

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne démocratique et populaire

وزارة التعليم العالي و البحث العلمي
Ministère de l'enseignement supérieur et de la recherche scientifique

جامعة سعد دحلب البليدة
Université SAAD DAHLAB de BLIDA

كلية التكنولوجيا
Faculté de Technologie

قسم الإلكترونيك
Département d'Électronique



Mémoire de Projet de Fin d'Études

Pour l'obtention du diplôme de Master en Système et Télécommunications

Thème

Etude et conception d'un système d'acquisition multi
capteurs pour le contrôle et la surveillance sur site ou à
distance

Présenté par

MENAD ABDELFETAH & KHERIF OUSSAMA

Proposé par :

L'encadreur : DERGUINI NOUREDDINE.
Le Co-promoteur : MOUNTASSAR MAAMOUN.

Année Universitaire 2020-2021

Remerciement

Tout d'abord, nous remercions dieu qui nous a permis d'atteindre ce succès, qui nous a donné la force et la volonté pour atteindre nos objectifs.

Nous voulons offrir tout merci et appréciation a notre dirigeant au niveau de centre de développement des technologies avancée, Mr : Derguini et notre dirigeant au niveau de l'université Dr : M.Maamoun pour leurs aides et leurs conseils durant toute la période de l'achèvement de ce projet

Nous remercions surtout nos parents pour tout ce qu'ils ont fait pour nous, tout ce qu'ils nous ont donné pour atteindre ce succès

On tient aussi à remercier après ce travail tous ceux qui ont joué un rôle dans ce succès de prêt ou de loin, la famille, les amis et les enseignants

Dédicace

Je dédie ce travail à mes chers parents qui nous ont tout données dès le début de notre parcours jusqu'à la fin, que dieu les bénis et les protège

A mes cher frères, sœurs qui nous ont encouragé et soutenu au long de notre parcours

je dédie spécialement mon binôme Oussama kherif qui a joué un grand rôle pour réaliser ce projet jusqu'à la fin

mes camarades Abderrahmane, Amar, Fouad, Abdelmounime et Oussama

mes chers amis durant les deux ans préparatoires au niveau de ENSH Khaled, Abdelatif et Imad.

Abdelfetah

Dédicace

Je dédie cette thèse à mes chers parents, aucune dédicace ne saurait exprimer mon respect à eux ma considération pour leurs soutiens, leurs sacrifices et leurs aides.

A mes cher adorables frère et sœurs qui m'ont aidé durant toute cette période

A mes amis et camarades , Abdelfetah, Abderrahmane, Amar, Lounis.

A mon binôme Abdelfetah qui a joué un grand rôle pour réaliser ce projet jusqu'à la fin

Une spéciale dédicace mon cher frère Fouad qui compté et qui compte encore énormément pour moi, tu as partagé avec moi les meilleures parties de ma vie, j'aurai tant aimé que tu sois présent pour ce succès, que dieu ai vos âmes dans sa sainte miséricorde.

Oussama

Résumé

Ce projet de Master 2 consiste à concevoir un système capable de faire des mesures en temps réel des phénomènes physiques existant travers une application qui permettra de contrôler une Plate-forme expérimentale (exemple : chauffage, ventilation...etc.) via le réseau WIFI ou bien une application web.

L'application en question doit permettre de visualiser les données issues des différents capteurs de la Plate-forme expérimentale (capteur de pression, capteur de température, capteur d'humidité, etc.), sur site ou distance.

Mots clés : phénomènes physiques, Plate-forme expérimentale, Capteurs, application

الملخص

يعتبر هذا المشروع , مشروع سنة ثانية ماستر, يهدف الى تصميم نظام قادر على إجراء قياسات في الوقت الفعلي للظواهر الفيزيائية من خلال تطبيق يسمح بالتحكم في منصة تجريبية (على سبيل المثال: التدفئة والتهوية وما إلى ذلك) عبر شبكة WIFI أو تطبيق ويب. يجب أن يتيح التطبيق المعني إمكانية تصور البيانات من أجهزة الاستشعار المختلفة للمنصة التجريبية (مستشعر الضغط ، ومستشعر درجة الحرارة ، ومستشعر الرطوبة ، وما إلى ذلك) ، في الموقع أو عن بُعد.

الكلمات المفتاحية: الظواهر الفيزيائية ، المنصة التجريبية ، الحساسات ، التطبيق

Abstract

This Master 2 project consists of designing a system capable of making real-time measurements of existing physical phenomena through an application that will allow the control of an experimental platform (example: heating, ventilation, etc.) via the WIFI network or else A web application.

The application in question must make it possible to visualize the data from the various sensors of the Experimental Platform (pressure sensor, temperature sensor, humidity sensor, etc.), on site or remotely.

Keywords: physical phenomena, Experimental platform, Sensors, application

AES : Advanced Encryption Standard

API : Application Programming Interface

AVR : Automatic Voltage Regulator

DC : Direct Current

DDR2 : Double Data Rate Two

EEPROM : Electrically Erasable Programmable Read-Only Memory

FTP : File Transfer Protocol

GND : Global Nearness Diagram

GPIO : General Purpose Input/Output

GPS : Global Position System

HTTP : Hyper Text Transport Protocol

I2C : Inter Integrated Circuit

I2S : Integrated Interchip Sound

IDE : Integrated Development Environment

IEEE : Institut of Electrical and Electronic Engineers

IPV4 : Internet Protocol Version 4

IPV6 : Internet Protocol Version 6

IR : Infra-rouge

LCD : Liquid Crystal Display

LiFi : Light Fidelity

LPDDR2 : Low Power Double Data Rate Two

LP-WPAN : Low Power Wide Area Network

Mac : Medium Access Control

MQTT : Message Queuing Telemetry Transport

NFC : Near-Field communication

noSQL : not only Structured Query Language

OSI : Open System Interconnection

OTA : Over The Air

PC : Personal Computer

PCB : Printed Circuit Board

PWM : Pulse Width Modulation

RAM : Random Acces Memory

RFID : Radio-Frequency Identification

ROM : Read Only Memory

SDIO : secure Digital Input Output

SDK : Software Development Kit

SPP : Serie Port Protocol

SRAM : Static Random Access Memory

SPI : Serial Peripheral Interface

SSL : Secure Socket Layer

TCP : Transmission Control Protocol

TOR : Tout ou Rien

TTL : Transistor-Transistor Logic

UART : Universal Asynchronous Receiver Transmitter

UDP : User datagram Protocol

URL : Uniform Resource Locator

WEP : Wired Equivalent Privacy

Wi-Fi : Wireless Fidelity

WPA : WI-Fi Protected Access

Table des matières

<u>Introduction générale</u>	1
<u>Chapitre 01</u>	2
<u>Notation sur le projet</u>	2
<u>1.Introduction</u>	2
<u>2.Structure générale de système</u>	2
<u>2.1.Description du système</u>	2
<u>2.2.Les capteurs</u>	3
<u>2.2.1.Catégories des capteurs</u>	3
<u>2.2.2.Classification des capteurs</u>	4
<u>2.3.Unité de traitement des données</u>	5
<u>2.3.1.Comparaison entre les cartes de traitement</u>	5
<u>2.3.2.Choix de carte</u>	8
<u>2.4.Connectivité sans fil</u>	8
<u>2.4.1.Protocole Wifi 802.11</u>	8
<u>2.4.2.Protocole Bluetooth IEEE 802.15.1</u>	11
<u>2.4.3.Protocole Zigbee IEEE 802.15.4</u>	12
<u>3.Conclusion</u>	13
<u>Chapitre 02</u>	14
<u>MATERIELS ET LOGICIEL</u>	14
<u>1.Introduction</u>	14
<u>2.Carte arduino</u>	14
<u>2.1.Les caractéristiques de la carte Arduino uno</u>	15
<u>2.2.La surface d'IDE arduino</u>	17
<u>3.Les composantes électroniques</u>	18
<u>3.1.Les capteurs utilisés</u>	18
<u>3.1.1.Capteur d'humidité et de température (DHT11)</u>	18
<u>3.1.2.Le capteur de pression BMP180</u>	19
<u>3.2.les composantes d'affichage et de contrôle des phénomènes</u>	21
<u>3.2.1.l'écran LCD (16*2)</u>	21
<u>3.2.2.Ventilateurs</u>	22
<u>3.2.3.Modules de relais</u>	23
<u>3.3.Le composant de communication sans fil (ESP8266)</u>	24
<u>3.3.1.Description de ESP82566</u>	24
<u>3.3.2.Les caractéristiques de ESP8266</u>	25
<u>3.3.3.Les broches de ESP8266</u>	26

<u>4.Conclusion :</u>	27
<u>Chapitre 03</u>	28
<u>SIMULATION ET REALISATION DE PROJET</u>	28
<u>1.Introduction</u>	28
<u>2.Matériel et logiciel utilisé</u>	29
<u>3.Principe de fonctionnement</u>	29
<u>4.Caractéristiques des composants</u>	31
4.1.DHT11	31
4.2.BMP180	31
4.3.Ecran LCD	31
4.4.Relais (le module relais)	31
4.4.1.Côté de commande	31
4.4.2.Côté principal	31
4.5.Arduino uno	32
4.6.Ventilateur (moteur)	32
<u>5.organigramme du système</u>	32
<u>6.Simulation et réalisation</u>	33
6.1.Simulation	33
6.2.Etapes de simulation	33
6.3.Schéma de simulation	33
<u>7.Description de simulation</u>	34
<u>8.Résultats de simulation :</u>	34
<u>9.Réalisation du système</u>	37
<u>10.Configuration de la carte Arduino avec les composantes</u>	38
<u>11.Partie matérielle</u>	38
<u>12.Etapes de montage avec carte Arduino (sur site)</u>	39
12.1.Résultats de montage	40
12.2.Récupération des données avec Matlab	41
12.2.1. Organigramme de communication Matlab-Arduino	42
12.2.2. Affichage des données sur Matlab	42
<u>13.Etapes de montage avec Node MCU (à distance)</u>	43
13.1.Résultat de montage	45
<u>14.Application mobile et réalisation finale</u>	45
14.1.Introduction	45
14.2.Définition du système d'exploitation mobile 'Android'	45

<u>14.3.Firebase</u>	45
<u>14.4.Création d'un projet Firebase</u>	46
<u>14.5.Environment de développement MIT App Inventor</u>	47
<u>14.5.1.Création de l'application mobile</u>	47
<u>15.Récupération des données de notre montage</u>	50
<u>15.1.Résultats d'affichage</u>	50
<u>16.Conclusion</u>	52
<u>Conclusion générale</u>	52
<u>Annexes</u>	54
<u>Références bibliographiques</u>	63

Liste des figures

Chapitre 1

FIGURE 1.1 : Schéma synoptique du système d'acquisition.....	2
FIGURE 1.2 : PRINCIPE DES CAPTEURS	3
FIGURE 1.3 : SIGNAL DE SORTIE DES CAPTEURS.....	4
FIGURE 1.4 : ARDUINO MEGA	5
FIGURE 1.5 : ARDUINO YUN.	6
FIGURE 1.6 : ARDUINO UNO	6
FIGURE 1.7 : NETDUINO	7
FIGURE 1.8 : RASPBERRY PI 3 B+	8
FIGURE 1.9 : LES COUCHES DE RESEAU WIFI	9
FIGURE 1.10 : ESP8266.....	10
FIGURE 1.11 : NODE MCU	10
FIGURE 1.12 : LES COUCHES DE RESEAU BLUETOOTH	11
FIGURE 1.13 : MODULE BLUETOOTH HC-05.....	12
FIGURE 1.14 : MODULE BLUETOOTH (CSR8645)	13

Chapitre 2

FIGURE 2.1 : CARTE ARDUINO UNO.....	14
FIGURE 2.2 : MICROCONTROLEUR ATmega328	15
FIGURE 2.3 : LE CAPTEUR DHT11.....	18
FIGURE 2.4 : BMP180	20
FIGURE 2.5 : AFFICHER LCD 16*2	22
FIGURE 2.6 : VENTILATEUR 12V.	23
FIGURE 2.7 : RELAIS 5V.	23
FIGURE 2.8 : MODULE CARTE WIFI ESP8266.	24
FIGURE 2.9 : LES BROCHES DE ESP8266.....	26

Chapitre 3

FIGURE 3.1 : SCHEMA DE SYSTEME D'ACQUISITION	28
FIGURE 3.2 : SCHEMA SYNOPTIQUE DE SYSTEME.....	30

FIGURE 3. 3: ORGANIGRAMME DE DHT11.....	32
FIGURE 3. 4: SCHEMA DE SIMULATION SUR ISIS PROTEUS.....	33
FIGURE 3. 5: L'ESPACE ARDUINO IDE.....	34
FIGURE 3. 6: LIEN HEXADECIMALE DE CODE ARDUINO.....	35
FIGURE 3. 7: L'EMPLACEMENT DE FICHIER DE CODE ARDUINO.....	35
FIGURE 3. 8: PREMIERE RESULTAT DE SIMULATION.....	36
FIGURE 3. 9: DEUXIEME RESULTAT DE SIMULATION.....	36
FIGURE 3. 10: SCHEMA SYNOPTIQUE DE FONCTIONNEMENT DE MONTAGE.....	37
FIGURE 3. 11: LES ETAPES DE CONFIGURATION DES COMPOSANTES AVEC ARDUINO.....	38
FIGURE 3. 12: MONTAGE DE ARDUINO- DHT11-VENTILATEUR-LCD.....	40
FIGURE 3. 13: RESULTATS DE MONTAGE DHT11-LCD-VENTILATEUR.....	41
FIGURE 3. 14: RESULTATS DE MONTAGE DHT11-ESP8266-BMP180-LCD-VENTILATEURS.....	41
FIGURE 3. 15: ORGANIGRAMME DE COMMUNICATION SERIE MATLAB-ARDUINO.....	42
FIGURE 3. 16: ORGANIGRAMME DE COMMUNICATION SERIE MATLAB-ARDUINO.....	43
FIGURE 3. 17: MONTAGE NODE MCU DHT11 BMP180.....	44
FIGURE 3. 18: RESULTATS DE MONTAGE NODE MCU DH11 BMP180.....	44
FIGURE 3. 19: ORGANIGRAMME DE TRANSFERT DES DONNEES.....	50
FIGURE 3. 20: LES DONNEES DANS FIREBASE.....	50
FIGURE 3. 21: LES DONNEES SUR L'APPLICATION.....	51

Liste des tableaux

TABLEAU 3. 1: MATERIELS ET LOGICIELS UTILISES.....	29
---	----

Introduction générale

Dans la vie quotidienne, on constate toujours dans de nombreux domaines des problèmes d'estimation et d'évaluation des phénomènes météorologiques tels que la température, l'humidité et la pression, par exemple, dans le domaine de la médecine, ces variables doivent être soigneusement prises en compte, ainsi que dans le domaine de la l'industrie ou même dans la vie quotidienne, nous avons donc besoin d'évaluer les phénomènes météorologiques qui touchent également de nombreux autres domaines, tels que l'environnement, l'architecture, la conversion d'énergie...etc.

En s'appuyant sur le développement technologique qui a permis de développer de nouveaux outils et technologies qui peuvent nous permettre d'évaluer et de mesurer ces phénomènes avec précision et haute performance sans aucun soucis à l'aide des logiciels et des composantes électroniques et de l'internet qui est considéré comme l'élément le plus important de cette révolution technologique et qui passe les communications entre les différentes destinations via des protocoles de communication (TCP,UDP.....etc.), et suivants des technologies différentes aussi comme Wifi, Zigbee, LoRa, Bluetooth...etc.

Dans le cadre de notre projet, nous essayons de réaliser système capable de faire des mesures de température et d'humidité et de pression et d'assurer la visualiser et le contrôle via un réseau wifi et à l'aide d'une application web, ce système est composé des capteurs pour la mesure des grandeurs physique, d'une carte à microcontrôleur Arduino UNO pour l'acquisition et le traitement des données et d'un module Wifi permettant le transfert des grandeurs mesurées vers un ordinateur distant via internet.

Ce mémoire est organisé en trois chapitres répartis comme suit :

Le premier chapitre est consacré aux notions générales des systèmes d'acquisition multi capteurs, la structure générale de système et des différents outils qu'on peut utiliser pour la réalisation de notre système

Dans le deuxième chapitre exposera la conception des différents blocs de notre système et les composants utiliser dans le montage, notamment l'alimentation, le contrôleur, les capteurs, les relais et les branchements des liaisons entre les différentes parties qui constituent la réalisation de notre projet.

Le dernier chapitre traitera la programmation, la simulation et réalisation de notre système ainsi que les tests et les résultats obtenus sur site et à distance.

Enfin, une conclusion générale et perspective

Chapitre 01

Notation sur le projet

1. Introduction

Avant de commencer le travail, il nous faut tout d'abord commencer à éclairer les chemins de travail et de réalisations du système, et aussi comprendre la logique de fonctionnement de notre système et qu'elles sont les différentes composantes et méthodes qu'on pourrait utiliser, et tout pour bien choisir les méthodes, les besoins et les composantes des bases pour engager la réalisation.

2. Structure générale de système

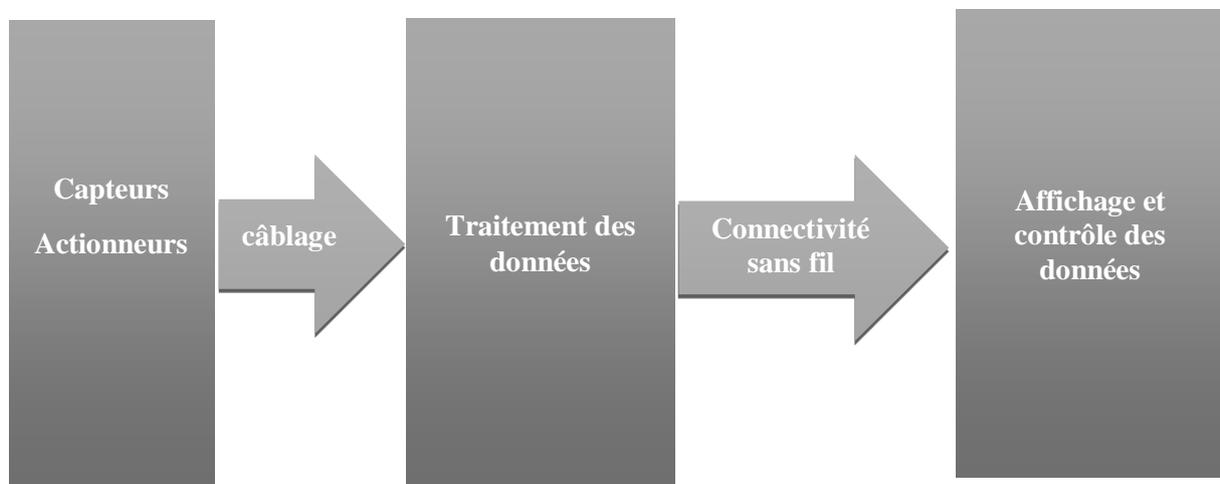


Figure 1. 1: Schéma synoptique du système d'acquisition

2.1. Description du système

Comme l'objectif de notre travail est de réaliser un système qui est capable de faire des mesures en temps réel des phénomènes physiques, et les contrôler à distance ou sur site, le figure (1.1) montre la structure de ce système.

