

BADJI MOKHTAR- ANNABA UNIVERSITY

UNIVERSITE BADJI MOKHTAR ANNABA



جامعة باجي مختار - عنابة

Année : 2018

Faculté: Sciences de l'Ingéniorat

Département: Electronique

MEMOIRE

Intitulé :

Réalisation d'une station météorologique

Présenté en vue de l'obtention du diplôme de : MASTER

Domaine : Sciences et Technologie

Filière :Électronique

Spécialité: Réseaux et télécommunication

Par :

MAAFA ABDELKADER

DEVANT Le JURY

Président : D .MESSADEG

Grade : Pr UBM Annaba

Directeur de mémoire: M. ZADAM

Grade : MCB UBM Annaba

Examineurs: M .FEZZARI

Grade : Pr UBM Annaba

M.KADDECHE

Grade : Pr UBM Annaba

Remerciements

Je tiens à remercier, dans un premier temps, ALLAH, qui nous aide et nous donne la patience et le courage durant ces longues années d'étude.

Je veux offrir tout merci et appréciation à mon dirigeant, Pr : M. Zadam pour son aide et ses conseils tout au long de l'achèvement du projet.

Je remercie mes parents et toute ma famille, de m'avoir soutenu et encouragé pendant toutes mes études et surtout pendant ces années loin de la maison.

Pour finir, j'exprime mes remerciements mes plus sincères à tous mes enseignants au département d'électronique à l'université Badji Mokhtar Annaba, particulièrement mon jury.

ملخص

هذا المشروع هو محطة أرصاد جوية، حيث يتم قياس درجة الحرارة والرطوبة والضغط الجوي وسرعة الرياح ووجود المطر وإرسال كل هذه المعلومات إلى موقع على شبكة الإنترنت .

يتم ذلك عن طريق بطاقة اريينو مع بعض المستشعرات و بطاقة اترنت ، ثينغ سبيك هو الموقع الذي يوفر مساحة مجانية لانترنت الأشياء حيث يمكن لأي شخص الإطلاع على هذه المعلومات

Résumé

Ce projet est une station météorologique, où il est mesuré la température l'humidité, la pression atmosphérique, vitesse du vent, et présence de la pluie et envoyer toutes ces informations vers un site web sur l'internet.

Ceci est fait par une carte Arduino et plusieurs capteurs et Ethernet Shield . Thingspeak est un site web qui fournit un espace sur le Web gratuite pour l'internet des objets où n'importe qui peut voir ces informations.

Abstract

This project is a weather station, where it is measured the temperature, humidity, atmospheric pressure, wind speed, and presence of rain and send all this information to a website on the internet.

This is done by an Arduino board and several sensors and an Ethernet Shield. Thingspeak is a website that provides a free web space for the Internet of Things where anyone can see this information.

Liste des figures

Figure 1 : Image réelle du capteur de température et d'humidité DHT11.

Figure 2 : Câblage du capteur de température et d'humidité DHT11 avec l'Arduino UNO.

Figure 3 : Image réelle du capteur de Pression BMP280.

Figure 4 : Câblage du capteur de pression BMP280 avec l'Arduino UNO.

Figure 5 : Image réelle du capteur d'eau ST045.

Figure 6 : Câblage du capteur d'eau ST045 avec l'Arduino UNO.

Figure 7 : Image réelle de l'anémomètre professionnel

Figure 8 : Image réelle de l'anémomètre fabriqué

Figure 9 : Image réelle de moteur à courant continu et la turbine

Figure 10 : Câblage du moteur à courant continu l'Arduino UNO.

Figure 11 : Graphe représente la relation entre la vitesse du vent et la différence de tension entre les extrémités du moteur.

Figure 12 : Image réelle d'une carte Arduino UNO

Figure 13 : Image réelle de Ethernet Shield 2

Figure 14 : Image réelle de Arduino avec Ethernet Shield.

Figure 15 : Schéma synoptique de système à réaliser.

Figure 16 : Interfaçage de logiciel de simulation fritzing

Figure 17 : Schéma électrique

Figure 18 : Montage en plaque d'essai.

Figure 19 : Image réelle du montage réalisé connecté par USB.

Figure 20 : Fenêtre d'affichage des résultats de mesures des capteurs sur le moniteur sérié

Figure 21 : Nombre des objets connectés à l'horizon 2020.

