet de réduire la complexité des circuits imprimés à réaliser [29].

Chaque circuit est reconnu par son adresse et peut être soit transmetteur soit receveur de l'information. On utilise un adressage sur 7 bits pour identifier les nœuds, ce qui permet (en théorie) jusqu'à 128 nœuds sur le réseau. Ces circuits peuvent être : Un ordinateur, un microcontrôleur,

une mémoire, un périphérique (affichage, convertisseur A/D, capteurs...) etc.

Le protocole utilisé par I2C est dit [29]:

- Synchrone, car on utilise un mécanisme de requête/réponse.
- Half-duplex, car on peut communiquer dans les deux sens, mais pas simultanément.
- Maitre-esclaves, car seul un nœud maitre peut communiquer avec les esclaves du réseau.
- Multi-maitres, car plusieurs maitres peuvent être présents sur le réseau (Figure II.4).

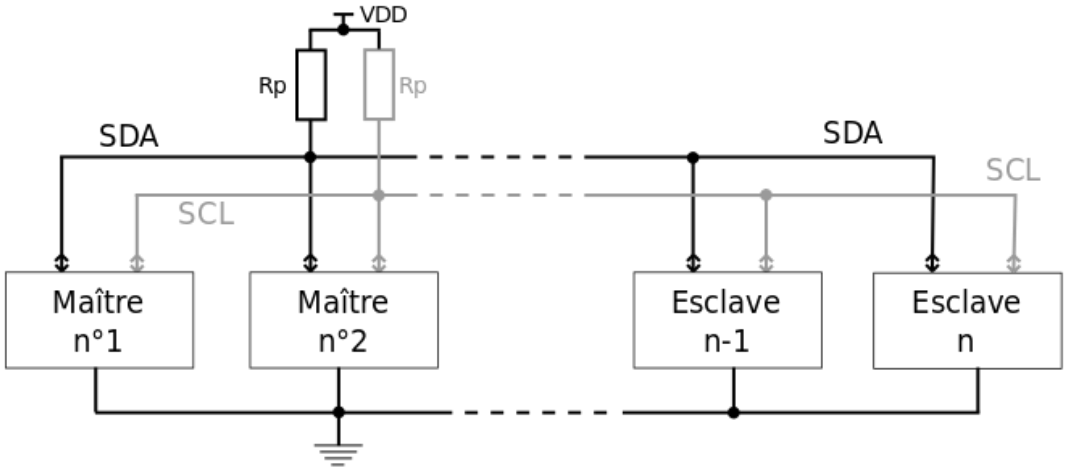


Figure II.4. Architecture I²C avec plusieurs maîtres et plusieurs esclaves.

II.4.4. Limites de l'ESP8266 et solutions

L'ESP8266 est un microcontrôleur puissant et peu couteux, malgré les nombreuses fonctionnalités qu'il propose, il présente quelques limites à son utilisation :

a) Nombre d'entrées/sorties :

L'ESP8266 dans sa version 12E dispose de 16 ports E/S (appelé aussi GPIO) [27] :

- Les ports GPIO 0,2 et 15 sont utilisés pour la sélection du mode de démarrage (mode normal ou programmation).
- Les ports 1 et 3 sont utilisés pour les communications serial (RX, TX).
- Les ports GPIO 6, 7, 8, 9, 10 et 11 sont utilisés pour la communication avec la mémoire flash externe SPIFS.
- Et le GPIO 16 est utilisé pour sortir du mode veille prolongé.

Ce qui nous laisse 5 ports E/S pour connecter les capteurs, les actionneurs et l'affichage ce qui est très insuffisant. Il existe certaines méthodes pour pouvoir utiliser les pins de sélection du mode de démarrage (0,2 et 15) et les pins Rx, TX (1 et 3) comme E/S à usage générale (donc 10 ports au total) mais cela reste assez limité.

❖ **Solution :**

Il existe une solution peu coûteuse et qui consiste à utiliser, en parallèle, un ou plusieurs extenseurs de port comme par exemple le circuit PCF8574 qui est un extenseur de port E/S 8bit. Grâce au protocole I²C, on peut connecter 8 circuits PCF8574 sur le même bus, ce qui permet d'ajouter 64 ports E/S au microcontrôleur ESP8266.

Il existe une variante PCF8574A qui utilise un adressage 8bit diffèrent, ce qui permet d'ajouter 8 autres circuit au bus I²C avec un total de 128 port E/S. Ces ports peuvent être utilisés pour gérer un grand nombre d'actionneurs et de capteurs numériques [39].

La figure II.5 présente un exemple d'utilisation du PCF8574 qui permet grâce à seulement 2 lignes (SDA et SCL) de gérer un afficheur LCD 16x2 ainsi que 8 boutons poussoirs.

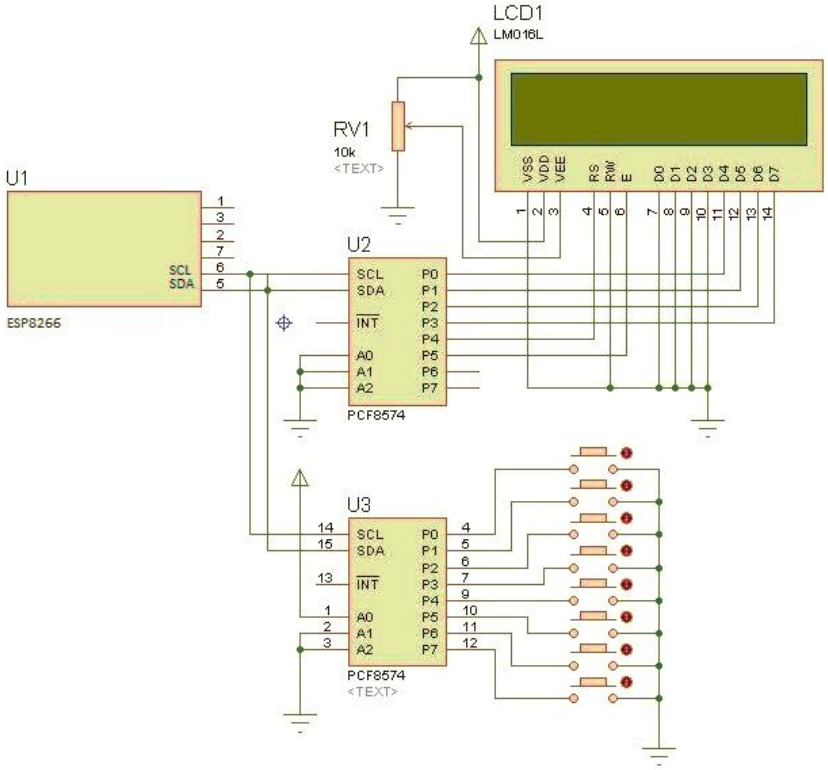


Figure II.5. Exemple d'utilisation du circuit PCF8574 réalisé avec le logiciel de simulation Proteus ISIS.

b) Convertisseur A/N

L'ESP8266 intègre un convertisseur Analogique/numérique 10 bits avec une tension d'entrée maximale de 1V. Après quelques essais, nous avons réalisé que même en ajoutant un filtrage numérique, les valeurs mesurées par le convertisseur sont instables à cause des perturbations provoquées par l'antenne Wifi du microcontrôleur [27].

❖ Solution

La solution est d'utiliser un convertisseur A/N externe comme le MPC3021 qui est un convertisseur 10 bits de petite taille et de faible puissance, la communication avec le microcontrôleur se fait par le biais du bus I²C [30].

Il existe une autre solution qui consiste à utiliser un autre microcontrôleur (par exemple l'ATMEGA 328 [28]) comme esclave I²C, ce microcontrôleur sera programmé pour mesurer l'entrée analogique pour ensuite la transmettre au microcontrôleur principal. Pour effectuer la conversion des signaux 3.3V et 5V, nous allons utiliser un circuit « Level Shifter » comme l'AN10441 [35].

II.4.5. L'affichage

Pour l'affichage il existe plusieurs technologies d'afficheurs, dont les afficheurs LCD ou OLED. Dans ce projet nous allons utiliser un afficheur LCD car c'est la solution la moins couteuse, la communication avec le microcontrôleur se fera à l'aide d'un extendeur de port I²C comme présenté dans la figure II.5.

II.4.6. Utilisation du bus I²C

Afin d'étendre les fonctionnalités de notre microcontrôleur, nous avons configuré notre bus I²C comme le montre la figure II.6.

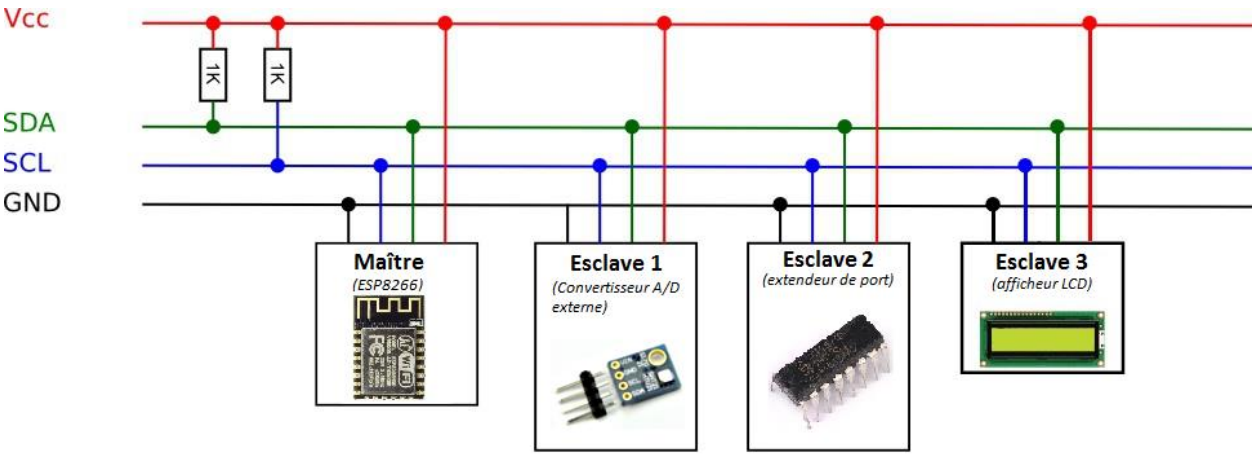


Figure II.6. Utilisation du bus I²C dans notre carte de commande.

II.4.7. Programmation du microcontrôleur

L'ESP8266 peut se programmer de plusieurs façons [27]:

- Avec des scripts Lua, interprétés ou compilés, avec le firmware NodeMCU.
- En C, avec l'IDE Arduino.
- En JavaScript, avec le firmware Espruino.
- En MicroPython, avec le firmware MicroPython.
- En C, avec le SDK d'Espressif.
- En C, avec le SDK esp-open-sdk3 basé sur la chaîne de compilation GCC.

Pour la programmation du microcontrôleur, nous allons utiliser l'environnement de développement d'Arduino car cet IDE présente beaucoup d'avantages :

- Gratuit et open-source.
- Multiplateforme (Windows, mac et linux).
- Facilité d'installation et programmation claire et intuitive.
- Disponibilité de beaucoup d'exemples de programmes et de cours.
- Disponibilité d'une grande quantité de librairies qui permettent de faciliter la programmation.

- Communauté importante sur internet : forums, cours et tutoriaux.

La figure II.7 présente l'interface de l'IDE Arduino avec le programme de notre microcontrôleur. Il est séparé en 7 fichiers afin de bien l'organiser.

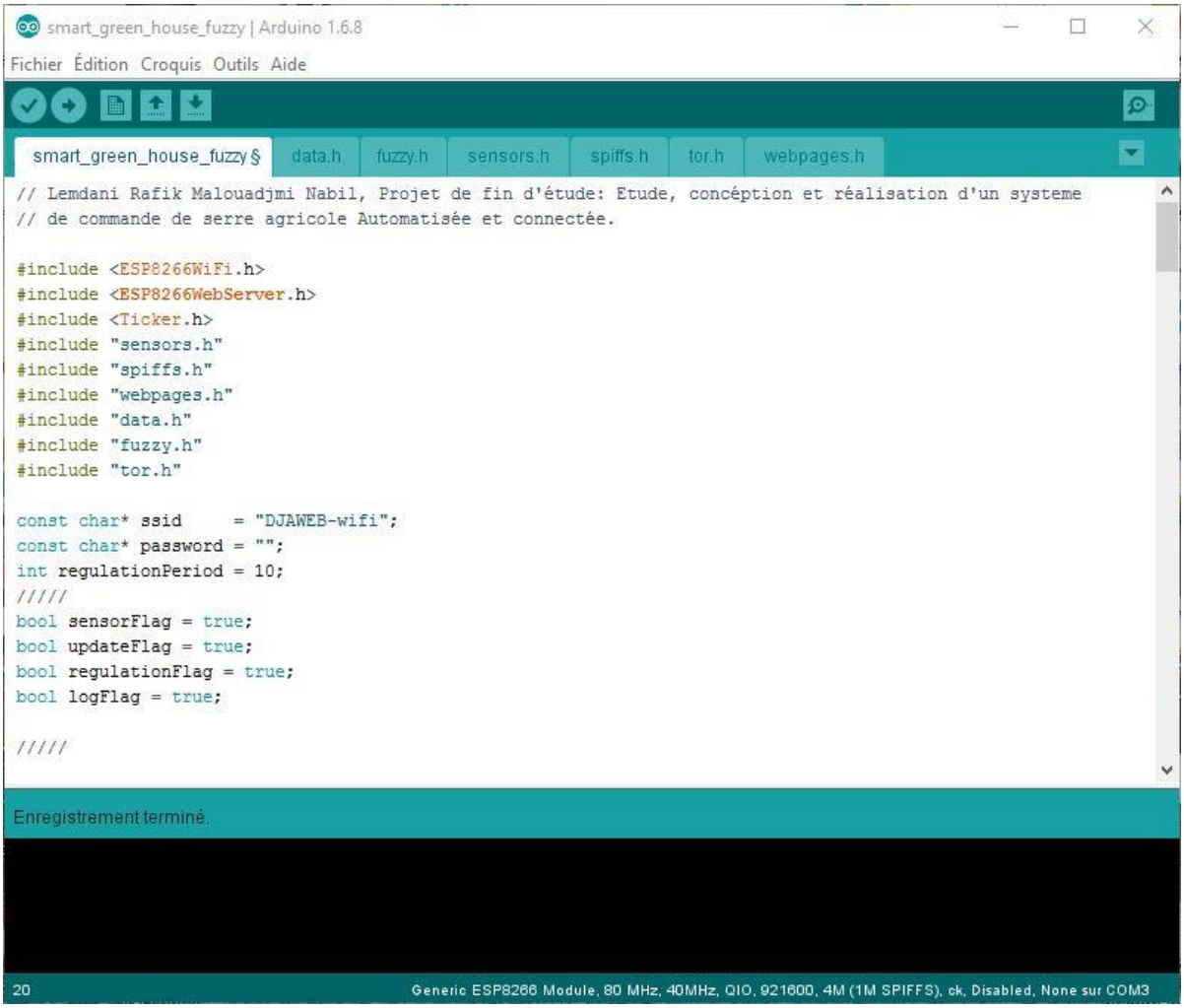


Figure II.7. Interface de l'environnement de développement Arduino avec la structure de notre programme.

II.5. Capteurs utilisés

II.5.1. La photorésistance

Une photorésistance (figure II.8) est un composant dont la résistivité dépend de la luminosité ambiante. Autrement dit, c'est une résistance dont la valeur change en fonction de la lumière qu'elle reçoit.

On la désigne aussi par LDR (Light Dépendent Résistor ou résistance dépendante de la lumière), il existe différents types de photorésistances, chacune ayant des valeurs de résistance différentes en fonction de la luminosité ambiante. Le type le plus classique de photorésistances est de 1M ohms (obscurité) / 12K ohms (pleine lumière) [31].

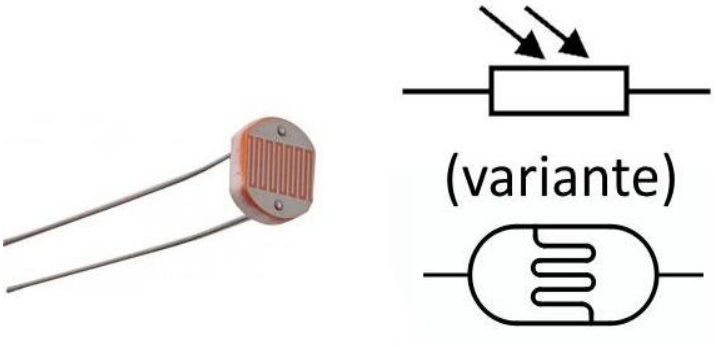


Figure II.8. Une Photorésistance avec son symbole.

❖ Principe de fonctionnement de la photorésistance

Un cristal de semi-conducteur à température basse contient peu d'électrons libres. La conductivité du cristal est très faible, proche de celle d'un isolant. Lorsque la température du cristal augmente, de plus en plus d'électrons qui étaient immobiles dans les liaisons covalentes s'échappent et peuvent participer à la conduction.

A température constante, si le même cristal semi-conducteur est soumis à une radiation lumineuse, l'énergie apportée par les photons peut suffire à libérer certains électrons utilisés dans les liaisons covalentes entre atomes du cristal. Plus le flux lumineux sera intense, plus le LDR est inversement proportionnelle à la lumière reçue. La sensibilité dépend de la fréquence de la radiation lumineuse [31].

❖ Propriétés des capteurs LDR

Avantage :

- Capteur sensible.
- Larges gammes spectrales.
- De faible coût.
- Facile à mettre en œuvre.

Inconvénients :

- Non linéaire.
- La vitesse de variation de la résistance avec l'éclairement est faible et non symétrique. (Temps de montée de l'ordre de 35 ms et temps de descente de l'ordre de 10 ms).

II.5.2. Capteur de gaz DFRobot SEN0159

Ce module (Figure II.9) est basé sur le capteur de gaz MG-811 permettant de détecter la présence de CO_2. La tension de sortie du module diminue à mesure que la concentration de CO_2 augmente. Avec un potentiomètre on peut régler le seuil de tension. Il intègre un circuit chauffant fournissant au capteur la meilleure température de fonctionnement.

Ce capteur est doté d'un circuit de conditionnement intégré, permettant d'amplifier le signal de sortie [36].

Figure II.9. Capteur de CO2 SEN0159 [36].

❖ Caractéristiques du capteur [36]

- Sortie : analogique (et une sortie numérique).
- Alimentation : 5V.
- Potentiomètre et module capteur MG-811 ultra-sensible intégrés.
- Un connecteur de haute qualité
- Circuit de conditionnement pour amplifier le signal de sortie.
- Circuit chauffant intégré.
- Dimensions : 32 x 42 mm
- Haute sensibilité et temps de réponse rapide.

Etant donné la cherté de ce capteur et son indisponibilité sur le marché algérien, nous avons procéder à son remplacement par un potentiomètre afin de simuler son comportement dans notre système.

II.5.3. Capteur de température LM35

Le LM35 permet de transformer l'effet du réchauffement ou du refroidissement en signal électrique. Il délivre en sortie une tension proportionnelle à la température mesurée.
Ce capteur est fabriqué par TEXAS INSTRUMENTS est très utilisée dans le domaine industriel, il est précis et fiable avec un cout réduit.
Le LM35 ne nécessite aucun étalonnage externe pour fournir des précisions typique de± 0.5 °C à +25 °C et ± 1 °C dans la plage de mesure -55 ° C à +150 °C, chaque 10mV représente un degré Celsius [32].

Tableau II.2. Caractéristiques du capteur LM35 [32].

Tension d'alimentation	4 à 30 volts
Sensibilité	10mV/°C
Précision	±1°C
Courant au repos	60-µA

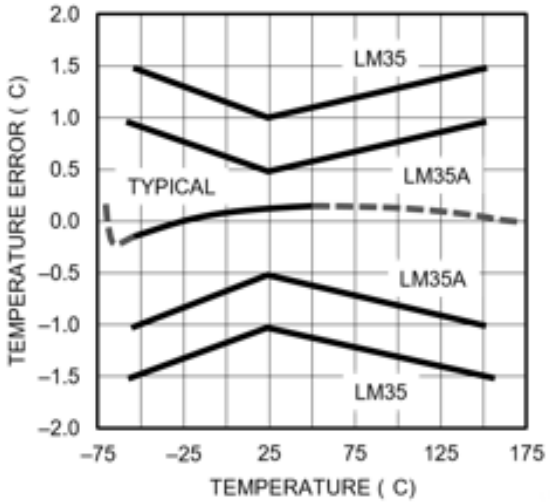


Figure II.10. Précision des différentes versions de LM35 [32].

Nous avons choisi ce capteur pour les raisons suivantes :

- Simplicité d'utilisation.
- Il ne nécessite aucun étalonnage externe.
- Précisions typique ± 1°C dans la plage de mesure -55 °C à +150°C.
- Peu coûteux.
- Disponible sur le marché.

II.5.4. Capteur d'humidité DHT11

Le capteur DHT11 (Figure II.11) fournit une information numérique proportionnelle à la température et l'humidité mesurée. Il est constitué d'un capteur de température à base de thermistances CTN, d'un capteur d'humidité résistif et un microcontrôleur qui s'occupe de faire les mesures, de les convertir et de les transmettre [37].

Il s'interface grâce au protocole OneWire qui permet de transmettre et de recevoir des données sur un seul fil [38], cette technologie utilisée par le capteur DHT11 garantie une grande fiabilité, une excellente stabilité à long terme et un temps de réponse très rapide.

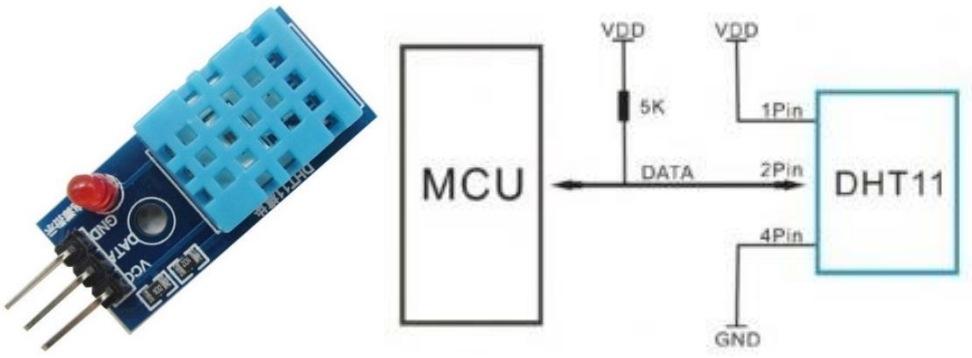


Figure II.11. Capteur DHT11 et son schéma de brochage [37].

❖ Caractéristiques techniques du DHT11

Ce capteur présente les caractéristiques suivantes [37]:

- Alimentation : 5V
- Consommation : 0.5 mA en nominal / 2.5 mA maximum
- Etendue de la mesure de température : 0°C à 50°C ± 2°C
- Etendue de la mesure de l'humidité : 20-90%RH ±5%RH
- Période de mesure: 2s
- Dimensions: 12x15.5x5.5mm

II.5.5. Le potentiomètre

Ce sont des résistances que l'on peut faire varier manuellement ou avec un tournevis (les potentiomètres ajustables). Ils peuvent être aussi appelés rhéostat ou résistance variable. Le symbole est aussi parfois utilisé pour exprimer que dans un circuit, la valeur de la résistance est commandée. Un potentiomètre est un élément résistif possédant trois bornes (Figure II.12):

- Deux correspondent aux extrémités du corps de la résistance,
- La dernière correspond au curseur qui peut se déplacer sur le corps de la résistance.

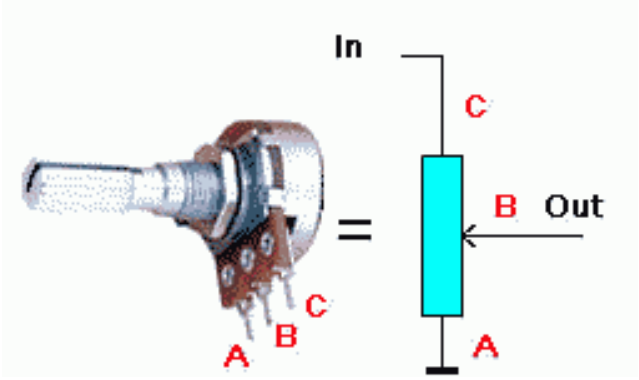


Figure II.12. Un potentiomètre et son symbole.

❖ Caractéristiques d'un potentiomètre

Loi de variation

Cette loi lie la variation en pourcentage de la résistance totale à la position du curseur. Dans la plupart des cas cette loi est linéaire.

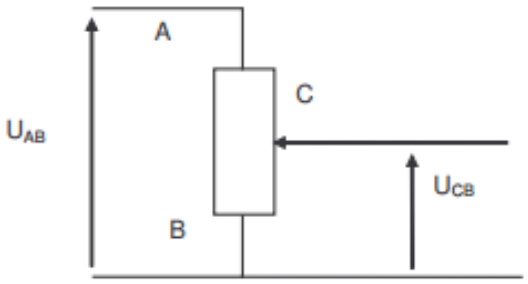


Figure II.13. Le montage d'un potentiomètre.

Si on appelle α la position du curseur, la différence de potentiels U_{CB} est nulle si α=0 (curseur en B), égale à U_{AB} si α=1(curseur en A), comprise entre 0 et U_{AB} lorsque le curseur est dans une position intermédiaire ($0 \leq \alpha \leq 1$) (Figure II.13).

Nous avons utilisé des potentiomètres pour simuler le comportement du capteur de CO_2 et de l'anémomètre, car ce sont des capteurs très couteux et indisponibles sur le marché algérien.

II.6. Partie puissance

II.6.1. Le moteur DC

Le moteur à CC (figure II.14) est un actionneur électromécanique très utilisé notamment dans les applications à vitesse variable. L'avantage principal des machines à courant continu réside dans leur adaptation simple aux moyens permettant de régler ou de faire varier leur vitesse, leur couple et leur sens de rotation.

Les machines électriques à courant continu sont constituées : d'un stator et d'un rotor.

- Le stator qui est à l' origine de la circulation d'un flux magnétique longitudinal fixe, crée soit par des enroulements statiques (bobinage) soit par des aimants permanents ; à l`arrière du stator se trouve la partie porte balais et les balais assurant les contacts électriques avec le rotor. Il est aussi appelé inducteur.
- Le rotor est constitué d'un ensemble de bobines reliées à un collecteur rotatif. Le collecteur rotatif permet d'inverser la polarité du champ magnétique crée par le stator avant que celui-ci ne soit en phase avec celui crée par le rotor. Grace à ce dispositif, les champs rotoriques et statiques sont toujours en quadrature provoquant ainsi la rotation du rotor. Les enroulements rotoriques sont aussi appelés enroulements d'induit, ou communément induit.

Figure II.14. Un moteur à courant continue.

II.5.2. L'extracteur

Les extracteurs (figure II.15) permettent de forcer l'aération dans la serre afin de garder les paramètres climatiques (température, humidité et taux de CO_2 dans l'air) au voisinage des consignes choisies.

Dans la réalisation de notre prototype, nous avons utilisé un ventilateur d'alimentation de PC.

Pour la variation de la puissance de ventilation, nous contrôlerons la vitesse du moteur par commande MLI à l'aide d'un convertisseur statique continu-continu (hacheur).