Ce type des systèmes est diviser en trois unités qui communiquent entre elles suivant des protocoles de communications filaires ou sans fils.

Comme première étape, nous adopterons la liaison filaire via l'unité de traitement suivant des protocoles de communications (I2c, SPI,UART), pour le transfert des données issus des capteurs. Une fois la communication est établie, nous passerons à la

deuxième étape, en utilisant la liaison sans fil suivant des protocoles communications (Wifi, Bluetooth, zigbee, RFID, NFC, LiFi...etc.)

2.2. Les capteurs

Les capteurs sont des dispositifs électroniques qui servent à convertir une grandeur physique analogique (pression, température, humidité, lumière...etc.) en un signal logique, analogique ou numérique rendu transmissible et exploitable par un système de conditionnement, la figure 1.2 illustre ce principe.

Le capteur fait une partie d'une chaîne de mesure qui se trouve au contact direct du mesurande, La tension ou l'intensité de signal électrique est l'image du mesurande par une loi continue qu'on souhaite linéaire.[2]

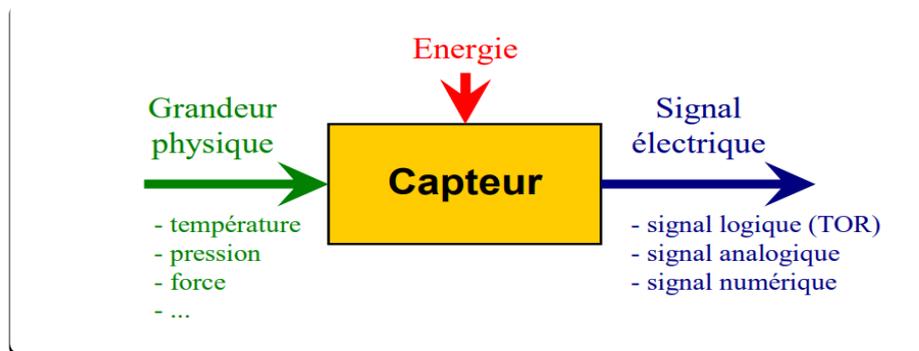


Figure 1. 2: principe des capteurs

2.2.1. Catégories des capteurs

○ Les capteurs numériques

Les capteurs numériques sont des capteurs qui donnent en sortie une valeur finie en sortie, ils sont hautement sensibles, par exemple, si une grandeur physique croît de manière linéaire, la sortie du capteur qui va la mesurer donnera soit une information du type "TOR " (Tout Ou Rien), un train d'impulsion ou un échantillonnage.[3]

○ Les capteurs analogiques

Les capteurs analogiques servent à convertir une grandeur physique en un autre type soit impédance, capacité, inductance ou de tension. Un signal est dit analogique si l'amplitude de la grandeur physique qu'il représente peut prendre une infinité de valeurs dans un intervalle donné.

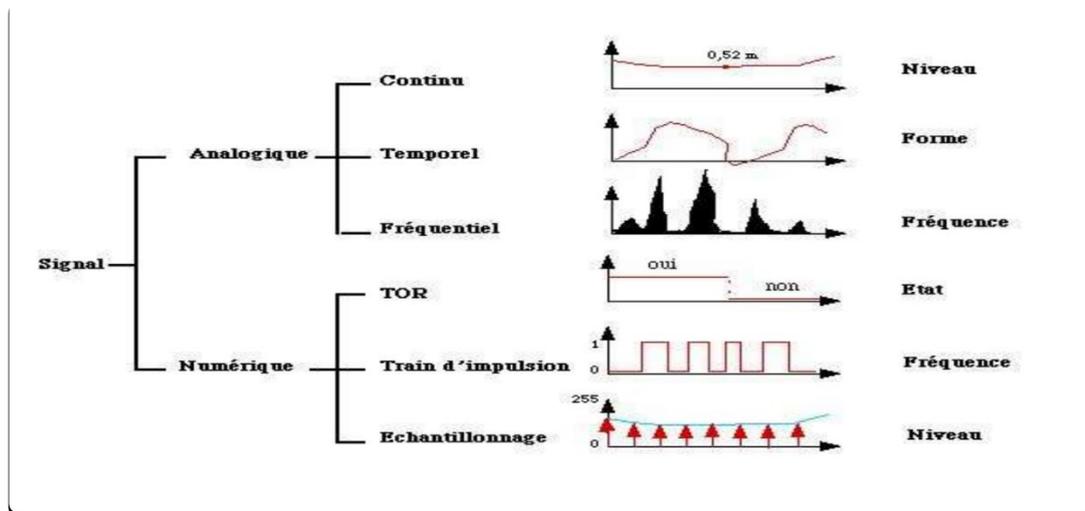


Figure 1. 3 : signal de sortie des capteurs.

2.2.2. Classification des capteurs

La classification des capteurs suit des critères qui aident à choisir les capteurs avec précision, tout dépend le domaine d'utilisation et le milieu de travail.

Nous pouvons classer les capteurs suivant les critères suivants.

- le type de mesurande qu'il peut capter (capteur d'humidité, de température, de pression...etc.).
- Par son rôle dans le processus industriel (contrôle de produit finis, de sécurité...etc.).
- Par le signal qu'il fournit en sortie qui peut être numérique, analogique, logique .
- Par leur principe de traduction du mesurande (capteur résistif, capteur capacitif, piézoélectrique...etc.).
- Par leur principe de fonctionnement, capteur Actif ou Passif.

- Par la précision et la plage d'utilisation qu'il offre.

2.3. Unité de traitement des données

L'unité de traitement s'appuie principalement sur les microcontrôleurs grâce à ces caractéristiques (RAM, ROM, la fréquence de l'horloge), ou il est disponible dans toutes les cartes de traitement comme Arduino, Raspberry, netduino...etc.

2.3.1. Comparaison entre les cartes de traitement

Pour choisir une carte de traitement il faut connaître ces caractéristique et ses privilèges et aussi ces avantages et inconvénients.

- **Arduino Méga**
- alimentation : via port USB ou 7 à 12 V.
- connecteur d'alimentation.
- microprocesseur:ATMega2560.
- mémoire flash : 256 kb.
- 54 broches d'E/S dont 14 PWM.
- 16 entrées analogiques 10 bits.
- intensité par E/S : 40 mA.
- cadencement : 16 MHz.
- gestion des interruptions.
- fiche USB.
- Prix : 4800 DA.[4]



Figure 1. 4 :Arduino Méga

- **Arduino Yun**
- mémoire RAM : 64 Mb DDR2.
- mémoire flash : 32 kb.
- 20 broches d'E/S dont 6 PWM.
- 6 entrées analogiques 10 bits.
- intensité par E/S : 40 mA.
- cadencement : 16 MHz .
- bus série, I2C et SPI.
- gestion des interruptions.
- fiche micro USB.
- 1 port Ethernet, 1 port USB et 1 port micro-SD.
- Dimensions : 72 x 53 x 17 mm
- Prix : 4800 DA. [5]



Figure 1. 5: Arduino YUN.

- **Arduino Uno**
- Alimentation : via port USB ou 7 à 12 V.
- Microprocesseur : ATmega328.
- Mémoire flash : 32 kb.
- 14 broches d'E/S dont 6 PWM.
- 6 entrées analogiques 10 bits.
- Intensité par E/S : 40 mA.
- Cadencement : 16 MHz.
- Bus série, I2C et SPI gestion des interruptions.
- fiche USB B - dimensions : 74 x 53 x 15 mm.



Figure 1. 6: Arduino Uno

- Prix : 2500 DA.[6]

- **Netduino**

- Vitesse : 120MHz.
- Mémoire Flash : 192 KB.
- RAM : 60 Ko.
- I_{max} : 25 mA par broche.
- Alimentation : 3.3V mais 5V support.
- Prix : 10000 DA.[7]



Figure 1. 7: Netduino

- **Raspberry Pi 3 B+**

- Processeur : Broadcom BCM2837B0 A53 64-bit 4GHz.
- Mémoire : 1Go LPDDR2.
- Port USB: 4 ports USB2.
- Alimentation: 5V, 2,5A.
- Wifi : 2,4Ghz et 5Ghz, 802.11.b/g/n/ac.
- Bluetooth : 4,2.
- Prix : 11000 DA.[8]



Figure 1. 8: Raspberry Pi 3 B+

2.3.2. Choix de carte

Après avoir exposés les différentes cartes, nous avons choisi la carte Arduino Uno parmi les autres cartes grâce a les avantages qu’il offre pour notre système et son prix abordable aussi et la facilité d’utilisation

2.4. Connectivité sans fil

Pour que nous puissions envoyer les données traitées vers l’unité de contrôle, nous devons utiliser un protocole de communication comme Wifi, Bluetooth, Zigbee, RFID...etc.

2.4.1. Protocole Wifi 802.11

Protocole Wifi 802.11 : Est un protocole de communication sans fil de courte portée ou le signal s’atténue en fonction de distance entre l’émetteur et le récepteur et les obstacles entre ces deux derniers, il utilise deux plages de fréquence 2,4Ghz et 5Ghz dont la deuxième fréquence (5Ghz) offre presque deux fois la puissance de la première fréquence (2,4Ghz).

Comme tous les réseaux sans fil le Wifi a une normalisation qui doit respecter le modèle OSI, cette normalisation est différente d’une norme a l’autre mais elle garde toujours la logique de modèle OSI.[9]

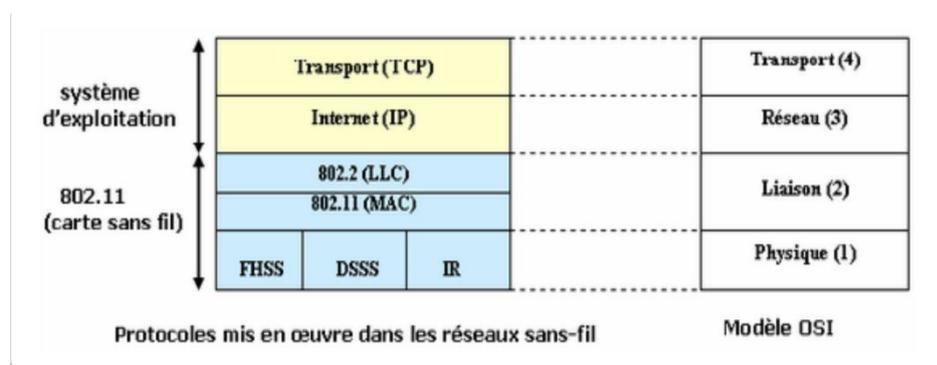


Figure 1. 9: les couches de réseau Wifi

2.4.1.1. Les modules de Wifi

Il existe plusieurs types de module qui offre une communication Wifi comme.

- **ESP8266** : est un circuit intégré a microcontrôleur avec connexion Wifi, qui offre des caractéristiques intéressantes.
 - Norme : 802.11 b/g/n.
 - Fréquence : entre 2,4Ghz et 2,5Ghz.
 - Puissance d'émission :
 - 802.11 b: +20 dBm
 - 802.11 g: +17 dBm
 - 802.11 n: +14 dBm
 - Sensibilité de réception :
 - 802.11 b: -91 dbm (11 Mbps)
 - 802.11 g: -75 dbm (54 Mbps)
 - 802.11 n: -72 dbm (MCS7)
 - Protocoles réseau : IPv4, TCP, UDP, HTTP, FTP
 - Tension de fonctionnement : 2,5V ou 3,6V
 - Courant de fonctionnement : 80mA

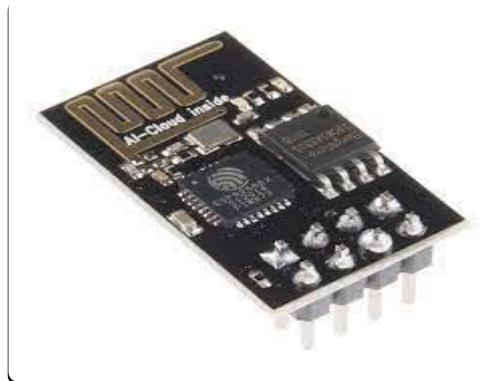


Figure 1. 10: ESP8266

Il existe un autre type de ESP8266, le module Node MCU, il possède des broches analogiques et numériques et on peut le programmer directement.[10]

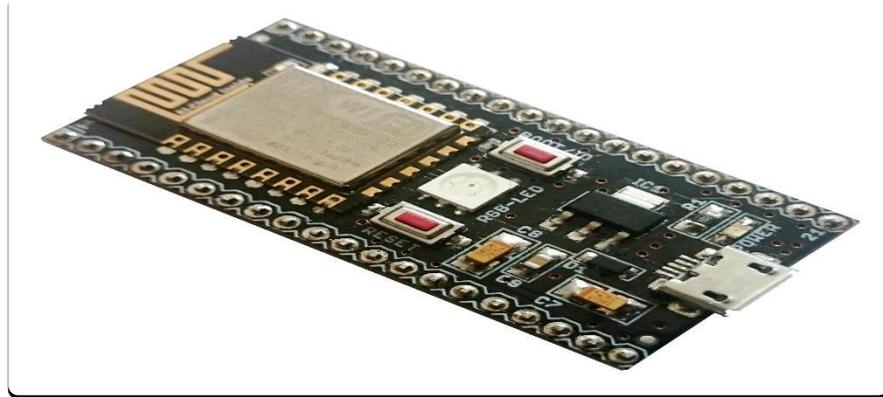


Figure 1. 11: Node MCU

- **ESP32** : ce module possède des caractéristiques développées que les deux premiers types, avec des avantages importants, mais ça n'empêche pas que les deux autres sont des modules de bonne qualité.

L'ESP32 est caractérisé par :

- Mémoire : 520 Ko SRAM
- Fréquence : 160 ou 240Mhz
- interface MAC Ethernet
- mémoire Flash : 4Mb
- ROM : 448Kb
- Protocole de réseau : IPv4, IPv6, SSL, TCP/UDP/HTTP/FTP/MQTT.
- Tension de fonctionnement : 2,5V ou 3,6V.[11]

2.4.2. Protocole Bluetooth IEEE 802.15.1

Est un protocole de communication sans fil, de courte portée et qu'il fonctionne dans la bande de fréquence 2,4Ghz, il est largement utilisé dans la vie quotidienne dans de nombreux domaines.

Et Comme tous les réseaux sans fil et le Wifi aussi, Bluetooth a une normalisation qui doit respecter le modèle OSI aussi.[12]

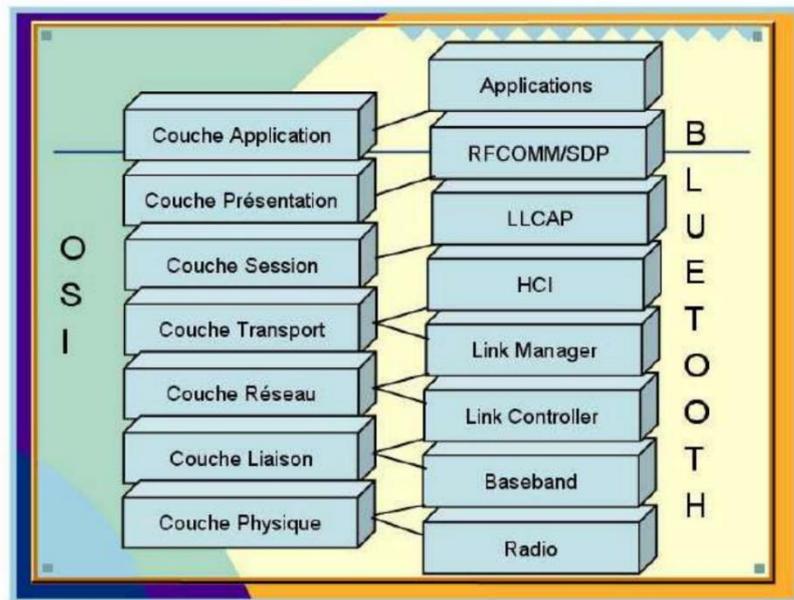


Figure 1. 12: Les couches de réseau Bluetooth

2.4.2.1. Le module Bluetooth HC-05

Module Bluetooth HC-05 : Est un module Bluetooth SPP (protocole de port série) facile à utiliser, sa communication s'effectue via une communication série, ce qui facilite l'interface avec le contrôleur ou le PC, ce module est caractérisé par :

- Tension de fonctionnement: 4V à 6V
- Méthode de communication: communication série
- Courant de fonctionnement: 30mA
- Portée: <100m
- norme IEEE 802.15.1
- Débit en bauds: 9600,19200,38400,57600,115200,230400,460800.[13]



Figure 1. 13: module Bluetooth HC-05

2.4.2.2. Le module (CSR8645)

Ce module peut nous permettre de recevoir et envoyer les données avec Arduino, il est largement utilisé vu ces caractéristiques qui sont les suivantes :

- Tension de travail: DC 3.3V-4.2V
- Spécification Bluetooth: Bluetooth V4.1
- Courant de travail: $\leq 30\text{mA}$
- Courant de veille: $< 50\mu\text{A}$
- Rapport signal audio sur bruit: $> 75\text{dB}$
- Plage de température de fonctionnement: $-40\text{ }^{\circ}\text{C} \sim 85\text{ }^{\circ}\text{C}$
- Plage d'humidité de travail: 0% -95% HR. [14]

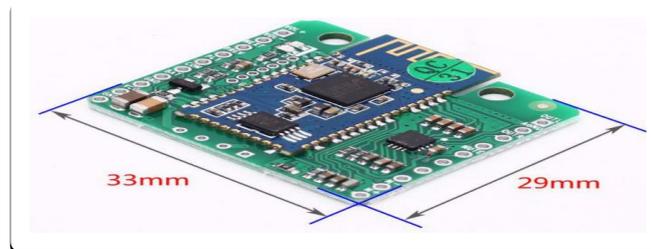


Figure 1. 14: module Bluetooth (CSR8645) .

2.4.3. Protocole Zigbee IEEE 802.15.4

Zigbee c'est un Protocol de communication sans fil comme le WIFI et Bluetooth, est un LP-WPAN (*Löw Power – Wireless Personal Area Network*), c'est un réseau sans fil à bas débit et à courte portée qui utilise les ondes hertziennes pour transporter des messages entre deux ou plusieurs entités réseaux.