Figure 22 : Logo de site ThingSpeak

Figure 23 : Courbe graphique exprime le changement de la Température.

Figure 24 : Courbe graphique exprime le changement de l'Humidité.

Figure 25 : Courbe graphique exprime le changement de la Pression.

Figure 26 : Courbe graphique exprime le changement de la Vitesse du vent.

Figure 27 : Courbe graphique exprime la présence de la pluie.

Liste des abréviations

ADC (Analog to Digital Converter) : Convertisseur Analogique/Numérique.

APIs (Application Program Interface) : Interface de Programme d'Application.

DC (Direct Current) : Courant Continu.

IoT (Internet of Things) : IdO(Internet des Objets).

I/O (Input/Output) : Entrée/Sortie.

IP (Internet Protocol) : Protocole Internet.

I2C (Inter-Integrated Circuit) : Circuit Inter-Intégré.

PC (Personal Computer) : Ordinateur Personnel.

PWM (Pulse Width Modulation) : Modulation de Largeur d'Impulsion.

TCP (Transmission Control Protocol) : Protocole de Contrôle de Transmission.

DHCP (Dynamic Host Configuration Protocol) : Protocole de Configuration

Dynamique des Hôtes.

MAC (Media Access Control) : Contrôle d'accès au support .

Sommaire

Introduction générale	1
Chapitre I : Généralités sur les capteurs.....	3
I .1 Introduction.....	3
I .2. Définition principale du capteur.....	3
I .3.Les capteurs	3
I .3.1 Le capteur de température et le capteur d'humidité (DHT11).....	3
I .3.2 Le capteur de pression(BMP280)	5
I .3.3 Le capteur d'eau(ST045)	7
I .3.4 L'anémomètre.....	7
Chapitre II : Unité de traitement et transmission des données	11
II .1. Unité de traitement des données.....	11
II .1. 2. Carte Arduino UNO.....	11
II .1. 1.3 Les avantages d'Arduino.....	12
II .2. Unité de transmission des données.....	13
II .2. 1.1 La connexion du Ethernet Shield avec le router	14
Chapitre III: Conception et réalisation du station météo.....	14
III .1.Introduction.....	15
III .2.Schéma en bloc du système à réaliser.....	15
III .2.1. Schéma électrique du montage sur Fritzing.....	15
III .3. Présentation de l'interfaçage Arduino-PC par câble USB et écran LCD.....	17
III .4. Choix du support d'affichage.....	20
III .4.1 Avantages de l'Internet des objets.....	20
III .4.2 Choix du site d'affichage.....	20
III .5. Installation de la carte à l'extérieur de Laboratoire - les résultats Obtenus.....	21
III .6. Conclusion.....	22
Conclusion générale.....	25
Annexes	26
Bibliographie	31

Introduction générale

De nos jours, la nécessité d'obtenir des informations en temps réel sur les paramètres environnementaux locaux est très importante. Avec ce projet nous fournissons ces informations de manière simple à toutes les personnes qui veulent être informé sur la situation météorologique d'une manière régulière.

Dans notre travail nous avons réalisé une station météo qui mesure les phénomènes physiques existants. Cette station sera connectée par Ethernet au réseau internet. Dans l'ensemble, la station comprenant plusieurs capteurs tels que: capteur de température, capteur d'humidité, capteur de pression, Capteur d'eau et capteur anémomètre. Une carte d'acquisition à base d'Arduino UNO est pour but de traiter les données mesurées de ces capteurs et transférer vers un site web à travers un Ethernet Shield.

Notre projet expérimental consiste donc à réaliser un système de mesure en temps réel de l'ensemble des grandeurs physiques existantes. Les données sont traitées dans l'unité de traitement et de commande à base de l'Arduino UNO, afin de les transférer par un protocole TCP/IP de transfert des données à travers Ethernet Shield. L'affichage des résultats est assuré par des courbes sur le site web choisi s'appel: 'ThingSpeak'.

Chapitre I: Généralités sur les capteurs

I .1.Introduction

Dans ce chapitre nous allons donner une description détaillée des capteurs utilisés et le mode de fonctionnement de ces capteurs dans notre station et comment peut-on connecter avec la carte Arduino UNO.

I .2. Définition principale du capteur

Un capteur est un dispositif ayant pour tâche de transformer une mesure physique observée en une mesure généralement électrique qui sera à son tour traduite en une donnée binaire exploitable et compréhensible par un système d'information.