Figure II.15. Type d'extracteur utilisé dans notre prototype.

II.6.3. La pompe d'arrosage

Une pompe est un dispositif permettant d'aspirer et de refouler un fluide, il existe une panoplie de pompes adaptées à chaque besoin, le choix d'une pompe s'effectue toujours selon son usage et la détermination des paramètres de performances

Pour notre système d'irrigation, nous avons besoin d'une pompe d'arrosage, c'est l'organe qui fournit au sol l'eau dont les plantes ont besoin pour garder l'humidité du sol au niveau de la consigne.

Pour notre prototype, nous avons utilisé une pompe d'essuie-glace pour voitures (figure II.16), alimentée par une source de tension 12v et contrôlée par un circuit de puissance L293D.

Figure II.16. Pompe d'arrosage utilisée.

II.6.4. Le servomoteur

Le servomoteur (figure II.17) est un moteur à courant continu ou un moteur sans balai (brushless), pour produire un mouvement précis en réponse à une commande externe, C'est un actionneur qui mélange l'électronique, la mécanique et l'automatique [40].

Un servomoteur est composé :

- d'un moteur à courant continu ;
- d'une boite à vitesse ;
- d'un capteur de retour de position (très souvent un potentiomètre) ;

- une carte électronique de commande.

Un servomoteur est capable d'atteindre des positions prédéterminées dans les instructions qui lui ont été données, puis de les maintenir.

Le servomoteur à l'avantage d'être asservi en position angulaire, cela signifie que l'axe de sortie du servomoteur respectera la consigne d'instruction que vous lui avez envoyée en son entrée [40].

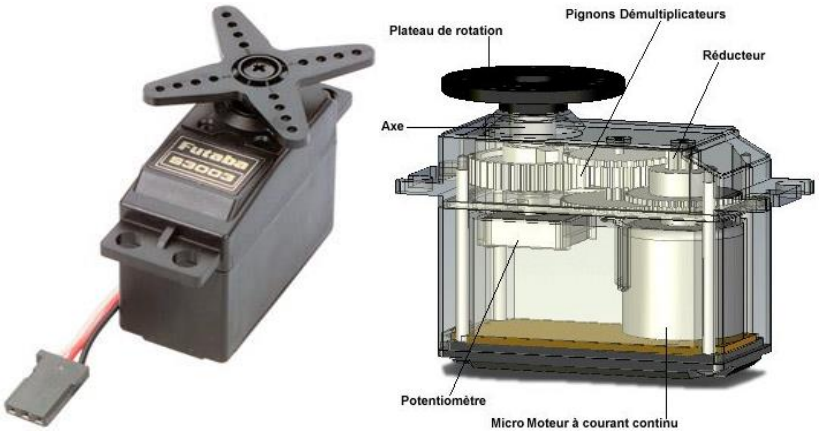


Figure II.17. Un servomoteur.

Les servomoteurs sont pilotés par un fil de commande et alimentés par deux autres fils. Habituellement, ces 3 fils sont rassemblés dans une prise au format standard.

Pour notre projet, nous avons utilisé le servomoteur Futaba S3003. Ses principales caractéristiques sont [40] :

- Dimensions : 40.4x19.8x36mm.
- Poids : 37.2g.
- Couple : 3.2kg.cm sous 4.8V.
- Couple : 4.1kg.cm sous 6V.
- Vitesse : 0.23s/60° sous 4.8V.
- Vitesse : 0.19s/60° sous 6V.
- Alimentation : 4.8 à 6V.

II.6.5. Circuit L293D

Le L293D est un double pont-H, grâce à ses 4 canaux, il est possible de l'utiliser pour commander quatre moteurs distincts (dans un seul sens.) Le circuit peut fournir jusqu'à 600mA par canal. En raccordant les sorties de façon appropriée, il est possible de constituer deux pont-H. Il est ainsi possible de commander deux moteurs distincts, dans les deux sens et indépendamment l'un de l'autre.

Le L293D est aussi connu pour être un excellent Pont-H facile à mettre en pratique. Avec deux signaux de commande Input 1 et Input 2 fournit par le microcontrôleur, il est possible d'inverser la direction du courant dans le pont-H et donc renverser le sens de rotation du moteur qui y est raccordé. En utilisant les différentes combinaisons d'Input 1 et Input 2 il devient possible de démarrer, stopper ou inverser le courant [33].

❖ Caractéristique du L293D

Les caractéristiques du L293D se présentent dans le tableau ci-dessous :

Tableau II.3. Caractéristique du L293D [33].

Caractéristique	Valeur	Caractéristique	Valeur
Nombre de pont H	2	Courant max	600(*2)
Courant de pointe max	1200	VS max	36V
VSS MAX	7V	Perte de tension	1.3 à 1.4 V

❖ Brochage du L293D

La figure II.18 représente le schéma de brochage du circuit L293D. Le tableau II.4 présente une description de chaque broche :

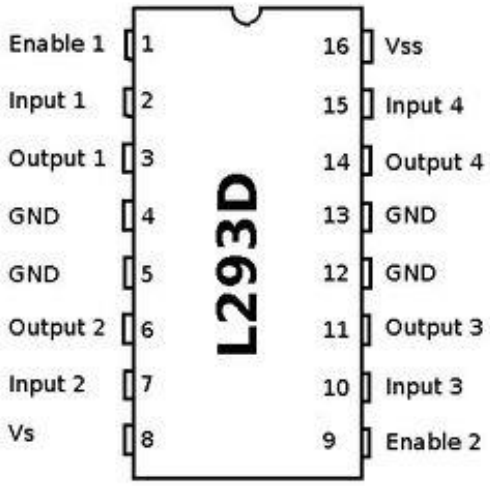


Figure II.18. Le schéma de brochage du L293D [33].

Tableau II.4. Brochage du L293D [33].

Broche	Nom	Description
1, 9	Enable1, 2	L'activation/désactivation du premier et second pont H.
2, 7, 10, 15	Input 1, 2, 3, 4	La commande des sorties des Ponts-H.
3, 6, 11, 14	Output1, 2, 3, 4	Raccorder aux charges (les moteurs).
4, 5, 12, 13	GND	Doit être raccordée à la masse (GND).
8	VS	Alimentation de puissance des moteurs.
16	VSS	Alimentation de la logique de commande (5V).

II.6.6. L'humidificateur

Nous avons utilisé un humidificateur ultrasonique, qui permet d'augmenter le pourcentage d'humidité présente dans l'air. Son principe de fonctionnement basé sur des ondes sonores qui, en faisant vibrer la membrane métallique de l'humidificateur, éclate l'eau en microgouttes qui permet de créer un nuage de vapeur fraîche comme le fog system : les humidificateurs sont très pratiques

pour refroidir l'air car le degré d'humidité dans une serre est inversement proportionnel à la température de l'air.

II.6.6.1. Description du Shengmeiyu Mini-humidifier

C'est un humidificateur domestique de petite taille et de faible puissance (figure II.19), il convient parfaitement aux serres de jardin comme notre serre prototype, son utilisation est très simple, il suffit de remplir son réservoir d'eau et de l'alimenter avec une source de tension.

Figure II.19. L'humidificateur Shengmeiyu Mini-humidifier.

II.6.7. Le relais

Le relais (Figure II.20) joue le rôle d'un interrupteur commandé. Un relais électromécanique est doté d'un bobinage en guise d'organe de commande. La tension appliquée à ce bobinage va créer un courant, ce courant produisant un champ électromagnétique à l'extrémité de la bobine (électro-aimant). Ce champ magnétique va être capable de faire déplacer un élément mécanique métallique monté sur un axe mobile, qui déplacera alors des contacts électriques.

Figure II.20. Le relai utilisé dans notre projet.

Les principales caractéristiques techniques du relai utilisé dans notre projet sont :

- Signal de commande 5V ;
- Maximum AC courant et voltage : 10A 250VAC ;
- Maximum DC courant et voltage: 10A 30VDC ;
- Il existe un contact normalement ouvert et un contact normalement fermé ;
- Pour que la bobine du relais soit excitée, vous devez avoir une entrée de 0 dans la broche de signal.

❖ Brochage du relai

- \+ : 5V source ;
- \- : Masse ;
- S : Signale de commande en 3.3V (dans notre cas) ;
- NC : normalement fermé ;
- NO : normalement ouvert.

II.6.8. L'alimentation

Nous avons besoin d'une alimentation avec les caractéristiques suivantes :

- Entrée : une tension alternative de 220V à 50Hz ;
- Sorties : trois tensions continuent de 12V et 5V pour les actionneurs et 3.3V pour la partie commande.

Remarque : La sortie 12V doit fournir un courant assez élevé pour alimenter notre chauffage à résistances.

Pour la réalisation de notre projet, nous avons utilisé une alimentation d'ordinateur de bureau. Elle présente les caractéristiques suivantes :

- Dimensions : 140 x 86 x 150mm ;
- Ventilateur 120mm ;
- Puissance nominale : 400 W;
- Tensions de sortie :
  - +3.3V à 13A ;
  - +5V à 28A ;
  - +12V à 14V ;
  - -12V à 0.5A.

Pour allumer l'alimentation, la ligne PS-ON (fil vert) doit être mis au zéro volt.

Figure II.21. Alimentation de PC utilisée.

II.7. Conclusion

La conception d'un projet peut être faite de plusieurs manières pour répondre à un même cahier des charges. Le choix de la solution adéquate est primordial et est fonction de plusieurs critères.

Pour concevoir le dispositif de commande de la serre agricole, nous avons opté pour la réalisation d'une carte électronique de commande autour du microcontrôleur ESP8266. La puissance de calcul de ce microcontrôleur (32bits) est largement suffisante pour gérer des algorithmes de régulation complexes.

Grace au bus I²C nous avons proposé des solutions qui nous permettront de gérer un nombre important de capteurs et d'actionneurs, avec la possibilité d'ajouter un affichage LCD.

Les composants électroniques de la serre agricole étant choisi, on va entamer l'étude des méthodes de commande et de régulation pour le contrôle du système dans le prochain chapitre.

CHAPITRE 3

Commande et régulation automatique

III.1. Introduction

Dans ce chapitre nous allons entamer en premier lieu l'étude des commandes automatiques utilisées dans ce domaine, aussi nous allons effectuer une comparaison théorique entre les différentes méthodes de commande.

Afin d'implémenter la commande floue dans notre système, nous allons utiliser la Toolbox fuzzy logic de Matlab qui nous permettra de créer les règles d'inférences floues et de simuler les valeurs de sorties selon les valeurs d'entrées du système.

Enfin nous allons simuler un sous-système proche de notre serre prototype sous Matlab Simulink pour visualiser l'évolution des paramètres de température avec et sans régulation.

III.2. La régulation automatique

La régulation automatique est la technique utilisée pour contrôler une ou plusieurs grandeurs physiques d'un système telles que la température, l'humidité, la pression, le pH, la concentration de co2, ….etc. en vue d'en imposer le comportement et de maintenir ces grandeurs à des niveaux prédéfinis dans un environnement perturbé. (Figure III.1) [41].

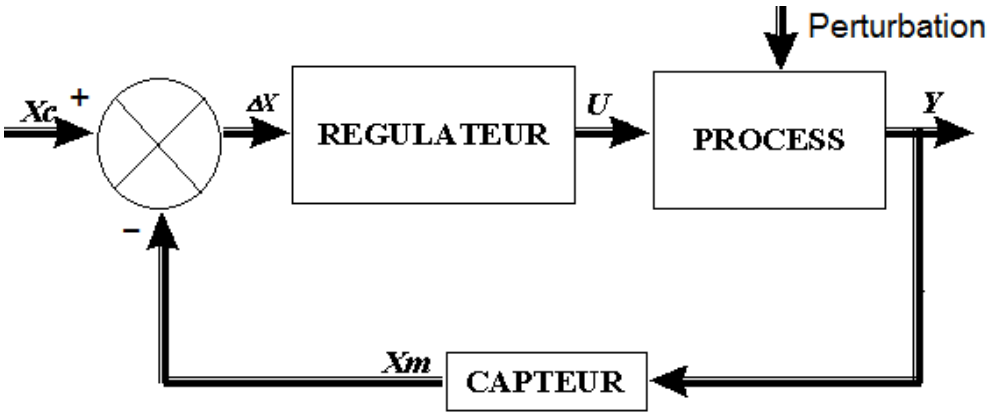


Figure III.1. Présentation générale d'une régulation en boucle fermée [41].

III.2.1. Notion de système

Un système ou processus peut être défini comme un ensemble d'éléments exerçant collectivement une fonction déterminée. Un système communique avec l'extérieur par l'intermédiaire de grandeurs, fonction du temps, appelées signaux. (Figure III.2)

Un système peut être représenté schématiquement de la manière suivante:

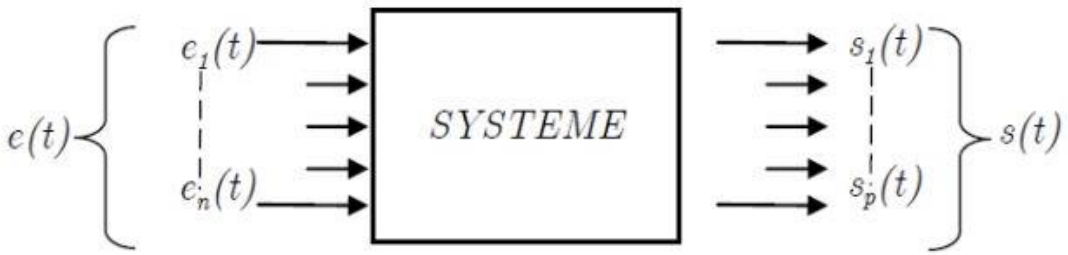


Figure III.2. Présentation générale d'un système en boucle ouverte [41].

Où e(t), est un ou plusieurs signaux d'entrée (excitation, cause ou sollicitation) et s(t) est un ou plusieurs signaux de sortie (ou réponses), t étant la variable temps. En général, ces signaux (d'entrée et de sortie) ne sont pas de même nature. Les grandeurs d'entrée sont les grandeurs qui agissent sur le système. Il en existe deux types :

- Commandes : Celles que l'on peut maîtriser.

- Perturbations : Celles que l'on ne peut pas maîtriser.

Les signaux de sortie d'un système sont aussi appelés réponse du système.

Pour observer les grandeurs de sortie, on utilise des capteurs. C'est l'information de ces capteurs qui va permettre d'élaborer la commande [41].

III.2.2. La régulation numérique

En régulation numérique, le régulateur est réalisé sous la forme d'un algorithme de traitement, programmé en langage évolué, s'exécutant à intervalles réguliers h [s]. « h » est la période échantillonnage. Cela signifie que la grandeur réglée y(t) est échantillonnée (figure III.3).

Lorsqu'une conversion A/N est effectuée aux intervalles k.h, l'algorithme du régulateur est alors exécuté et délivre une grandeur de commande u(k.h) également à intervalles réguliers h.

L'avantage principal de la régulation numérique est la souplesse d'emploi [41].

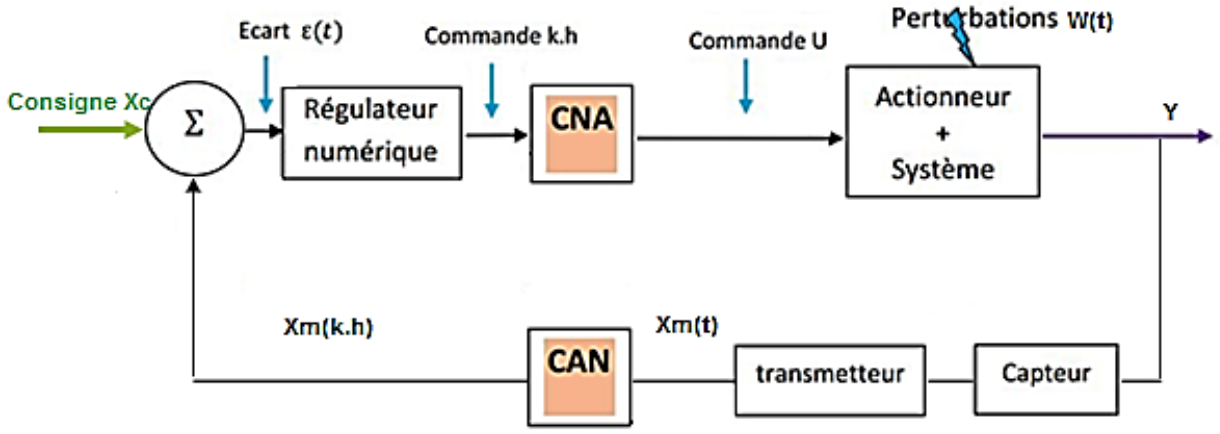


Figure III.3. Représentation par schéma fonctionnel du mode de fonctionnement d'une régulation numérique.

III.2.3. Régulation Tout Ou Rien

La plus simple des techniques de contrôle est la régulation Tout Ou Rien (TOR). Elle est utilisée quand la dynamique du procédé est très lente (grande constante de temps). Cette technique de régulation est considérée comme une régulation discontinue car la commande envoyée aux actionneurs varie instantanément. Sa réalisation impose de se fixer une limite inférieure et une limite supérieure de la grandeur réglée. Lorsque la mesure atteint la limite inférieure, l`actionneur, commande par le régulateur TOR ou tout simplement un relais, prend une position particulière (arrêt ou marche pour un chauffage, ventilateur,…). La figure III.4 illustre l'organigramme de l'algorithme TOR [42].

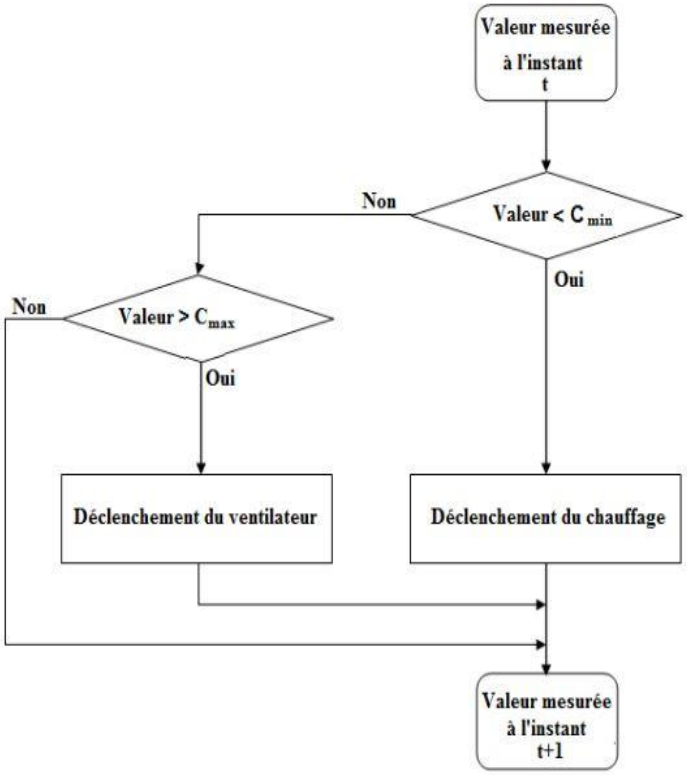


Figure III.4. Organigramme d'un exemple de régulation TOR. Légende: Cmin : consigne minimale du paramètre climatique; Cmax : consigne maximale du paramètre climatique.

Cette régulation est tout à fait acceptable pour les systèmes thermiques stables de faible puissance ou de forte inertie thermique. Il en résulte des permutations moins nombreuses du régulateur et de l'organe de puissance. Le système devient moins précis. Ce qui montre bien la nécessité du recours vers les régulateurs à commande variable car ils permettent de rationnaliser l'énergie utile pour commander les actionneurs [43].

III.2.4. Régulation PID

Le régulateur PID (proportionnel, intégrateur, dérivateur ou proportionnel, intégral, dérivé) est un système de contrôle, permettant d'effectuer un asservissement en boucle fermée d'un système industriel ou « procédé » (figure III.5). C'est le régulateur le plus utilisé dans l'industrie, et il permet de régler un grand nombre de grandeurs physiques. Le contrôleur fournit un terme proportionnel (Kp), un terme intégral (Ki) et un terme dérivatif (Kd) [43].

Ce contrôleur a une fonction de transfert :

$$G_c(s) = K_P + \frac{K_i}{S} + K_d S$$

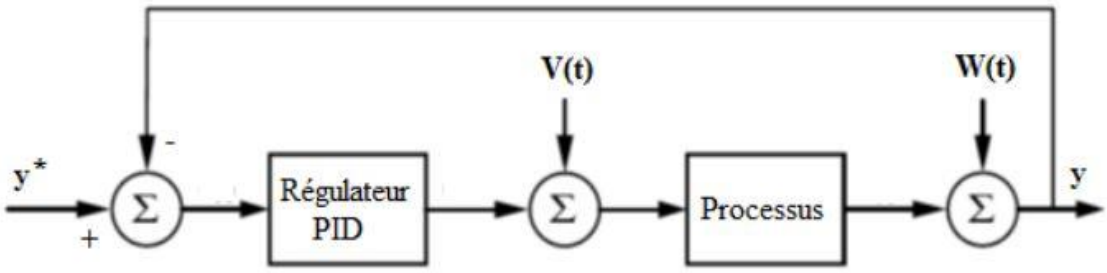


Figure III.5. Schéma fonctionnel d'un processus réglé par un régulateur PID classique où on y trouve les perturbations V(t) et W(t) [43].

III.2.5. Régulation par logique floue

Le climat à l'intérieur d'une serre agricole est un système dynamique très complexe, la température et l'humidité à l'intérieur sont non linéaires et fortement couplées par les lois de la thermodynamique [45].

❖ Définition

La logique floue est une branche des mathématiques et, à ce titre, toute une série de notions fondamentales sont développées. Ces notions permettent de justifier et de démontrer certains principes de base. Dans ce qui suit, on ne retiendra que les éléments indispensables à la compréhension du principe du réglage par la logique floue [44], Ces éléments sont :

- les variables floues
- les règles d'inférences.

III.2.5.1. Variables floues

Contrairement aux variables binaires qui sont définies par les deux états « vrai » ou « faux », les variables floues présentent toute une gradation entre la valeur « vrai » et la valeur « faux ».

Dans une partition floue forte, chaque ensemble correspond à un concept linguistique (figure III.6), par exemple Très faible, Faible, Moyen, Elevé, Très élevé. Pour le raisonnement les variables sont manipulées par les termes linguistiques ainsi définis, les ensembles flous assurent la correspondance avec l'univers numérique.

Cette opération s'appelle la « fuzzification » qui a pour but de donner à la variable associée un sous ensemble flou particulier.

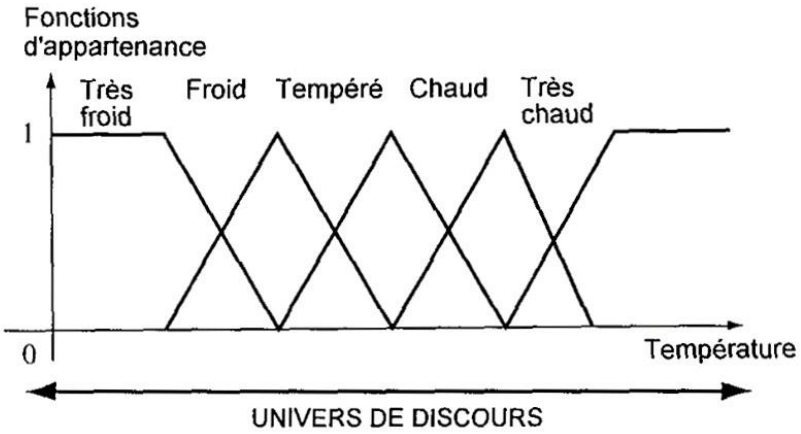


Figure III.6. Exemple de partition floue d'un univers de discours [45].

III.2.5.2. Règles d'inférence

On appelle règles d'inférence, l'ensemble des différentes règles reliant les variables floues d'entrée d'un système aux variables floues de sortie de ce système. Ces règles se présentent sous la forme :

Si condition 1 et/ou condition 2 (et/ou...) alors action sur les sorties.

Les différentes règles d'inférences produisent chacune, une valeur. Ces différentes valeurs doivent être combinées afin d'obtenir la (ou les) variable(s) de sortie. Ensuite la (ou les) variable(s) floue(s) de sortie doivent être converties en une grandeur de commande (débit, température...) afin d'être appliquée au système à régler. On appelle cette dernière étape : la défuzzification [44].

La figure III.7 présente la structure d'une commande floue.

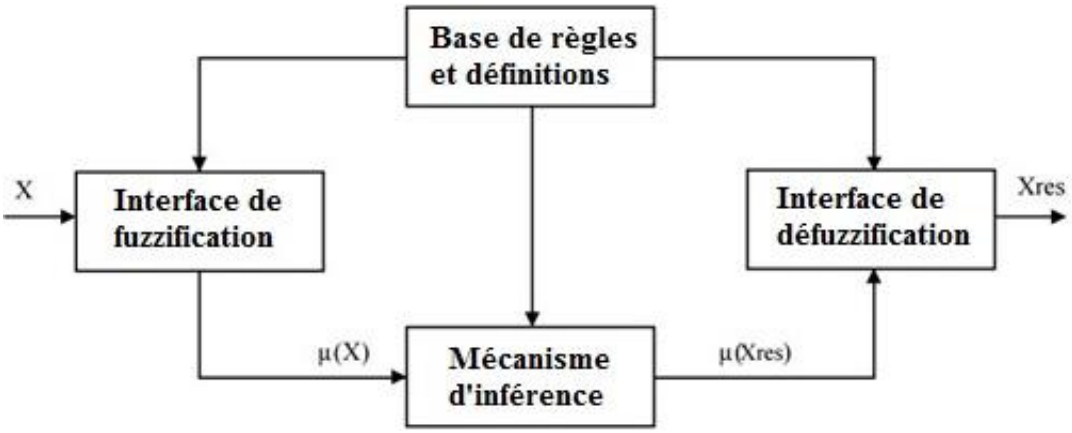


Figure III.7. Structure d'une commande floue [44].

III.2.5.3. La défuzzification

Il s'agit de convertir cette information en une grandeur physique. Plusieurs façons de faire peuvent être envisagées mais, en pratique on utilise surtout les deux méthodes suivantes :

- Défuzzification par calcul du centre de gravité.

- Défuzzification par calcul du maximum.

En général, c'est la défuzzification par calcul du centre de gravité qui est la plus utilisée [44].

❖ Défuzzification par calcul du centre de gravité

Il s'agit de calculer le centre de gravité de la fonction d'appartenance de la variable de sortie (figure III.8).

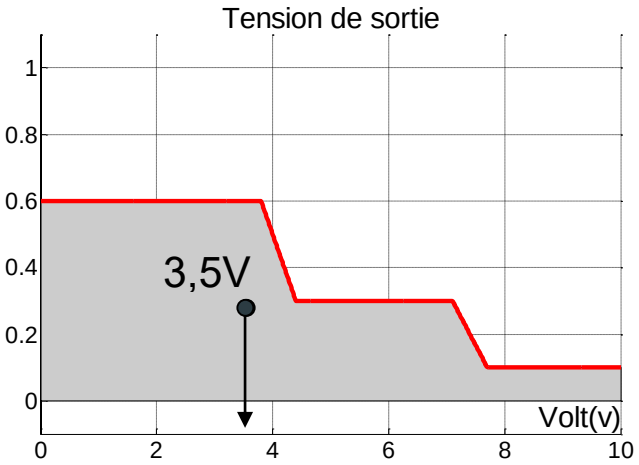


Figure III.8. Représentation de la méthode de défuzzification Centre de gravité (COG).