2.4.3.1. Utilisation de protocole Zigbee

L'utilisation de protocole zigbee en augmentation continue grâce a les caractéristiques qu'il offre, ce protocole est utilisé spécialement dans :

- Système domotique
- Surveiller et contrôler les appareils
- Contrôle de climatisation
- Contrôle d'éclairage
- Caméra de surveillanceetc.

2.4.3.2. Le modules Xbee de Zigbee

Ces modules XBee se basent sur la norme ZigBee pour mettre en place des communications radio. Cette norme est adaptée aux réseaux personnels sans fil à faibles débits. Elle utilise : 16 canaux dans la bande de fréquences 2,4 GHz, 10 canaux dans la bande de fréquences 900 MHz, 1 canal dans la bande de fréquences 868 MHz, il fonctionne avec un courant de 50 mA et une tension de 3,3V, et sa sensibilité est de (-92dBm). [15]

3. Conclusion

Dans ce chapitre on a vu les différentes notions de notre système, qu'est ce qu'on a besoin comme équipement pour qu'on puisse démarrer à créer un système d'acquisition multi capteurs et le contrôler à distance ou sur un site, et on a vu aussi les critères de comment choisir les composantes. On a choisi la carte Arduino Uno grâce à la simplicité d'utilisation, la disponibilité, et aussi puisque on a l'habitude de travaillé avec cette carte. D'autre part on s'est arrêté après beaucoup de recherche à utiliser module de communication Wifi, le ESP8266 est compatible avec l'objectif de notre travail, en plus de ça ce module est disponible partout a un prix réduit, ainsi qu'il possède plusieurs types.

Chapitre 02
MATERIELS ET LOGICIEL

1. Introduction

La réalisation d'un montage est basée majoritairement sur le choix des composantes de base de qu'on peut utiliser, c'est pour ça il est vachement intéressant de bien choisir les composantes suivant des critères de choix de chaque montage ou chaque entreprise ou chaque personne.

Dans ce chapitre nous allons étudier les différentes composantes qu'on a choisies suivant les prix, les performances et la compatibilité entre composantes et qui doivent servir pour réaliser notre système.

2. Carte arduino

la carte arduino est une interface programmable équipé d'un microcontrôleur qui est conçu pour régir des opérations à l'aide d'un programme téléversé par un logiciel (Arduino IDE), il existe plusieurs types de cartes arduino comme Mega, UNO, Leonardo, DUE, et dans ce projet on va utiliser la carte arduino Uno sur la base des avantages qu'il nous procure

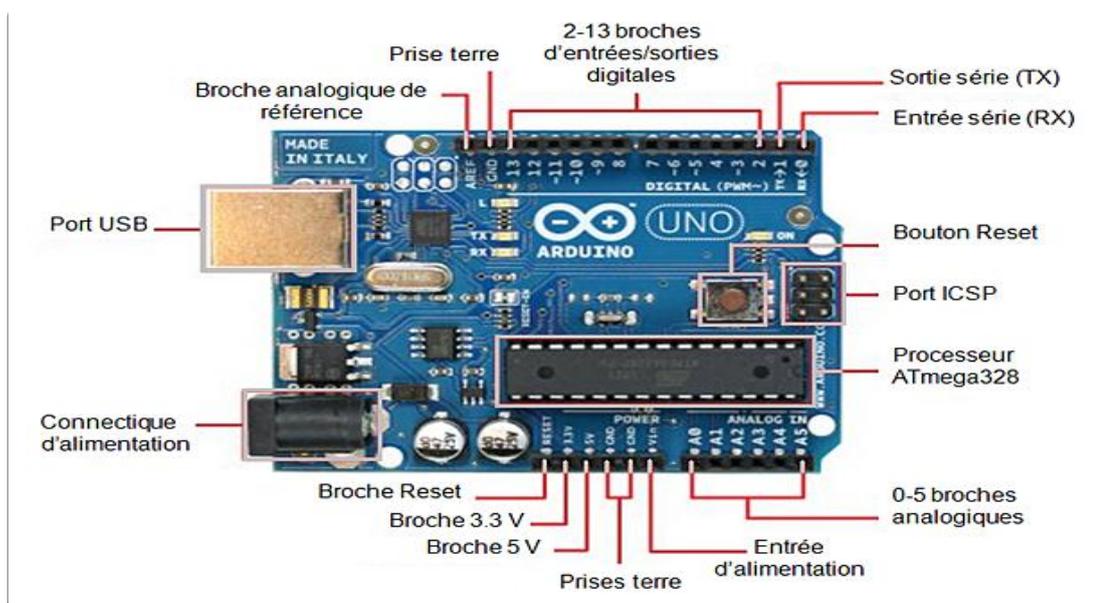


Figure 2. 1: carte arduino Uno.

2.1. Les caractéristiques de la carte Arduino uno

La majorité des cartes Arduino ont ces composants en communs avec des caractéristiques propres à chaque composant à savoir :

- **Un microcontrôleur ATmega328**

C'est un microcontrôleur ATMEL de la famille AVR 8bits. Contient trois mémoires, une mémoire flash de 32Ko, mémoire SRAM de données (volatiles) de 2Ko et une autre EEPROM données (non volatiles) de 1Ko , 3 ports entrées-sorties Tout Ou Rien PortB, PortC, PortD (soit 23 broches en tout I/O), Timer0 et Timer2 des minuteurs/Compteurs (comptage 8 bits), Timer1 (comptage 16bits) Chaque timer peut être utilisé pour générer deux signaux PWM. (6 broches OCxA/OCxB).

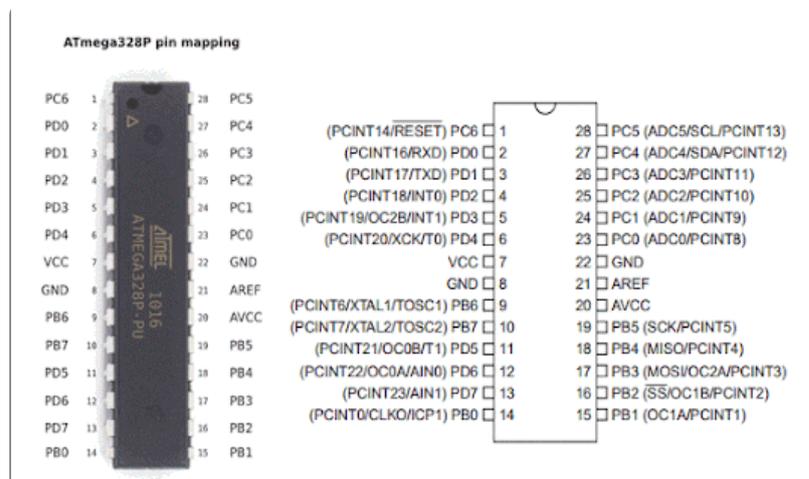


Figure 2. 2: microcontrôleur Atega328

- **l'alimentation :**

La carte Arduino uno est dotée d'une alimentation [6V, 20V] limites et [7V, 12V] recommandée, évidemment, il est déconseillé d'utiliser les tension maximal et minimal. Au risque de voir celle-ci mal fonctionner ou encore que celle-ci soit endommagée par une surtension.

- **La vitesse d'horloge :**

La vitesse de l'horloge de la carte arduino est cadencé a 16Mhz

- **Les entrées / sorties numériques (les broches de 0 à 13) :**

La carte Arduino uno possède 14 broches d'entrées/sorties numériques, qui peuvent se programmer à l'aide des instructions suivantes , `pinMode ()`, `digitalWrite ()` et `digitalRead ()`.

- **Broche PWM :**

Les broches 3, 5, 6, 9, 10,11 peuvent fonctionner en mode PWM pour faire varier la puissance du signal envoyé sur 8 bits à l'aide de l'instruction `analogWrite ()`.

- **Entrées/sorties série :**

Les broche 0(Rx) et 1(Tx), permet de recevoir et transmettre les données séries de niveau TTL

- **Broches analogiques :**

Cette carte dispose 6 entrées analogiques (A0 à A5), où chacune peut fournir une mesure avec fonction `analogRead ()` du langage Arduino sur une résolution de 10 bits (de 0 à 1023). Par défaut, ces broches mesurent une tension comprise entre le 0V correspondant au niveau 0 et le 5V correspondant au niveau 1023. Notons qu'il est possible de modifier le niveau supérieur de la plage de mesure en modifiant la tension sur la broche AREF ou en utilisant l'instruction `analogReference ()` du langage Arduino.

- **bus I2C :**

le bus est exploité via les broches SDA(A5)/SCL(A4).

- **Port USB :**

Le port USB sert à alimenter la carte arduino à partir le PC et aussi à transférer le programme de logiciel de programmation vers la carte arduino.

- **courant de sortir :**

- Courant maximal dans les branches entrée/sortie est de 40mA

Notons aussi qu'il s'agit de ne pas dépasser au maximum dans toutes les broches entrée/sortie un courant de 200mA.

- Courant maximal fourni de la broche 5V et 3.3V : c'est de 500mA et 50mA respectivement

Arduino fournit un environnement de développement (IDE) avec un éditeur de source, les opérations de compilation et de chargement dans la mémoire du microcontrôleur étant ramenées à des clics sur des boutons dans l'IDE,

la programmation avec IDE arduino est basée sur le langage C grâce au compilateur utilisé (AVR GCC) (compilateur C/C++ pour processeur AVR).[16]

2.2. La surface d'IDE arduino

La programmation avec IDE arduino est basée sur le langage C grâce au compilateur utilisé (AVR GCC) (compilateur C/C++ pour processeur AVR).

L'un des avantages du langage Arduino est de supporter toutes les syntaxes standards du langage C et quelques-unes des outils du C++. En plus, la disponibilité des bibliothèques permet de faciliter la communication avec le matériel connecté à la carte (Afficheurs LCD, capteurs, ... etc.). Pour écrire un programme avec le langage Arduino, il faut respecter certaines règles. Notons que l'exécution d'un programme Arduino s'effectue d'une façon séquentielle, c'est-à-dire que les instructions sont exécutées les unes après les autres. Avant tout, le compilateur doit vérifier l'existence de deux structures obligatoires à tout programme en langage Arduino qui sont la partie d'initialisation et de configuration des entrées/sorties (la fonction `setup()`) et la partie principale qui s'exécute en boucle (la fonction `loop()`).

- Void `setup()` : c'est la fonction de déclaration et d'initialisation et qui est exécutée une seule fois au démarrage donc cette partie doit contenir les initialisations (temps, démarrage de communication...etc.).

- Void `loop()` : cette fonction est une boucle sans fin, et qui est exécutée en boucle une fois le void `setup()` est exécuté.

Dans cette surface on peut écrire notre programme et le téléverser vers la carte arduino via le port USB.

3. Les composantes électroniques

3.1. Les capteurs utilisés

3.1.1. Capteur d'humidité et de température (DHT11)

Le composant DHT11 est un capteur d'humidité et de température numérique à faible cout, ce capteur peut être interfacé facilement avec les microcontrôleurs comme celle de Raspberry, Arduino...etc.

Ce capteur utilise une thermistance a coefficient de température négatif pour mesurer les valeurs de température, et un capteur capacitif pour mesurer l'humidité Le condensateur de détection d'humidité a deux électrodes avec un substrat de rétention d'humidité comme diélectrique entre elles. La modification de la valeur de capacité se produit avec le changement des niveaux d'humidité.[17]

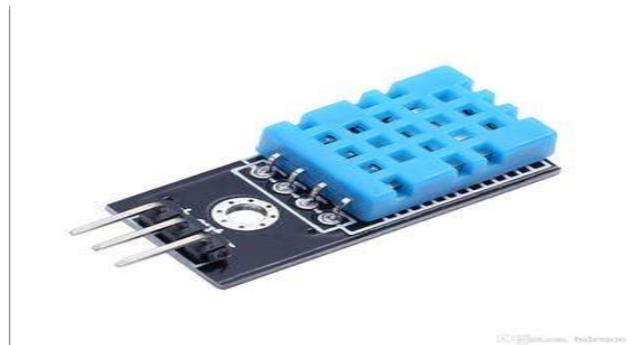


Figure 2. 3: le capteur DHT11.

3.1.1.1. Les Caractéristiques de DHT11

- Humidité : de 20 à 80% avec une précision de +/- 5%
- Température : de 0 à 50°C avec une précision de +/- 2°C
- Fréquence de mesure : 1hz
- Alimentation : de 3 à 5V

- Stabilité : +/- 1% par an.[17]

3.1.1.2. Les applications du DHT11

le capteur d'humidité et de température DHT11 possède une large utilisation dans plusieurs domaines comme les chauffages, ventilations et climatisations, équipement de test et d'inspection, automobile, automatique, Contrôle, stations météorologiques, appareils ménagers, régulateur d'humidité, humidité médicale ...etc. donc il est très intéressant de s'appuyer sur ce capteur dans ce projet vu ces avantages.

- Faible coût
- stabilité à long terme
- mesure de l'humidité relative et de la température
- excellente qualité
- Réponse rapide
- forte capacité anti-interférence
- sortie de signal est numérique .

3.1.2. Le capteur de pression BMP180

Le BMP180 est un capteur numérique de pression et de température et d'altitude, de la marque BOSCH, l'avantage de ce capteur c'est qu'il est soudé sur un PCB avec une interface I2C, ce qui facilite la communication entre le microcontrôleur de la carte de traitement et le capteur, les données de pression et de température de ce capteur sont compensées par les données d'étalonnage de son EEPROM , Le BMP180 est basé sur la technologie piézorésistive , il possède une haute précision et une linéarité importante Ainsi qu'une stabilité à long terme.[18]

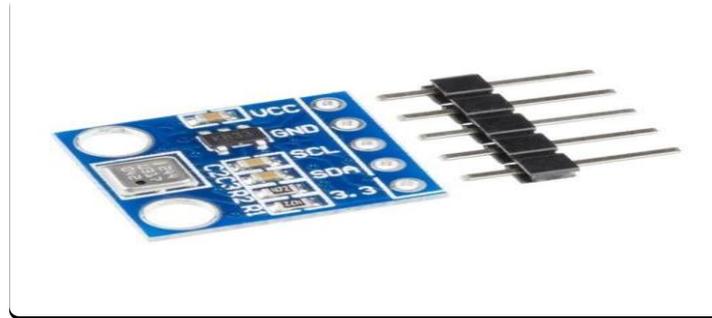


Figure 2. 4: BMP180

3.1.2.1. Les caractéristiques générales de BMP180

Le capteur BMP180 est capable de mesurer la pression atmosphérique l'altitude et la température.

- Plage de pression : 30kPa a 110kPa avec une précision de +/-0.12hPa entre (-500m et 9000m par rapport au niveau de la mer)
- Faible consommation d'énergie qui est 3 μ A
- Taux de conversion de pression de 5msec
- Température de fonctionnement : -40°C à 80°C
- Courant de crête : 1000 μ A
- Tension d'entrer : entre 1,6V et 3,6V.[18]

3.1.2.2. Les applications du BMP180

Le BMP180 est un capteur de pression barométrique a une faible consommation et de haute précision, il optimisé pour l'utilisation dans :

- Amélioration de la navigation GPS (navigation à l'estime, détection de pente, etc.)
- Navigation intérieure et extérieure
- Loisirs et sports
- Prévisions météo
- Indication de la vitesse verticale (vitesse de montée/descente)

3.1.2.3. Les avantages du BMP180

Le capteur BMP180 possède des avantages intéressants comme :

- Mesure de température incluse
- Interface I2C
- Entièrement calibré
- Sans plomb, sans halogène et conforme RoHS
- Pas cher

3.1.2.4. Les inconvénients du BMP180

le seul inconvénient de BMP180 est qu'il peut communiquer avec les micro-contrôleurs suivant le protocole I2C seulement.

3.2. les composantes d'affichage et de contrôle des phénomènes

3.2.1. l'écran LCD (16*2)

l'écran LCD est un afficheur de 16 colonnes et 2 lignes, cet écran sert à afficher les informations récupérées par les capteurs et traité par le microcontrôleur de la carte arduino, il possède 16 broches pour le model normal et quatre broches pour le model basé sur le protocole I2C.

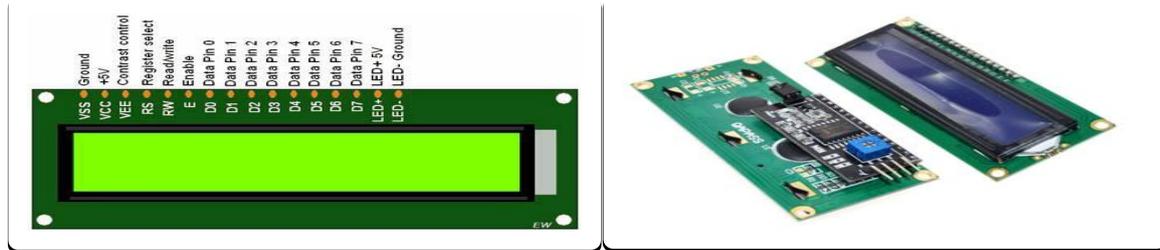


Figure 2. 5: afficher LCD 16*2

3.2.1.1. Montage de l'écran LCD

Il est très simple de faire le montage de l'afficheur LCD avec la carte arduino, il suffit juste faire le choix entre l'écran LCD simple qui a 16 broches et le model I2C qui a seulement 4 broches.

Dans notre projet on va utiliser l'écran LCD I2C pour minimiser le câblage, et aussi ce type il contient un potentiomètre pour régler la luminosité de l'écran cet écran peut être broncher avec l'Arduino uno suivant les étapes suivantes .

- Vdd : 5V .
- GND : GND .
- SDA : SDA de carte arduino.
- SCL : SCL de la carte arduino.

Notons qu'il existe deux broches SDA et aussi SCL dans la carte Arduino UNO, et les données de notre système peuvent être aussi afficher dans le téléphone ou bien un PC, suivant un protocole de communication (wifi, Bluetooth, zigbee , LoRa...etc.).[19]

3.2.2. Ventilateurs

Les ventilateurs nous permettent d'aérer l'espace pour changer les paramètres climatiques (température, humidité) suivant les besoins et l'état du climat souhaiter, nous pouvons les commander à travers le microcontrôleur de la carte arduino ou même de l'ESP8266.

Dans notre projet, nous allons utiliser deux ventilateurs pour contrôler la température et l'humidité.



Figure 2. 6: Ventilateur 12V.

3.2.3. Modules de relais

Le relais est circuit qui joue le rôle d'une interrupteur qui peut être commander à travers une tension continue ou alternatif de faible puissance , il est utilisé pour commander un circuit de puissance fort à l'aide d'un circuit de puissance faible, donc il est utilisé pour diviser deux partie, une de commande(circuit de commande) et une autre de puissance(circuit contrôlé), dans la partie de commande on passe un courant électrique qui aide la bobine de a générer champ électromagnétique, Ce champ magnétique peut être capable de faire déplacer l'armature, donc il ferme le circuit.