Trois des capteurs utilisés ici sont des capteurs électroniques, de la marque shenzhen [<http://www.lctech-inc.com/>] , Ils utilisent le protocole I2C , tous ces capteurs contient convertisseur analogique numérique (ADC) transforme une tension comprise entre 0V et +5V en un nombre allant de 0 à 1023.

I .3.Les capteurs

I .3.1 Le capteur de température et le capteur d'humidité (DHT11)

DHT11 est un capteur numérique de température et d'humidité, le capteur composite contient une sortie de signal numérique calibré de la température et de l'humidité, application d'une technologie de collecte de modules numériques dédiés et de la technologie de détection de température et d'humidité, Le capteur comprend un sens résistif de composants humides et un dispositif de mesure de température, et connecté avec un microcontrôleur 8 bits haute performance.

Le seul inconvénient réel de ce capteur est que vous ne pouvez obtenir de nouvelles données qu'une fois toutes les 2 secondes, alors lorsqu' on déclare la bibliothèque de ce capteur dans le programme, la lecture de ce capteur peut durer jusqu'à 2 secondes [1].

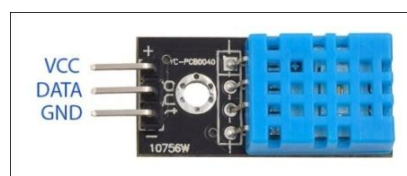


Figure 1 : Image réelle du capteur de température et d'humidité DHT11

Il suffit de connecter la première broche à gauche (Vcc) à l'alimentation de 3 à 5 V, la deuxième broche doit brancher à la carte Arduino, au pin 8 numérique. La troisième broche est montée à la masse GND (voir la figure 2).

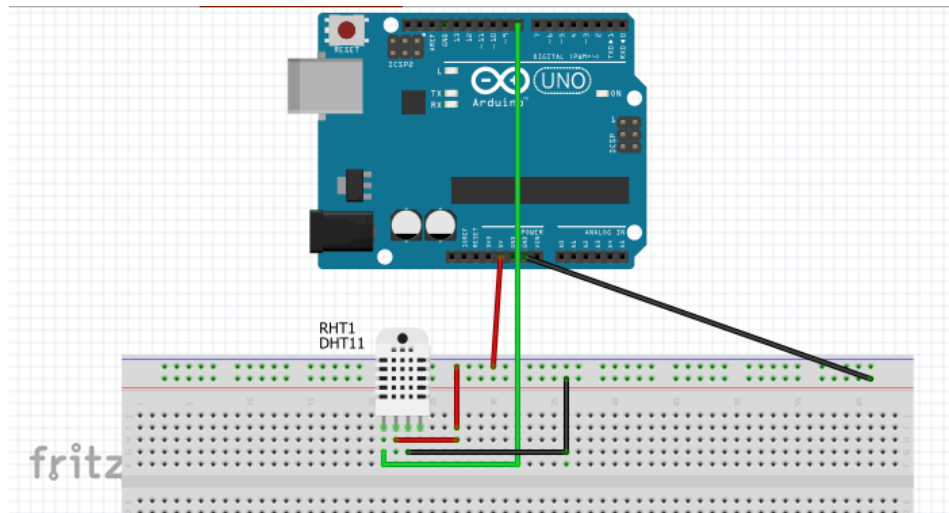


Figure 2: Câblage du capteur de température et d'humidité DHT11 avec l'Arduino UNO.

Principales caractéristiques de DHT11

- ❖ À bas prix.
- ❖ Puissance de E / S est 3.5 à 5.5 V.
- ❖ 2.5 mA max courant d'utilisation pendant la conversion (lors de la demande de données).
- ❖ Bon pour 0-100% de lectures d'humidité avec 2-5% de précision.
- ❖ Bon pour les lectures de température de -40 à 80 ° ± Précision ± 0,5 ° C.
- ❖ Pas plus de 0,5 Hz de fréquence d'échantillonnage (une fois toutes les 2 secondes).
- ❖ Taille du corps 12 mm × 15.5mm × 5.5 mm .
- ❖ Poids : 2,4 g. [2]

I .3.2 Le capteur de pression(BMP280)

Le capteur de pression BMP280, fabriqué par shenzhen, est la meilleure solution de détection à faible coût pour mesurer la pression atmosphérique. Parce que la pression change avec l'altitude, vous pouvez aussi l'utiliser comme altimètre.