Le calcul du centre de gravité permet d'obtenir une seule valeur pour la grandeur de sortie. Son calcul est cependant relativement complexe puisqu'il nécessite le calcul d'une intégrale, ou dans le cas simple de fonctions d'appartenance en raies, d'une somme pondérée [44].

III.2.5.4. Les avantages et inconvénients de la logique floue

La logique floue permet de raisonner non pas sur des variables numériques, mais sur des variables linguistiques, c'est-à-dire, sur des variables qualitatives (grand, petit, moyen, loin, près, fort, etc.). Le fait de raisonner sur ces variables linguistiques va nous permettre de pouvoir manipuler des connaissances en langage naturel. Tout ce que l'on a à rentrer au système, ce sont des règles d'inférences exprimées en langage naturel.

La logique floue permet donc de contrôler des systèmes complexes non forcement modélisable de façon "intuitive". Néanmoins, cette méthode présente divers inconvénients. Tout d'abord, le fait d'exprimer ses connaissances sous la forme de règles en langage naturel (et donc qualitative) ne permet pas de prouver que le système aura un comportement optimal. Cette méthode ne peut donc pas garantir que le système soit stable, précis ou optimal, ni même qu'elle ne peut garantir que les règles entrées par le programmeur ne soient pas contradictoire.

Les performances sont donc mesurées a posteriori et ne peuvent être calculées a priori, les réglages se font donc par essais/erreurs [44].

III.2.6. Choix du régulateur

Il faut trouver des compromis, le régulateur idéal n'existe pas. En général on se fixe un cahier des charges à respecter sur la robustesse, en termes de dépassement, temps de réponse, et l'erreur statique suite à un changement de consigne ou de perturbations.

III.2.6.1. Etude comparative des régulateurs PID et flou

Le tableau ci-dessous compare le comportement des deux régulateurs PID et floue parmi d'autres types de régulation dans différentes situations :

Tableau III.1. Comparatif des principales familles de régulateurs [41].

Critères	Type de régulation				
	PID	Prédictif	Modèle interne	Flou	PID auto-adaptatif
Simplicité d'utilisation en monovariable	bien	médiocre	moyen	moyen	bien
Simplicité d'utilisation en multivariable	bien	bien	moyen	bien	médiocre
Mesures à échantillonnage lent et irrégulier	médiocre	moyen	bien	médiocre	médiocre
Performances en multivariable	médiocre	moyen	bien	moyen	médiocre
Processus à non-minimum de phase	médiocre	bien	bien	moyen	médiocre
Processus instable	moyen	moyen	moyen	médiocre	médiocre
Processus non linéaire au sens paramétrique	médiocre	moyen	bien	bien	moyen
Processus non linéaire à structure brutalement évolutive	moyen	moyen	bien	moyen	moyen
Processus à paramètres évolutifs dans le temps	médiocre	moyen	moyen	moyen	bien
Contraintes sur la commande	médiocre	bien	moyen	médiocre	médiocre
Contraintes sur une grandeur réglée	moyen	bien	bien	moyen	moyen
Processus non modélisable mathématiquement	moyen	médiocre	médiocre	bien	médiocre

La régulation par correcteur PID est sans doute l'une des régulations les plus utilisées dans le milieu industriel, mais elle présente de mauvaises performances dans le cas des processus non linéaires ou les processus dont on ne connaît pas la formule mathématique exacte.

L'utilisation de la commande floue est particulièrement intéressante, lorsqu'on ne dispose pas de modèle mathématique précis du processus à commander ou lorsque ce dernier présente de trop fortes non- linéarités ou imprécisions comme c'est le cas pour le climat intérieur d'une serre agricole [44].

Dans notre projet, nous appliquerons la commande floue aux volets et aux extracteurs pour abaisser la température et diminuer le taux d'humidité et du Co2 à l'intérieur de la serre. Le chauffage, l'humidification ainsi que l'arrosage des plantes se fera par régulateur Tout ou rien.

III.2.7. Présentation du Toolbox fuzzy logic

La Toolbox fuzzy logic de Matlab possède un éditeur qui permet de créer des systèmes d'inférence flous : des FIS (pour fuzzy inférence système). Cette boite à outils permet de générer des fichiers (.fis).

Cette boite à outils possède 3 éditeurs (de fis. de règles et de fonction d'appartenance) qui permettent de saisir l'ensemble des données du FIS ainsi que 2 interfaces graphiques qui permettent de visualiser les inférences directement sur la base de règles, ainsi que des surfaces de contrôle (figure III.9) [48].

Figure III.9. Les différentes fenêtres du Toolbox fuzzy logic de Matlab [48].

III.2.7.1. Mise en œuvre de la commande floue

III.2.7.2. Recensement des variables

Nous avons cinq variables d'entrées : température intérieure, température extérieure, humidité, vitesse du vent, taux de CO2, et deux sorties : la vitesse d'extraction de l'air et l'angle d'ouverture du volet (figure III.10).

Pour des raisons qui tiennent à la technologie, chaque variable a sa propre gamme de variation (univers du discours) qui sera décrite dans la fenêtre range.

Pour les entrées :

- Température intérieure varie de 0 à +50 °C.
- Température extérieure varie de 0 à +50 °C.
- Humidité varie de 0 à +100%.
- Taux de CO2 de 0 à +1500 ppm.
- Vitesse du vent de 0 à +100 % (100% correspond à 50 km/h).

Pour les sorties :

- Vitesse d'extraction de l'air de 0 à +100% (100% correspond à la vitesse max du moteur).
- Angle d'ouverture du volet de 0° à 110°.

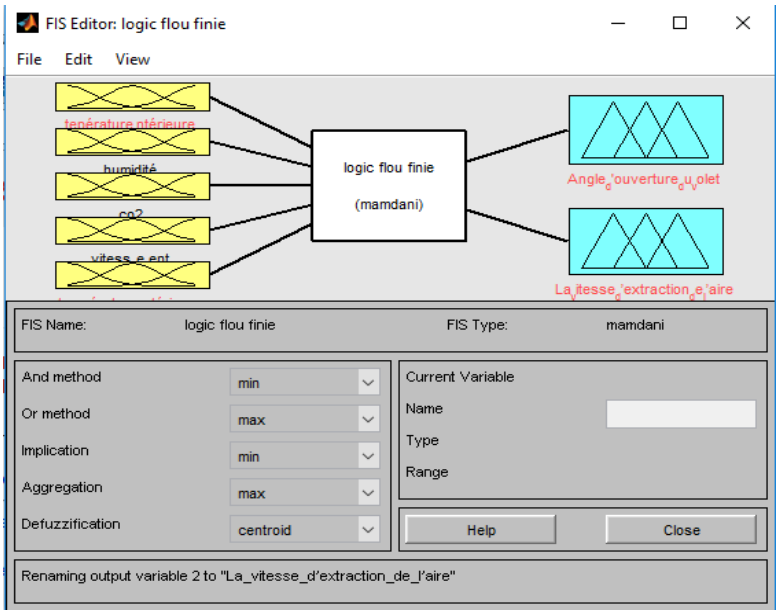


Figure III.10. Les entrées/sorties de notre système sur la fenêtre FIS Editor.

III.2.7.3. Fuzzyfication des variables

Chaque univers de discours est découpé en ensembles flous qui sont commodes. Pour avoir un élément central, il faut prendre un nombre d'ensembles impairs (2N+1 ensemble).

Nous définirons les plages de variations (les ensembles flous) et les fonctions d'appartenance pour les entrées et les sorties, et nous appellerons chaque partie de la fonction d'appartenance par un nom significatif. Nous prendrons comme exemple le cas de la fonction d'appartenance de température intérieure (figure III.11) :

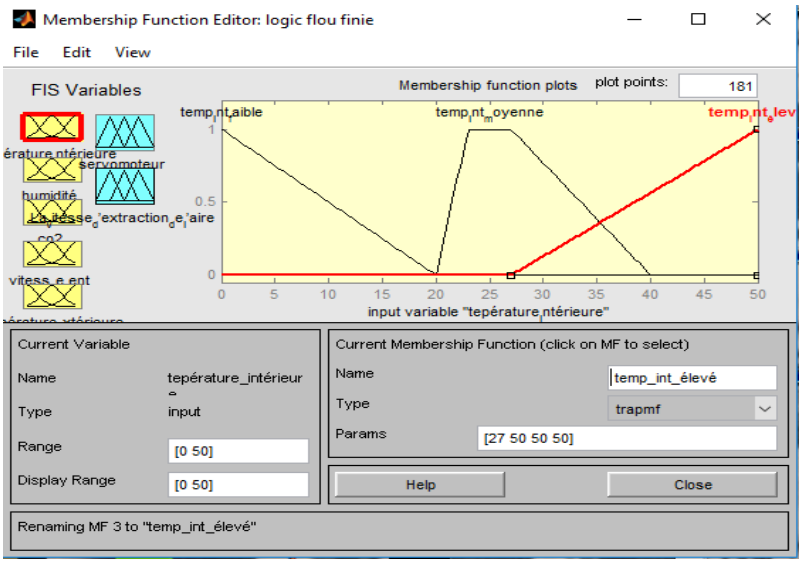


Figure III.11. La fonction d'appartenance de la température intérieure.

Pour les entrées nous avons choisi cinq ensembles flous de forme trapézoïdale :

- Température intérieure, trois ensembles flous trapézoïdaux : temp_int_faible, temp_int_moyenne, temp_int_élevée.

- Température extérieure varie de 0 à +50°C, trois ensembles flous trapézoïdaux : temp_ex_faible, temp_ex_moyenne, temp_ex_élevée.
- Humidité varie de 0 à +100%, trois ensembles flous trapézoïdaux : faible, normal, élevé.
- Taux de CO2 de 0 à +1500ppm, trois ensembles flous trapézoïdaux : CO2_faible, CO2_normal, CO2_élevé.
- Vitesse de vent de 0 à +100%, scinque ensembles flous trapézoïdaux : v_trés_faible, v_faible, v_moyenne, v_élevé, v_trés_élevé.

Pour les sorties nous avons choisi deux ensembles flous de forme trapézoïdale :

- La vitesse d'extraction de l'air : v_ex_trés_faible, v_ex_faible, v_ex_moyenne, v_ex_élevée.
- L'angle d'ouverture du volet : ang_trés_petit, ang_petit, ang_moyenne, ang_grand, ang_trés_grand.

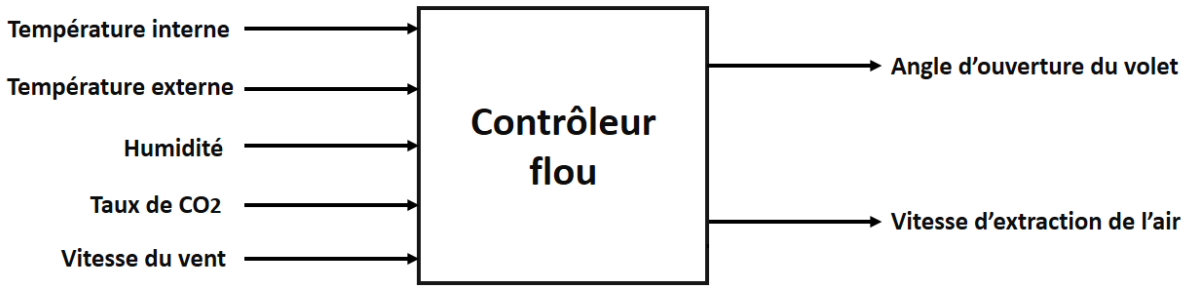


Figure III.12. Le régulateur flou avec ses 5 entrées et ses 2 sorties.

Le paramètre "Vitesse du vent" : il permet au régulateur flou de choisir entre la ventilation naturelle par l'ouverture des volets ou la ventilation forcée par l'extracteur d'air.

Si la vitesse du vent est élevée, le régulateur choisira l'extracteur d'air afin d'éviter l'endommagement des volets par des vents violentes. Par contre, si la vitesse du vent est faible, le régulateur choisira la ventilation naturelle pour économiser l'énergie.

Le paramètre "Température extérieure" : il permet au régulateur de comparer la température intérieur avec la température extérieur. Si la température extérieure est supérieure à celle de l'intérieur, le régulateur évitera d'utiliser la ventilation pour réguler la température.

III.2.7.4. Règles d'inférences

Après avoir définit les fonctions d'appartenance, nous avons implémenté 29 règles d'inférences, de telle façon à réaliser notre cahier des charges, par exemple si (température intérieure est élevé) ou (humidité est faible) ou (CO2 est faible) le régulateur va automatiquement laisser l'angle d'ouverture du volet « très petit » et la vitesse d'extraction de l'air « très faible » cela se traduit par la commande suivante:

Si (température_intérieure est faible) ou (humidité est faible) ou (CO2 est faible) alors (l'angle d'ouverture du volet est trés_petit) et (La vitesse d'extraction de l'air est trés_faible) (1).

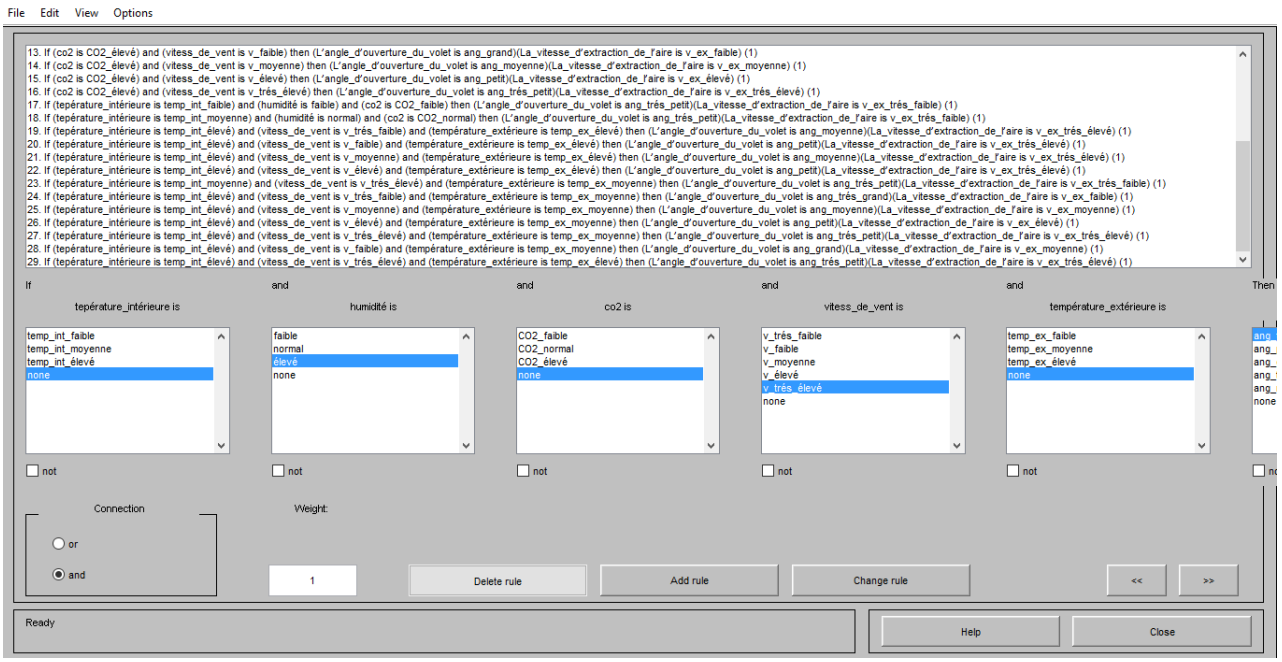


Figure III.13. Implémentation des règles d'inférence.

III.2.7.5. Exploitation du "Rule viewer"

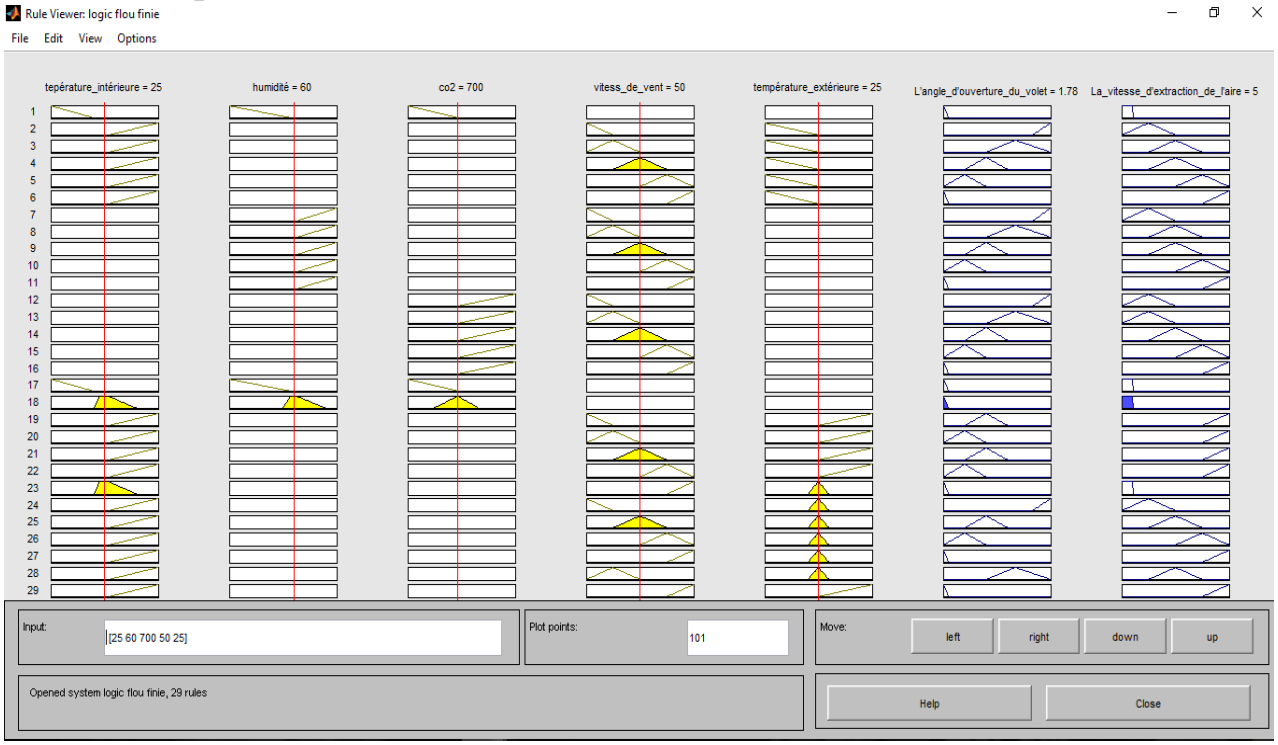


Figure III.14. Simulation des sorties du régulateur par le Rule viewer.

La fenêtre Rule viewer permet de simuler les sorties du système en variant les entrées, il se base sur les règles d'inférences implémentées dans le Rule editor.

En cliquant sur View (View rules), nous faisons apparaitre le Rule viewer qui montre que pour une température inférieure de 25 °C, humidité de 60%, taux de CO2 de 700 ppm, vitesse de vent de 50% et la température extérieure de 25°C, il faut fournir à la sortie, 1.78° pour l'angle d'ouverture de volet et 5% pour la vitesse d'extraction de l'air.

Il est possible de modifier la valeur de chacune des entrées en tirant le repère à la souris. On constate l'influence sur les sorties défuzzifiées.

III.2.8. Fonctionnement du régulateur TOR

Le régulateur TOR permet de contrôler les actionneurs suivants (Figure III.15) :

- Le chauffage ;
- L'humidificateur ;
- La pompe d'arrosage ;
- L'éclairage.

Tous ces actionneurs ont des valeurs de démarrage et d'arrêt selon la configuration choisie.

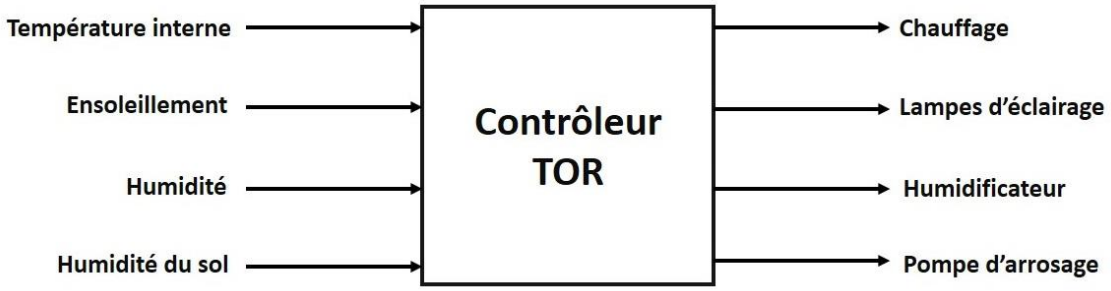


Figure III.15. Le régulateur TOR avec ses 4 entrées et ses 4 sorties.

Remarque : L'arrosage automatique s'active seulement quand le taux d'ensoleillement est en dessous d'une certaine valeur. Ceci est dans le but d'arroser les plantes pendant la nuit afin d'éviter l'évaporation de l'eau.

III.3. Simulation sous Simulink

Simulink est une plate-forme de simulation et de modélisation de systèmes dynamiques intégré à MATLAB. Elle fournit un environnement graphique et un ensemble de bibliothèques contenant des blocs de modélisation qui permettent le design précis, la simulation, l'implémentation et le contrôle des systèmes [49].

Remarque : Dans cette simulation, le système de ventilation sera remplacé par un bloc de climatisation.

III.3.1. Simulation du Tout ou Rien

Notre objectif est de réaliser la régulation de la température sous serre en utilisant le mode de contrôle TOR pour le chauffage et la climatisation.

Nous avons modifié le model "Thermal Model of a House" disponible sur MATLAB (sldmo_househeat.mdl) [37] selon les dimensions et les caractéristiques thermiques de notre serre

agricole. Nous avons aussi ajouté un bloc de climatisation. La figure III.16 représente le model serre agricole.

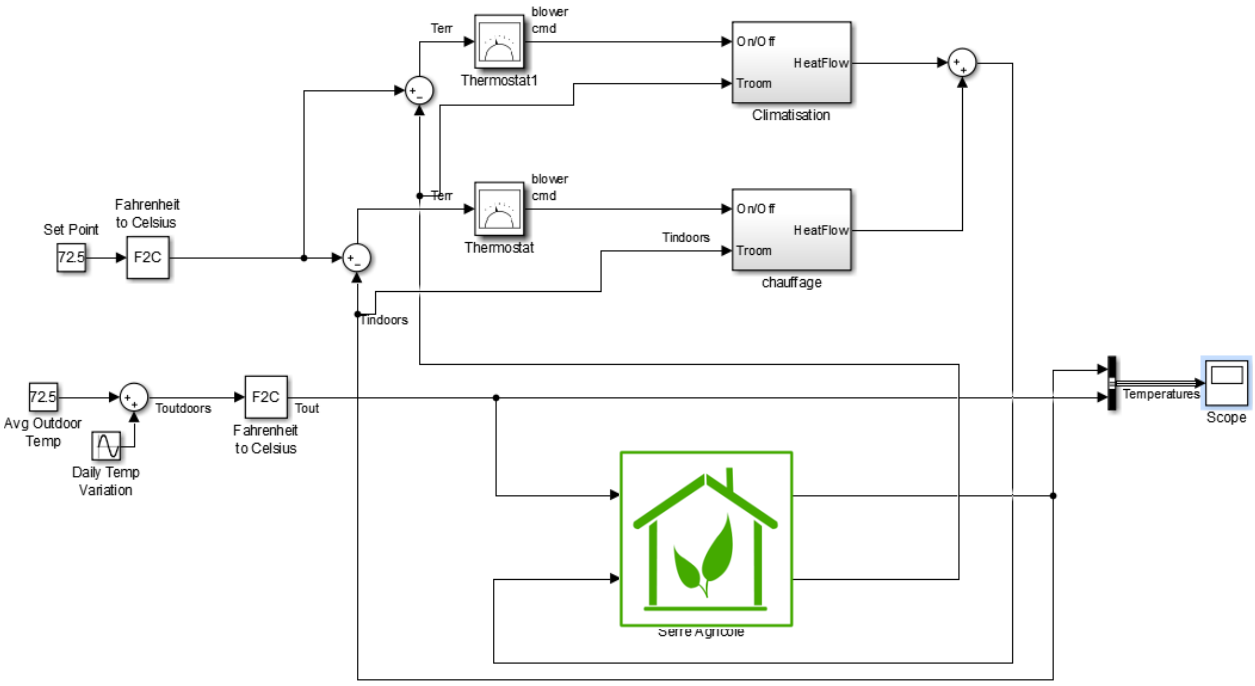


Figure III.16. Modèle Simulink utilisé pour la simulation de la commande TOR.

La figure III.17 représente la consigne et l'évolution des températures interne et externe sur une période de 48H. La figure III.18 représente l'évolution de la commande du chauffage et de la climatisation.

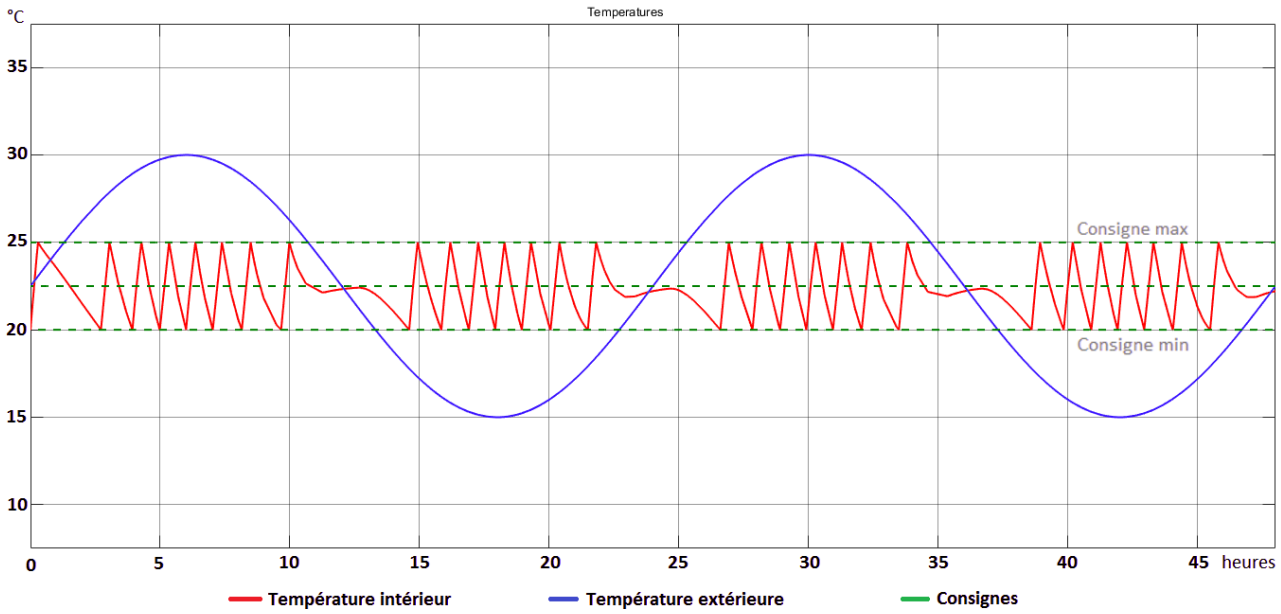


Figure III.17. Résultat de la simulation TOR.