Figure 2. 7: Relais 5V.

Dans ce projet nous avons choisi le relais 5V pour commander les ventilateurs 12V avec une tension de 5V, donc on va utiliser deux relais, un relais pour chaque ventilateur.

3.3. Le composant de communication sans fil (ESP8266)

3.3.1. Description de ESP8266

la carte ESP8266 est une carte wifi qui offre une solution de réseau complète et autonome, possède un microcontrôleur et dotée d'une pile TCP/IP, elle exige une conception compacte et des performances fiables dans l'industrie, cette carte démarre rapidement à partir du flash externe, le cache haute vitesse intégré permet d'augmenter les performances du système et optimiser la mémoire de système, aussi, ESP8266 peut être appliqué à toute conception de microcontrôleur en tant qu'adaptateur Wi-Fi via les interfaces SPI, SDIO ou I2C, UART.

ESP8266 possède un microcontrôleur 32 bits cadencé à 80MHz avec une mémoire Flash de 16 MO et une mémoire RAM de 32K + 80K. Il possède aussi 16 entrées/sorties (GPIO), dont une entrée analogique 10 bits.[20]

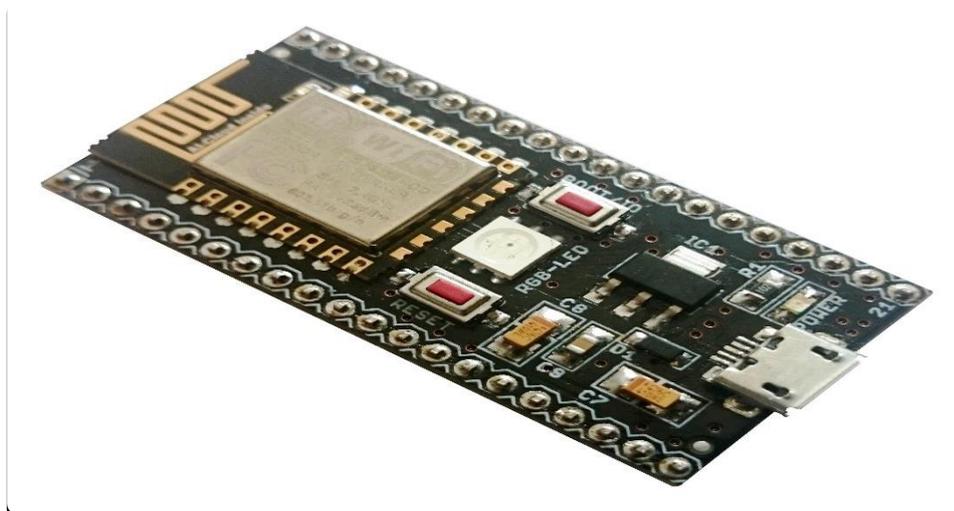


Figure 2. 8: module carte wifi ESP8266.

3.3.2. Les caractéristiques de ESP8266

la carte ESP8266 possède des caractéristiques très intéressantes qui nous donnent des choix multiples, c'est ce que nous allons les diviser sur les trois catégories suivantes (Wifi, matériels et logiciels)

3.3.2.1. Les caractéristiques de carte Wifi de ESP8266

- **Norme** : 802.11 b/g/n.
- **Fréquence** : entre 2,4Ghz et 2,5Ghz.
- **Puissance d'émission** :
 - 802.11 b: +20 dBm
 - 802.11 g: +17 dBm
 - 802.11 n: +14 dBm
- **Sensibilité de réception** :
 - 802.11 b: -91 dbm (11 Mbps)
 - 802.11 g: -75 dbm (54 Mbps)
 - 802.11 n: -72 dbm (MCS7)
- **Antenne** : Trace PCB, externe, connecteur IPEX, puce céramique.[20]

3.3.2.2. Les caractéristiques matérielles de ESP8266

- **Interface périphérique** : UART, SDIO, SPI, I2C, I2S, IR Contrôle, GPIO, PWM.
- **Alimentation** : de 3V à 3,6V
- **Courant de fonctionnement** : 80mA.[20]

3.3.2.3. Les caractéristiques logiciel de ESP8266

Mode Wi-Fi : station, softAP, SoftAP + station

Sécurité : WPA, WPA2

Cryptage : WEP, TKIP, AES

Mise à niveau du micrologiciel : Téléchargement UART / OTA (via le réseau)

Développement logiciel : SDK pour développement personnalisé / développement serveur cloud

Protocoles réseau : IPv4, TCP, UDP, HTTP, FTP

Configuration utilisateur : commandes AT, serveur cloud, application Android/iOS.[20]

3.3.3. Les broches de ESP8266

ESP8266 possède 30 broches divisé sur plusieurs utilisations comme est déjà mentionné, et grâce à la fonction de multiplexage plusieurs fonctions sont attribuées à la même broche.

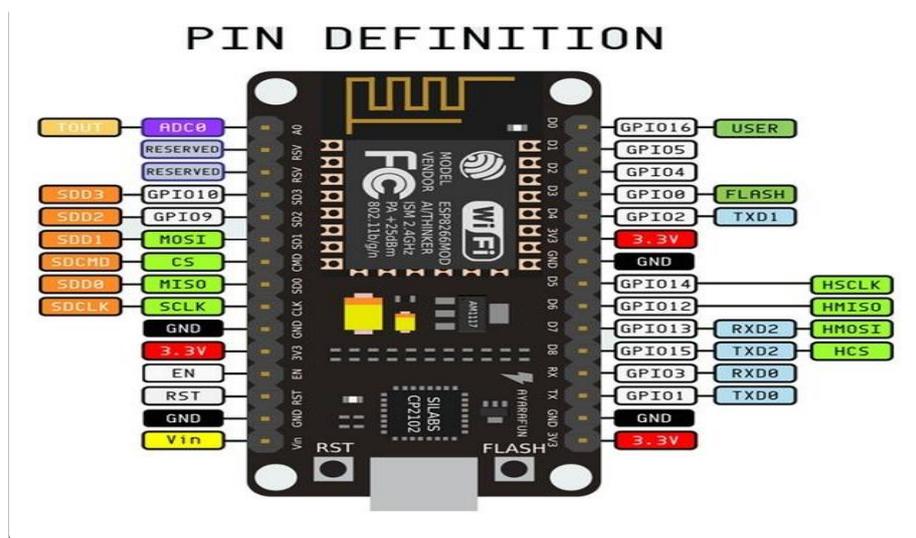


Figure 2. 9: Les broches de ESP8266.

4. Conclusion :

Dans ce chapitre on a vu les composantes que nous allons les utiliser dans ce projet, et les quelles qu'on a choisi suivant leurs performances, et caractéristique et la disponibilité de chaque un.

Ce chapitre il nous donne une idée pour la simulation et il nous ouvre la porte pour réaliser le système, donc après avoir choisir les composantes, il nous reste que à commencer la simulation et la réalisation de projet.

Chapitre 03

SIMULATION ET REALISATION DE PROJET

1. Introduction

Dans la plupart des domaines, comme le domaine médical, agronomique, industriel ...etc, nous trouvons des difficultés et des obstacles et même des problèmes pour contrôler et suivre les trois phénomènes caractéristiques de l'atmosphère (la température, la pression et l'humidité).

Pour essayer de trouver des solutions à ces problèmes , et puisque on cherche toujours à faciliter et minimiser le travail dans la vie quotidienne on va faire dans cette partie la simulation et la réalisation de notre système d'acquisition multi-capteurs pour le contrôle et la surveillance sur site ou à distance, en s'appuyant sur des capteurs qui peuvent mesurer la température ainsi que la pression et l'humidité, et les contrôler à travers une application web, laquelle nous permettra aussi de surveiller notre espace de loin ou sur un site.

Dans ce chapitre nous allons présenter l'architecteurs de notre système, et plus particulièrement comment simuler et réaliser ce système.

Commençons par la structure et les étapes de simulation et réalisation du système. Ce système se compose de quatre étapes essentielles, les capteurs qui servent à convertir les grandeurs physiques vers des signaux électriques. L'unité de traitement qui est le cerveau de ce système et qui sert à le programmer, puis l'unité de surveillance et de contrôle, cette unité a un rôle très important et indissociable, qui nous permet en effet de gérer notre système à distance ou sur site à travers l'internet, et finalement le système d'affichage qui donne les résultats des mesures de système.

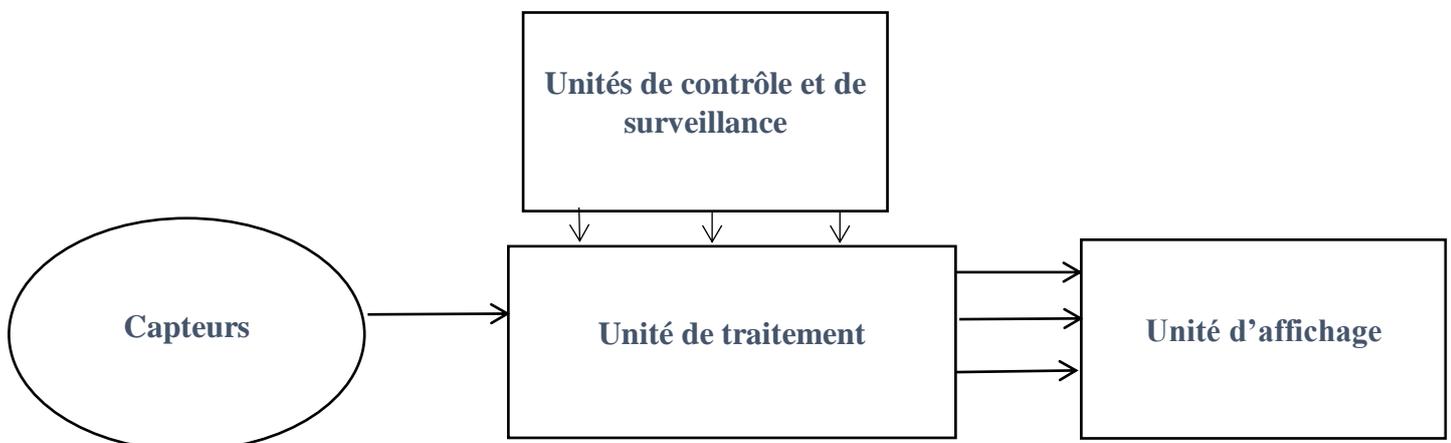


Figure 3. 1: Schéma de système d'acquisition

2. Matériel et logiciel utilisé

Pour effectuer notre projet, nous avons besoin de certains éléments et de logiciels, au début pour faire la simulation, ensuite pour réaliser le montage.

Voici ci-dessous les matériaux et les logiciels à utiliser :

Matériels	Logiciels
<ul style="list-style-type: none"> ○ Carte arduino UNO. ○ DHT11 : Capteur de température et d’humidité. ○ écran LCD 16*2. ○ LEDs. ○ Transistors NPN. ○ Relais 5V. ○ Moteurs 12V. ○ Résistances 220 ohm et 1k ohm. ○ Une diode pour protéger le relais. ○ Des piles (5V et 12V). ○ BMP180 	<ul style="list-style-type: none"> ○ Proteus 8.11 ○ Arduino IDE

Tableau 3. 1: Matériels et logiciels utilisés

3. Principe de fonctionnement

Avant de commencer la simulation et la réalisation de ce projet, il est intéressant de comprendre le principe de fonctionnement.

Comme il est montré dans la figure(3.1), ce projet est basé sur plusieurs tâches, commençons par la première tâche des capteurs , le DHT11 qui nous propose un double service, à savoir il mesure la température et l’humidité et il envoie ces informations de mesure vers le microcontrôleur de l’Arduino. La même chose vaut pour le capteur de pression BMP180, sauf que ce dernier est basé sur le protocole de communication I2C .

Passons ensuite vers l’unité d’affichage où nous utilisons un écran LCD (16*2) pour afficher les résultats de mesure, là on va afficher la température et l’humidité et la pression en degrés Celsius, en pourcentage et en hecto-pascale respectivement.

Et pour que nous puissions contrôler la température et l’humidité il faut ajouter un système composé d’un ventilateur qui peut marcher à l’aide d’un relais, et on ajoute une LED pour nous aider à déterminer si les variables ont dépassé les valeurs de seuil ou pas.

Et finalement on utilisera une carte de Wifi (ESP8266) pour que nous puissions contrôler et surveiller notre système et le contrôler à l'aide d'une page Web.

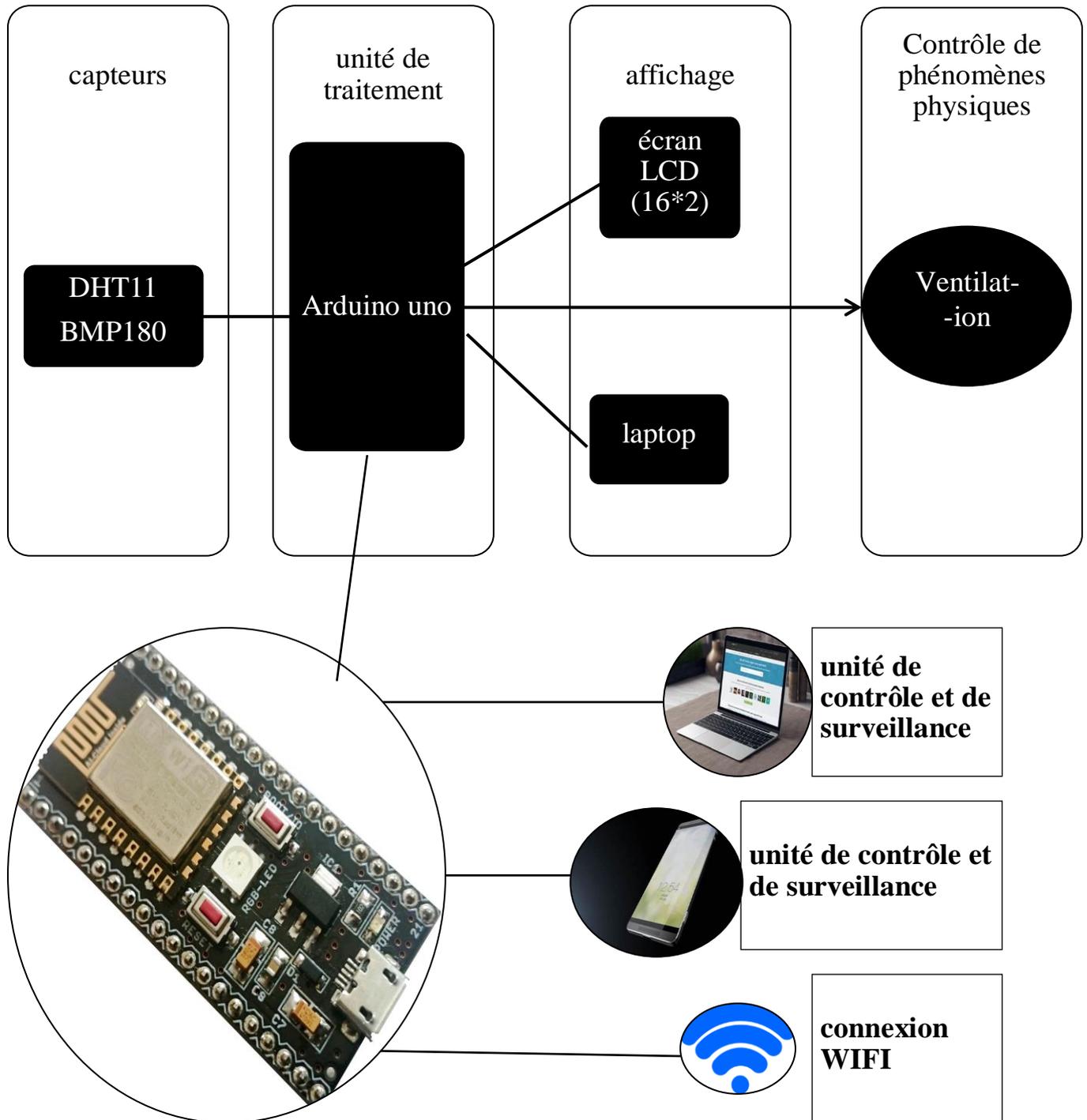


Figure 3. 2: schéma synoptique de système.

4. Caractéristiques des composants

4.1. DHT11

- Fréquence de mesure : 1hz
- Alimentation : de 3 à 5V
- Stabilité : +/- 1% par an

4.2. BMP180

- Alimentation : 3.3V
- précision de $\pm 0.12\text{hPa}$ (de -40°C à 65°C)

4.3. Ecran LCD

- Alimentation : 5V

4.4. Relais (le module relais)

4.4.1. Côté de commande

- Vcc : 5V
- GND : GND
- IN : avec une entrée digitale d'Arduino

4.4.2. Côté principal

- NO : normalement ouvert (-)
- COM : commun (+)
- NC : normalement fermée (-)

4.5. Arduino uno

- Alimentation : 9V recommandée.

4.6. Ventilateur (moteur)

- Alimentation : 12V.

5. organigramme du système

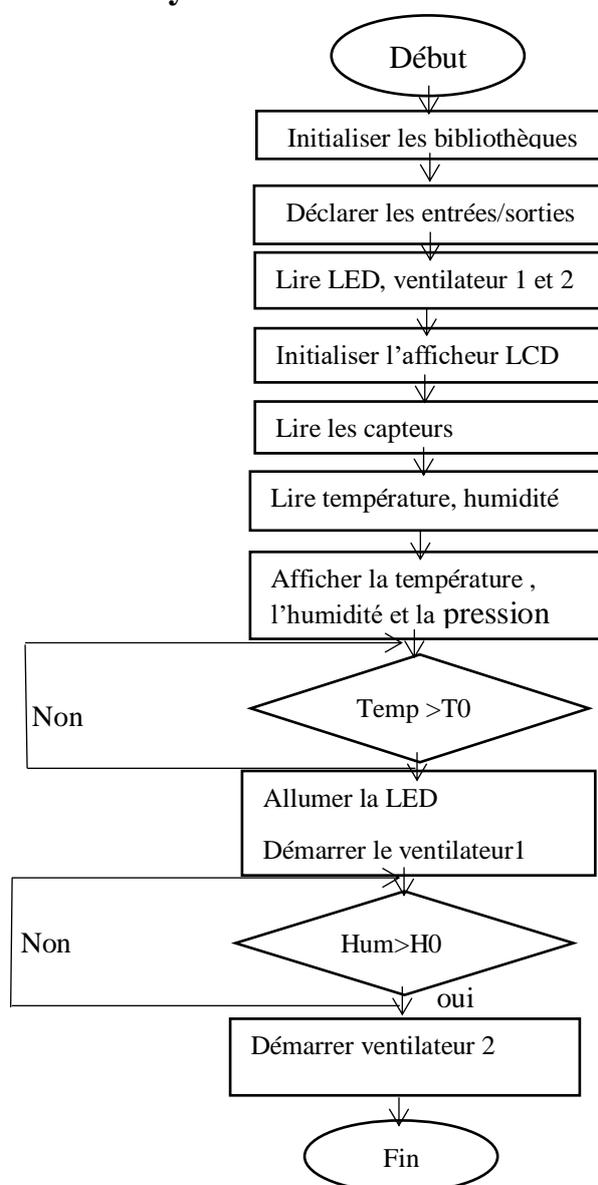


Figure 3. 3: Organigramme de DHT11.