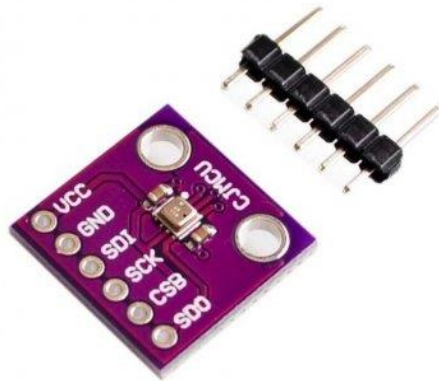


Figure 3 : Image réelle du capteur de Pression BMP280.

L'utilisation du capteur BMP 280 est facile. Si on utilise un Arduino UNO, on connecte simplement la broche Vin à la broche de tension 5V, GND à la terre, SCL à la broche Analogue 5 et SDA à la broche Analogue 4 .Ensuite, on fait déclarer la bibliothèque Arduino BMP180/BMP280 et un code d'exemple pour le calcul de la pression doit l'ajouter dans le programme.

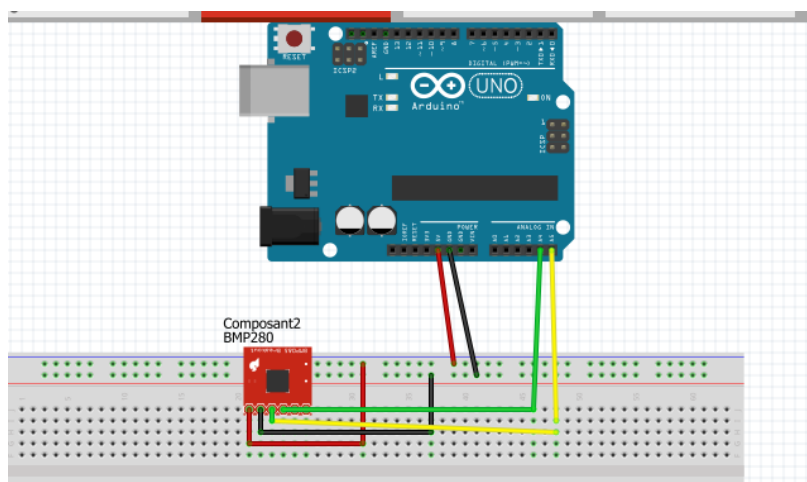


Figure 4 : Câblage du capteur de pression BMP280 avec l'Arduino UNO.

Principales caractéristiques de BMP280

- ❖ Alimentation maxi : 5,6V
- ❖ mesure de la température de -40°C à 85°C avec une précision de +/- 1°C
- ❖ mesure de la pression de 300hPa à 1100 hPa avec une précision de +/-1hPa (de 0°C à 65°C) ou +/- 1hPa (de -40°C à 0°C)
- ❖ mesure d'humidité relative de 0% à 100%
- ❖ sortie numérique sur bus I2C ou bus SPI [3].

I .3.3 Le capteur d'eau(ST045)

ST045 Est un capteur d'eau analogique , ce capteur travaille sur le principe de mesure de la taille des traces de gouttelettes d'eau à travers la ligne avec une série de fils parallèles exposés à la quantité d'eau pour simuler la deuxième plasticité basée sur les valeurs analogiques de sortie du capteur troisième puissance faible consommation haute sensibilité directement connecté à un microprocesseur ou à d'autres circuits logiques .

Ce module didactique délivre une tension analogique en fonction du niveau d'eau grâce à ses pistes imprimées. Le capteur délivre "700" lorsque le niveau est au maximum et "450" lorsque le niveau est au plus bas [4].



Figure 5: Image réelle du capteur d'eau ST045.

Il suffit de connecter la première broche à gauche (GND) à la masse, la deuxième broche doit brancher à l'alimentation de 3 à 5 V. La troisième broche (signal) doit brancher à la carte Arduino, juste au pin A3 analogique (voir la figure 6).