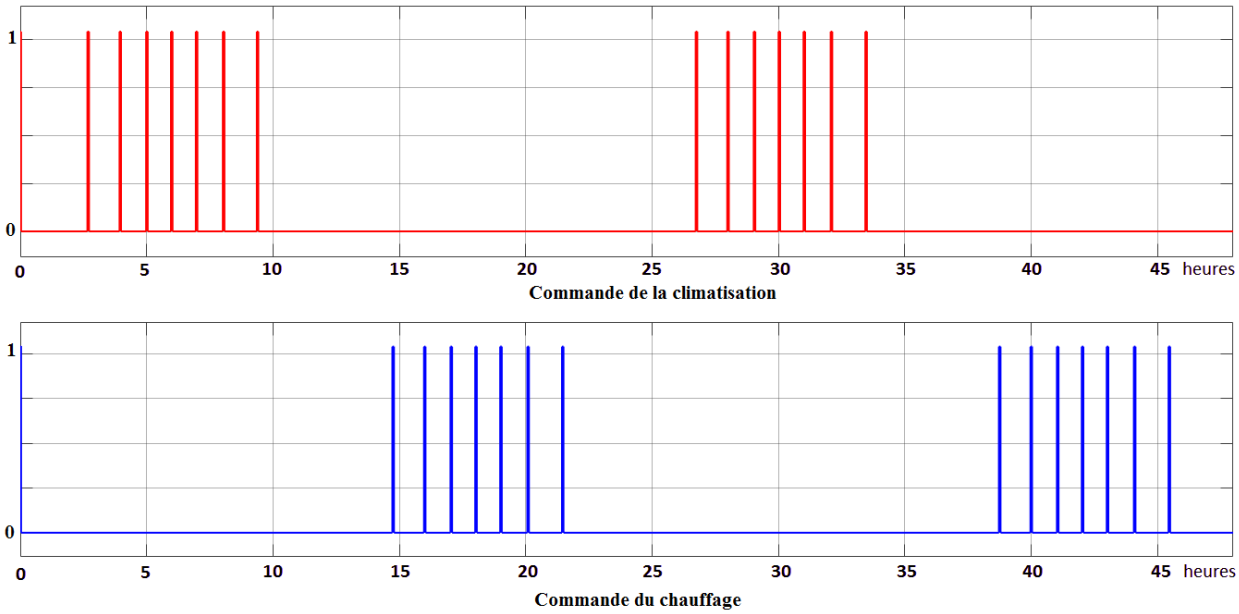


Figure III.18. Evolution de la commande TOR.

Afin de minimiser le nombre de commutations des actionneurs et d'économiser la consommation de l'énergie, nous avons choisis un intervalle d'hystérésis ± 2,5°C par rapport à la consigne de 22,5°C.

On peut remarquer que, la température interne suit la température externe, dès que la température interne dépasse la consigne de +2,5°C, le bloc de climatisation s'active pour abaisser la température. Pendant la deuxième alternance (cycle nuit), la température extérieure descend en dessous de la consigne de -2,5°C, le bloc de chauffage s'active pour augmenter la température de la serre.

L'intervalle de température entre 20°C et 25°C est idéal pour la culture de la tomate [23].

Remarque : L'effet de serre et l'humidité ne sont pas pris en compte dans cette simulation.

III.3.2. Simulation de la commande floue

Dans ce qui suit, nous allons intégrer au modèle Simulink précèdent, le régulateur flou que nous avons réalisé dans le Toolbox Fuzzy logic, pour cela nous avons suivis les étapes suivantes :

1- Exporter le fichier .fis vers l'espace de travail (Workspace).
2- Ajouter un bloc « Fuzzy Logic Controller » au modèle Simulink.
3- Introduire le nom du fichier .fis dans les paramètres du bloc Fuzzy.
4- Ajouter la sortie « extracteur » du régulateur flou comme entrée au model thermique de la serre agricole.

La figure III.19 représente la simulation de l'évolution des températures interne et externe par régulation floue sur une période de 48H.

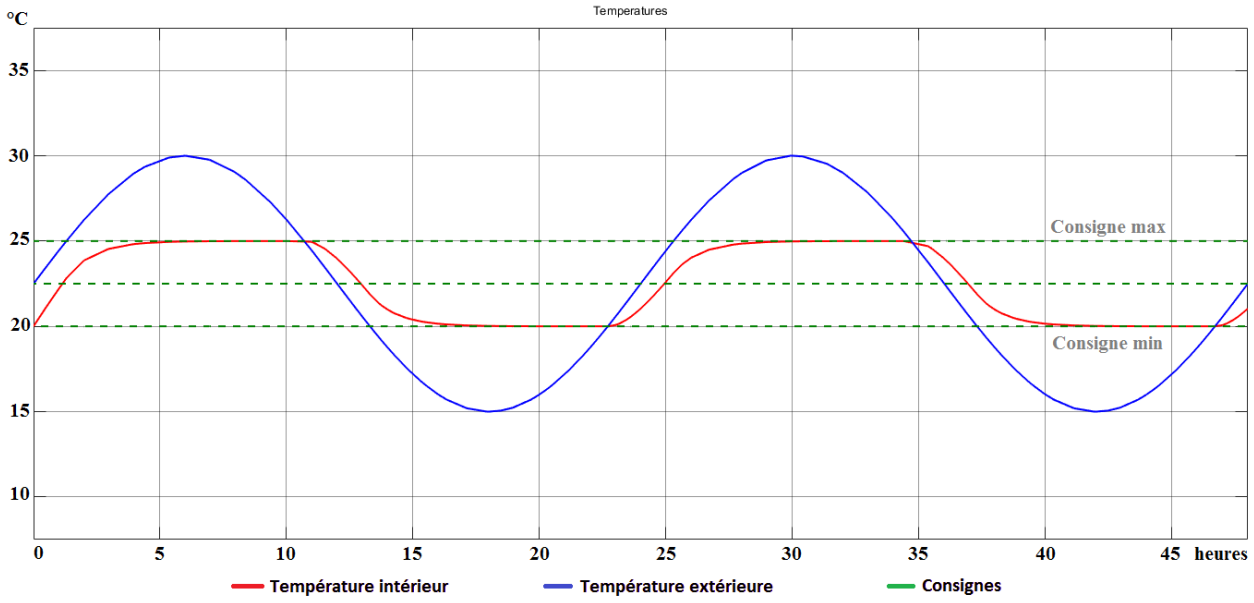


Figure III.19. Résultat de la simulation floue.

Nous remarquons que, pendant le cycle jour, la température externe est supérieure à la consigne, le régulateur flou agit sur le système de climatisation afin de réguler la température intérieure et l'empêcher de dépasser le seuil limite.

Pendant la deuxième alternance, la température externe diminue, dans ce cas, le chauffage commandé par le régulateur flou, maintient la température interne à la limite inférieure de l'intervalle de consigne.

Par manque de temps dans la réalisation de ce projet, nous n'avons pas pu réaliser le circuit hacheur qui permet la variation de la puissance du chauffage. Le régulateur flou va donc commander seulement le système de climatisation. La commande du chauffage se fera par régulation TOR.

Remarque : Afin de simuler seulement l'évolution de la température (figure III.20), les entrées température interne et température externe du régulateur flou sont raccordées avec les températures interne et externe du système mais les autres entrées du régulateur sont raccordées avec des valeurs constantes.

III.3.3. Simulation de la commande floue et TOR

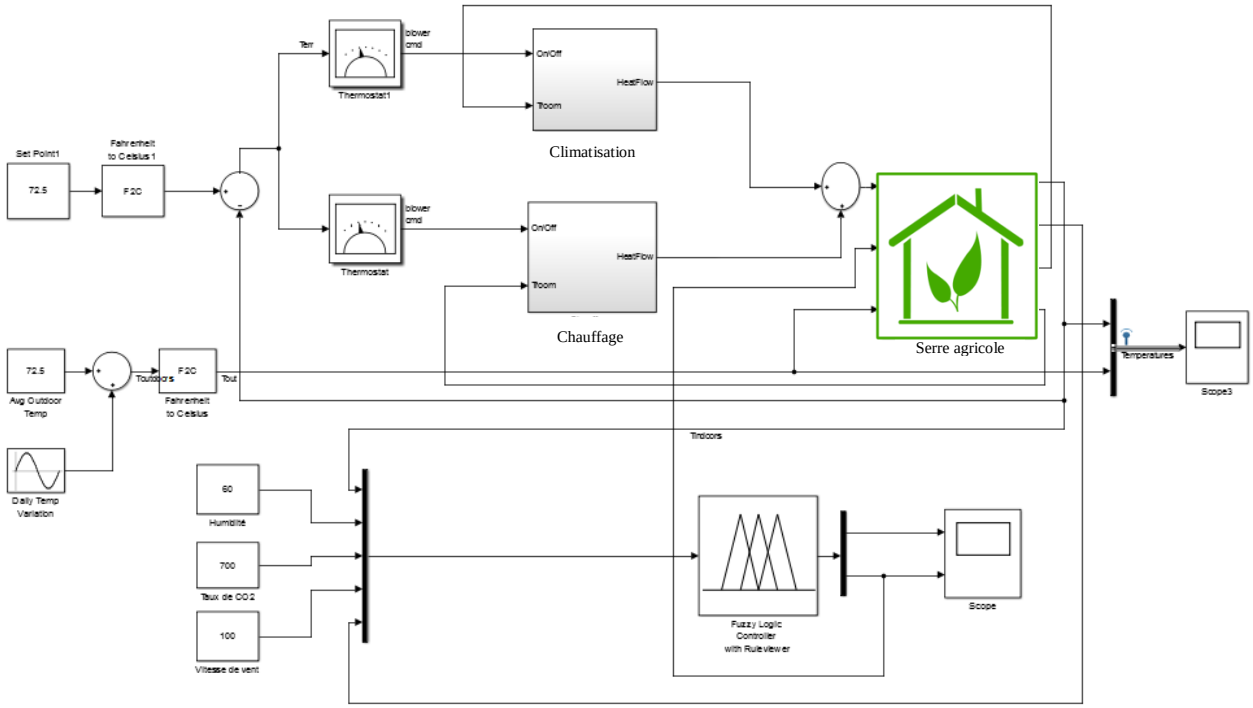


Figure III.20. Model Simulink avec régulateur flou.

La figure III.21 représente la simulation de l'évolution des températures interne et externe sur une période de 48H. Dans cette simulation les deux régulateurs (TOR et flou) fonctionnent en même temps, afin d'éviter toute interférence avec le régulateur TOR, nous avons configuré le régulateur flou avec une marge de +2.5°C par rapport à la consigne de 22.5°C.

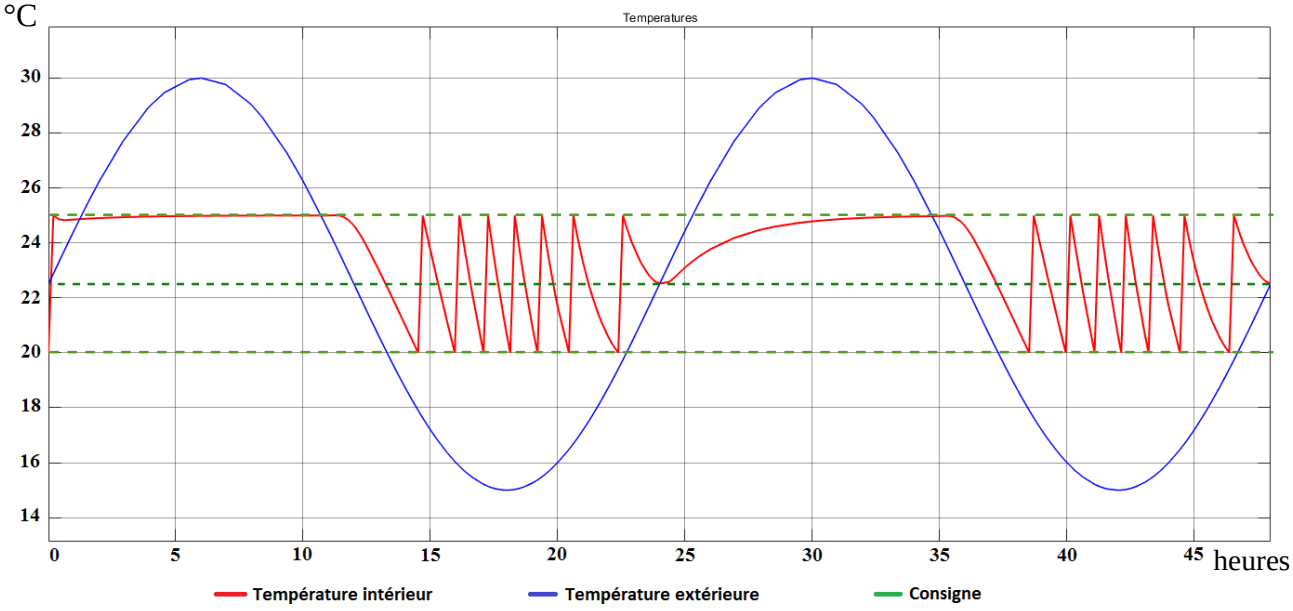


Figure III.21. Résultat de la simulation TOR et floue.

Nous remarquons que, pendant la première alternance (cycle jour), la température externe est supérieure à la consigne, le régulateur flou agit sur le système de climatisation afin de réguler la température intérieure et l'empêcher de dépasser le seuil limite (consigne +2,5°C).

Pendant la deuxième alternance (cycle nuit), le bloc de chauffage commandé par le régulateur TOR garde la température intérieure dans l'intervalle 20-25°C.

La figure III.22 représente l'évolution de la commande floue du système de climatisation sur la même période.

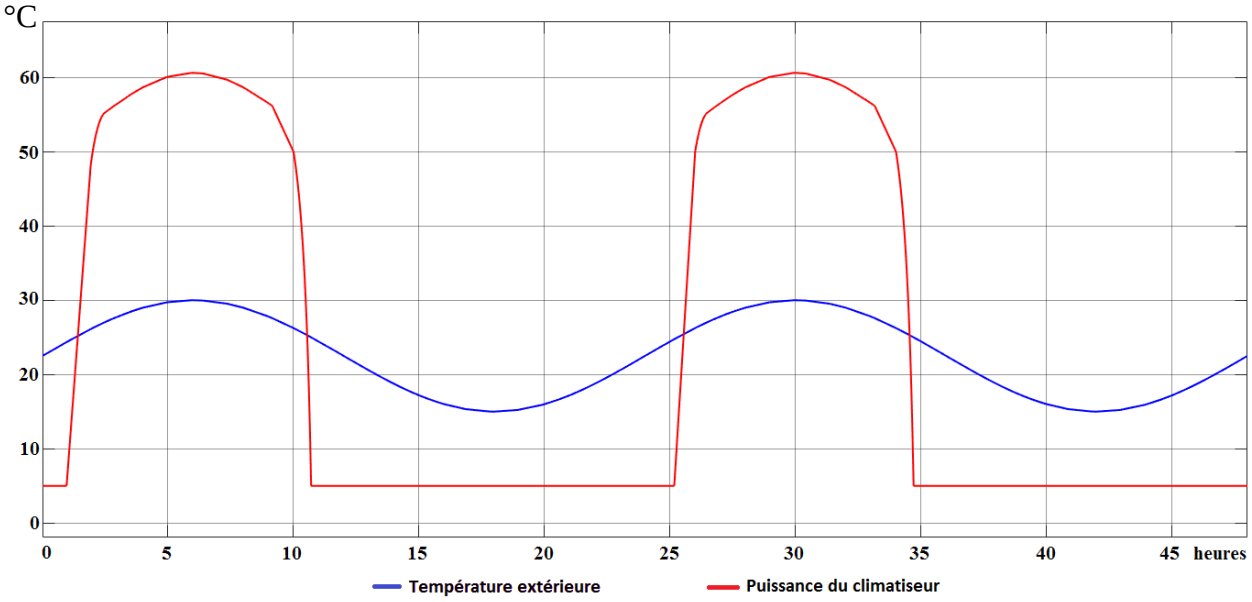


Figure III.22. Evolution de la commande floue du système de climatisation.

III.4. Implémentation de la logique floue

Afin d'implémenter l'algorithme de régulation floue dans le programme de la carte de commande, nous allons utiliser la bibliothèque eFLL (Embedded Fuzzy Logic Library) qui est écrite en langage C. Elle est conçue pour faciliter l'implémentation de la logique floue dans les systèmes embarqués.

Elle permet de gérer les méthodes d'implication Mamdani et Min-Max et la méthode de défuzzification par centre de gravité [46] (Voir l'annexe II pour le programme en langage C).

III.5. Conclusion

Dans ce chapitre, nous avons présenté dans un premier temps quelques notions sur la régulation automatique. Ensuite nous avons donné une brève description sur le fonctionnement des commandes TOR, PID et floue.

En second lieu, nous avons procédé à l'implémentation de la commande floue dans notre système à l'aide de la Toolbox fuzzy logic de Matlab.

Enfin nous avons simulé sous Matlab Simulink l'évolution de la température de la serre lors de l'implémentation de la commande TOR et la commande floue importé de la Toolbox fuzzy logic.

Dans le chapitre suivant, nous allons étudier plusieurs technologies de communication sans fil, ensuite nous allons présenter l'interface homme-machine qui nous permettra d'interagir avec notre système.

CHAPITRE 4

Réseau sans fil et logiciels

IV.1. Introduction

Dans ce chapitre, nous allons étudier et comparer les méthodes de communication sans fil ainsi que les moyens utilisés pour effectuer une communication via un serveur WEB.

Ensuite, nous allons présenter les outils qui nous permettrons de créer une interface homme-machine (IHM) sous forme d'une application WEB, ainsi qu'une description des différents outils utilisés pour son développement.

Enfin, nous allons présenter notre interface finale avec une description des différentes parties qu'elle comprend.

IV.2. Les réseaux sans fils

❖ Définition

Un réseau sans fil est un ensemble d'appareils connectés entre eux et qui peuvent envoyer et recevoir des données sans qu'aucune connexion « filaire » physique reliant ces différents composants entre eux ne soit nécessaire [50].

Il existe de multiples technologies de transmission sans fil ayant chacune leurs caractéristiques à savoir: la consommation, la portée, le débit, ... Ces systèmes se révèlent plus ou moins appropriés en fonction des cas de figure rencontrés.

IV.2.1. Avantages des réseaux sans Fil

- Réduit l'encombrement du câblage.
- Moins de risques de rupture de liens que dans le filaire (coupure du câble, détérioration desco nnecteurs et prises, etc.).
- Délais de mise en service du réseau plus faibles.
- Des économies à long terme (retour sur investissements)

IV.2.2. Etude comparative

Le tableau IV.1 récapitule certaines caractéristiques techniques de plusieurs technologies de communication sans fil.

Tableau IV.1. Comparatif des trois technologies de communication: Wifi, Bluetooth et Xbee [50].

Technologies de communication	Débit théorique (kbps)	Portée (m)	Consommation (mA)
Wifi norme 802.11b	11000	100	175
Bluetooth SN2100	3000	100	165
Xbee série 1	250	30 – 100	50

Les technologies de transmission sont utilisées différemment selon les besoins. En fonction des contraintes d'un projet, un concepteur sera amené à choisir l'un ou l'autre de ces systèmes.

Nous utiliserons dans notre projet la technologie Wi-Fi pour les raisons suivantes :

- C'est la technologie sans fil la plus fiable et la mieux adaptée pour relier des terminaux à internet [50].
- La facilité d'installation et d'utilisation.
- Le prix des microcontrôleurs utilisant cette technologie : 2$ pour l'ESP8266 contre 25$ pour le XBee XB24 utilisant la technologie ZigBee.

IV.2.3. Réseau local sans fil Wi-Fi

Le Wi-Fi est un ensemble de protocoles de communication sans fil régis par les normes du groupe IEEE 802.11. En 2015, les versions les plus utilisées de la norme IEEE 802.11 sont les 802.11g, 802.11n. Un réseau Wi-Fi permet de relier par ondes radio plusieurs appareils au sein d'un réseau informatique afin de permettre la transmission de données entre eux. La portée peut atteindre plusieurs dizaines de mètres en intérieur (généralement entre une vingtaine et une centaine de mètres) s'il n'y a aucun obstacle gênant (mur en béton par exemple) entre l'émetteur et l'utilisateur [51].

IV.2.3.1. Les normes Wi-Fi

Le tableau suivant présente les principales révisions de la norme 802.11 et leur signification :

Tableau IV.2. Les principales révisions de la norme 802.11 et leurs signification [51].

Norme	Description
802.11b	La norme 802.11b était la norme Wi-Fi la plus répandue, elle propose un débit théorique crête de 11 Mbit/s (6 Mbit/s réels) avec une portée pouvant aller jusqu'à 300 mètres (en théorie) dans un environnement dégagé. La plage de fréquences utilisée est la bande des 2,4 GHz.
802.11g	La norme 802.11g offre un plus haut débit (54 Mbit/s théoriques, 25 Mbit/s réels) dans la bande de fréquence des 2,4 GHz. La norme 802.11g offre une compatibilité ascendante avec la norme 802.11b.
802.11n	Le 802.11n a été conçue pour pouvoir utiliser les bandes de fréquences de 2,4 GHz ou 5 GHz, avec un débit théorique qui peut atteindre les 450 Mbit/s.

IV.2.4. Le modèle OSI

Le modèle OSI est un standard de communication conçu dans les années 1970, en réseau, de tous les systèmes informatiques. C'est un modèle de communications entre ordinateurs proposé par l'ISO qui décrit les fonctionnalités nécessaires à la communication et l'organisation de ces fonctions [52].

Le modèle OSI comporte 7 couches (figure IV.1) :

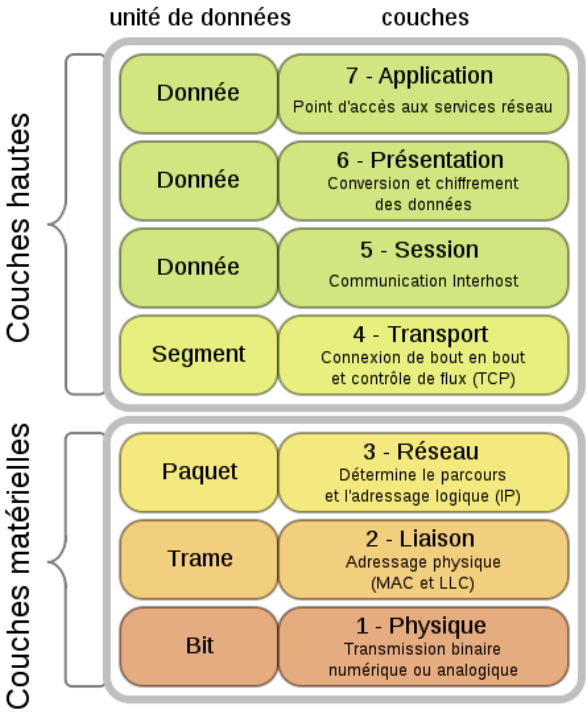


Figure IV.1. Diagramme du modèle OSI [52].

IV.2.5. Les protocoles de communication

Un protocole de communication définit l'ensemble des règles qui précisent les modalités de fonctionnement d'une communication entre deux ordinateurs. Voici quelques protocoles de communication utilisé dans notre projet :

1. Le protocole HTTP

C'est un protocole de communication client-serveur développé pour le World Wide Web. HTTPS (avec S pour secured, soit « sécurisé ») est la variante du HTTP sécurisée. Le HTTP est un protocole de la couche application. Il peut fonctionner sur n'importe quelle connexion fiable, dans les faits on utilise le protocole TCP comme couche de transport [52].

2. Le protocole TCP/IP

TCP/IP est un protocole de transport fiable, en mode connecté, c'est-à-dire qu'il permet l'établissement d'une session de communication entre deux parties qui veulent échanger des données. Il se base sur la notion d'adressage IP, c'est-à-dire le fait de fournir une adresse IP à chaque machine du réseau afin de pouvoir acheminer des paquets de données [52].

3. Le protocole UDP

L'User Datagram Protocol (UDP, en français protocole de datagramme utilisateur) est un des principaux protocoles de télécommunication utilisés par Internet. Il fait partie de la couche transport du modèle OSI comme le protocole TCP, son rôle est de permettre la transmission de données de

manière très simple entre deux entités, chacune étant définie par une adresse IP et un numéro de port [52].

4. Le WebSocket

WebSocket est une technologie évoluée qui permet d'ouvrir une connexion bidirectionnelle permanente entre un client et un serveur, afin de résoudre certains problèmes posés par le caractère unidirectionnel et déconnecté du protocole HTTP. Avec cette API on peut envoyer des messages à un serveur et recevoir ses réponses de manière événementielle sans avoir à aller consulter le serveur pour obtenir une réponse [52] [53].

IV.2.6. L'environnement client-serveur

Il désigne un mode de communication à travers un réseau entre plusieurs programmes : l'un, qualifié de client, envoie des requêtes ; l'autre ou les autres, qualifiés de serveurs, attendent les requêtes des clients et y répondent.

IV.2.6.1. Le Serveur HTTP

Un serveur HTTP ou (moins précisément) serveur Web est un logiciel servant des requêtes respectant le protocole de communication client-serveur HTTP, qui a été développé pour le World Wide Web [51].

Un « serveur web » peut faire référence à des composants logiciels (*software*) ou à des composants matériels (*hardware*) ou à des composants logiciels et matériels qui fonctionnent ensemble :

1. **Au niveau des composants matériels :** un serveur web est un ordinateur qui stocke les fichiers qui composent un site web (par exemple les documents HTML, les images, les feuilles de style CSS, les fichiers JavaScript) et qui les envoie à l'appareil de l'utilisateur qui visite le site. Cet ordinateur est connecté à Internet et est généralement accessible via un nom de domaine.
2. **Au niveau des composants logiciels :** un serveur web contient différents fragments qui contrôlent la façon dont les utilisateurs peuvent accéder aux fichiers hébergés. On trouvera à minima un serveur HTTP. Un serveur HTTP est un logiciel qui comprend les URL et le protocole HTTP (le protocole utilisé par le navigateur pour afficher les pages web).

IV.2.6.2. Utilisation d'un Raspberry PI comme serveur Web

Le Raspberry Pi est un nano-ordinateur monocarte à processeur ARM [54], qui peut être configuré comme serveur WEB avec toutes les fonctionnalités de base grâce au logiciel Apache, il a l'avantage de consommer très peu de courant et prend beaucoup moins d'espace en comparaison avec un ordinateur standard. Sa puissance de calcul est suffisante pour héberger une petite application web comme celle utilisé dans notre projet.

Cette solution est avantageuse d'un point de vue économique, il faut savoir que les services d'hébergement web ne sont pas gratuits. Contrairement au Raspberry qui demande juste une connexion internet.

IV.2.6.3. Client HTTP

Un client HTTP est un logiciel conçu pour se connecter à un serveur HTTP. Dans le cas le plus courant, il s'agit d'un navigateur Web. Les autres clients sont généralement des programmes automatisant les transferts. Il existe des bibliothèques logicielles dans de nombreux langages de programmation qui permettent de faire des requêtes HTTP [52].