6. Simulation et réalisation

6.1. Simulation

Pour effectuer la simulation de ce système, il nous faut au moins deux logiciels, le premier pour faire le schéma de simulation (Proteus) ,et le deuxième servant à programmer la carte Arduino (Arduino IDE).

6.2. Etapes de simulation

- Télécharger les composants listées sur Proteus, soit directement à partir de la bibliothèque ou via un fichier zip
- Faire le schéma de système
- Préparer le code Arduino
- Télécharger le code Arduino sur la carte Arduino de Proteus.

6.3. Schéma de simulation

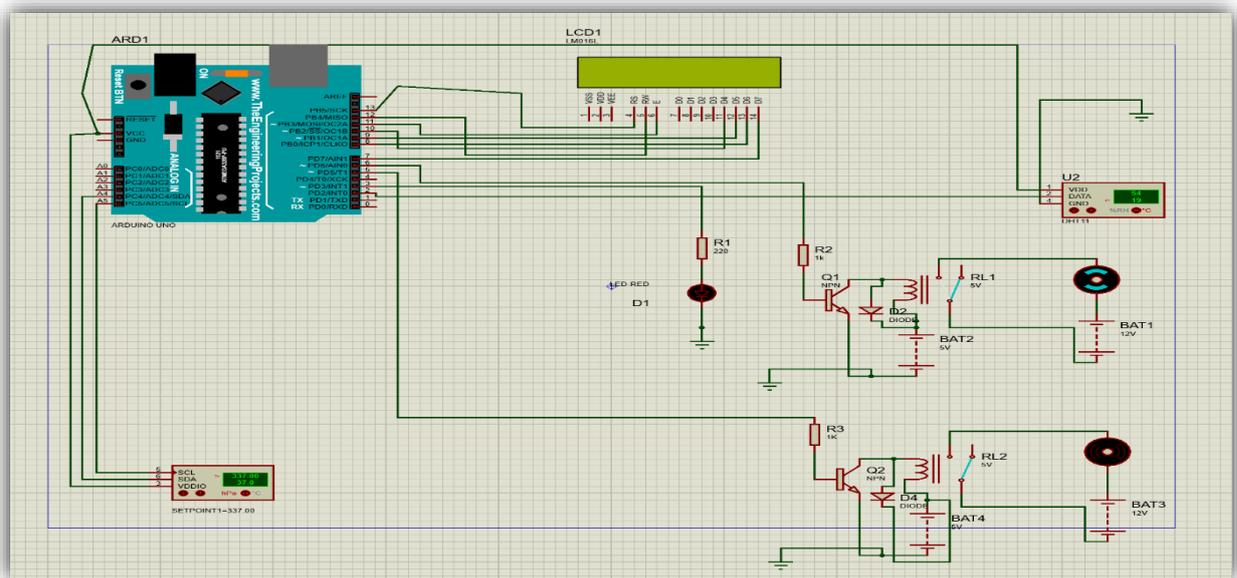


Figure 3. 4: Schéma de simulation sur ISIS Proteus.

7. Description de simulation

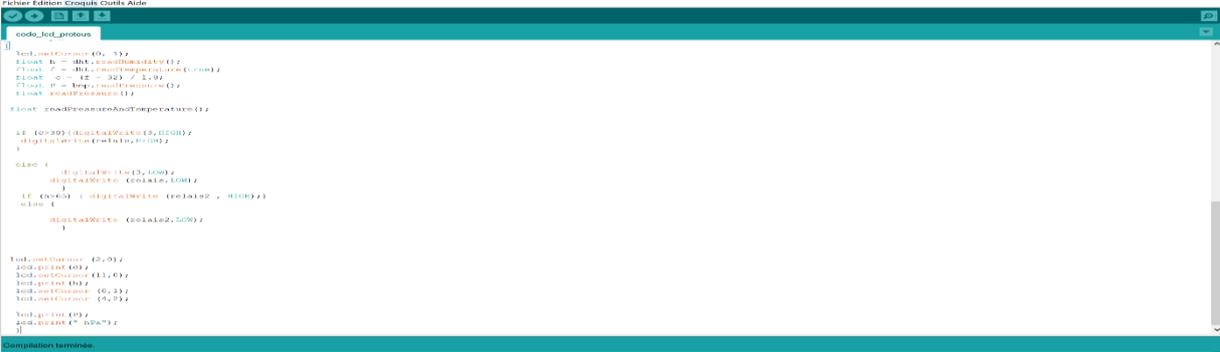
Comme montré sur le schéma, le système est basé sur les éléments mentionnés ci-dessus, premièrement pour les capteurs, le DHT11 à 3 broches (Vcc ,GND et DATA), ils sont branchés sur les pins (5V,GND et digitale 2) respectivement, de l'autre côté le BMP180 aussi à trois broches (SCL ,SDA ,Vcc) qui sont connectés avec les pins (A4,A5 et 5V) respectivement. Ce capteur fonctionne selon le protocole I2C, et donc on doit savoir qu'il n'y a que quatre broches qui peuvent fonctionner selon ce protocole, à savoir : A4 et A5 , SCL et SDA.

Deuxièmement l'écran LCD, il suffit de relier les pins (Rs,Rw,E,D4,D5,D6,D7) avec les pins digitales (13,12,11,10,9,8,7)

Finalement la LED, celle-ci est reliée avec la pin digitale (3), et les moteurs avec les pins digitales 6 et 5. Comme il est montré sur le schéma, les moteurs sont reliés avec un transistor qui sert à amplifier le courant qui circule dans le collecteur et par la même occasion dans la bobine, ainsi qu'une résistance qui vient limiter le courant dans la base du transistor, et finalement une diode pour protéger le transistor.

8. Résultats de simulation :

Après avoir fait les étapes précédentes, nous passons à l'étape de programmation, pour laquelle nous allons utiliser le logiciel Arduino IDE pour programmer le microcontrôleur de notre système suivant l'algorithme précédent.



```
code_led_moteurs
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
25
```

Et pour lancer la simulation de ISIS Proteus, nous devons compiler le code Arduino et copier le lien d'emplacement du fichier hexadécimal de code comme il est montré dans la **figure (3.6)** et le coller comme il est montré dans la **figure(3.7)**, ou bien insérer directement le fichier, puis il suffit d'appuyer sur «Run the simulation » .

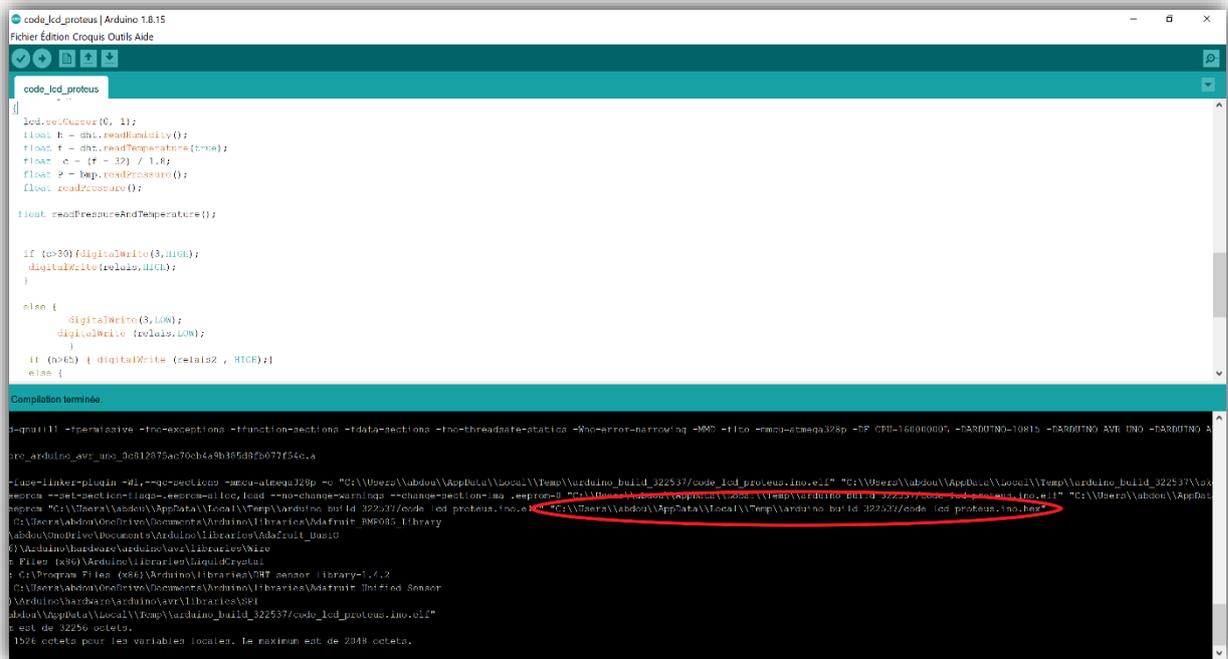


Figure 3. 6: lien hexadécimale de code arduino.

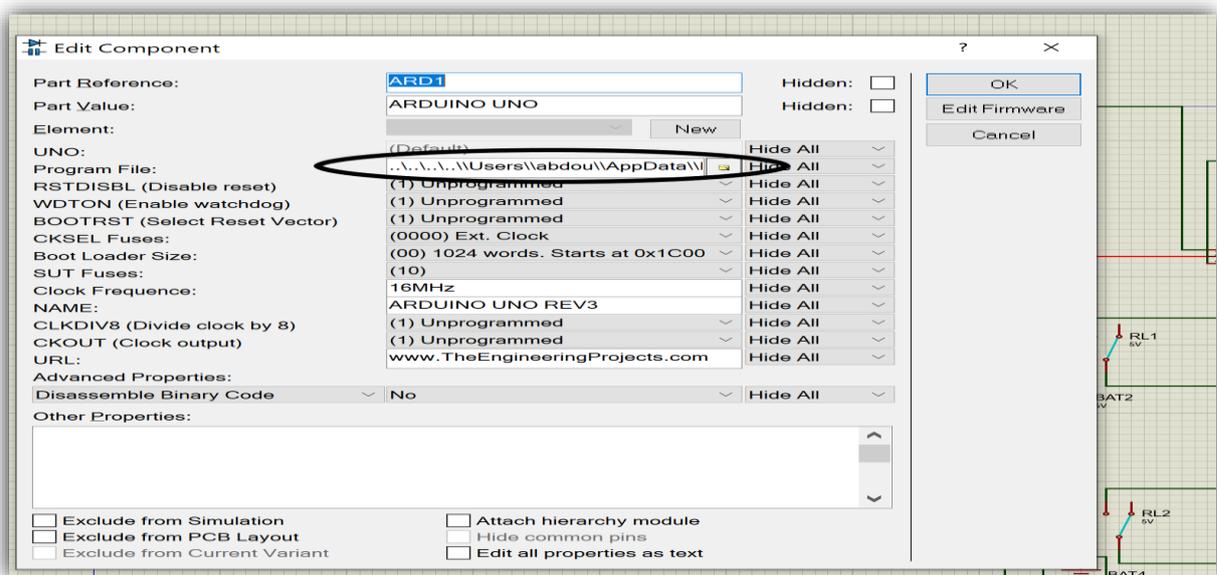


Figure 3. 7: l'emplacement de fichier de code arduino.

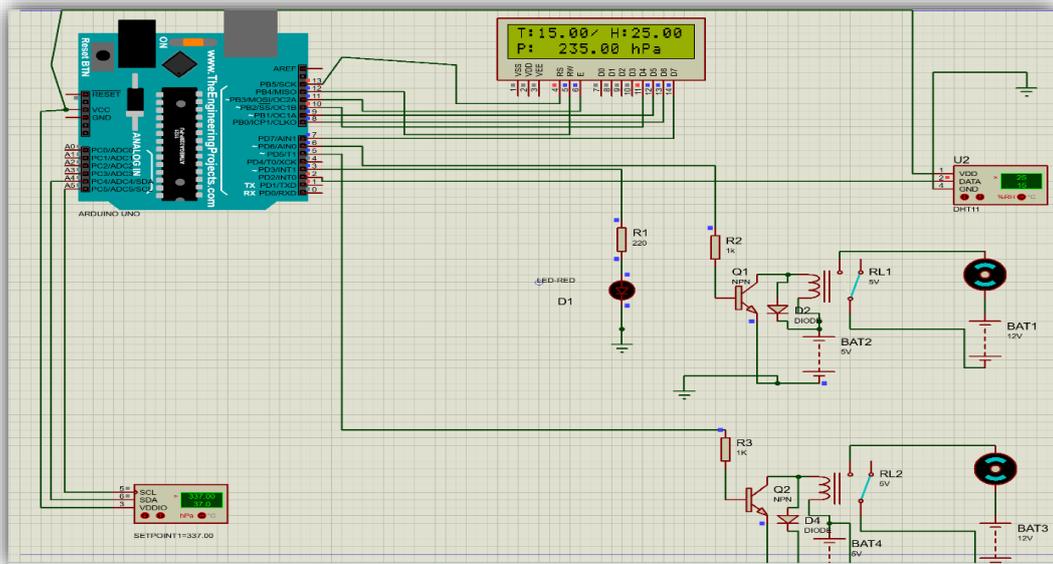


Figure 3. 8: premier résultat de simulation.

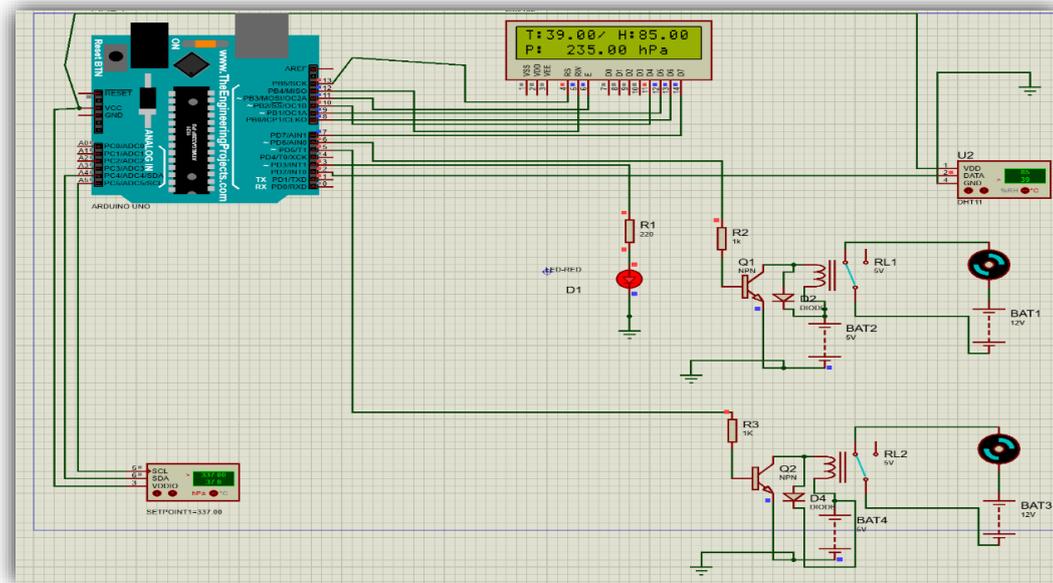


Figure 3. 9: deuxième résultat de simulation.

D’après la figure(3.8) et figure(3.9), nous pouvons constater que lorsque les valeurs de température et d’humidité sont plus élevés que les valeurs des seuils respectifs, comme affichés sur l’écran LCD, les moteurs fonctionnent et les relais sont fermés (figure (3.8)), en rajoutant la LED qui est faite pour indiquer la montée de la température.

Sinon, comme la figure(3.9) le montre, lorsque les valeurs de température et d'humidité sont plus basses que les seuils respectifs, les moteurs ne fonctionnent pas à cause des relais qui sont ouverts, de même la LED est éteinte.

9. Réalisation du système

La réalisation de ce système est composée de trois étapes principales dont la première est de configurer les capteurs avec la carte Arduino suivant les caractéristiques de chaque capteur et de configurer le système de commande et d'affichage, ainsi que l'introduction du programme dans le microcontrôleur de la carte Arduino pour obtenir les mesures et exécuter les commandes et les ordres suivant l'organigramme.

La deuxième étape concerne la transmission des données et les mesures à distance vers un ordinateur ou un smartphone, ce qui se fait grâce à un module de wifi (ESP8266)

Et la troisième étape consiste à commander la chaîne à partir d'une application web.

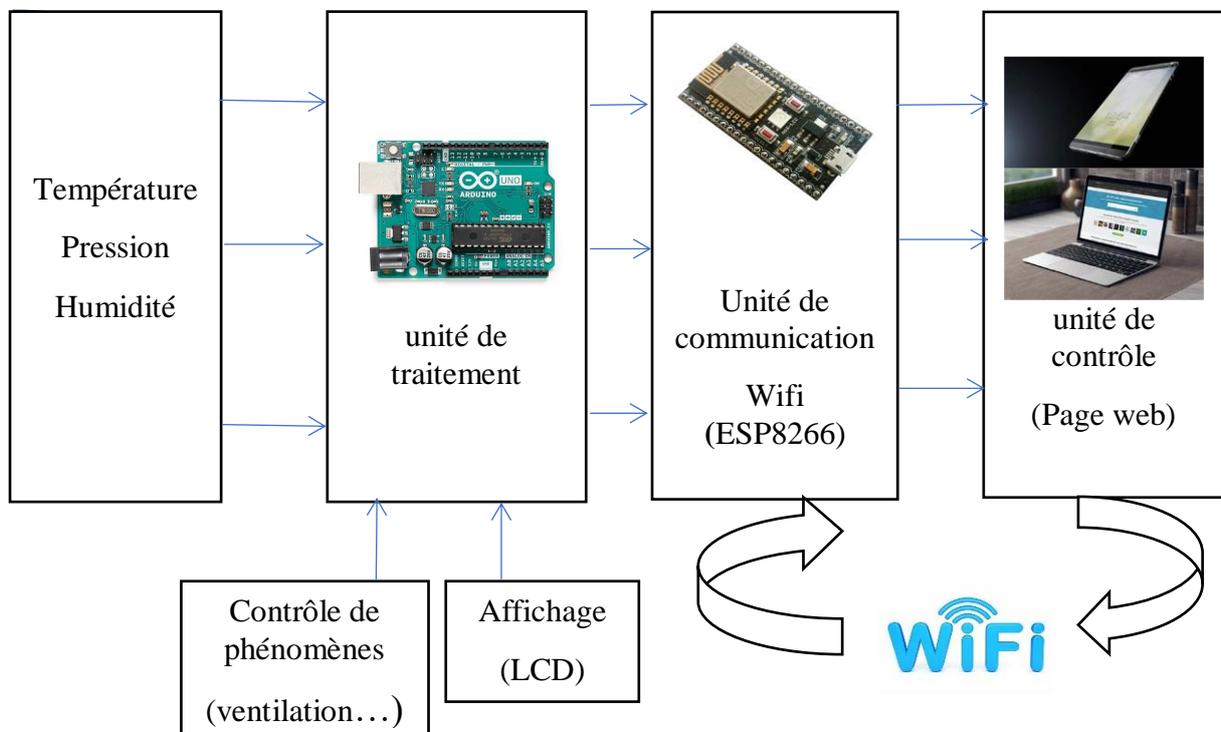


Figure 3. 10: schéma synoptique de fonctionnement de montage.