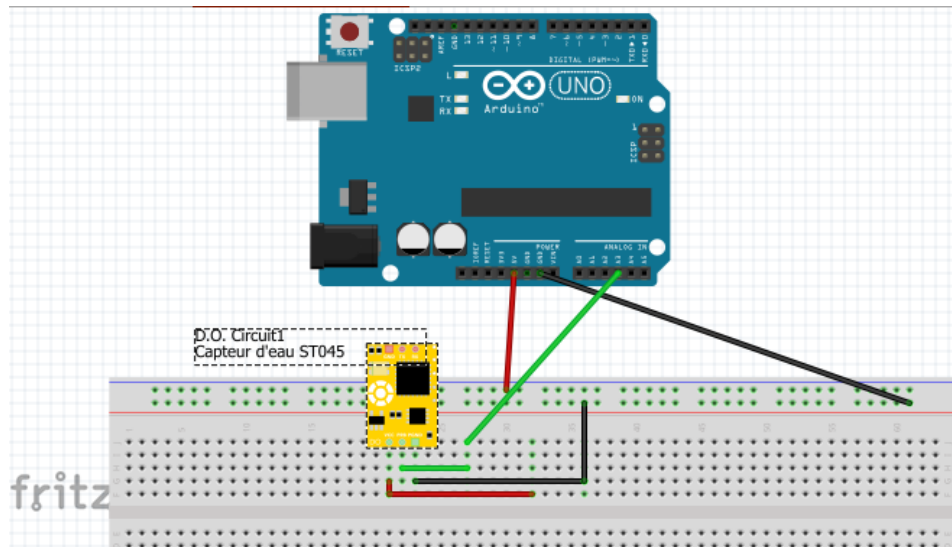


Figure 6: Câblage du capteur d'eau ST045 avec l'Arduino UNO.

I.3.4 L'anémomètre

L'anémomètre est un appareil électronique permet de mesurer la vitesse du vent, pour éviter la turbulence de l'air causée par la présence des bâtiments et de la végétation l'anémomètre doit installer au sommet d'un mât de 10 mètres. C'est à cette hauteur que les mesures standards sont effectuées.



Figure 7 : Image réelle de l'anémomètre professionnel

Vu l'indisponibilité de l'anémomètre professionnel sur le marché j'ai réalisé une installation avec un moteur DC, son principe est de connaître la vitesse du vent à travers la différence de tension entre les extrémités du moteur..



Figure 8 : Image réelle de l'anémomètre fabriqué

L'anémomètre fabriqué (voir figure 9) est un moteur a courant continu (dans notre cas il fonctionne comme un dynamo) avec turbine qui permet a moteur tourne a partir l'air .Lorsque le moteur tourne ,il génère une différence de tension , à partir de laquelle la vitesse du vent est calculée.

La différence de tension max qui ce moteur génère est 5 V, et la vitesse du vent max qui peut notre anémomètre mesurer est 195 Km /h.



Figure 9 : Image réelle de moteur a courant continu et la turbine

Il suffit de connecter la première broche (0V) de moteur à la pin A1 , la deuxième broche (+5V) doit brancher à la carte Arduino, au pin A0 analogique (voir la figure 10).

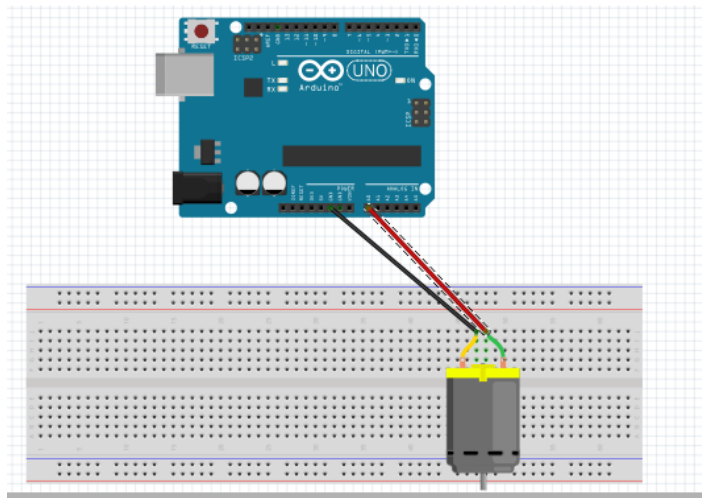


Figure 10: Câblage du moteur à courant continu l'Arduino UNO

Ces mesures ont été prises en comparant les paramètres d'un anémomètre professionnel et l'anémomètre fabricant. [5]

Vitesse du vent (km /h)	Différence de tension(mV)
20	82
30	125
40	190
50	250
60	320
70	390
80	430
90	480