IV.2.6.4. Navigateur Web

Un navigateur web (web browser en anglais) est un logiciel permettant d'accéder à une page web et de l'afficher sur l'écran de l'utilisateur. Un tel logiciel doit donc être capable de lire et d'interpréter des fichiers html. Les navigateurs web accèdent au web à travers le protocole de communication HTTP. On dit que c'est un "client HTTP", qui se connecte à un serveur.

IV.3. Les applications WEB

❖ Définition

Une application web (aussi appelée Web App, de l'anglais) est une application manipulable grâce à un navigateur web. De la même manière que les sites web, une application web est généralement placée sur un serveur et se manipule en actionnant des widgets (composant de l'interface graphique) à l'aide d'un navigateur web, via un réseau informatique (Internet, réseau local, etc.) [53].

IV.3.1 Avantages des applications WEB

Contrairement à un logiciel traditionnel, l'application web est placée sur un serveur et les utilisateurs y accèdent par un simple navigateur : il suffit d'une connexion à Internet. Il en découle les avantages suivants :

- Il n'est plus nécessaire d'installer un logiciel sur chaque ordinateur.
- Une application web peut être utilisée par différents types de terminaux (ordinateurs, laptops, tablettes, smartphones) (Figure IV.2).
- Pour mettre à jour l'application, il suffit de modifier l'application sur le serveur et tous les postes accèdent instantanément à la nouvelle version.

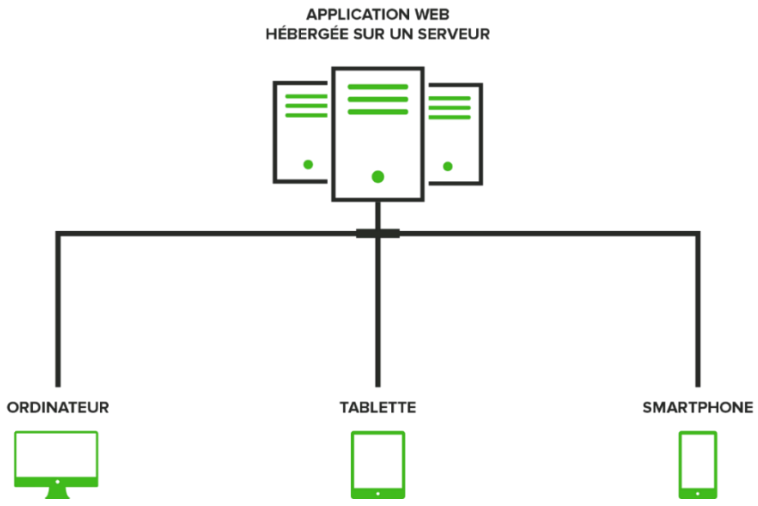


Figure IV.2. Utilisation d'une application web sur plusieurs types de terminaux sur le réseau.

IV.3.2. Utilisation de l'application Web

Afin d'interagir avec notre système et de gérer son état de façon optimale, il est nécessaire de développer une interface homme-machine qui se présentera sous forme d'une application web. Cette application devra offrir à l'utilisateur des fonctionnalités équivalentes aux logiciels directement installés sur les ordinateurs.

Les technologies utilisées pour développer les applications web sont les mêmes que celles employées dans la création des sites internet, voici la liste des outils utilisés pour la programmation de notre application :

1. **Langage HTML**

Le HTML (HyperText *Mark-Up Language*) est un langage dit de « structuration » dont le rôle est de formaliser l'écriture d'un document avec des balises de formatage. Les balises permettent d'indiquer la façon dont doit être présenté le document et les liens qu'il établit avec d'autres documents [53].

2. **Langage CSS**

Le terme CSS (Cascading Style Sheets) est un langage informatique utilisé pour mettre en forme les fichiers HTML. Ainsi, les feuilles de style, aussi appelé les fichiers CSS, comprennent du code qui permet de gérer le design d'une application en HTML [53].

3. **Langage Python**

Python est un langage de programmation objet et multiplateformes. Il favorise la programmation impérative structurée, fonctionnelle et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions [55].

4. **Flask**

Flask est une structure logicielle open-source de développement web en Python associé à un système de templates. Son avantage principal est d'être léger, afin de garder la souplesse de la programmation Python [55].

5. **Langage JavaScript**

JavaScript est un langage de programmation de scripts principalement utilisé dans les pages et applications web interactives, ce langage permet une interaction avec l'utilisateur en fonction de ses actions [53].

6. **MySQL**

MySQL est un serveur de bases de données relationnelles Open Source. Il fait partie des logiciels de gestion de base de données les plus utilisés au monde [53].

7. **Logiciel Apache**

Apache est un logiciel libre qui permet à un ordinateur (comme le Raspberry Pi) d'être utilisé comme un serveur http [56], il regroupe les applications suivantes :

- Le serveur web Apache.
- Le serveur de bases de données MySQL.
- L'outil phpMyAdmin permettant de gérer des bases MySQL.

IV.4. Présentation de l'interface Homme-Machine

Afin d'avoir le contrôle de notre serre à distance, nous avons développé une interface Homme-Machine sous forme d'application web (figure IV.3), ce type d'interface est très utilisé de nos jours car il offre la possibilité de contrôler un système depuis n'importe quel ordinateur connecté à internet sans avoir besoin d'installer des logiciels ou pilotes supplémentaires.

Pour avoir accès à cette application, il suffit juste d'ouvrir le navigateur web et d'y entrer l'adresse de l'application : dans notre cas ça sera l'adresse IP du serveur « 192.168.8.15 » car l'application est hébergée en local.

Figure IV.3. Page principale de notre interface Homme-Machine.

Apres avoir validé l'adresse, nous arrivons sur la page principale de notre interface Homme-machine, notre application est constituée en 3 parties : Tableau de bord, Commande et Configuration. Elles sont accessibles depuis le menu « 1 » (figure IV.3).

❖ Partie Tableau de bord

Cette partie nous donne beaucoup d'informations sur l'évolution des paramètres de notre système (figure IV.3):

Nous avons 7 gauges colorées qui nous donne la valeur actuelle de chaque paramètre climatique, ces gauges sont actualisées chaque seconde et change de couleur selon la valeur des consignes du régulateur.

A l'aide du menu numéro « 2 », nous pouvons sélectionner entre plusieurs graphes qui nous montrent l'évolution de notre système sur une plage horaire que nous pouvons choisir : 30 min, 1 heure, 6 heures, 1 jour, une semaine ou toute la durée de l'enregistrement (Figure IV.4).

En passant le curseur de la souris sur un point du graphe, un rectangle apparait avec la date et l'heure de l'enregistrement ainsi que la valeur détaillé des paramètres enregistrés.

Figure IV.4. Graphe qui montre l'évolution des paramètres de température intérieure et extérieure sans régulation.

❖ Partie Commande

Cette page nous permet de choisir entre la commande automatique par régulation ou la commande manuelle, elle est séparée en deux sections (Figure IV.5):

1. **Commande automatique :** Dans cette section, nous pouvons activer ou désactiver les régulateurs flous et TOR ainsi que l'enregistrement des paramètres sur la base de données.
2. **Commande manuelle :** Cette section nous permet de commander manuellement les actionneurs de la serre grâce au boutons colorés et aux curseurs situé au milieu. Ces boutons nous donnent l'état actuel de chaque actionneur par leur couleur : Vert pour éteint et orange pour actionné.

Lorsque la commande automatique est activée, les boutons de la section manuelle change de couleur selon l'état de chaque actionneur, ce qui nous donne l'état de fonctionnement global de notre serre.

Le voyant en haut à droite nous montre l'état de connexion de la carte de commande.

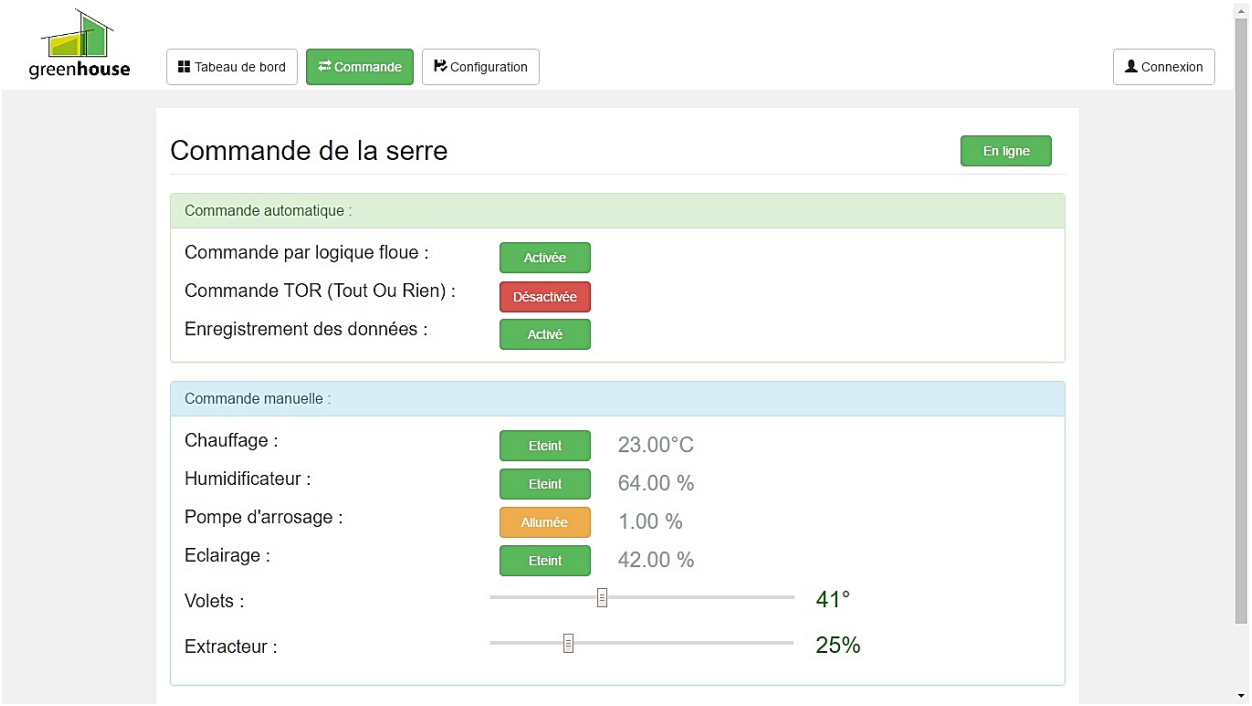


Figure IV.5. Partie Commande de la serre qui nous montre l'état de chaque actionneur.

❖ **Partie Configuration**

Cette page nous permet de configurer tous les paramètres de notre serre agricole : le type de plantation, les consignes de régulation, la période d'éclairage artificiel, la fréquence d'enregistrement…etc. (Figure IV.6)

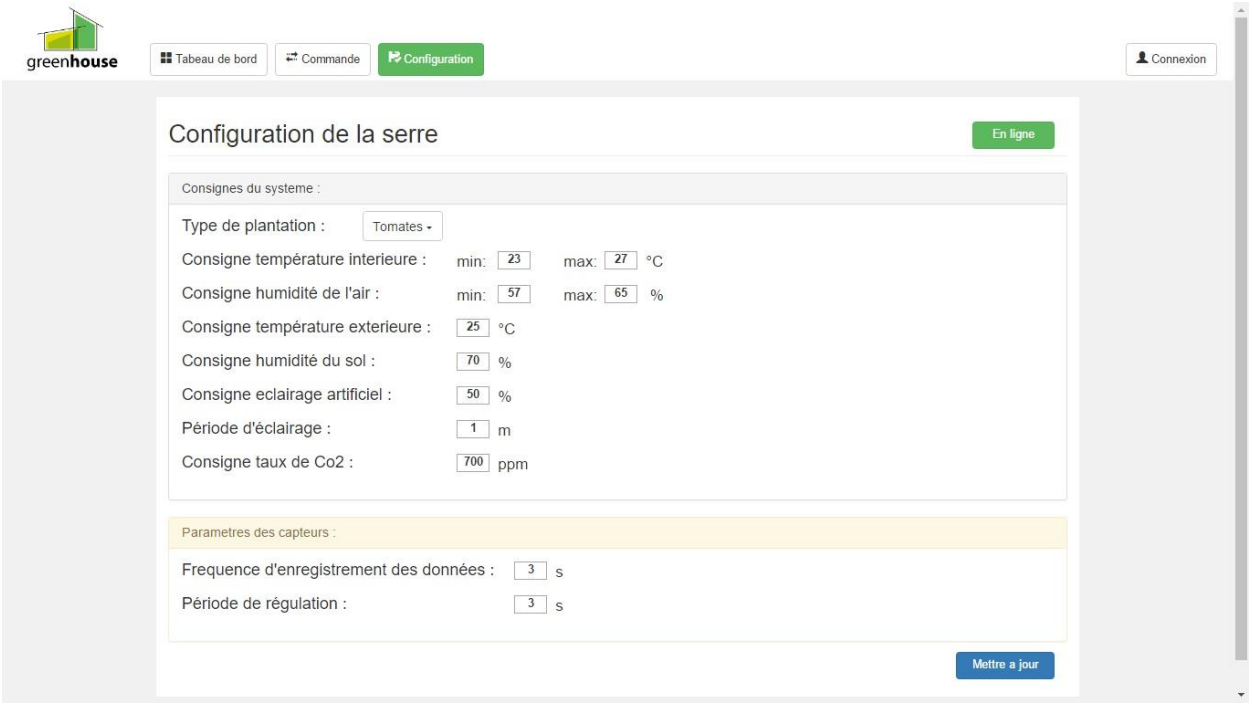


Figure IV.6. Page de Configuration de la serre.

Remarque : Les pages de Commande et de configuration sont protégées par identifiant et mot de passe afin de sécuriser notre système et d'empêcher les personnes extérieures d'y accéder.

IV.5. Conclusion

Dans ce chapitre, nous avons présenté les moyens de communication sans fil, par une étude comparative de quelques technologies de communication utilisées dans ce domaine. Ensuite nous avons défini quelques protocoles de communication utilisés dans notre projet.

Ensuite nous avons étudié les avantages de l'utilisation des applications Web dans la conception d'une interface homme-machine et les outils qui nous ont permis de la développer.

Enfin nous avons présenté notre interface finale avec une description des différentes parties qu'elle comprend.

Dans le chapitre suivant, nous présentons les étapes de la réalisation de notre prototype, étudier l'évolution des paramètres climatiques à l'intérieur et à l'extérieur de la serre et la réalisation PCB de la carte de commande.

CHAPITRE 5

Réalisation et tests

V.1. Introduction

Dans ce chapitre, nous présentons en premier lieu, les différentes étapes de la construction de la serre prototype, du montage de la structure jusqu'au placement des capteurs et actionneurs ainsi que la carte de commande et l'alimentation.

En second lieu, nous avons réalisé le schéma du circuit électronique à l'aide du logiciel EasyEDA avec quoi nous avons aussi réalisé le circuit imprimé de la carte de commande. Ensuite nous allons interpréter les résultats des tests expérimentaux réalisés et obtenus avec et sans contrôle climatique.

En dernier lieu nous allons entamer une étude socioéconomique afin de déterminer le cout de revient de notre projet. Nous terminerons par la présentation des organigrammes fonctionnels de la carte de commande et de la partie serveur.

V.2. Réalisation de la serre

V.2.1. Construction de la structure en bois

Nous avons conçu la structure de notre prototype de serre avec des baguettes de bois (section 20x20).

Figure V.1. La structure de notre serre prototype.

La figure V.1 représente la structure en bois de notre serre avec les dimensions 600x600x700, divisé en deux parties : une partie pour l'électronique (600x150x500) et l'autre partie pour la culture des plantes (600x450x700).

V.2.2. Montage des parois en plexiglass

Nous avons découpées 8 pièces en plexiglass que nous avons fixé sur la structure en bois, ces pièces vont constituer la paroi externe de la serre (figure V.2).

Figure V.2. Montage des parois en plexiglass.

V.2.3. Montage des capteurs et actionneurs

Nous avons installé les capteurs (DHT11, LM35, capteur d'humidité de sol, etc.), les actionneurs (pompe, servomoteur, humidificateur, etc.), l'éclairage et l'alimentation (Figure V.3).

Figure V.3. Montage des capteurs et actionneurs.

V.2.4. Montage de la partie électrique

La figure V.4 présente la partie électrique de notre serre, elle contient :

- La carte de commande.
- L'alimentation.
- Les circuits de puissance (Relais, L293D).
- La pompe d'arrosage.

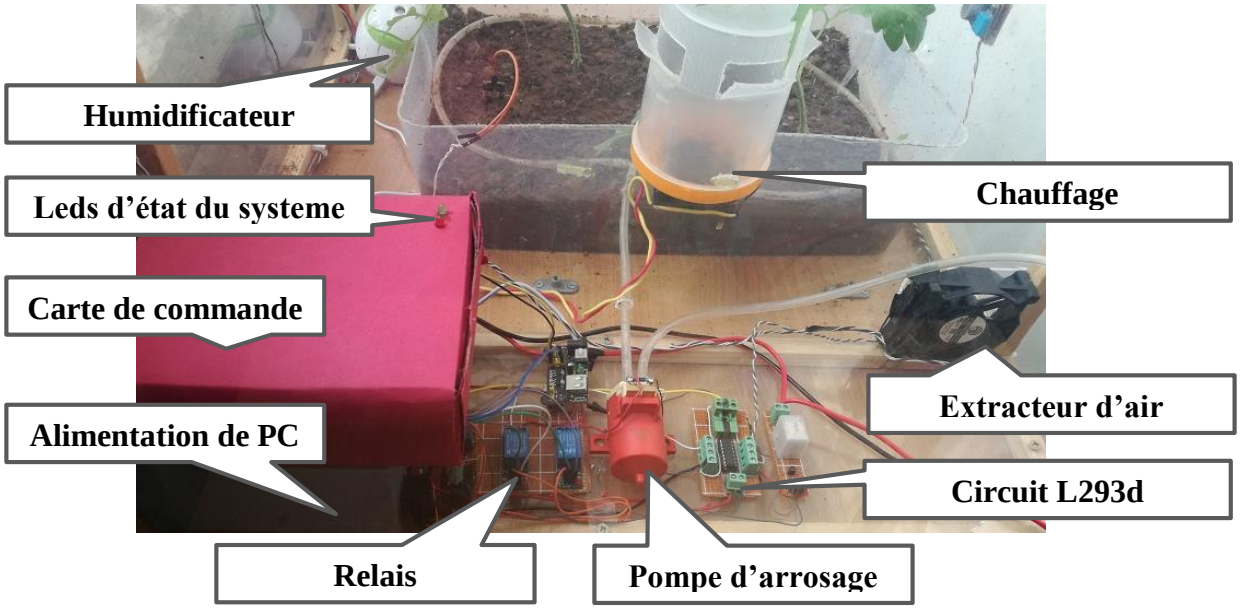


Figure V.4. Description de la partie électrique de la serre.

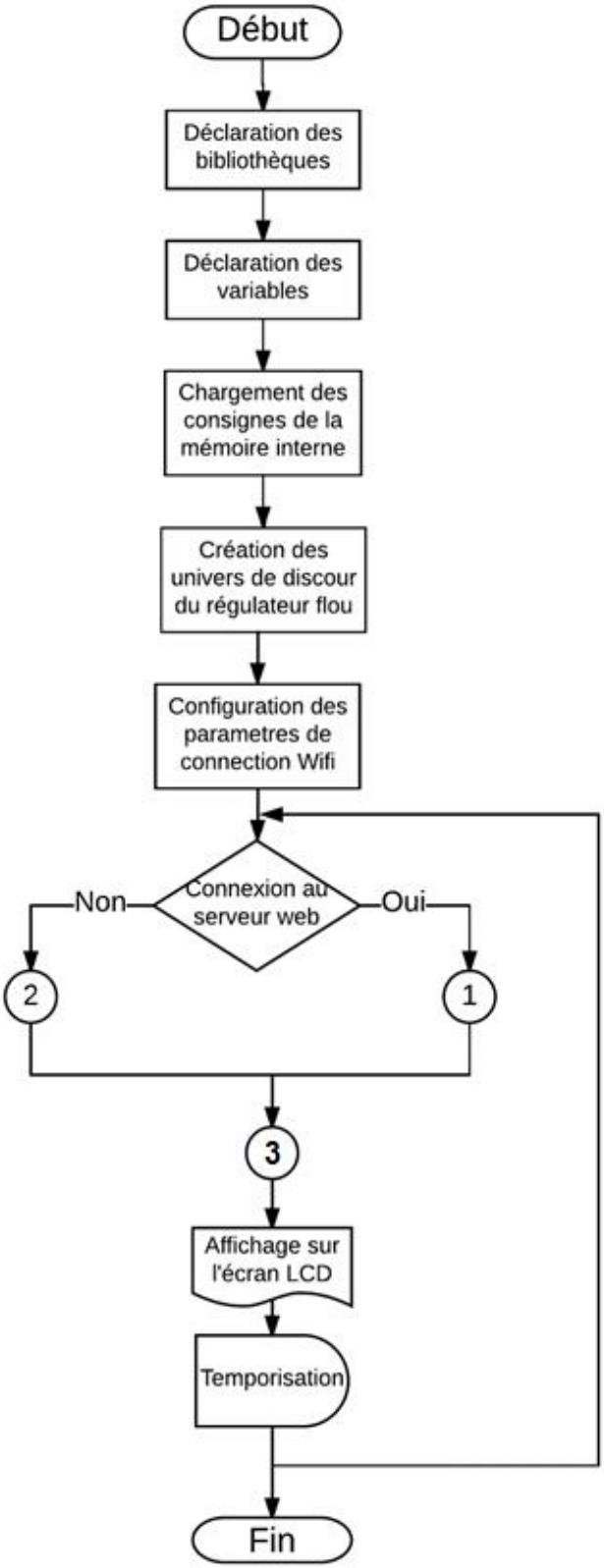
V.2.5. Plantation des tomates

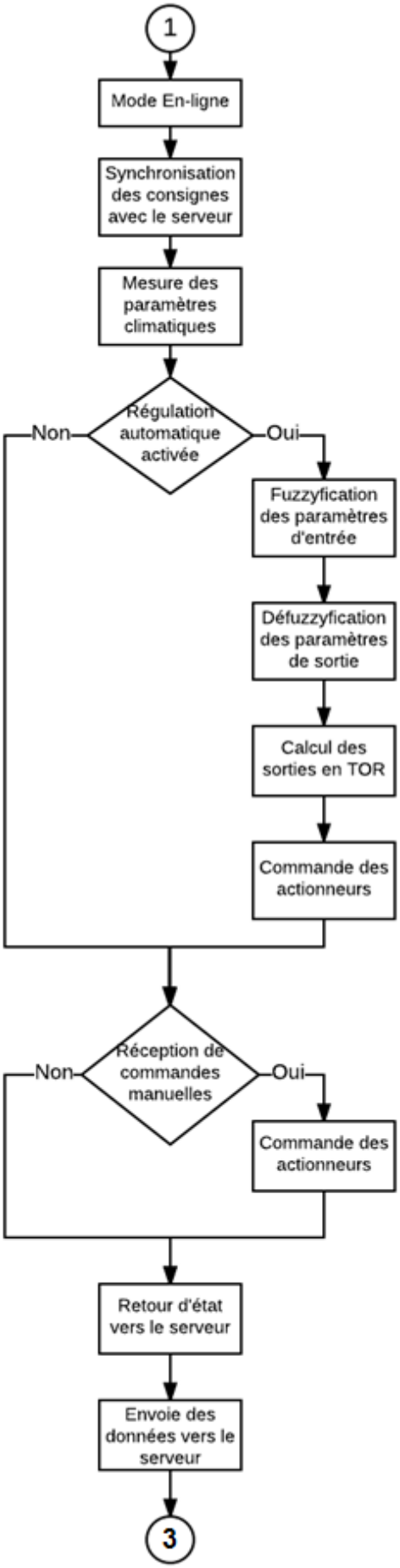
Figure V.5. Plantation des tomates le 04/05/2017.

V.3. Organigramme générale du système

V.3.1. Organigramme de la carte de commande

La figure V.6 présente l'organigramme du programme principal de la carte de commande. Elle peut fonctionner en mode "En ligne" ou "Hors ligne" selon la disponibilité du réseau.




1
Mode En-ligne
Synchronisation des consignes avec le serveur
Mesure des paramètres climatiques
Régulation automatique activée
Non
Oui
Fuzzyfication des paramètres d'entrée
Défuzzyfication des paramètres de sortie
Calcul des sorties en TOR
Commande des actionneurs
Réception de commandes manuelles
Non
Oui
Commande des actionneurs
Retour d'état vers le serveur
Envoie des données vers le serveur
3

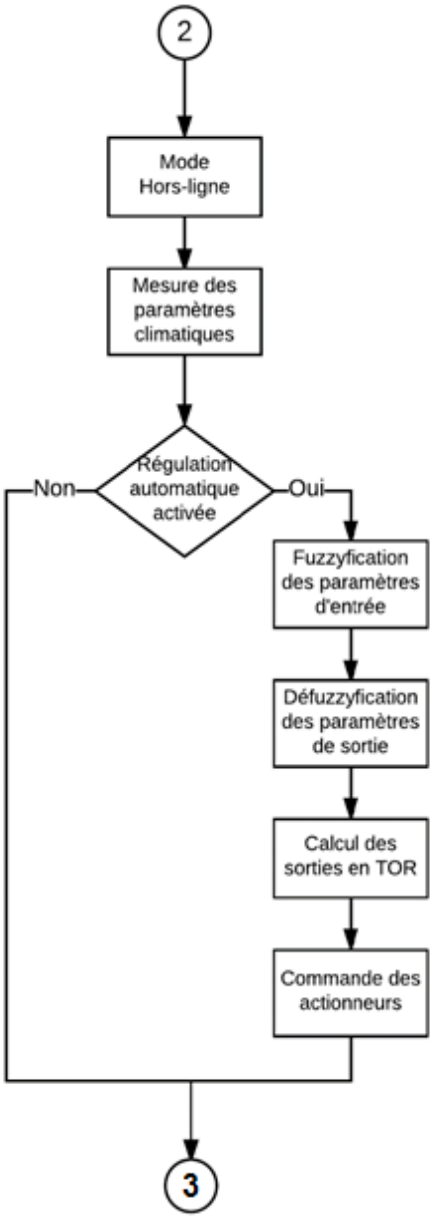


Figure V.6. Diagramme global du programme de la carte de commande.

Lors de la mise sous tension de la carte de commande, le programme commence par la déclaration des bibliothèques, l'initialisation des variables et le chargement des consignes de régulation depuis la mémoire interne, ces consignes sont ensuite utilisées pour créer les univers de discours du régulateur flou ainsi que les états marche/arrêt du régulateur TOR.

Apres la phase d'initialisation, le programme fonctionnera en mode "En ligne" ou "Hors Ligne" selon la disponibilité d'un réseau Wifi et d'une connexion internet.

Dans le mode « En ligne », le programme commence par synchroniser les consignes de régulation avec le serveur, ensuite il fait l'acquisition des données des capteurs, ces données vont être utilisées pour les calculs des paramètres de sorties par les algorithmes de régulation afin de commander les actionneurs. Enfin ces données vont être transmises vers le serveur qui va les sauvegarder dans la base de données.