10. Configuration de la carte Arduino avec les composantes

La configuration de la carte Arduino avec les composantes consiste à établir les différentes connexions et liaisons entre les composantes et la carte Arduino.

La programmation d'une carte Arduino en utilisant Arduino IDE suit la chaîne suivante :

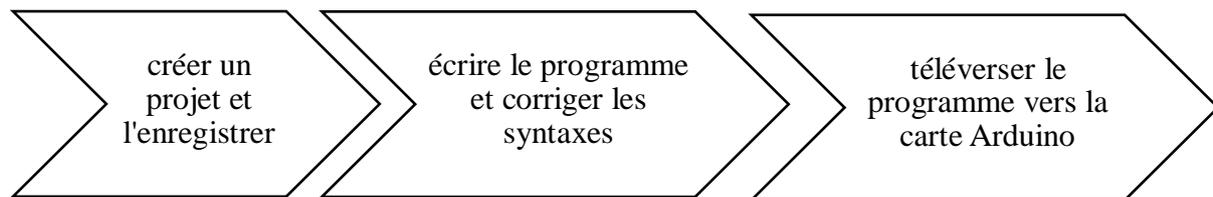


Figure 3. 11: les étapes de configuration des composantes avec Arduino.

11. Partie matérielle

Dans les travaux pratiques, les outils utilisés ont un rôle majeur dans la réalisation d'un travail de haute qualité. Nous devons par conséquent faire attention à chaque fois à la qualité et à l'état des outils qu'on doit utiliser, ainsi qu'à leur bon fonctionnement en vue d'atteindre les objectifs.

Donc voici ci-dessous le matériel qu'on peut utiliser pour réaliser notre système :

- Carte Arduino UNO.
- DHT11 : pour la température et l'humidité.
- BMP180 : capteur de pression.
- ESP8266 : module Wifi
- Ecran LCD (I2C) : pour l'affichage des données .
- Resistance de 200ohm : pour limiter le courant vers la LED.
- Alimentation pour la carte Arduino (9V).
- Plaque d'essai.

- Transistors (dans ce cas on a choisi NPN « BC547 »).
- Diode de protection.
- Relais 5V.
- Ventilateur 12V (moteur).
- Pile 12V pour faire fonctionner le ventilateur.
- Des fils pour faire les connexions entre les composants.
- Multimètres pour tester les composants.

On peut minimiser les fils de connexion en remplaçant le montage de relais avec le transistor et la diode ainsi que la résistance avec un module de relais, pour que nous puissions utiliser plusieurs appareils qui s'alimentent du même voltage (5V) on doit alimenter la plaque d'essai à partir de la carte Arduino (5V,GND). On peut faire ça aussi avec les composantes qui s'alimentent avec (3.3V), et comme la plaque d'essai nous offre deux côtés d'alimentation dans ces deux extrémités il vaut mieux les utiliser comme deux alimentations suivant les sorties de la carte Arduino.

12. Etapes de montage avec carte Arduino (sur site)

Afin d'éviter l'imbrication des étapes, on doit suivre une méthode de montage bien structurée. Dans un premier temps, il est préférable de commencer à alimenter la plaque d'essai à partir la carte Arduino.

Après l'alimentation de la plaque d'essai, nous commençons à connecter le reste des composants selon les étapes suivantes.

- DHT11: ($V_{cc}=5V$, $GND=GND$, $DATA=digital\ 2$).
- BMP180: ($V_{cc}=3.3V$, $GND=GND$, $SDA=A4$, $SCL=A5$).
- Écran LCD (i2c): ($V_{cc}=5V$, $GND=GND$, $SDA=A4$, $SCL=A5$).
- LED : avec résistance de 200ohm entre la sortie digitale 3 et la LED.
- Ventilateur : concernant le ventilateur on a deux branchements à faire :

- Module relais1: ($V_{cc}=5V$, $GND=GND$, $IN=digital\ 5$)
- Module relais 2: ($V_{cc}=5V$, $GND=GND$, $IN=digital\ 8$)
- Ventilateurs: il nous suffit de brancher le côté positif des ventilateurs avec la broche (NO) et négatif avec le côté négatif de la pile (12V), et on doit brancher le (+) de la pile avec la broche (COM) des relais.

Comme le montre la figure(3.12) notre montage est prêt, il ne manque que l'alimentation, dans notre cas on a choisi un transformateur 9V, mais on peut également utiliser une pile de 9V pour alimenter la carte Arduino.

12.1. Résultats de montage

Après avoir fait une révision générale du branchement de ce montage et de brancher l'alimentation on peut voir les résultats de ce montage comme ci-dessous.

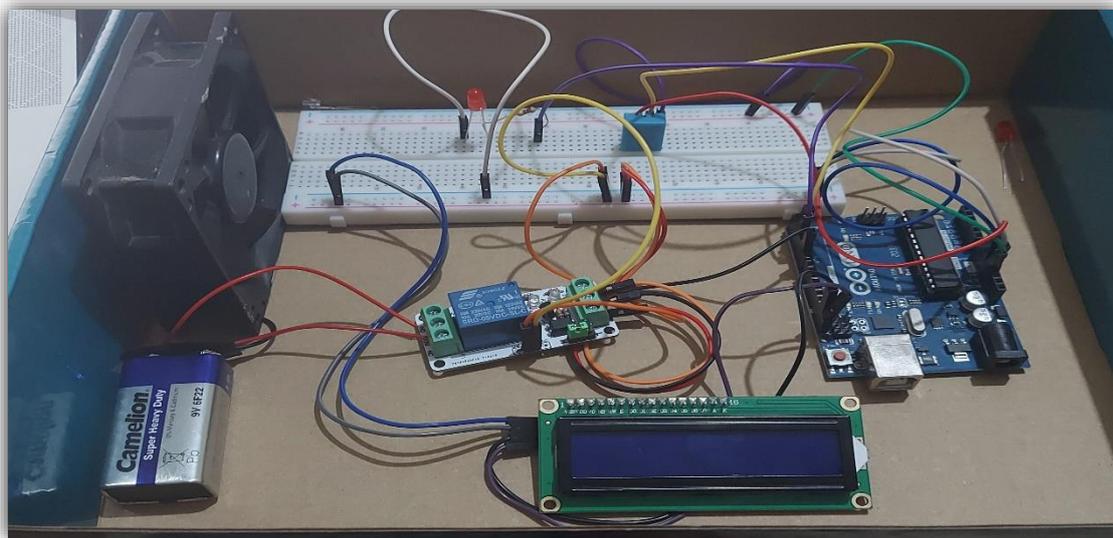


Figure 3. 12: Montage de Arduino- DHT11-ventilateur-LCD.

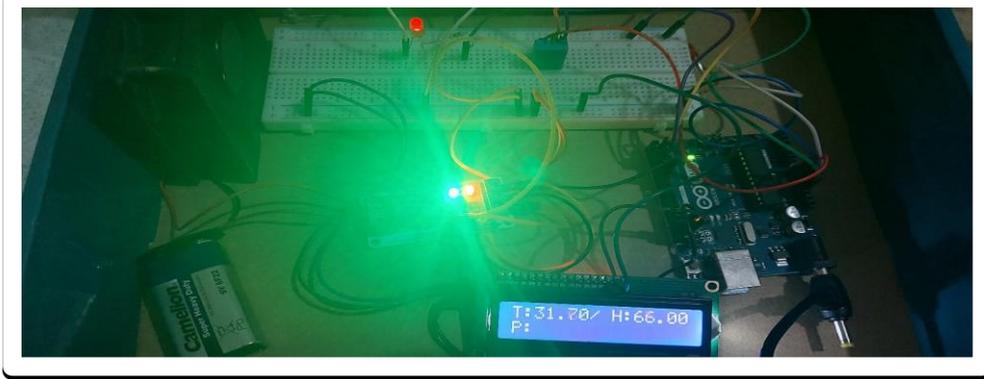


Figure 3. 13: résultats de montage DHT11-LCD-ventilateur.

A partir la figure (3.13) et figure (3.14) on peut remarquer que le montage fonctionne parfaitement comme en simulation. On peut de même remarquer que les données affichées sur l'écran LCD montrent plus ou moins les mêmes valeurs réelles, avec une petite incertitude qui dépend de la précision des capteurs utilisés.

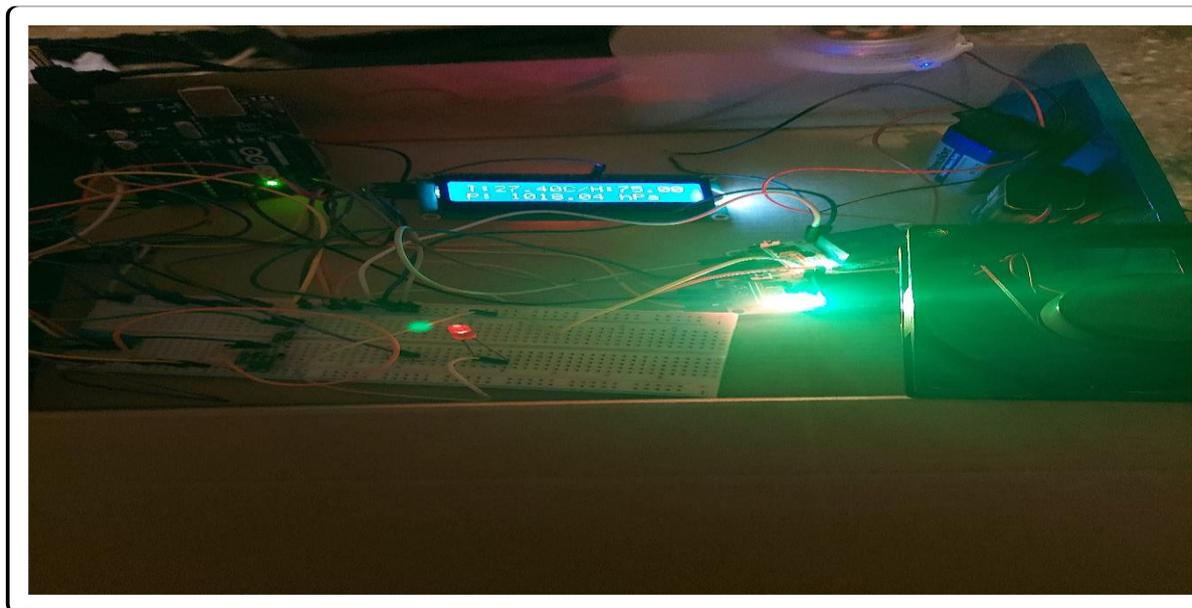


Figure 3. 14: résultats de montage DHT11-ESP8266-BMP180-LCD-ventilateurs.

12.2. Récupération des données avec Matlab

Pour récupérer les données de Arduino Uno avec Matlab, il nous faut tout d'abord créer une communication entre notre carte Arduino Uno et Matlab en utilisant un algorithme de récupération des données.

12.2.1. Organigramme de communication Matlab-Arduino

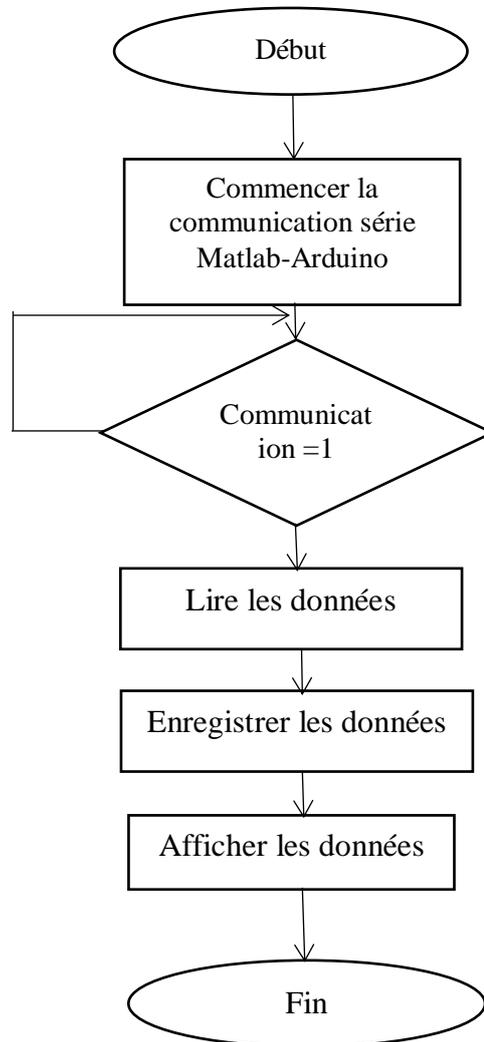


Figure 3. 15: organigramme de communication série Matlab-Arduino.

12.2.2. Affichage des données sur Matlab

Comme il est identifié sur la figure ci-dessous qui représente l'acquisition des données en temps réel, on a des pointes d'intersection entre les courbes des données récupérées de l'Arduino Uno et les droites qui représentent les seuils, d'où la ligne noire de paramètre de température représentant les moments où la Led d'avertissement s'allume ou s'éteint, sinon la droite orange est faite pour mentionner les moments où la ventilation commence et s'arrête.

Au deuxième paramètre, qui représente les valeurs d'humidité dans l'intervalle de temps, on a deux points d'intersection qui représentent le début et la fin de la ventilation respectivement.[21]

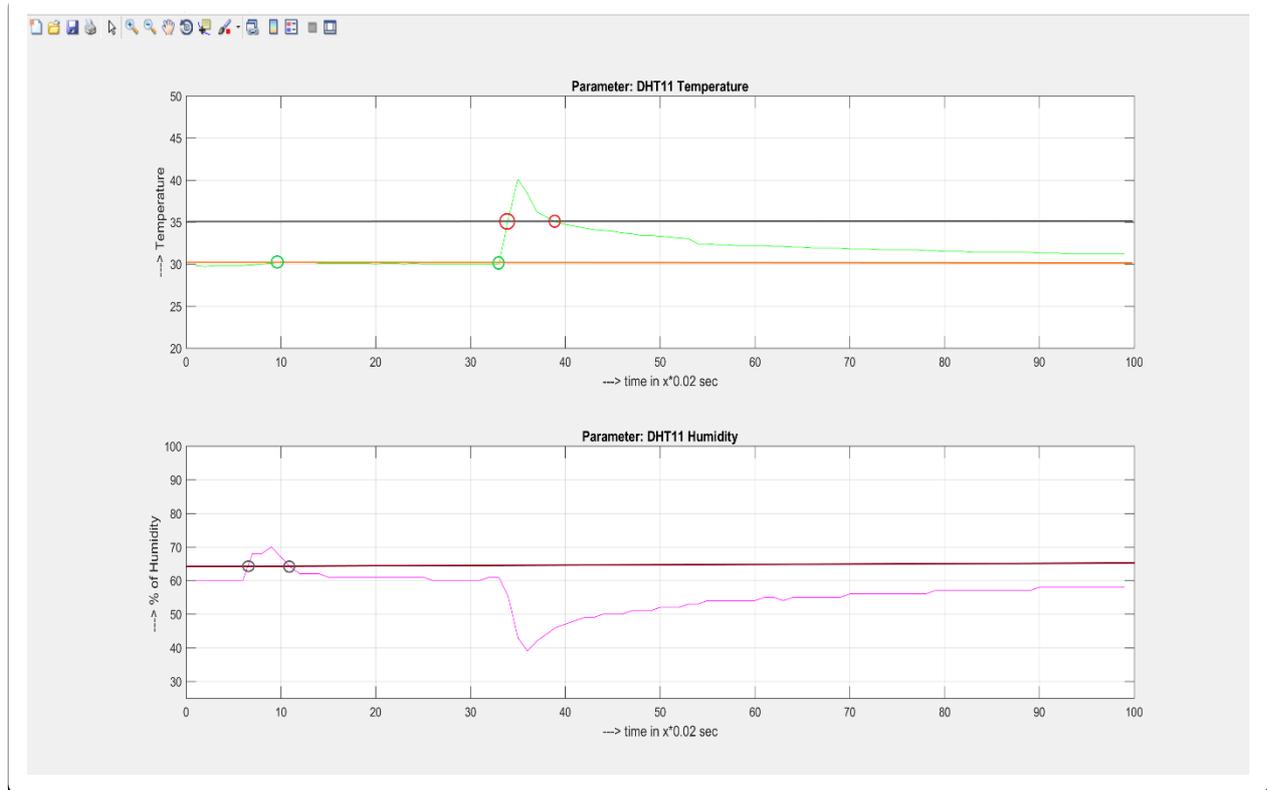


Figure 3. 16: organigramme de communication série Matlab-Arduino.

13. Etapes de montage avec Node MCU (à distance)

Dans cette partie on a le choix d'utiliser soit (Arduino Uno avec ESP8266) ou Node MCU. Dans notre cas on va utiliser Node MCU, ce qui est plus pratique puisque, comme nous avons déjà parlé dans la description de cette carte, elle nous offre des privilèges de ne pas utiliser la carte Arduino Uno. Le seul inconvénient de cette carte c'est qu'elle nous offre une minorité des broches, mais dans notre cas, les broches de Node MCU sont suffisantes pour réaliser le montage.

Voici ci-dessous les étapes de montage de notre système.

- DHT11: ($V_{cc}=3V3$, $GND=GND$, $DATA=digital\ 6$).
- BMP180: ($V_{cc}=3.3V$, $GND=GND$, $SDA=D2$, $SCL=D1$).