Tableau : La relation entre la vitesse du vent et la différence de tension entre les extrémités du moteur

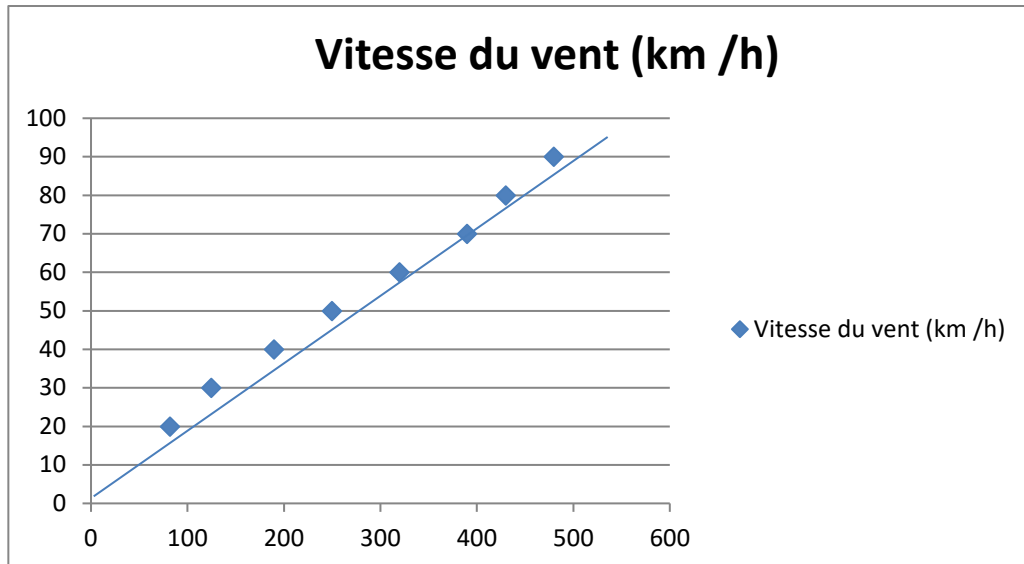


Figure 11 : Graphe représente la relation entre la vitesse du vent et la différence de tension aux bornes du moteur.

À partir la courbe on peut déduire la relation entre vitesse du vent et la différence de potentiel :

La pente de la courbe = $\frac{\Delta y}{\Delta x} = 38/20 = 0.190$

Vitesse du vent = 0.190 * Différence de tension

Chapitre II: Unité de traitement et transmission des données

II .1. Unité de traitement des données

Cette station météorologique se décompose en deux modules :

Le premier est la station météo, c'est à dire qu'il est traite et charge de la mesure des paramètres environnementaux, nous l'appellerons la station météo qui contenant la carte Arduino et les différents capteurs.

Les principales fonctionnalités devaient être :

- la mesure de l'humidité de l'air.
- la mesure de la température de l'air.
- la mesure de la pression atmosphérique.
- la mesure de la vitesse du vent.
- détection de la présence de pluie.

II .1. 2. Carte Arduino UNO

La carte Arduino Uno est une carte à microcontrôleur construite autour de l'ATmega 328. Elle possède 14 broches d'entrée/sortie numériques (dont 6 peuvent servir de sorties PWM), 6 entrées analogiques, un oscillateur à quartz de 16 MHz, un connecteur USB, un jack d'alimentation, une embase ICSP, et un bouton d'initialisation (reset). La carte Uno contient tout ce qui est nécessaire au fonctionnement du microcontrôleur. Pour l'utiliser, il suffit de la relier à un ordinateur avec un câble USB, ou encore de l'alimenter à l'aide d'un bloc secteur externe ou de piles. [6]

Ce carte comporte plusieurs entrées analogiques, des entrées et des sorties numériques. Le convertisseur analogique numérique possède 10 bits et sa tension de pleine échelle est par défaut de 5 V mais peut être réglée entre 2 et 5V.



Figure 12: Image réelle d'une carte Arduino UNO.

Caractéristiques techniques:

- Atmega328 Microcontrôleur.
- 5-12 V entrées voltage.
- 14 Digital E/S pins.
- 6 PWM sortie.
- 6 entrées analogiques.
- 16 Mhz Clock Signal.
- 32 kB mémoire flash.