Dans le mode "Hors Ligne", le programme procède à la mesure, la régulation et la commande des actionneurs.

Remarque : la régulation automatique peut être désactivée depuis l'application web afin de commander le système manuellement.

V.3.2. Organigramme de la partie serveur

La figure V.7 présente l'organigramme du programme du serveur web, il permet de faire le lien entre la carte de commande et l'interface Homme-machine. Il permet aussi de gérer la base de données (enregistrement et visualisation des données).

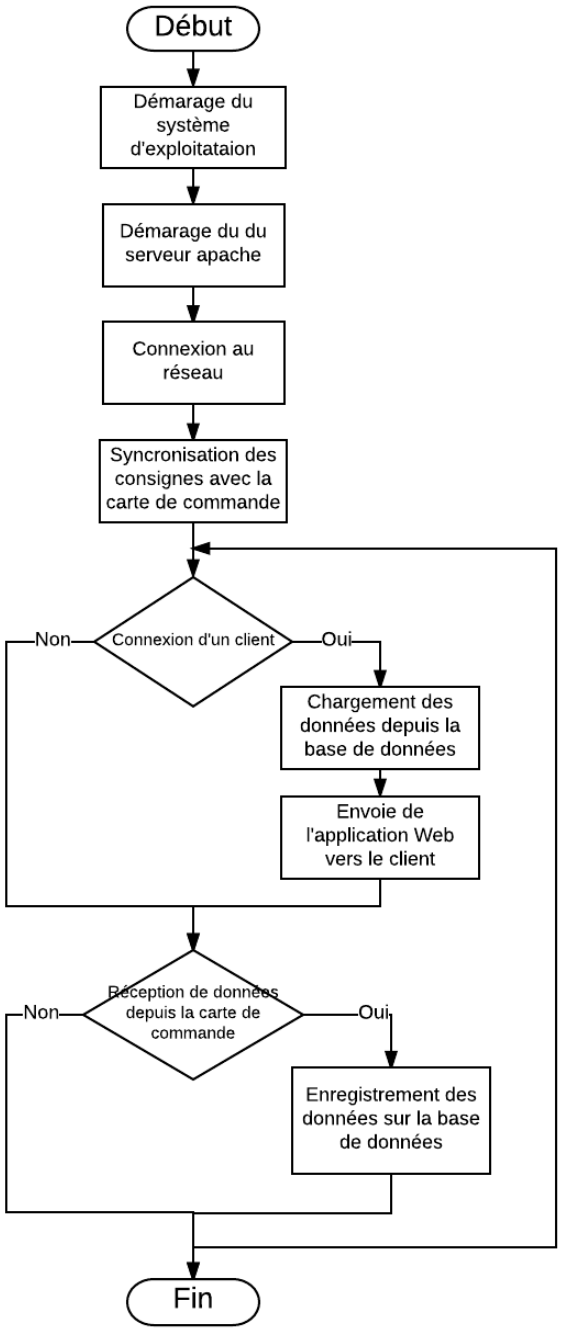


Figure V.7. Diagramme du programme de la partie serveur.

V.4. Conception de la carte PCB de commande

La première figure (figure V.8) présente la face A, réalisée à l'aide du logiciel EasyEDA, elle montre la disposition des différents composants de la partie commande et leurs connexions sur la couche supérieure, La seconde figure (figure V.9) présente la face B, elle montre les connexions de la couche inférieure (Voir l'annexe I pour le schéma électrique).

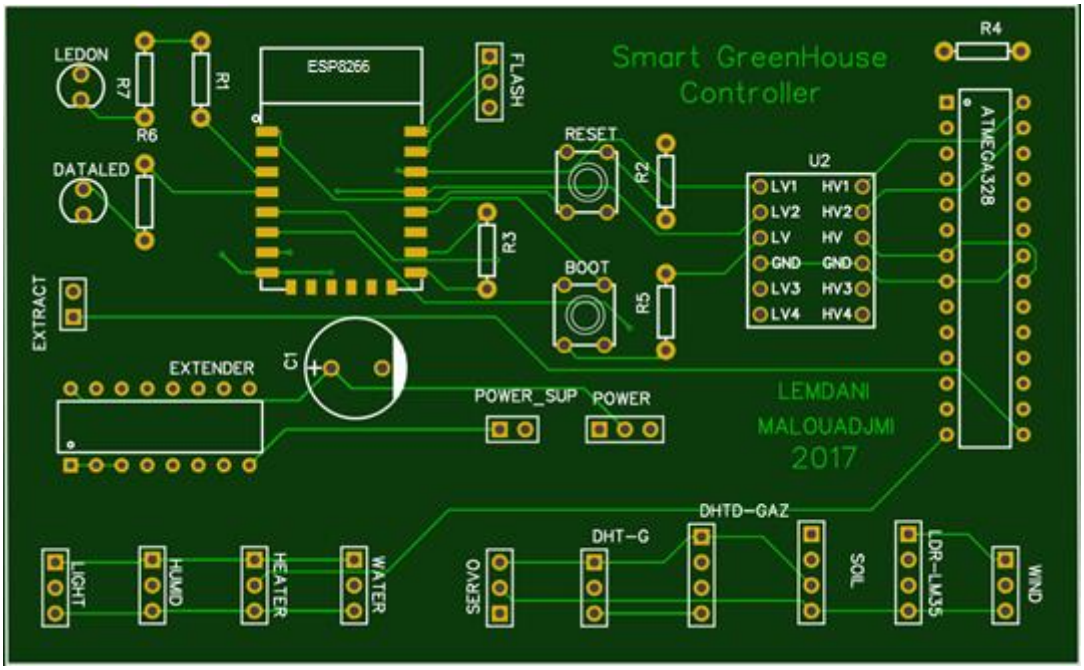


Figure V.8. Carte de commande face A.

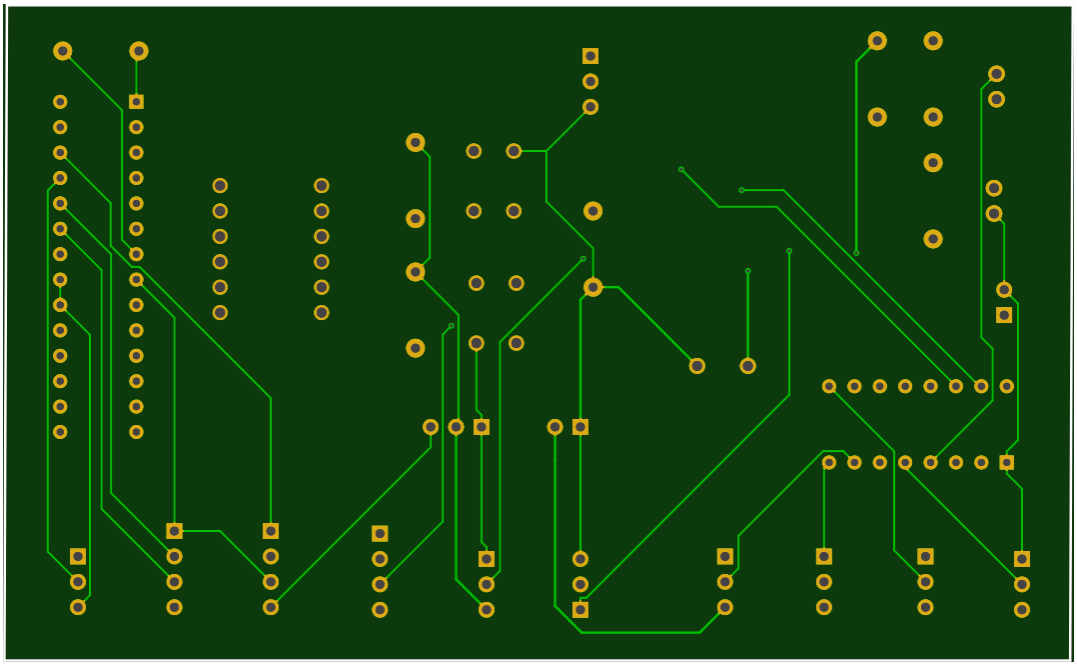


Figure V.9. Carte de commande face B.

Les composants constituants la carte de commande :

- Un ESP8266 avec son adaptateur.
- Un ATMEGA328

- Un PCF8574
- Un I²C Level Shifter.
- Des PINs de connexion femelles
- 1 Condensateur 2200uF.
- Résistances 230 Ohm et 10k Ohm
- 2 Leds.

V.5. Le projet final

La figure V.10 présente une description des différents composants de la serre agricole après assemblage:

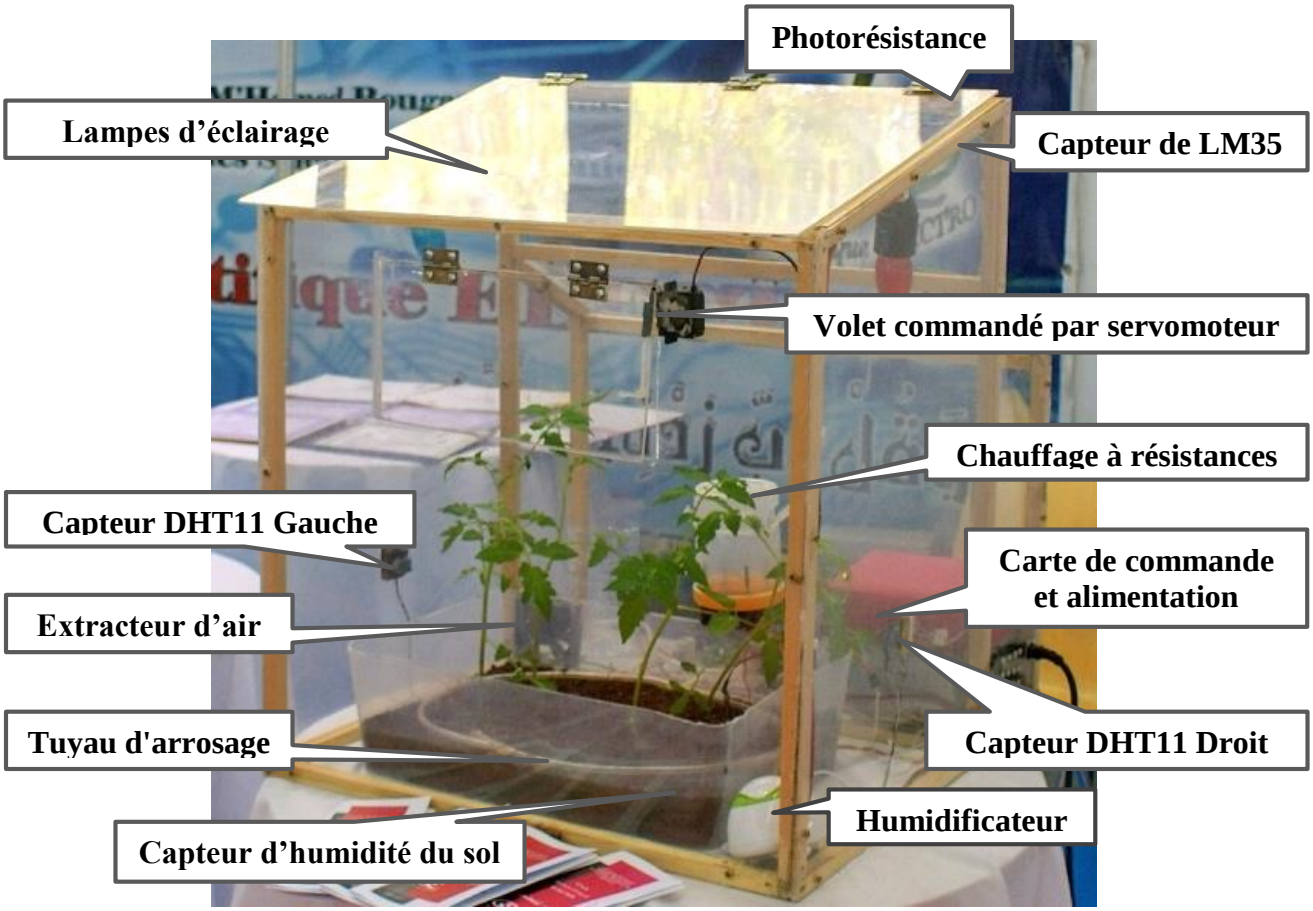


Figure V.10. Description des différentes parties de la serre prototype.

Figure V.11. La serre agricole finale exposée au salon national des produits de la recherche (SAFEX 2017).

V.6. Evaluation des performances du système

Afin de tester le bon fonctionnement et les performances de notre système, nous avons réalisé deux enregistrements de 24h dans les conditions réelles avec et sans contrôle climatique.

V.6.1. Evolution des paramètres climatique sans contrôle

La figure V.12 et la figure V.13 présentent l'évolution des paramètres climatiques suivants : l'humidité, les températures intérieure et extérieure par rapport au taux d'ensoleillement durant 24h sans contrôle automatique. Cet enregistrement a été effectué le 20/05/2017.

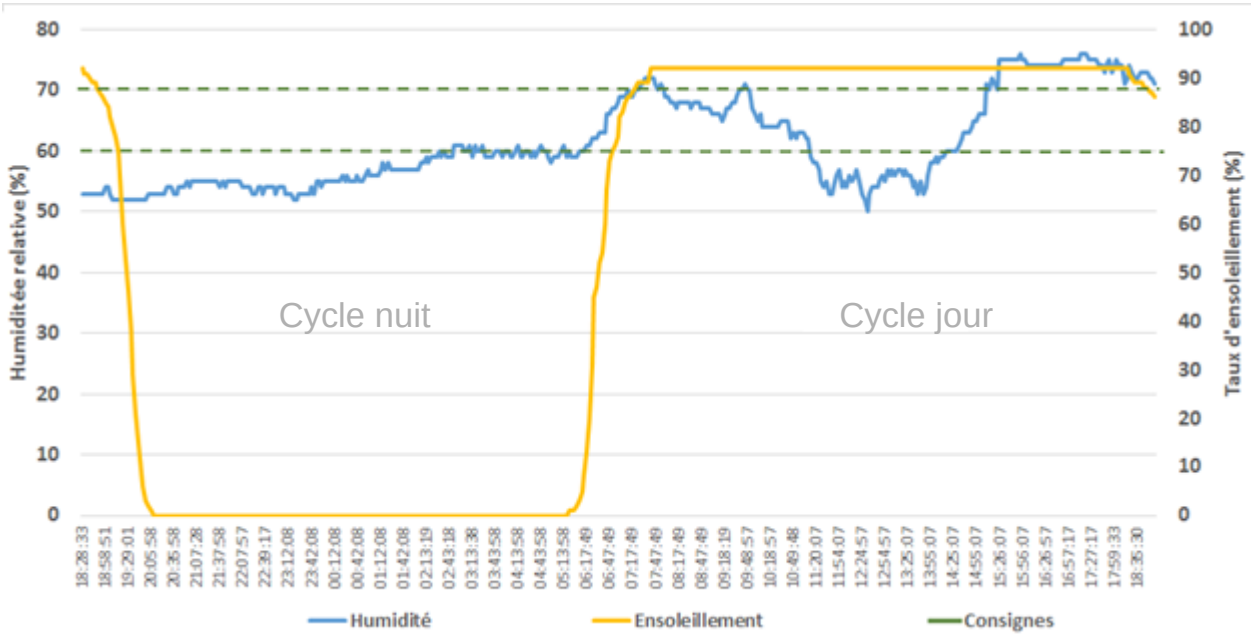


Figure V.12. Enregistrement de l'évolution de l'humidité relative par rapport au taux d'ensoleillement sans contrôle.

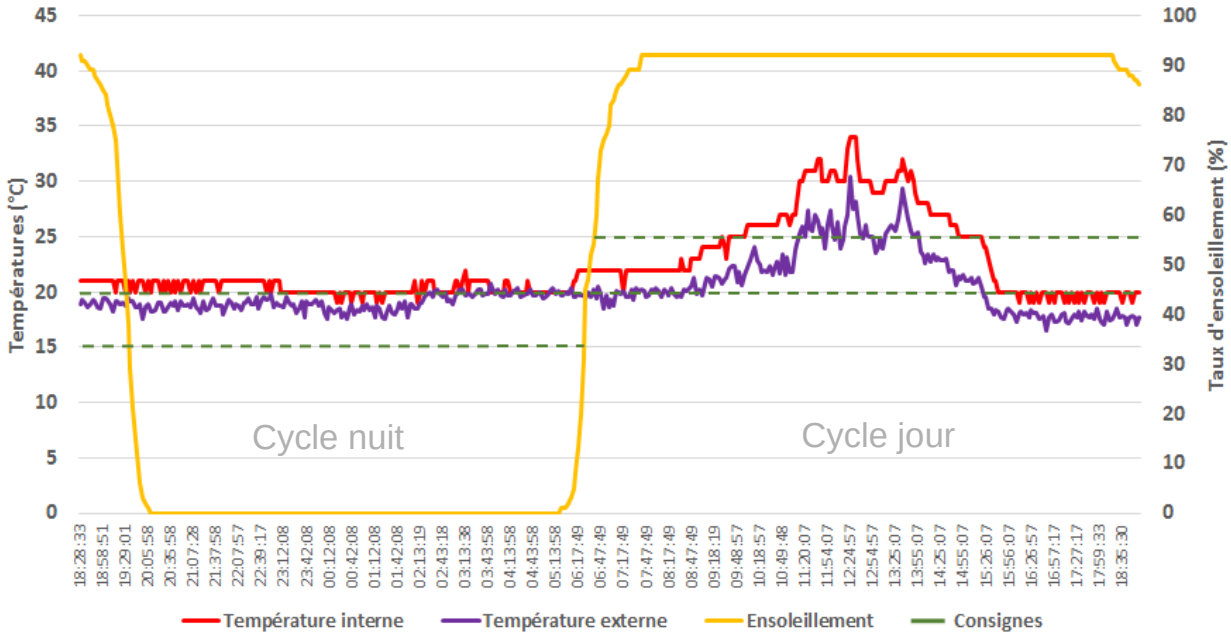


Figure V.13. Enregistrement de l'évolution des températures intérieur et extérieur par rapport au taux d'ensoleillement sans contrôle.

Interprétation :

1) Durant le début du cycle nuit (entre 20 :10 et 2 :20), l'humidité intérieure de la serre est inférieure à la consigne de 60%, elle augmente progressivement jusqu'à l'atteindre vers 2 :43 et dépasse la consigne maximale de 70% au lever du soleil. Durant ce même cycle, la température interne est supérieure à la consigne.

2) Durant le début du cycle jour, l'humidité reste dans l'intervalle de consigne, elle diminue fortement vers 11 :13 à cause de l'augmentation de la température. A partir de 15 :26, la température diminue et l'humidité augmente jusqu'à dépasser l'intervalle de consigne. Cette augmentation est due à une humidité externe élevée.

V.6.2. Evolution des paramètres climatique avec contrôle

Dans cet enregistrement (figure V.14 et figure V.15), nous évalueront les deux commandes réalisées (Tout Ou Rien et flou) durant les cycles jour et nuit. Nous allons le comparer avec l'enregistrement sans contrôle. Cet enregistrement a été effectué le 23/05/2017.

Les régulateurs TOR et flou sont configurés avec les consignes suivantes :

- **Cycle nuit :** Humidité : Min : 50% Max : 60%.
  Température : Min : 15°C Max : 20°C.
- **Cycle jour :** Humidité : Min : 50% Max : 60%.
  Température : Min : 20°C Max : 25°C.

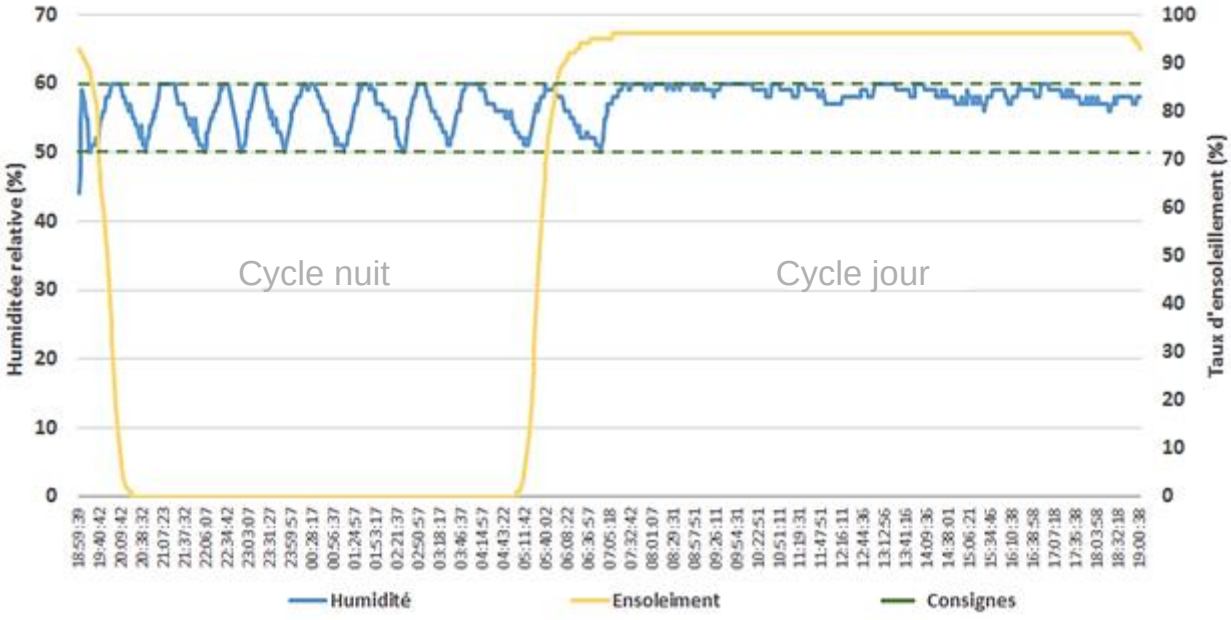


Figure V.14. Enregistrement de l'évolution de l'humidité relative par rapport au taux d'ensoleillement avec contrôle.

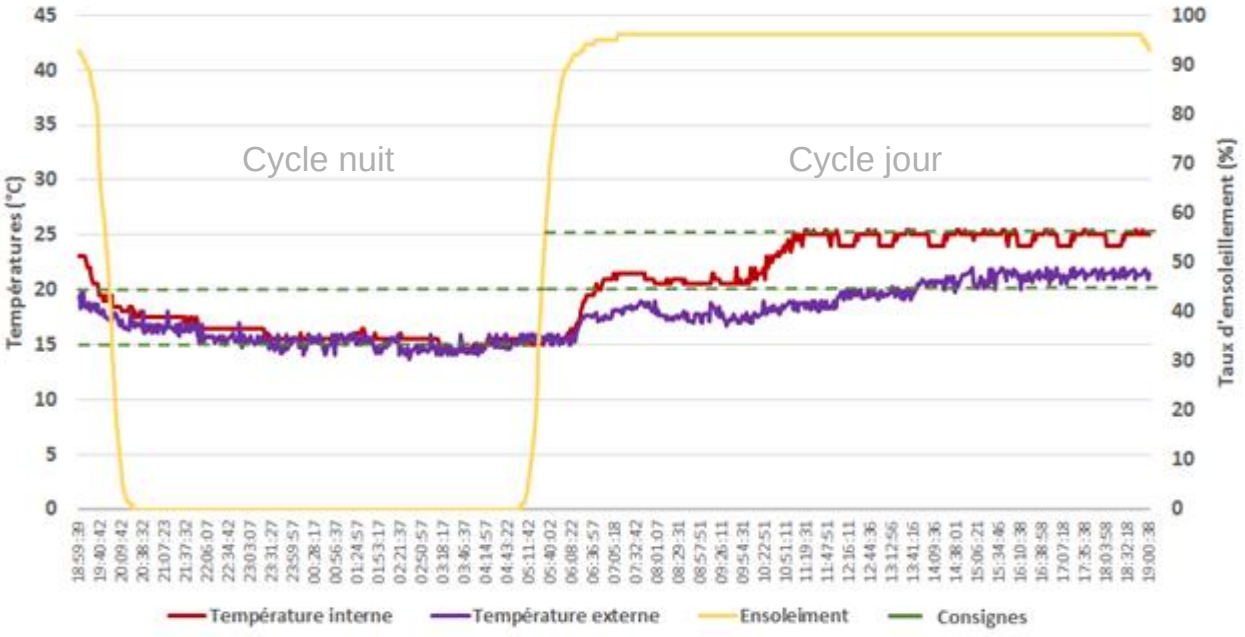


Figure V.15. Enregistrement de l'évolution des températures intérieur et extérieur par rapport au taux d'ensoleillement avec contrôle.

Interprétation :

3) Nous constatons que durant le cycle « nuit », comme il n'y a pas de photosynthèse, le paramètre de l'humidité n'excède pas une certaine valeur inférieure à 50%. Dans ce cas le régulateur « Tout Ou Rien » agit pour augmenter ce paramètre et le garder dans l'intervalle 50-60%. Durant ce même cycle, la température reste plus ou moins stable dans l'intervalle 15-20°C à l'intérieur comme à l'extérieur de la serre.

4) Durant le cycle « jour », la photosynthèse agissant, l'humidité augmente et dépasse le seuil de 60%. Le régulateur flou commande alors l'extracteur d'air ainsi que le volet qui agissent pour réduire l'humidité excédentaire. Par effet de serre, la température intérieure dépasse celle de l'extérieur, mais par la régulation celle-ci ne peut dépasser une valeur seuil préprogrammée de 25°C.

D'après ces enregistrements, on peut conclure que le climat à l'intérieur de la serre est fortement influencé par les conditions externes (température, humidité, rayonnement solaire). Les différentes commandes (chauffage, humidification...) d'une serre ont pour objectif l'amélioration du climat interne de la serre.

V.7. Etude socioéconomique

Pour estimer le coût de fabrication de la plateforme de commande ainsi que la serre agricole, nous avons fait une étude socioéconomique afin d'estimer le prix de fabrication :

Tableau V.1. Estimation du prix de revient de notre projet.

Composant	Prix unitaire (DA)	Quantité	Total (DA)
ESP8266	1400	1	1400
PCF8475	480	1	480

ATMEGA 328	800	1	800
I²C level shifter	200	1	200
Capteur DHT11	680	2	1360
Capteur LM35	200	1	200
Capteur d'humidité du sol	450	1	450
Pompe d'arrosage	450	1	450
Relai	100	4	400
L293	260	1	260
Alimentation PC	1000	1	1000
Lampes d'éclairage	60	2	120
Humidificateur	2200	1	2200
Résistance chauffante	40	1	40
Condensateur 2200uF	40	1	40
Fils électriques	10	10	100
Bouton poussoir	20	2	40
Câble d'alimentation	200	1	200
Rouleau d'étain	800	1	800
Potentiomètre	40	1	40
Pin de connexion	40	5	200
Raspberry PI 2	7000	1	5000
Feuilles en plexiglass	3000	1	3000
		Total	25 780 DA

D'après le tableau, le prix de revient est de 25 780.00 DA, ce montant n'inclut pas le prix du capteur de CO_2 et le capteur de vent qui sont assez couteux et indisponible sur le marché.