- Ventilateurs : concernant le ventilateur on a deux branchements à faire :
 - Module relais1: (Vcc=5V, GND=GND, IN=digital 7)
 - Module relais 2: (Vcc=5V,GND=GND,IN=digital 3)
 - Ventilateurs: il nous suffit de brancher le cote positif des ventilateurs avec la broche (NO) et négatif avec le cote négatif de la pile (12V), et on doit brancher le (+) de la pile avec la broche (COM) des relais.

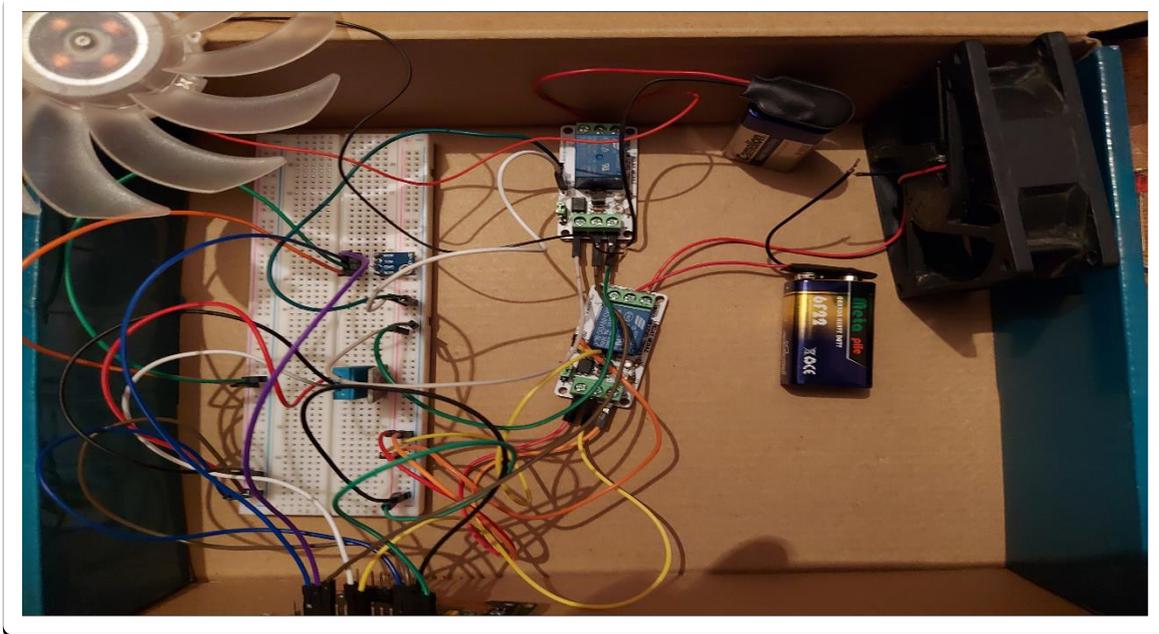


Figure 3. 17: Montage Node MCU DHT11 BMP180.

13.1. Résultat de montage

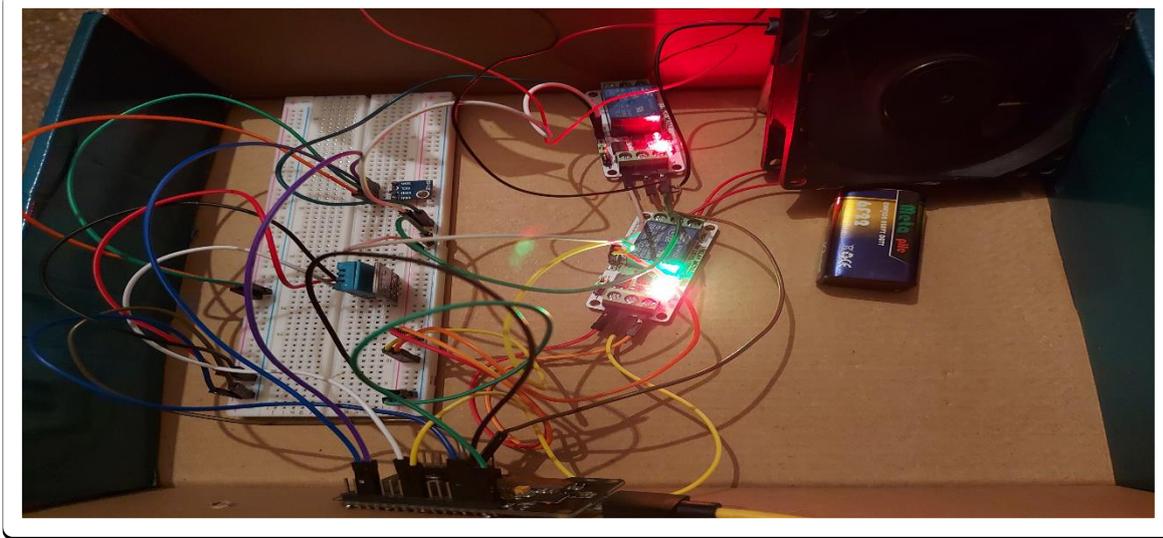


Figure 3. 18: résultat de montage Node MCU DH11 BMP180.

14. Application mobile et réalisation finale

14.1. Introduction

Dans cette partie, nous allons voir comment on peut créer une application mobile qui marche sous un système Android, quelle est la méthode et comment on peut récupérer les données de notre système et en fin comment on peut contrôler les ventilateurs à travers cette application.

Mais avant de commencer il nous faut d'abord définir, c'est quoi un système Android

14.2. Définition du système d'exploitation mobile 'Android'

Android est un système d'exploitation crée et développé par la société Google, très courant dans le domaine des smartphones, car il équipe la majorité des smartphones. Ce système permet aux utilisateurs de personnaliser leurs smartphones, et de télécharger des applications (WhatsApp, GPS, navigateurs internet...etc.).

14.3. Firebase

Dans ce projet on va créer une base des données en utilisant la plateforme Firebase qui peut être accessibles par les différents utilisateurs.

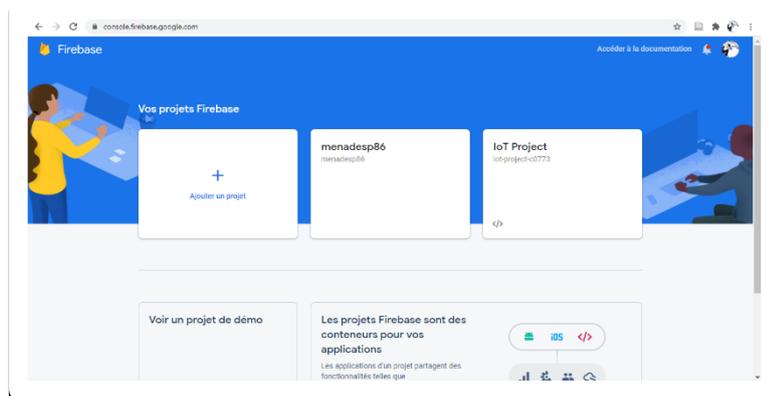
Google Firebase est une plateforme de développement d'applications pour le web ou pour mobiles. C'est un backend en tant que service (Baas). Elle fournit des outils sous forme de services pour la création d'applications mobiles.

La base des données de Firebase utilise la technologie en temps réel (noSQL) (Realtime DataBase), Hébergée dans le Cloud, elle sert à stocker et synchroniser les données utilisateurs à l'aide d'une simple API, Firebase fournit à l'application les valeurs en temps réel des données et les rafraîchit automatiquement.[22]

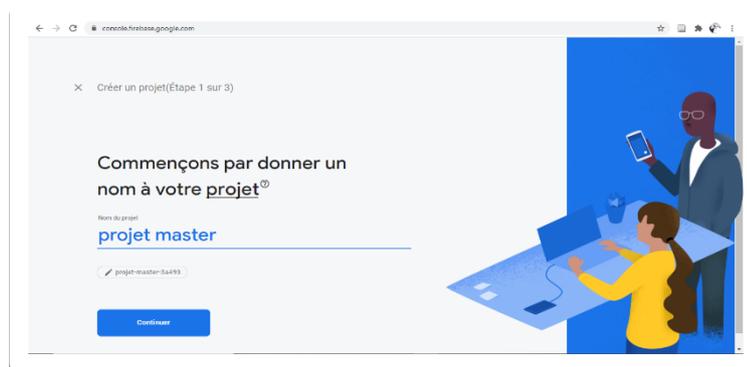
14.4. Création d'un projet Firebase

Dans cette partie on va voir les étapes et comment on peut créer un projet sur Firebase, c'est très simple de s'inscrire dedans.

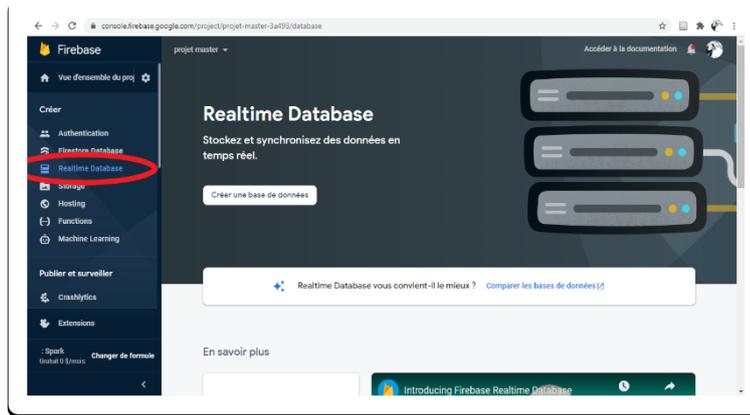
- Commencer par ajouter un projet



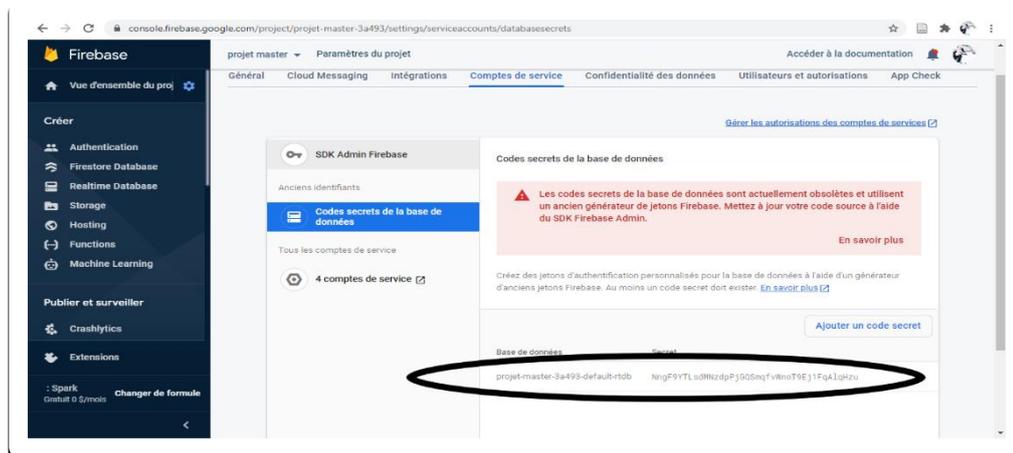
- Donner un nom au projet



- Créer une base des données



Après avoir fait ces étapes on a donc créé une base des données pour notre projet, on doit seulement programmer les sorties que nous voulons enregistrer dans cette page avec Arduino IDE en s'appuyant sur l'HOST et le AUTH qui sont fournis par cette base des données.[22]

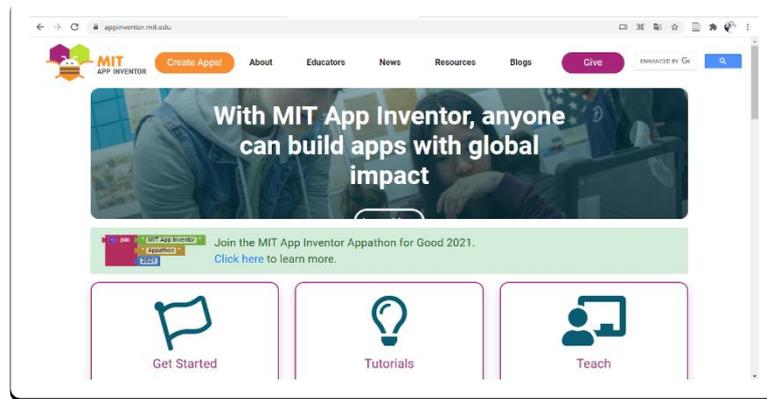


14.5. Environnement de développement MIT App Inventor

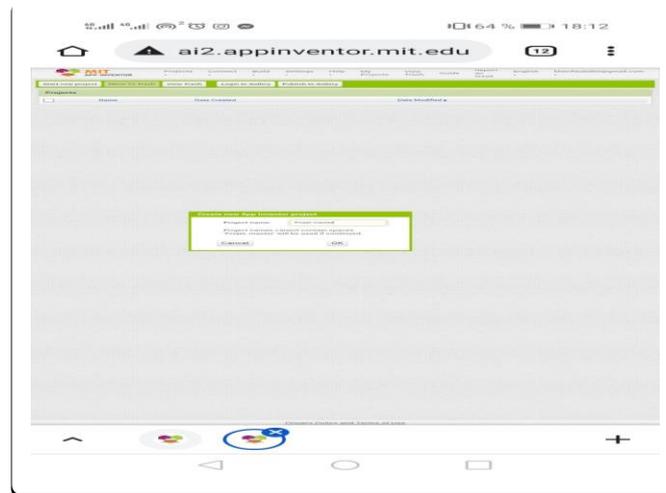
MIT App Inventor est un outil qui nous permet d'obtenir une application Android facilement. En vue de créer une application, MIT App Inventor permet aux utilisateurs de glisser-déposer des blocs pour créer un objet d'application, lequel peut fonctionner sur des appareils Android.

14.5.1. Création de l'application mobile

- Au début il nous faut créer un profil dans MIT App inventor



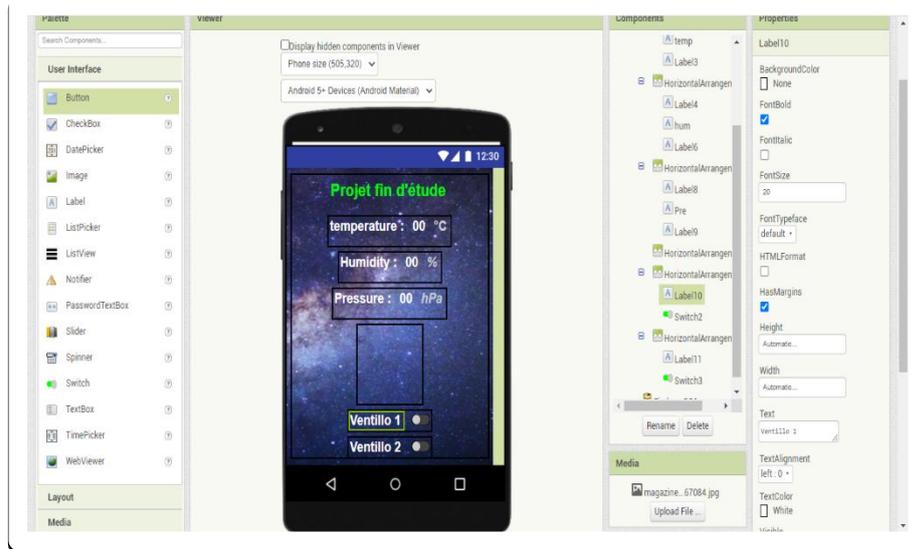
- après avoir fait la première étape d'enregistrement, il nous reste à donner un nom pour le profil.



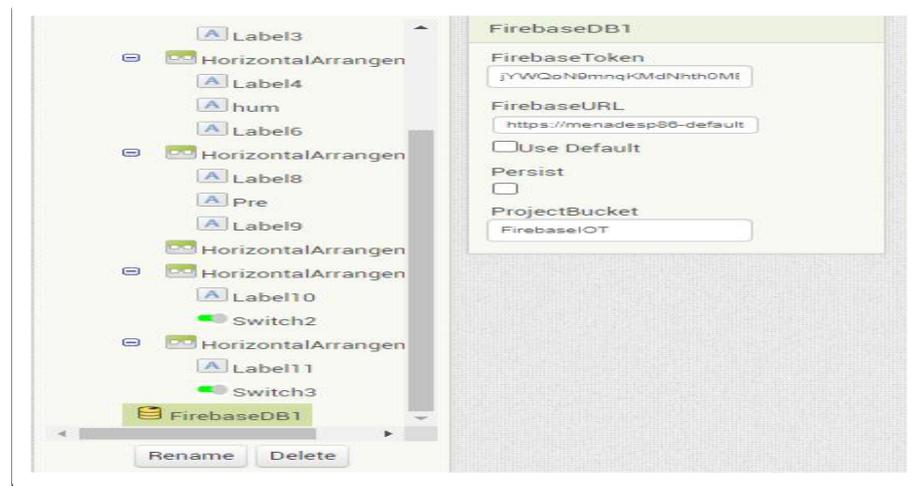
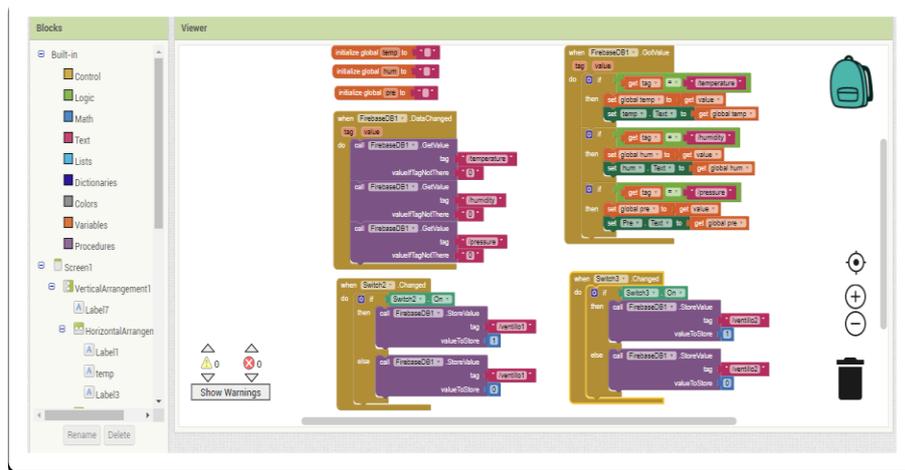
- **Configuration de l'application**

on a deux opérations à faire dans cette partie, la première consiste à configurer l'affichage, et la deuxième consiste à configurer les fonctions.

On commence par l'affichage, comme il est affiché dans la photo ci-dessous, on a créé trois variables pour l'humidité, la pression et la température et deux boutons pour faire fonctionner ou arrêter les ventilateurs.



Après ça, il nous reste à relier l'application avec la base des données en utilisant l'URL et l'authentificateur de la base des données Firebase.



Finalement, notre application est prête pour récupérer les data de notre système, et contrôler les ventilateurs de notre montage. [23]

15. Récupération des données de notre montage

Après avoir fait la partie programmation avec Arduino IDE et avoir fait le montage comme mentionné, et avoir créé une application en la reliant avec la base des données, il ne reste qu'à téléverser le programme.

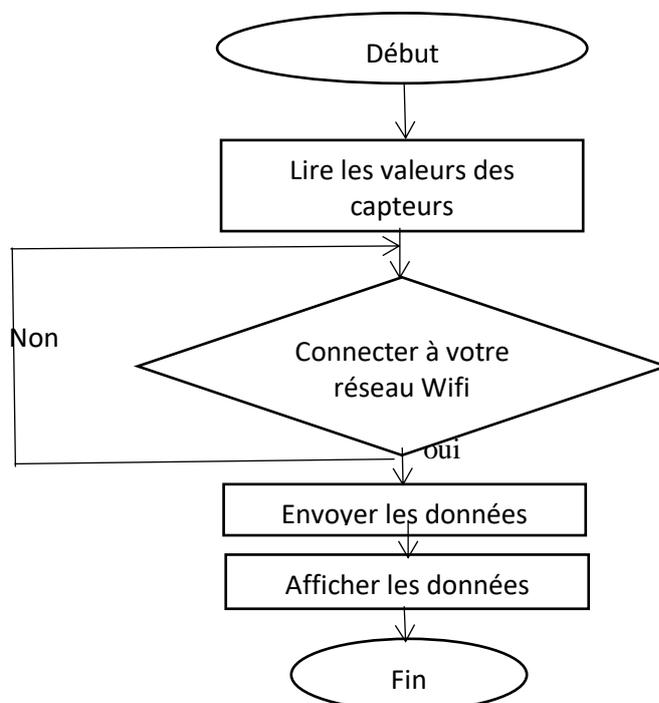


Figure 3. 19: organigramme de transfert des données

15.1. Résultats d'affichage

Finalement, on a réussi à faire toutes les étapes précédentes, il ne nous reste qu'à afficher les données. Premièrement on va afficher les données sur Firebase ensuite sur l'application Android.



Figure 3. 20: les données dans Firebase.



Figure 3. 21: les données sur l'application.

16. Conclusion

Dans ce chapitre on a vu comment on peut simuler et réaliser notre système et comment on peut récupérer les data dans une application.

Premièrement on a réalisé notre système et on a vu comment on peut le contrôler sur site via l'afficher LCD. Ensuite on a réalisé un système d'acquisition pour le contrôler à distance et on a vu que les données sont transférées vers l'application en temps réel.

Comme il est affiché dans la figure(3.20) notre montage fonctionne bien, il affiche les valeurs d'humidité, de pression et de température en temps réel.

Les boutons (ventillo1 et ventillo2) dans l'application qui représente les boutons pour faire fonctionner ou arrêter les ventilateurs qui servent à contrôler la température et l'humidité, fonctionnent parfaitement.

Conclusion générale

Les systèmes d'acquisition multi-capteurs sont de plus en plus utilisés surtout avec la révolution de l'internet des objets et de la domotique, car ces systèmes jouent des rôles largement intéressants dans la sécurité, la prévention et la gestion des besoins en facilitant les choses dans plusieurs domaines.

Dans le domaine médical par exemple, on doit choisir avec soin les composantes, dont la précision et la sensibilité sont très abordables. Le choix des composantes dans ce domaine ne doit donc pas être sous-estimé en raison du danger et de la sensibilité dans ce domaine

Dans ce projet on a abordé ce sujet au regard des besoins de développer des idées dans ce domaine. Le projet propose donc d'élaborer une chaîne d'acquisition multi-capteurs qui sert à mesurer la température, l'humidité et la pression, et de contrôler ces phénomènes sur site à travers un afficheur et des ventilateurs, et de les consulter à distance à travers une application mobile qui affiche les valeurs de chaque phénomène en temps réel et qui sert aussi à commander les ventilateurs au moyen d'un seul clic.

Dans le premier chapitre on a abordé le sujet en faisant l'étude des notions de base d'un système d'acquisition multi capteurs et des composantes de base de ce système. On a commencé par montrer la logique de fonctionnement de système d'acquisition multi-capteurs, ensuite on a abordé les différentes couches de ce système en parlant sur les cartes programmables et les capteurs ainsi que les carte qui servent à relier le système a une connexion Wifi ou Bluetooth ou Zigbee et on a définie chaque protocole.

Dans le deuxième chapitre on a vu les différentes composantes qu'on peut utiliser pour réaliser un tel système comme la Arduino Uno, module ESP8266 et Node MCU ainsi que le BMP180 et on a vu comment on peut choisir les composantes suivant des critères de choix comme le prix, la précision et la sensibilité, A partir ces derniers on s'est arrêté a choisir la carte Arduino Uno suivant les critères de choix ou on a trouvé qu'elle est très courante et disponible dans le marché, et aussi puisque on a déjà travaillé avec cette carte, de l'autre côté, on a choisi les capteurs suivant les caractéristiques général de chaque capteur et aussi le prix et la facilité d'utilisation avec la carte Arduino Uno et Node MCU.

Dernièrement, dans le troisième chapitre on s'est basé sur les deux premiers chapitres pour commencer la simulation et la réalisation de projet. Ainsi on a d'abord commencé par la simulation en s'appuyant sur la surface Arduino IDE et sur ISIS Proteus. Tout d'abord nous

avons rédigé un organigramme de programme qui avait le rôle d'introduire la programmation, Après avoir remarqué que les résultats de simulation étaient logiques, on est passé à la réalisation de projet, ou on a préparé les composants et les testé. En lisant les datasheets de chaque composant on a fait les branchements de ces composants avec la carte Arduino Uno puis Node MCU en suivant les étapes de simulation. Après avoir fini le branchement on a vu comment on a créé une base des données en utilisant Firebase et aussi comment on a créé une application mobile en utilisant MIT App Inventor. L'envoi et la récupération des données avec cette application était nickel, et on a réussi à contrôler les ventilateurs à travers cette application

D'autre part, le stage de fin d'étude a été une occasion pour se familiariser avec le milieu du travail, ce qui nous a permis d'acquérir des notions et des méthodes d'études et de travail intéressantes.

Annexes

Annexe 1 : code de projet avec Node MCU (à distance) .

```
#include "FirebaseESP8266.h"

#include <ESP8266WiFi.h>

#include <SFE_BMP180.h>

#include <Wire.h>

#include "DHT.h"

#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

SFE_BMP180 pressure;

float temperature;

#define ALTITUDE 100

#define FIREBASE_HOST "menadesp86-default-rtdb.firebaseio.com"

#define FIREBASE_AUTH "jYWQoN9mnqKMdNht0MBFMzDKEs2RbKQaaUIZNi"

#define WIFI_SSID "AndroidAP38E9"

#define WIFI_PASSWORD "12346789"

int led = D0;

const int DHTPIN = D6;

#define DHTTYPE DHT11

DHT dht(DHTPIN, DHTTYPE);

#define relais D7

#define relais2 D3

FirebaseData firebaseData;

FirebaseData ledData;

FirebaseData ventillo1;

FirebaseData ventillo2;

FirebaseJson json;

void setup()

{

    pressure.begin();

    Serial.begin(115200);

    dht.begin();

    pinMode(led,OUTPUT);
```

```

pinMode(relais,OUTPUT);

WiFi.begin(WIFI_SSID, WIFI_PASSWORD);

Serial.print("Connecting to Wi-Fi");

while (WiFi.status() != WL_CONNECTED)

{

Serial.print(".");

delay(300);

}

Serial.println();

Serial.print("Connected with IP: ");

Serial.println(WiFi.localIP());

Serial.println();

Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);

Firebase.reconnectWiFi(true);

}

void loop() {

char status;

double T,P,p0,a;

status = pressure.startTemperature();

if (status != 0)

{

delay(status);

status = pressure.getTemperature(T);

if (status != 0)

{

Serial.print("temperature: ");

Serial.print(T,2);

Serial.print(" deg C, ");

Serial.print((9.0/5.0)*T+32.0,2);

Serial.println(" deg F");

status = pressure.startPressure(3);

```

```

if (status != 0)
{
delay(status);

status = pressure.getPressure(P,T);

if (status != 0)
{

Serial.print("absolute pressure: ");

Serial.print(P,2);

Serial.print(" mb, ");

Serial.print(P*0.0295333727,2);

Serial.println(" inHg");

p0 = pressure.sealevel(P,ALTITUDE);

Serial.print("relative (sea-level) pressure: ");

Serial.print(p0,2);

Serial.print(" mb, ");

Serial.print(p0*0.0295333727,2);

Serial.println(" inHg");

a = pressure.altitude(P,p0);

Serial.print("computed altitude: ");

Serial.print(a,0);

Serial.print(" meters, ");

Serial.print(a*3.28084,0);

Serial.println(" feet");

}

else Serial.println("error retrieving pressure measurement\n");

}

else Serial.println("error starting pressure measurement\n");

}

else Serial.println("error retrieving temperature measurement\n");

}

else Serial.println("error starting temperature measurement\n");

float h = dht.readHumidity();

```

```

float t = dht.readTemperature();

float f = dht.readTemperature(true);

Serial.print(F("Humidity: "));

Serial.print(h);

Serial.print(F("% Temperature: "));

Serial.print(t);

Serial.print(F("C ,"));

Serial.print(f);

Serial.println(F("F "));

if (h>50){ digitalWrite (relais2,HIGH);}

else {digitalWrite (relais2, LOW);}

if (isnan(h) || isnan(t) || isnan(f) {

    Serial.println(F("Failed to read from DHT sensor!"));

    return;

}

lcd.clear();

lcd.print("T:");

lcd.print(t);

lcd.print("C");

if (Firebase.setFloat(firebaseData, "/FirebaseIOT/temperature", t))

{

    Serial.println("PASSED");

    Serial.println("PATH: " + firebaseData.dataPath());

    Serial.println("TYPE: " + firebaseData.dataType());

    Serial.println("ETag: " + firebaseData.ETag());

    Serial.println("-----");

    Serial.println();

}

else

{

    Serial.println("FAILED");

    Serial.println("REASON: " + firebaseData.errorReason());

```

```

Serial.println("-----");

Serial.println();

}

if (Firebase.setFloat(firebaseData, "/FirebaseIOT/humidity", h))

{

Serial.println("PASSED");

Serial.println("PATH: " + firebaseData.dataPath());

Serial.println("TYPE: " + firebaseData.dataType());

Serial.println("ETag: " + firebaseData.ETag());

Serial.println("-----");

Serial.println();

}

else

{

Serial.println("FAILED");

Serial.println("REASON: " + firebaseData.errorReason());

Serial.println("-----");

Serial.println();

}

if (Firebase.getString(ledData, "/FirebaseIOT/led")){

Serial.println(ledData.stringData());

if (ledData.stringData() == "1") {

digitalWrite(led, HIGH);

}

else if (ledData.stringData() == "0"){

digitalWrite(led, LOW);

}

}

if (Firebase.getString(ventillo1, "/FirebaseIOT/ventillo1")){

Serial.println(ventillo1.stringData());

if (ventillo1.stringData() == "1") {

digitalWrite(relais, HIGH);

```

```

}

else if (ventillo1.stringData() == "0"){

digitalWrite(relais, LOW);

}

if (Firebase.setFloat(firebaseData, "/FirebaseIOT/pressure",P))

{

Serial.println("PASSED");

Serial.println("PATH: " + firebaseData.dataPath());

Serial.println("TYPE: " + firebaseData.dataType());

Serial.println("ETag: " + firebaseData.ETag());

Serial.println("-----");

Serial.println();

}

else

{

Serial.println("FAILED");

Serial.println("REASON: " + firebaseData.errorReason());

Serial.println("-----");

Serial.println();

}

}

}

delay(100);}}

```

Annexe 2 : création de l'application mobile avec le site MIT APP INVENTOR .

MIT APP INVENTOR : est une application développée par Google. Elle est actuellement entretenue par le Massachusetts Institute of Technology (MIT) , Elle simplifie le développement des applications sous Android et le rend accessible même pour les novices et ceux qui ne sont pas familiers avec les langages de programmation.

Cette plate-forme de développement est utilisable par toutes et tous à condition de posséder un compte Gmail.

La programmation est réalisée sans taper une seule ligne de code, mais simplement en associant et en paramétrant des briques logicielles toutes faites (langage Scratch) est voici les étapes pour créer l'application.

- Lorsque vous utilisez App Inventor, vous arrivez sur une page présentant les projets App Inventor de votre compte.
- cliquez sur le bouton « New Project » et acceptez les conditions, puis cliquez sur "Continuer" bouton.
- Entrez le nom du nouveau projet
- Dessiner l'interface de l'application.
- le site affiche un écran de téléphone dans lequel nous pouvons placer, en les faisant glisser, les éléments que nous voulons utiliser.
- Il y a des éléments graphiques comme des boutons, des labels... et des éléments non graphiques comme des capteurs (ex. géo localisation...) ou des fonctions permettant d'effectuer des actions (lecteur audio, appareil photo...) ou Experimental (FIREBASE)...
- Sur le côté droit il y a des propriétés pour ajuster les couleurs ,les tailles et les arrière-plans.
- Programmation de l'application.
- cliquer sur « Blocks » en haut et à droite de la page
- Les éléments de notre "Screen1" apparaissent bien (en bas à gauche).

les blocs disponibles classés en grandes fonctions :

- Contrôle : les éléments de test, de boucles,
- Logique : éléments liées à de la logique, "vrai", "faux", ...
- Math : des outils mathématiques (addition, ...)
- Texte : pour gérer les textes
- L'algorithme a été entièrement réalisé. Sauvegarder votre programme
- Collez le code Firebase token et Firebase URL tiré de Firebase.

- Il suffit de saisir le code ou de scanner le QR Code pour que l'application soit visible sur le smartphone et vous pouvez ensuite la tester.

Annexe 3 : création la base des données avec Firebase.

Pour créer un nouveau projet Firebase, suivez les étapes mentionnées ci-dessous:

- Ouvrez la page de Firebase..
- cliquez sur "Ajouter un projet" .
- Entrez le nom du projet Firebase et acceptez les conditions, puis cliquez sur "Continuer" bouton.
- Ouvrez la section Base de données en cliquant sur le bouton "realtime database "option dans le menu de gauche.
- Créer une base de données, puis cliques sur suivant.
- Pour copier le Firebase token et URL utilisé dans notre application, nous allons dans les paramètres situés en haut a droite.
- En suit cliquez sur comptes de service.

Références bibliographiques

Références :

- [1] « Transmission série », *Wikipédia*. sept. 05, 2021. Consulté le: sept. 23, 2021. [En ligne]. Disponible sur:
https://fr.wikipedia.org/w/index.php?title=Transmission_s%C3%A9rie&oldid=186080337
- [2] TS IRIS (Physique Appliquée) Christian BISSIERES « capteurs.pdf ».p.1
- [3] « Généralités sur les capteurs », p.2. Consulté le : juin. 10, 2021.[En ligne].Disponible sur :
https://sti.discip.ac-caen.fr/IMG/pdf/Generalites_sur_les_capteurs.pdf
- [4] « ArduinoMega2560Datasheet.pdf ». Consulté le: sept. 23, 2021. [En ligne]. Disponible sur:
<https://www.robotshop.com/media/files/PDF/ArduinoMega2560Datasheet.pdf>
- [5] « 0900766b81409bb2.pdf ». Consulté le: sept. 23, 2021. [En ligne]. Disponible sur:
<https://docs.rs-online.com/51ed/0900766b81409bb2.pdf>
- [6] B. Cottenceau, « Carte ARDUINO UNO Microcontrôleur ATMega328 », p. 29.
- [7] « 2170767.pdf ». Consulté le: sept. 23, 2021. [En ligne]. Disponible sur:
<https://www.jameco.com/jameco/products/prodds/2170767.pdf>
- [8] « Raspberry-Pi-Model-Bplus-Product-Brief.pdf ». Consulté le: sept. 23, 2021. [En ligne]. Disponible sur: <https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-Model-Bplus-Product-Brief.pdf>
- [9] « wifiNB.pdf ». Consulté le: sept. 23, 2021. [En ligne]. Disponible sur: <http://perso.univ-mlv.fr/present/supports/pdfsrc/wifiNB.pdf>
- [10] « ESP12E Datasheet.pdf ». Consulté le: sept. 23, 2021. [En ligne]. Disponible sur:
https://components101.com/sites/default/files/component_datasheet/ESP12E%20Datasheet.pdf
- [11] « esp32_datasheet_en.pdf ». Consulté le: sept. 23, 2021. [En ligne]. Disponible sur:
https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf
- [12] « Formation-Interface-communication-25.pdf ». Consulté le: sept. 24, 2021. [En ligne]. Disponible sur: <https://www.electronique-mixte.fr/wp-content/uploads/2018/07/Formation-Interface-communication-25.pdf>
- [13] « HC-05 Datasheet.pdf ». Consulté le: sept. 24, 2021. [En ligne]. Disponible sur:
https://components101.com/sites/default/files/component_datasheet/HC-05%20Datasheet.pdf
- [14] « CSR8645.pdf ». Consulté le: sept. 24, 2021. [En ligne]. Disponible sur:
<https://ecksteinimg.de/Datasheet/CSR8645.pdf>
- [15] J. Francomme, F. Virolleau, J. Pang, Y. X. Phang, et T. Val, « ZigBee, de la théorie à la pratique : création d'un réseau ZigBee avec transmission de données », p. 19.
- [16] « Cottenceau - Carte ARDUINO UNO Microcontrôleur ATMega328.pdf ».
- [17] « DHT11 Humidity & Temperature Sensor », p. 10.
- [18] « BST-BMP180-DS000-09.pdf ».
- [19] « ADM1602K-NSW-FBS-3.3v.pdf ». Consulté le: sept. 24, 2021. [En ligne]. Disponible sur:
<https://www.sparkfun.com/datasheets/LCD/ADM1602K-NSW-FBS-3.3v.pdf>
- [20] « NodeMCU ESP8266 Pinout, Specifications, Features & Datasheet ». <https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and-datasheet> (consulté le sept. 24, 2021).
- [21] « Comm - Formation Arduino MatlabSimulink.pdf ».
- [22] « Firebase », *Firestore*. <https://firebase.google.com/?hl=fr> (consulté le sept. 24, 2021).
- [23] « MIT App Inventor | Explore MIT App Inventor ». <https://appinventor.mit.edu/> (consulté le sept. 24, 2021).