V.8. Conclusion

Après la phase de montage de la structure et l'installation des équipements à l'intérieur et à l'extérieur de la serre, nous avons pu constituer une base de données climatique. A partir des résultats obtenus des tests sur les conditions réels, nous avons constaté que la commande appliquée sur le système donne des résultats satisfaisants.

Puis, nous avons réalisé avec un organigramme fonctionnel qui décrit le fonctionnement global de notre système.

Enfin, nous avons conclus ce chapitre avec une étude socioéconomique afin de déterminer le cout de réalisation, ce cout parait acceptable compte tenu des résultats obtenus.

Conclusion générale

Afin d'augmenter la qualité et la quantité des produits agricoles issue de la culture sous serres et de faire face à un marché très concurrentiel, il est nécessaire d'avoir un contrôle climatique optimal.

Pour cela, nous avons développé une plateforme pour l'automatisation et le contrôle à distance des serres agricoles. Elle est composée d'une partie électronique constituée d'une carte de commande autour du microcontrôleur ESP8266 et d'une partie logicielle sous forme d'une application Web pour le contrôle du système via internet.

Nous avons organisé ce mémoire en cinq chapitres :

Dans le premier chapitre, nous avons déterminé les paramètres les plus agissants sur la croissance de la culture sous serre. Pour mener à bien notre travail et d'améliorer nos connaissances dans le domaine agricole, nous avons effectué un état de l'art sur les types de serres agricoles les plus populaires et les différents composants nécessaires à la gestion efficace des paramètres climatiques. Ces connaissances nous ont aidés par la suite, à l'élaboration de la commande climatique afin de répondre au cahier des charges.

Dans le 2eme chapitre, nous avons présenté la problématique en évoquant le besoin, les exigences et les objectifs du cahier des charges. Après cela nous sommes passés vers le choix de la solution technologique de la partie commande. Nous avons opté pour le microcontrôleur ESP8266 afin de piloter notre système. Par la suite nous avons choisis les capteurs et les actionneurs que nous avons utilisés dans notre réalisation.

Dans le 3eme chapitre, nous avons présenté dans un premier temps quelques notions sur la régulation automatique ainsi qu'une brève description sur le fonctionnement des commandes TOR, PID et floue. En second lieu, nous avons procédé à l'implémentation et à la simulation de la commande floue et TOR dans plusieurs cas de figures à l'aide du « Toolbox fuzzy logic » et de MATLAB/Simulink.

Dans le 4eme chapitre, nous avons en premier lieu, effectué une étude comparative des différentes technologies de communication sans fil, afin de choisir la plus adaptée à nos besoins. Nous avons aussi défini quelques protocoles de communication utilisés dans notre projet.

En second lieu, nous avons entamé la conception de l'interface homme-machine en présentant les avantages de l'utilisation des applications WEB ainsi que les outils informatiques utilisés pour leur développement.

En dernier lieu, nous avons décrit l'interface finale, en présentant ses différentes fonctionnalités.

Dans le 5eme chapitre, nous avons entamé le montage de la structure ainsi que les équipements de régulation de la serre. Ensuite nous avons mis en marche notre système et effectuer les premiers tests de fonctionnement qui nous ont donnés des résultats concluants.

Puis, nous avons effectué des enregistrements de 24h sur les conditions réels afin d'évaluer les performances des commandes appliquées. Après cela nous avons conçu la carte de commande PCB afin de finaliser notre réalisation.

Enfin, nous avons décrit le fonctionnement global de notre système par un organigramme fonctionnel. Nous avons conclu ce chapitre par une étude socioéconomique afin de déterminer le cout de revient de notre projet.

Ce travail nous a permis de mettre à l'épreuve d'une part et d'améliorer d'autre part nos connaissances théoriques et pratiques en électronique, électrotechnique, mécanique et informatique. De plus, il a été une source de découverte de plusieurs domaines d'études tels que : l'agriculture

(gestion du climat, types et équipements de serres), la biotechnologie, l'électronique embarquée et l'utilisation de plusieurs outils comme Matlab/Simulink avec le Toolbox Fuzzy Logic, EasyEDA, Apache 2, langages de programmation : Python, JavaScript, PHP…

Enfin, signalons que ce travail a été valorisé par notre participation au Salon National des Produits de la Recherche Scientifique (SAFEX, Alger, du 18 au 21 mai 2017) organisé par la Direction Générale de la Recherche Scientifique et du Développement Technologique.

En perspective, nous proposons l'amélioration de notre projet par :

- Adapter notre plateforme pour la gestion des serres à grand volume ;
- Possibilité de contrôler plusieurs serres en réseau ;
- Utiliser des actionneurs et des capteurs industriels ;
- Intégrer un système d'irrigation fertilisante ;
- Ajouter d'autres actionneurs comme : le chauffage d'eau, l'enrichissement du CO_2…
- Contrôler plus de paramètres comme : la température du sol, le PH et la conductivité électrique ;
- Utiliser d'autres types de commandes avancées comme les réseaux de neurones.

Références bibliographiques

[1] F. Rodrigues, M. Berenguel, M. Arahal, "A hierarchical control system for maximizing profit in greenhouse crop production," Proceeding in Eurcapean Control Conference ECCO3, Cambridge, UK, 2003.

[2] M. Trigui, S. Barrington, L. Gauthier, Structures and environment, "A strategy for greenhouse climate control, " part i : model development. Journal of agricultural engineering research 78(4), p. 407-412, 2001.

[3] G.V. Straten, G.V. Willigenburg, E.V. Henten and R.V. Ooteghem, Optimal Control of Greenhouse Cultivation, Taylor and Francis Group, New York, Usa, 2011.

[4] H. Ait Bouh, "Conception, développement et mise en place d'un système de pilotage automatique du climat et de la fertirrigarion goutte à goutte sous serre Thèse de Doctorat", Faculté des Sciences, Mejnès, 2003.

[5] Y. El afou, "Contribution au contrôle des paramètres climatiques sous serre". Thèse de Doctorat Université Lille 1, 2014.

[6] K. Mesmoudi, "Etude Expérimentale et Numérique de la Température et de l'Humidité de l'Air d'un Abri Serre Installé dans les Haut Plateaux d'Algérie, Région des Aurès," Thèse de Doctorat Physique Energétique, option énergétique Université de Batna, 2010.

[7] C.A, Hermez, "Gestion des paramètres bioclimatiques dans les serres agricoles," Thèse d'Ingéniorat en Instrumentation électronique, Université des Sciences et de la technologie Houari Boumediene USTHB, 2011.

[8] Y. Bouteraa, "Automatisation d'une serre agricole, " Magister en Sciences Agronomiques, Ecole Nationale Supérieure D'agronomie-El Harrach, 2012.

[9] M. Kechir, H. Mahfoudi, "Acquisition et transmission sur bus CAN des paramètres bioclimatiques d'une serre agricole, " mémoire de master, Université des Sciences et de la technologie Houari Boumediene USTHB, 2012.

[10] Rapport de la société nationale d'Horticulture de France et l'Association des journalistes de jardin de d'Horticulture, "Serre tunnel," 2014. [En ligne]. Disponible : http://www.gerbeaud.com/jardin/amenagement/serre-tunnel,1128.html

[11] Catalogue de la Société ULMA-Agricola, "Serres-multi chapelle," 2008. [En ligne]. Disponible: https://www.interempresas.net/FeriaVirtual/Catalogos_y_documentos/1381/Serres-multichapelle-ULMA-Agricola.pdf

[12] Rapport de la société de marketing euro-serre, « les serres », 2016. [En ligne]. Disponible : http://www.serre-jardin.com/img/cata_euro_serre_2016.pdf

[13] R. Haxaire, J. C. Roy, T. Boulard, M. A. Lamrani, A. Jaffrin, "Etude numérique et expérimentale de la ventilation par convection naturelle dans une serre," In Colloque annuel-SFT, p. 64-69, 1998.

[14] Rapport de la société ULMA-Agricola, "l'équipement des serres," [En ligne]. Disponible : http://www.ulmaagricola.com/fr/serres/equipements

[15] R. Cadiergues, "L'éclairage Artificiel," Guide RefCad nR27. A, 2012. [En ligne]. Disponible : http://media.xpair.com/auxidev/nR27a.pdf

[16] J.P. Goure, L'optique dans les instruments: Généralités, Lavoisier, 2011. [En ligne]. Disponible : http://www.canna.fr/eclairage_pour_mes_plantes

[17] T.R. Bel hadj, "Conception d'un anémomètre à fil chaud à large bande passante, " thèse de doctorat, École de technologie supérieure, 2016. [En ligne]. Disponible : http://espace.etsmtl.ca/1675/1/BEL_HADJ_TAHER_Rafik.pdf

[18] M. Leroy, “La mesure au sol de la température et des précipitations,” La Météorologie, vol. 8, no. 39, p. 52, 2002.

[19] A. Cerqueus, M. Sevaux, H. Kerhascoet, J. Laurent. « Modélisation de la girouette d’un voilier : expérimentation avec LocalSolver », Université Bretagne-Sud, Lab-STICC, Lorient, 2016. [En ligne]. Disponible : http://www-labsticc.univ-ubs.fr/~sevaux/Publications/p-cerqueus-16-slides.pdf

[20] Rapport de l’unité d'affaire de la compagnie Premier Tech Horticulture, "Capteurs d'humidité volumétrique," juillet 2016. [En ligne]. Disponible : http://www.pthorticulture.com/fr/zone-du-savoir/capteurs-dhumidite-volumetrique/

[21] Catalogue de la Société SES AUTOMATION, "Catalogue Instrumentation," 2015 – 2016. [En ligne]. Disponible : http://www.ses-automation.fr/_document/catalogue-ses-automation-2015-2016.pdf

[22] Rapport de la société VAISALA, "Les technologies de capteurs infrarouges et leur impact sur la précision de la mesure du CO2 dans les applications CVC," 2014. [En ligne]. Disponible: http://www.vaisala.com/Vaisala%20Documents/Technology%20Descriptions/HVAC-CO2-Measurement-Accuracy-B211311FR-A.pdf

[23] R. Salazar, A. Rojano, I. Lopez, "A Model for the Combine Description of the Temperature and Relative Humidity Regime in the Greenhouse," Ninth Mexican International Conference on Artificial Intelligence, novembre 2010.

[24] A. Schmid and R. Holzer, Microcontrôleurs: cours et travaux pratiques: destiné aux étudiants de deuxième année en section de microtechnique "Microcontrôleurs-2007". Lausanne: EPFL, 2007.

[25] Espressif, "ESP8266 SDK API Reference", 2017. [En ligne]. Disponible : https://espressif.com/en/support/download /documents

[26] Espressif, "ESP8266EX Datasheet", 2017. [En ligne]. Disponible : https://espressif.com/en/support/download /documents

[27] Neil Kolban, "Kolban’s book on ESP8266 ", leanpub, 2016.

[28] Datasheet de l’ATMEGA328PU. [En ligne]. Disponible : http://www.atmel.com/Images/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf

[29] V. Himpe, Mastering the I2C Bus. Susteren: Elektor international media, 2011.

[30] Microchip, "MCP 3021 Datasheet", 2016. [En ligne]. Disponible : http://ww1.microchip.com/downloads/en/DeviceDoc/21805B.pdf

[31] Datasheet du capteur LDR. [En ligne]. Disponible :

[32] Datasheet du capteur de température lm35. [En ligne]. Disponible : https://www.ti.com/lit/ds/symlink/lm35.pdf?HQS=TI-null-null-alldatasheets-df-pf-SEP-wwe

[33] Datasheet du L293D. [En ligne]. Disponible : http://www.ti.com/lit/ds/symlink/l293.pdf

[34] R. Lefevre, "Qu'est-ce qu'un servomoteur, " Article d'école supérieur d'informatique SUPINFO, Grenoble, 2015. [En ligne]. Disponible : http://www.supinfo.com/articles/single/296-qu-est-ce-qu-servomoteur

[35] Datasheet de l'AN10441. [En ligne]. Disponible : www.nxp.com/documents/application_note/AN10441.pdf

[36] Datasheet capteur de gaz DFROBOT SEN0159. [En ligne]. Disponible : https://www.dirobots.com/fr/capteurs-gaz/501-capteur-co2-analogique.html

[37] Datasheet du DHT11. [En ligne]. Disponible : http://www.micropik.com/PDF/dht11.pdf

[38] Documentation du protocole 1-Wire. [En ligne]. Disponible : https://github.com/openremote/Documentation/wiki/1-Wire

[39] Datasheet du PCF8574A. [En ligne]. Disponible : http://www.ti.com/lit/ds/symlink/pcf8574.pdf

[40] Datasheet du Futaba S3003. [En ligne]. Disponible : http://etud.insaoulouse.fr/~bouvot/Cours/4AE/P%C3%A9riph%C3%A9riques/Datasheets/Moteurs%20roues%20tourelle/Futaba_S3003.pdf

[41] C. Sindjui, Le grand guide des systèmes de contrôle-commande industriels automatisme, instrumentation, réseaux locaux, régulation automatique. Paris: Lexitis éd, 2014.

[42] H. Ait Bouh, "Conception, développement et mise en place d'un système de pilotage automatique du climat et de la fertirrigarion goutte à goutte sous serre Thèse de Doctorat", Faculté des Sciences, Mejnès, 2003.

[43] J.F Bourgeois, Automatisme et régulation des équipements thermiques. Edition Technique Ingénieur, 1998.

[44] A. Bouisfi et Y. Belrhali, "Etude en simulation d'une régulation thermique par logique floue". Université Sidi Mohamed Ben Abdellah Fès, 2009.

[45] M.S. Aït Cheikh, B. Hadj Aïssa, A. Malek et M. Becherif, "Mise au point d'une régulation floue pour serre agricole à énergie solaire". Revue des Energies Renouvelables Vol. 13 N°3 2010.

[46] GitHub de la bibliothèque eFLL. [En ligne]. Disponible : https://github.com/zerokol/eFLL

[47] Description du model « Thermal Model of a House ». [En ligne]. Disponible : https://www.mathworks.com/examples/simulink/mw/simulink_product-sldemo_househeat-thermal-model-of-a-house

[48] Documentation du Fuzzy logic Toolbox. [En ligne]. Disponible : https://fr.mathworks.com/help/fuzzy/

[49] Documentation Simulink. [En ligne]. Disponible : https://www.mathworks.com/help/simulink/

[50] M. Gast, 802.11 réseaux sans fil: la référence. Paris: OReilly, 2005.

[51] P. Atelin, Wi-Fi réseaux sans fil 802.11. St Herblain: Éd. ENI, 2008.

[52] J. C. C. Dordoigne, TCP/IP et les protocoles Internet. Nantes: Éditions ENI, 2004.

[53] A. Cazes and Delacroix Joëlle, Développer une application web. Malakoff: Dunod, 2016.

[54] Documentation du Raspberry Pi. [En ligne]. Disponible : https://www.raspberrypi.org/documentation/

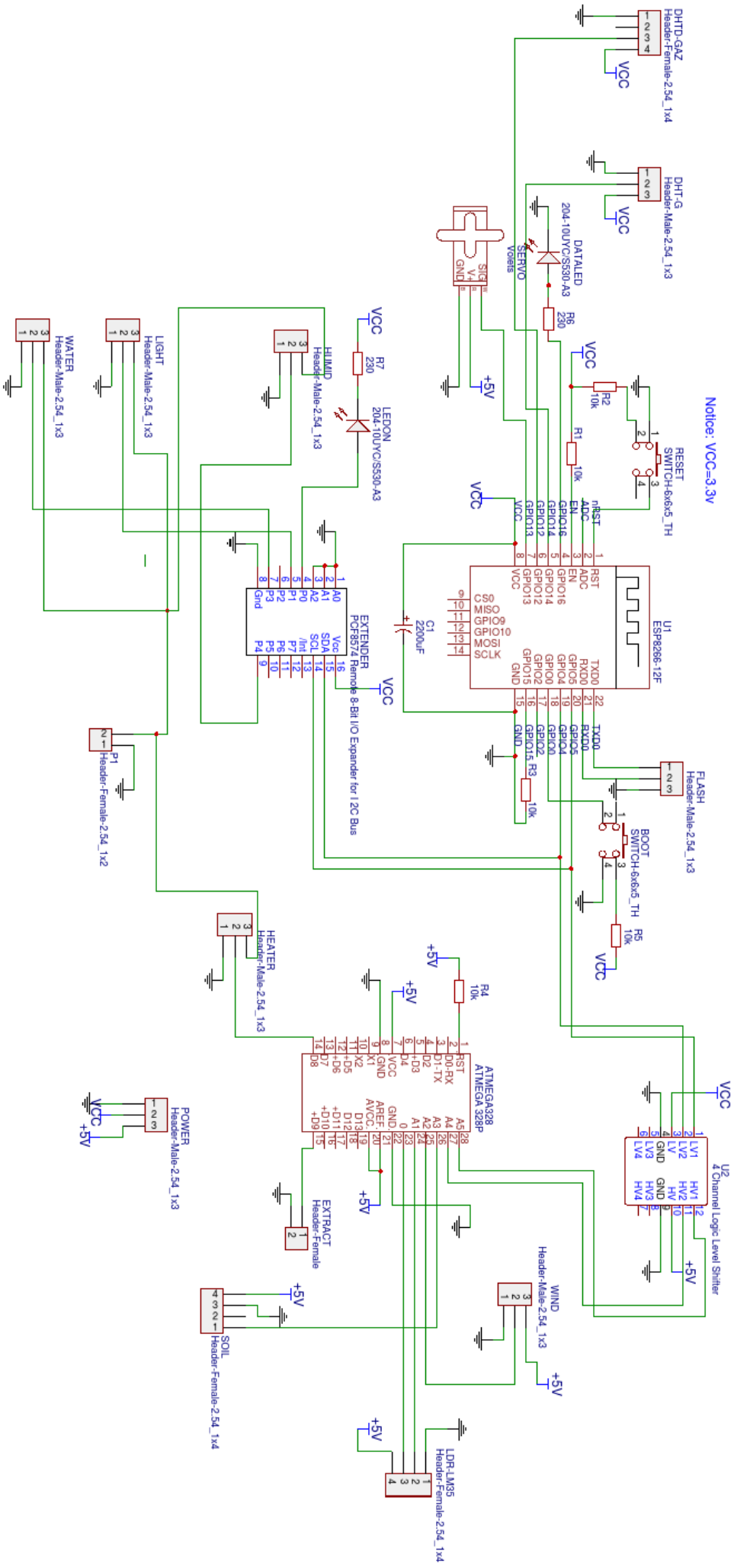
[55] M. Grinberg, Flask Web Development Developing Web Applications with Python. Sebastopol, CA: OReilly & Associates, 2014.

[56] Documentation du logiciel Apache 2. [En ligne]. Disponible : http://httpd.apache.org/docs/2.2/fr/

Annexe

Schéma électrique de la carte de commande réalisé avec le logiciel EasyEDA

Annexe I



```
#include <FuzzyRule.h>
#include <FuzzyComposition.h>
#include <Fuzzy.h>
#include <FuzzyRuleConsequent.h>
#include <FuzzyOutput.h>
#include <FuzzyInput.h>
#include <FuzzyIO.h>
#include <FuzzySet.h>
#include <FuzzyRuleAntecedent.h>
```

Déclaration des bibliothèques

```
Fuzzy* fuzzy = new Fuzzy();
```

Création de régulateur flou

```
extern int vent;
extern int co2;
extern float sol;
extern float temp_ext;
extern float h, t;
```

Déclaration des variables d'entrées

```
int seuilTemp = 23;
int seuilTemp_m = 27;
int seuilTemp_ext = 25;
int seuilHumi = 57;
int seuilHumi_m = 63;
int seuilCo2 = 700;
```

Déclaration des variables de seuil

```
void setup_fuzzy() {
```

Déclaration de la fonction d'initialisation de régulateur flou

```
  FuzzyInput* temperature = new FuzzyInput(1);
  FuzzySet*faible1 = new FuzzySet(0, 0, 0, seuilTemp);
  temperature->addFuzzySet(faible1);
  FuzzySet* normal1 = new FuzzySet(seuilTemp-3, seuilTemp,
  seuilTemp_m, seuilTemp_m+8);
  temperature->addFuzzySet(normal1);
  FuzzySet* eleve1 = new FuzzySet(seuilTemp_m, 30, 50, 50);
  temperature->addFuzzySet(eleve1);
  fuzzy->addFuzzyInput(temperature);
```

Création d l'univers de discours de la temperature intérieur

```
  FuzzyInput* temperature_extern = new FuzzyInput(1);
  FuzzySet* moin = new FuzzySet(0, 0, 0, 25);
  temperature_extern->addFuzzySet(moin);
  FuzzySet* meme = new FuzzySet(20, 25, 25, 30);
  temperature_extern->addFuzzySet(meme);
  FuzzySet* supieur = new FuzzySet(25, 50, 50, 50);
  temperature_extern->addFuzzySet(supieur);
  fuzzy->addFuzzyInput(temperature_extern);
```

Création d l'univers de discours de la temperature extérieur

```
FuzzyInput* humidite = new FuzzyInput(2);
  FuzzySet* faible2 = new FuzzySet(0, 0, 0, seuilHumi);
  humidite->addFuzzySet(faible2);
  FuzzySet*normal2 =new FuzzySet(seuilHumi-7,seuilHumi,
seuilHumi_m, seuilHumi_m+22);
  humidite->addFuzzySet(normal2);
  FuzzySet* eleve2 = new FuzzySet(seuilHumi_m,100,100,100);
  humidite->addFuzzySet(eleve2);
  fuzzy->addFuzzyInput(humidite);
```

Création de l'univers de discours de l'humidité de l'air

```
  FuzzyInput* Co2 = new FuzzyInput(3);
  FuzzySet* faible3 = new FuzzySet(0, 0, 0, 700);
  Co2->addFuzzySet(faible3);
  FuzzySet* normal3 = new FuzzySet(350, 700, 700, 1000);
  Co2->addFuzzySet(normal3);
  FuzzySet* eleve3 = new FuzzySet(700, 1500, 1500, 1500);
  Co2->addFuzzySet(eleve3);
  fuzzy->addFuzzyInput(Co2);
```

Création de l'univers de discours de taux de Co_2

```
  FuzzyInput* vitess_du_vent = new FuzzyInput(4);
  FuzzySet* tre_faible4 = new FuzzySet(0, 0, 0, 25);
  vitess_du_vent->addFuzzySet(tre_faible4);
  FuzzySet* faible4 = new FuzzySet(0, 25, 25, 50);
  vitess_du_vent->addFuzzySet(faible4);
  FuzzySet* moyen4 = new FuzzySet(25, 50, 50, 75);
  vitess_du_vent->addFuzzySet(moyen4);
  FuzzySet* eleve4 = new FuzzySet(50, 75, 75, 100);
  vitess_du_vent->addFuzzySet(eleve4);
  FuzzySet* tre_eleve4 = new FuzzySet(75,100,100,100);
  vitess_du_vent->addFuzzySet(tre_eleve4);
  fuzzy->addFuzzyInput(vitess_du_vent);
```

Création de l'univers de discours de vitesse du vent

```
  FuzzyOutput* servomoteur = new FuzzyOutput(1);
  FuzzySet* Stre_faible1 = new FuzzySet(0, 0, 0, 6);
  servomoteur->addFuzzySet(Stre_faible1);
  FuzzySet* S_faible1 = new FuzzySet(0, 23, 23, 46);
  servomoteur->addFuzzySet(S_faible1);
  FuzzySet* S_moyen1 = new FuzzySet(23, 46, 46, 69);
  servomoteur->addFuzzySet(S_moyen1);
  FuzzySet* S_eleve1 = new FuzzySet(46, 77, 77, 115);
  servomoteur->addFuzzySet(S_eleve1);
  FuzzySet* Stre_eleve1 = new FuzzySet(96,115,115,115);
  servomoteur->addFuzzySet(Stre_eleve1);
  fuzzy->addFuzzyOutput(servomoteur);
```

Création de l'univers de discours de la vitesse d'extracteur

```
  FuzzyOutput* ventilateur = new FuzzyOutput(2);
  FuzzySet* Stre_faible2 = new FuzzySet(0, 10, 10, 10);
  ventilateur->addFuzzySet(Stre_faible2);
  FuzzySet* S_faible2 = new FuzzySet(0, 25, 25, 50);
  ventilateur->addFuzzySet(S_faible2);
  FuzzySet* S_moyen2 = new FuzzySet(25, 50, 50, 75);
  ventilateur->addFuzzySet(S_moyen2);
  FuzzySet* S_eleve2 = new FuzzySet(50, 75, 75, 100);
  ventilateur->addFuzzySet(S_eleve2);
  FuzzySet* Stre_eleve2 = new FuzzySet(75, 100, 100, 100);
```

Création de l'univers de discours de l'ouverture de volet

```
  ventilateur->addFuzzySet(Stre_eleve2);
  fuzzy->addFuzzyOutput(ventilateur);
```

Suite de l'univers de discours de l'ouverture de volet extérieur

```
FuzzyRuleAntecedent* siteperaturefaibleORhumiditefaible = new
FuzzyRuleAntecedent();
siteperaturefaibleORhumiditefaible-
>joinWithOR(faible1,faible2);
FuzzyRuleAntecedent* siCo2faible = new FuzzyRuleAntecedent();
siCo2faible->joinSingle(faible3);
FuzzyRuleAntecedent*siteperaturefaibleORhumiditefaibleORCo2fa
ible=
new FuzzyRuleAntecedent();
siteperaturefaibleORhumiditefaibleORCo2faible-
>joinWithOR(siteperaturefaibleORhumiditefaible, siCo2faible);
FuzzyRuleConsequent*
thenservomoteurStre_faibleventilateurStre_faible =
 new FuzzyRuleConsequent();
thenservomoteurStre_faibleventilateurStre_faible-
>addOutput(Stre_faible1);
thenservomoteurStre_faibleventilateurStre_faible-
>addOutput(Stre_faible2);
FuzzyRule* fuzzyRule1 = new FuzzyRule(1,
siteperaturefaibleORhumiditefaibleORCo2faible ,
thenservomoteurStre_faibleventilateurStre_faible);
fuzzy->addFuzzyRule(fuzzyRule1);
```

Règle 1

```
  FuzzyRuleAntecedent* siteperaturefaibleEThumiditefaible =
new FuzzyRuleAntecedent();
  siteperaturefaibleEThumiditefaible->joinWithAND(faible1,
faible2);
  FuzzyRuleAntecedent*
siteperaturefaibleEThumiditefaibleANDCo2faible = new
FuzzyRuleAntecedent();
  siteperaturefaibleEThumiditefaibleANDCo2faible-
>joinWithAND(siteperaturefaibleEThumiditefaible,
siCo2faible);
 FuzzyRuleConsequent*
thenservomoteurStre_faibleandventilateurStre_faible12 = new
FuzzyRuleConsequent();
  thenservomoteurStre_faibleandventilateurStre_faible12-
>addOutput(Stre_faible1);
  thenservomoteurStre_faibleandventilateurStre_faible12-
>addOutput(Stre_faible2);
  FuzzyRule* fuzzyRule2 = new FuzzyRule(1,
siteperaturefaibleEThumiditefaible ,
thenservomoteurStre_faibleandventilateurStre_faible12 );
  fuzzy->addFuzzyRule(fuzzyRule2);
```

Règle 2

```
  FuzzyRuleAntecedent* siteperaturenormalEThumiditenormal =
new FuzzyRuleAntecedent();
  siteperaturenormalEThumiditenormal->joinWithAND(normal1,
normal2);
  FuzzyRuleAntecedent* siCo2normal = new
FuzzyRuleAntecedent();
  siCo2normal->joinSingle(normal3);
  FuzzyRuleAntecedent*
siteperaturenormalEThumiditenormalANDCo2normal = new
FuzzyRuleAntecedent();
  siteperaturenormalEThumiditenormalANDCo2normal-
>joinWithAND(siteperaturenormalEThumiditenormal,
siCo2normal);
 FuzzyRuleConsequent*
thenservomoteurStre_faibleand1ventilateurStre_faible13 = new
FuzzyRuleConsequent();
  thenservomoteurStre_faibleand1ventilateurStre_faible13-
>addOutput(Stre_faible1);
  thenservomoteurStre_faibleand1ventilateurStre_faible13-
>addOutput(Stre_faible2);
  FuzzyRule* fuzzyRule3 = new FuzzyRule(3,
siteperaturenormalEThumiditenormalANDCo2normal ,
thenservomoteurStre_faibleand1ventilateurStre_faible13 );
  fuzzy->addFuzzyRule(fuzzyRule3);
```

Règle 3

```
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtrefaible = new
FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventtrefaible-
>joinWithAND(eleve1, tre_faible4);
  FuzzyRuleAntecedent* sitemperature_extrenmoin = new
FuzzyRuleAntecedent();
  sitemperature_extrenmoin->joinSingle(moin);
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtrefaibleETtemperature_extrenm
oin = new FuzzyRuleAntecedent();

siteperatureeleveETvitessdeventtrefaibleETtemperature_extrenm
oin->joinWithAND(siteperatureeleveETvitessdeventtrefaible,
sitemperature_extrenmoin);

  FuzzyRuleConsequent*
thenservomoteurStre_eleveventilateurS_faible = new
FuzzyRuleConsequent();
  thenservomoteurStre_eleveventilateurS_faible-
>addOutput(Stre_eleve1);
  thenservomoteurStre_eleveventilateurS_faible-
>addOutput(S_faible2);
  FuzzyRule* fuzzyRule4 = new FuzzyRule(4,
siteperatureeleveETvitessdeventtrefaibleETtemperature_extrenm
oin , thenservomoteurStre_eleveventilateurS_faible );
  fuzzy->addFuzzyRule(fuzzyRule4);
```

Règle 4

```
    FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventfaible = new
FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventfaible->joinWithAND(eleve1,
faible4);
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventfaibleETtemperature_extrenmoin
= new FuzzyRuleAntecedent();

siteperatureeleveETvitessdeventfaibleETtemperature_extrenmoin
->joinWithAND(siteperatureeleveETvitessdeventfaible,
sitemperature_extrenmoin);

  FuzzyRuleConsequent*
thenservomoteurS_eleveventilateurS_moyen = new
FuzzyRuleConsequent();
  thenservomoteurS_eleveventilateurS_moyen-
>addOutput(S_eleve1);
  thenservomoteurS_eleveventilateurS_moyen-
>addOutput(S_moyen2);
  FuzzyRule* fuzzyRule5 = new FuzzyRule(5,
siteperatureeleveETvitessdeventfaibleETtemperature_extrenmoin
, thenservomoteurS_eleveventilateurS_moyen );
  fuzzy->addFuzzyRule(fuzzyRule5);
```

Règle 5

```
  FuzzyRuleAntecedent* siteperatureeleveETvitessdeventmoyen =
new FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventmoyen->joinWithAND(eleve1,
moyen4);

  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventmoyenETtemperature_extrenmoin
= new FuzzyRuleAntecedent();

siteperatureeleveETvitessdeventmoyenETtemperature_extrenmoin-
>joinWithAND(siteperatureeleveETvitessdeventmoyen,
sitemperature_extrenmoin);

  FuzzyRuleConsequent*
thenservomoteurS_eleveventilateurS_moyen1 = new
FuzzyRuleConsequent();
  thenservomoteurS_eleveventilateurS_moyen1-
>addOutput(S_moyen1);
  thenservomoteurS_eleveventilateurS_moyen1-
>addOutput(S_moyen2);
  FuzzyRule* fuzzyRule6 = new FuzzyRule(6,
siteperatureeleveETvitessdeventmoyenETtemperature_extrenmoin
, thenservomoteurS_eleveventilateurS_moyen1 );
  fuzzy->addFuzzyRule(fuzzyRule6);
```

Règle 6

```
  FuzzyRuleAntecedent* siteperatureeleveETvitessdeventeleve =
new FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventeleve->joinWithAND(eleve1,
eleve4);
FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventeleveETtemperature_extrenmoin
= new FuzzyRuleAntecedent();
```

Règle 7

```
siteperatureeleveETvitessdeventeleveETtemperature_extrenmoin-
>joinWithAND(siteperatureeleveETvitessdeventeleve,
sitemperature_extrenmoin);
 FuzzyRuleConsequent*
thenservomoteurS_faible1ventilateurS_eleve = new
FuzzyRuleConsequent();
  thenservomoteurS_faible1ventilateurS_eleve-
>addOutput(S_faible1);
  thenservomoteurS_faible1ventilateurS_eleve-
>addOutput(S_eleve2);
  FuzzyRule* fuzzyRule7 = new FuzzyRule(7,
siteperatureeleveETvitessdeventeleveETtemperature_extrenmoin
, thenservomoteurS_faible1ventilateurS_eleve );
  fuzzy->addFuzzyRule(fuzzyRule7);
```

Suite de la Règle7

```
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtreeleve = new
FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventtreeleve-
>joinWithAND(eleve1, tre_eleve4);

  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtreeleveETtemperature_extrenmo
in = new FuzzyRuleAntecedent();

siteperatureeleveETvitessdeventtreeleveETtemperature_extrenmo
in->joinWithAND(siteperatureeleveETvitessdeventtreeleve,
sitemperature_extrenmoin);

  FuzzyRuleConsequent*
thenservomoteurStre_faible1ventilateurStre_eleve = new
FuzzyRuleConsequent();
  thenservomoteurStre_faible1ventilateurStre_eleve-
>addOutput(Stre_faible1);
  thenservomoteurStre_faible1ventilateurStre_eleve-
>addOutput(Stre_eleve2);
  FuzzyRule* fuzzyRule8 = new FuzzyRule(8,
siteperatureeleveETvitessdeventtreeleveETtemperature_extrenmo
in , thenservomoteurStre_faible1ventilateurStre_eleve );
  fuzzy->addFuzzyRule(fuzzyRule8);
```

Règle 8

```
  FuzzyRuleAntecedent* sihumiditeeleveETvitessdeventtrefaible
= new FuzzyRuleAntecedent();
  sihumiditeeleveETvitessdeventtrefaible->joinWithAND(eleve2,
tre_faible4);
FuzzyRuleConsequent*
thenservomoteurStre_eleveventilateurS_faible2 = new
FuzzyRuleConsequent();
  thenservomoteurStre_eleveventilateurS_faible2-
>addOutput(Stre_eleve1);
  thenservomoteurStre_eleveventilateurS_faible2-
>addOutput(S_faible2);
  FuzzyRule* fuzzyRule9 = new FuzzyRule(9,
sihumiditeeleveETvitessdeventtrefaible ,
thenservomoteurStre_eleveventilateurS_faible2 );
  fuzzy->addFuzzyRule(fuzzyRule9);
```

Règle 9

```
  FuzzyRuleAntecedent* sihumiditeeleveETvitessdeventfaible =
new FuzzyRuleAntecedent();
  sihumiditeeleveETvitessdeventfaible->joinWithAND(eleve2,
faible4);
FuzzyRuleConsequent*
thenservomoteurS_eleveventilateurS_moyen3 = new
FuzzyRuleConsequent();
  thenservomoteurS_eleveventilateurS_moyen3-
>addOutput(S_eleve1);
  thenservomoteurS_eleveventilateurS_moyen3-
>addOutput(S_moyen2);
  FuzzyRule* fuzzyRule10 = new FuzzyRule(10,
sihumiditeeleveETvitessdeventfaible ,
thenservomoteurS_eleveventilateurS_moyen3 );
  fuzzy->addFuzzyRule(fuzzyRule10);
```

Règle 10

```
  FuzzyRuleAntecedent* sihumiditeeleveETvitessdeventmoyen =
new FuzzyRuleAntecedent();
  sihumiditeeleveETvitessdeventmoyen->joinWithAND(eleve2,
moyen4);
  FuzzyRuleConsequent*
thenservomoteurS_eleveventilateurS_moyen4 = new
FuzzyRuleConsequent();
  thenservomoteurS_eleveventilateurS_moyen4-
>addOutput(S_moyen1);
  thenservomoteurS_eleveventilateurS_moyen4-
>addOutput(S_moyen2);
  FuzzyRule* fuzzyRule11 = new FuzzyRule(11,
sihumiditeeleveETvitessdeventmoyen ,
thenservomoteurS_eleveventilateurS_moyen4 );
  fuzzy->addFuzzyRule(fuzzyRule11);
```

Règle 11

```
  FuzzyRuleAntecedent* sihumiditeeleveETvitessdeventeleve =
new FuzzyRuleAntecedent();
  sihumiditeeleveETvitessdeventeleve->joinWithAND(eleve2,
eleve4);
  FuzzyRuleConsequent*
thenservomoteurS_faible1ventilateurS_eleve5 = new
FuzzyRuleConsequent();
  thenservomoteurS_faible1ventilateurS_eleve5-
>addOutput(S_faible1);
  thenservomoteurS_faible1ventilateurS_eleve5-
>addOutput(S_eleve2);
  FuzzyRule* fuzzyRule12 = new FuzzyRule(12,
sihumiditeeleveETvitessdeventeleve ,
thenservomoteurS_faible1ventilateurS_eleve5 );
  fuzzy->addFuzzyRule(fuzzyRule12);
```

Règle 12

```
  FuzzyRuleAntecedent* sihumiditeeleveETvitessdeventtreeleve
= new FuzzyRuleAntecedent();
  sihumiditeeleveETvitessdeventtreeleve->joinWithAND(eleve2,
tre_eleve4);
  FuzzyRuleConsequent*
thenservomoteurStre_faible1ventilateurStre_eleve6 = new
FuzzyRuleConsequent();
```

Règle 13

```
  thenservomoteurStre_faible1ventilateurStre_eleve6-
>addOutput(Stre_faible1);
  thenservomoteurStre_faible1ventilateurStre_eleve6-
>addOutput(Stre_eleve2);
  FuzzyRule* fuzzyRule13 = new FuzzyRule(13,
sihumiditeeleveETvitessdeventtreeleve ,
thenservomoteurStre_faible1ventilateurStre_eleve6 );
  fuzzy->addFuzzyRule(fuzzyRule13);
```

Suite de la Règle13

```
  FuzzyRuleAntecedent* siCo2eleveETvitessdeventtrefaible =
new FuzzyRuleAntecedent();
  siCo2eleveETvitessdeventtrefaible->joinWithAND(eleve3,
tre_faible4);

  FuzzyRuleConsequent*
thenservomoteurStre_eleveventilateurS_faible7 = new
FuzzyRuleConsequent();
  thenservomoteurStre_eleveventilateurS_faible7-
>addOutput(Stre_eleve1);
  thenservomoteurStre_eleveventilateurS_faible7-
>addOutput(S_faible2);
  FuzzyRule* fuzzyRule14 = new FuzzyRule(14,
siCo2eleveETvitessdeventtrefaible ,
thenservomoteurStre_eleveventilateurS_faible7 );
  fuzzy->addFuzzyRule(fuzzyRule14);
```

Règle 14

```
  FuzzyRuleAntecedent* siCo2eleveETvitessdeventfaible = new
FuzzyRuleAntecedent();
  siCo2eleveETvitessdeventfaible->joinWithAND(eleve3,
faible4);

  FuzzyRuleConsequent*
thenservomoteurS_eleveventilateurS_moyen8 = new
FuzzyRuleConsequent();
  thenservomoteurS_eleveventilateurS_moyen8-
>addOutput(S_eleve1);
  thenservomoteurS_eleveventilateurS_moyen8-
>addOutput(S_moyen2);
  FuzzyRule* fuzzyRule15 = new FuzzyRule(15,
siCo2eleveETvitessdeventfaible ,
thenservomoteurS_eleveventilateurS_moyen8 );
  fuzzy->addFuzzyRule(fuzzyRule15);
```

Règle 15

```
  FuzzyRuleAntecedent* siCo2eleveETvitessdeventmoyen = new
FuzzyRuleAntecedent();
  siCo2eleveETvitessdeventmoyen->joinWithAND(eleve3, moyen4);
  FuzzyRuleConsequent*
thenservomoteurS_eleveventilateurS_moyen9 = new
FuzzyRuleConsequent();
  thenservomoteurS_eleveventilateurS_moyen9-
>addOutput(S_moyen1);
  thenservomoteurS_eleveventilateurS_moyen9-
>addOutput(S_moyen2);
  FuzzyRule* fuzzyRule16 = new FuzzyRule(16,
siCo2eleveETvitessdeventmoyen ,
thenservomoteurS_eleveventilateurS_moyen9 );
  fuzzy->addFuzzyRule(fuzzyRule16);
```

Règle 16

```
  FuzzyRuleAntecedent* siCo2eleveETvitessdeventeleve = new
FuzzyRuleAntecedent();
  siCo2eleveETvitessdeventeleve->joinWithAND(eleve3, eleve4);
  FuzzyRuleConsequent*
thenservomoteurS_faible1ventilateurS_eleve10 = new
FuzzyRuleConsequent();
  thenservomoteurS_faible1ventilateurS_eleve10-
>addOutput(S_faible1);
  thenservomoteurS_faible1ventilateurS_eleve10-
>addOutput(S_eleve2);
  FuzzyRule* fuzzyRule17 = new FuzzyRule(17,
siCo2eleveETvitessdeventeleve ,
thenservomoteurS_faible1ventilateurS_eleve10 );
  fuzzy->addFuzzyRule(fuzzyRule17);
```

Règle 17

```
  FuzzyRuleAntecedent* siCo2eleveETvitessdeventtreeleve = new
FuzzyRuleAntecedent();
  siCo2eleveETvitessdeventtreeleve->joinWithAND(eleve3,
tre_eleve4);
  FuzzyRuleConsequent*
thenservomoteurStre_faible1ventilateurStre_eleve11 = new
FuzzyRuleConsequent();
  thenservomoteurStre_faible1ventilateurStre_eleve11-
>addOutput(Stre_faible1);
  thenservomoteurStre_faible1ventilateurStre_eleve11-
>addOutput(Stre_eleve2);
  FuzzyRule* fuzzyRule18 = new FuzzyRule(18,
siCo2eleveETvitessdeventtreeleve ,
thenservomoteurStre_faible1ventilateurStre_eleve11 );
  fuzzy->addFuzzyRule(fuzzyRule18);
```

Règle 18

```
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtrefaible19 = new
FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventtrefaible19-
>joinWithAND(eleve1, tre_faible4);
  FuzzyRuleAntecedent* sitemperature_extrenmeme = new
FuzzyRuleAntecedent();
  sitemperature_extrenmeme->joinSingle(meme);
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtrefaibleETtemperature_extrenm
eme = new FuzzyRuleAntecedent();

siteperatureeleveETvitessdeventtrefaibleETtemperature extrenm
eme->joinWithAND(siteperatureeleveETvitessdeventtrefaible19,
sitemperature_extrenmeme);

  FuzzyRuleConsequent*
thenservomoteurStre_eleveventilateurS_faible19 = new
FuzzyRuleConsequent();
  thenservomoteurStre_eleveventilateurS_faible19-
>addOutput(Stre_eleve1);
  thenservomoteurStre_eleveventilateurS_faible19-
>addOutput(S_faible2);
  FuzzyRule* fuzzyRule19 = new FuzzyRule(19,
siteperatureeleveETvitessdeventtrefaibleETtemperature_extrenm
oin , thenservomoteurStre_eleveventilateurS_faible19 );
  fuzzy->addFuzzyRule(fuzzyRule19);
```

Règle 19

```
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventfaible20 = new
FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventfaible20-
>joinWithAND(eleve1, faible4);
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventfaibleETtemperature_extrenmeme
= new FuzzyRuleAntecedent();
siteperatureeleveETvitessdeventfaibleETtemperature_extrenmeme
->joinWithAND(siteperatureeleveETvitessdeventfaible20,
sitemperature_extrenmeme);
  FuzzyRuleConsequent*
thenservomoteurS_eleveventilateurS_moyen20 = new
FuzzyRuleConsequent();
  thenservomoteurS_eleveventilateurS_moyen20-
>addOutput(S_eleve1);
  thenservomoteurS_eleveventilateurS_moyen20-
>addOutput(S_moyen2);
  FuzzyRule* fuzzyRule20 = new FuzzyRule(20,
siteperatureeleveETvitessdeventfaibleETtemperature_extrenmeme
, thenservomoteurS_eleveventilateurS_moyen20 );
  fuzzy->addFuzzyRule(fuzzyRule20);
```

Règle 20

```
  FuzzyRuleAntecedent* siteperatureeleveETvitessdeventmoyen21
= new FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventmoyen21->joinWithAND(eleve1,
moyen4);
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventmoyenETtemperature_extrenmeme
= new FuzzyRuleAntecedent();
siteperatureeleveETvitessdeventmoyenETtemperature_extrenmeme-
>joinWithAND(siteperatureeleveETvitessdeventmoyen21,
sitemperature_extrenmeme);
  FuzzyRuleConsequent*
thenservomoteurS_eleveventilateurS_moyen21 = new
FuzzyRuleConsequent();
  thenservomoteurS_eleveventilateurS_moyen21-
>addOutput(S_moyen1);
  thenservomoteurS_eleveventilateurS_moyen21-
>addOutput(S_moyen2);
  FuzzyRule* fuzzyRule21 = new FuzzyRule(21,
siteperatureeleveETvitessdeventmoyenETtemperature_extrenmeme
, thenservomoteurS_eleveventilateurS_moyen21 );
  fuzzy->addFuzzyRule(fuzzyRule21);
```

Règle 21

```
  FuzzyRuleAntecedent* siteperatureeleveETvitessdeventeleve22
= new FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventeleve22->joinWithAND(eleve1,
eleve4);
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventeleveETtemperature extrenmeme
= new FuzzyRuleAntecedent();
siteperatureeleveETvitessdeventeleveETtemperature_extrenmeme-
>joinWithAND(siteperatureeleveETvitessdeventeleve22,
sitemperature_extrenmeme);
  FuzzyRuleConsequent*
thenservomoteurS_faible1ventilateurS_eleve22 = new
FuzzyRuleConsequent();
```

Règle 22

```
  thenservomoteurS_faible1ventilateurS_eleve22-
>addOutput(S_faible1);
  thenservomoteurS_faible1ventilateurS_eleve22-
>addOutput(S_eleve2);
  FuzzyRule* fuzzyRule22 = new FuzzyRule(22,
siteperatureeleveETvitessdeventeleveETtemperature_extrenmeme
, thenservomoteurS_faible1ventilateurS_eleve22 );
  fuzzy->addFuzzyRule(fuzzyRule22);
```

Suite de la Règle22

```
    FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtreeleve23 = new
FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventtreeleve23-
>joinWithAND(eleve1, tre_eleve4);
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtreeleveETtemperature_extrenme
me = new FuzzyRuleAntecedent();
siteperatureeleveETvitessdeventtreeleveETtemperature_extrenme
me->joinWithAND(siteperatureeleveETvitessdeventtreeleve23,
sitemperature_extrenmeme);
  FuzzyRuleConsequent*
thenservomoteurStre_faible1ventilateurStre_eleve23 = new
FuzzyRuleConsequent();
  thenservomoteurStre_faible1ventilateurStre_eleve23-
>addOutput(Stre_faible1);
  thenservomoteurStre_faible1ventilateurStre_eleve23-
>addOutput(Stre_eleve2);
  FuzzyRule* fuzzyRule23 = new FuzzyRule(23,
siteperatureeleveETvitessdeventtreeleveETtemperature_extrenme
me , thenservomoteurStre_faible1ventilateurStre_eleve23 );
  fuzzy->addFuzzyRule(fuzzyRule23);
```

Règle 23

```
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtrefaible24 = new
FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventtrefaible24-
>joinWithAND(eleve1, tre_faible4);
  FuzzyRuleAntecedent* sitemperature_extrensupieur = new
FuzzyRuleAntecedent();
  sitemperature_extrensupieur->joinSingle(supieur);
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtrefaibleETtemperature_extrens
upieur = new FuzzyRuleAntecedent();
siteperatureeleveETvitessdeventtrefaibleETtemperature extrens
upieur-
>joinWithAND(siteperatureeleveETvitessdeventtrefaible24,
sitemperature_extrensupieur);
  FuzzyRuleConsequent*
thenservomoteurStre_eleveventilateurS_faible24 = new
FuzzyRuleConsequent();
  thenservomoteurStre_eleveventilateurS_faible24-
>addOutput(Stre_eleve1);
  thenservomoteurStre_eleveventilateurS_faible24-
>addOutput(S_faible2);
  FuzzyRule* fuzzyRule24 = new FuzzyRule(24,
siteperatureeleveETvitessdeventtrefaibleETtemperature_extrens
upieur , thenservomoteurStre_eleveventilateurS_faible24 );
  fuzzy->addFuzzyRule(fuzzyRule24);
```

Règle 24

```
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventfaible25 = new
FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventfaible25-
>joinWithAND(eleve1, faible4);
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventfaibleETtemperature_extrensupi
eur = new FuzzyRuleAntecedent();
siteperatureeleveETvitessdeventfaibleETtemperature_extrensupi
eur->joinWithAND(siteperatureeleveETvitessdeventfaible25,
sitemperature_extrensupieur);
  FuzzyRuleConsequent*
thenservomoteurS_eleveventilateurS_moyen25 = new
FuzzyRuleConsequent();
  thenservomoteurS_eleveventilateurS_moyen25-
>addOutput(S_eleve1);
  thenservomoteurS_eleveventilateurS_moyen25-
>addOutput(S_moyen2);
  FuzzyRule* fuzzyRule25 = new FuzzyRule(25,
siteperatureeleveETvitessdeventfaibleETtemperature_extrensupi
eur , thenservomoteurS_eleveventilateurS_moyen25 );
  fuzzy->addFuzzyRule(fuzzyRule25);
```

Règle 25

```
  FuzzyRuleAntecedent* siteperatureeleveETvitessdeventmoyen26
= new FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventmoyen26->joinWithAND(eleve1,
moyen4);
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventmoyenETtemperature_extrensupie
ur = new FuzzyRuleAntecedent();
siteperatureeleveETvitessdeventmoyenETtemperature_extrensupie
ur->joinWithAND(siteperatureeleveETvitessdeventmoyen26,
sitemperature_extrensupieur);
  FuzzyRuleConsequent*
thenservomoteurS_eleveventilateurS_moyen26 = new
FuzzyRuleConsequent();
  thenservomoteurS_eleveventilateurS_moyen26-
>addOutput(S_moyen1);
  thenservomoteurS_eleveventilateurS_moyen26-
>addOutput(S_moyen2);
  FuzzyRule* fuzzyRule26 = new FuzzyRule(26,
siteperatureeleveETvitessdeventmoyenETtemperature_extrensupie
ur , thenservomoteurS_eleveventilateurS_moyen26 );
  fuzzy->addFuzzyRule(fuzzyRule26);
```

Règle 26

```
  FuzzyRuleAntecedent* siteperatureeleveETvitessdeventeleve27
= new FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventeleve27->joinWithAND(eleve1,
eleve4);
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventeleveETtemperature_extrensupie
ur = new FuzzyRuleAntecedent();
siteperatureeleveETvitessdeventeleveETtemperature_extrensupie
ur->joinWithAND(siteperatureeleveETvitessdeventeleve27,
sitemperature_extrensupieur);
```

Règle 27

```
  FuzzyRuleConsequent*
thenservomoteurS_faible1ventilateurS_eleve27 = new
FuzzyRuleConsequent();
  thenservomoteurS_faible1ventilateurS_eleve27-
>addOutput(S_faible1);
  thenservomoteurS_faible1ventilateurS_eleve27-
>addOutput(S_eleve2);
  FuzzyRule* fuzzyRule27 = new FuzzyRule(27,
siteperatureeleveETvitessdeventeleveETtemperature_extrensupie
ur , thenservomoteurS_faible1ventilateurS_eleve27 );
  fuzzy->addFuzzyRule(fuzzyRule27);
```

Suite de la Règle27

```
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtreeleve28 = new
FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventtreeleve28-
>joinWithAND(eleve1, tre_eleve4);
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtreeleveETtemperature_extrensu
pieur = new FuzzyRuleAntecedent();
siteperatureeleveETvitessdeventtreeleveETtemperature_extrensu
pieur->joinWithAND(siteperatureeleveETvitessdeventtreeleve28,
sitemperature_extrensupieur);
  FuzzyRuleConsequent*
thenservomoteurStre_faible1ventilateurStre_eleve28 = new
FuzzyRuleConsequent();
  thenservomoteurStre_faible1ventilateurStre_eleve28-
>addOutput(Stre_faible1);
  thenservomoteurStre_faible1ventilateurStre_eleve28-
>addOutput(Stre_eleve2);
  FuzzyRule* fuzzyRule28 = new FuzzyRule(28,
siteperatureeleveETvitessdeventtreeleveETtemperature_extrensu
pieur , thenservomoteurStre_faible1ventilateurStre_eleve28 );
  fuzzy->addFuzzyRule(fuzzyRule28);
```

Règle 28

```
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtreeleve29 = new
FuzzyRuleAntecedent();
  siteperatureeleveETvitessdeventtreeleve29-
>joinWithAND(eleve1, tre_eleve4);
  FuzzyRuleAntecedent*
siteperatureeleveETvitessdeventtreeleveETtemperature_extrensu
pieur1 = new FuzzyRuleAntecedent();
siteperatureeleveETvitessdeventtreeleveETtemperature extrensu
pieur->joinWithAND(siteperatureeleveETvitessdeventtreeleve28,
sitemperature_extrensupieur);
  FuzzyRuleConsequent*
thenservomoteurStre_faible1ventilateureleve29 = new
FuzzyRuleConsequent();
  thenservomoteurStre_faible1ventilateureleve29-
>addOutput(Stre_faible1);
  thenservomoteurStre_faible1ventilateureleve29-
>addOutput(S_eleve2);
  FuzzyRule* fuzzyRule29 = new FuzzyRule(29,
siteperatureeleveETvitessdeventtreeleveETtemperature_extrensu
pieur1 , thenservomoteurStre_faible1ventilateureleve29 );
  fuzzy->addFuzzyRule(fuzzyRule29);
```

Règle 29

```
}

void fuzzifier() {

  fuzzy->setInput(1, t);
  fuzzy->setInput(2, h);
  fuzzy->setInput(3, co2);
  fuzzy->setInput(4, vent);
  fuzzy->setInput(5, temp_ext);

fuzzy->fuzzify();

  volets = fuzzy->defuzzify(1);
  servo_driver(volets);
  extracteur = fuzzy->defuzzify(2);
  extra_driver(byte(extracteur));

}
```

Déclaration de la fonction de la Fuzzyfication et la défuzzification des entrées sorties

Ajout au régulateur floue les entées fuzzifier

Fuzzyfication des entrées

La défuzzification et l'ajout des sorties au régulateur floue