II .1. 1.3 Les avantages d'Arduino

Le système Arduino, nous donne la possibilité d'allier les performances de la programmation à celles de l'électronique. Plus précisément, nous allons programmer des systèmes électroniques. Les principaux avantages de l'électronique programmée sont :

- Simplifie grandement les schémas électroniques.
- Diminué le coût de la réalisation.
- La charge de travail à la conception d'une carte électronique.
- Environnement de programmation clair et simple.
- Multiplateforme : tourne sous Windows, Macintosh et Linux.
- Nombreuses bibliothèques disponibles avec diverses fonctions implémentées.
- Logiciel et matériel open source et extensible.
- Nombreux conseils, tutoriaux et exemples en ligne (forums, site perso, etc.).
- Existence de « shield » (boucliers en français). [7]

II .2. Unité de transmission des données

Le deuxième module nommé base météo qui contenant la Ethernet Shield avec l'unité de stockage et le router a pour tâche de stocker les données et de l'envoi vers un site web avec mise à jour régulièrement.



Figure 13 : Image réel de Ethernet Shield 2

Le module Ethernet Shield permet à une carte Arduino de se connecter à internet. Il est basé sur le composant Wiznet 5500 et communique via le bus SPI. Ce module nécessite l'utilisation de la version 2 de la bibliothèque Ethernet fournit avec le logiciel Arduino.



Figure 14 : Image réel de Arduino avec Ethernet Shield

Caractéristiques principales:

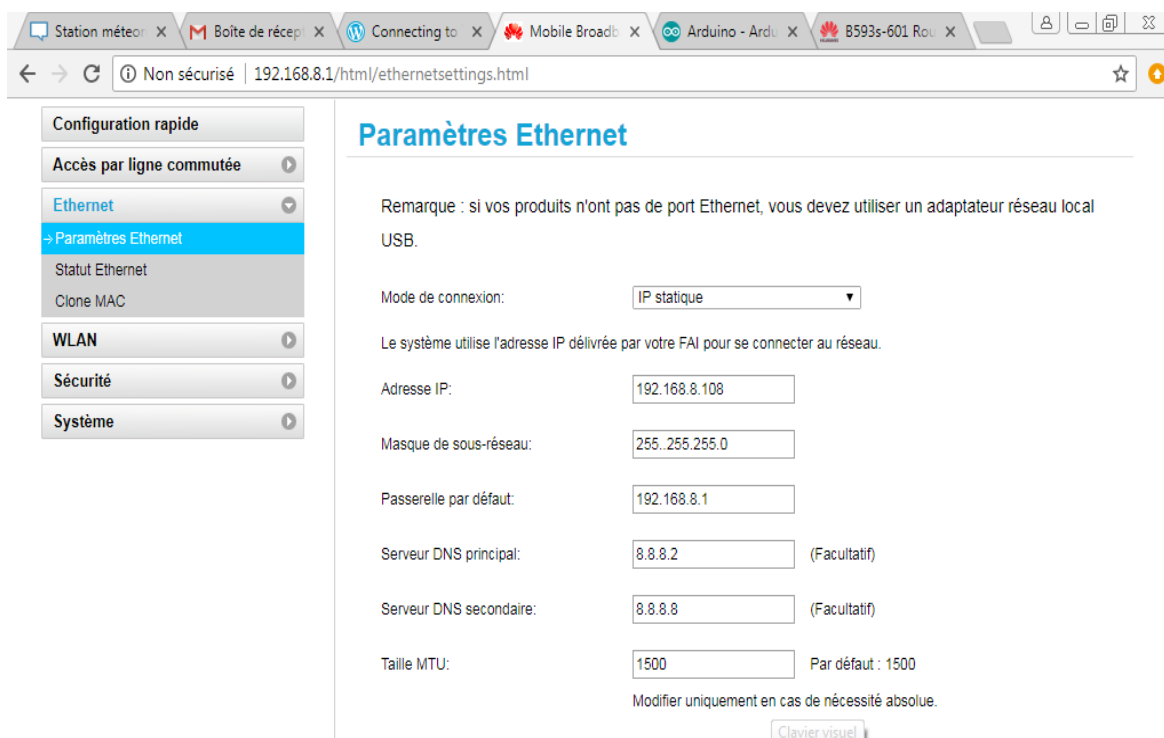
- version: 2
- circuit intégré: Wiznet 5500
- support pour micro-SD
- socle ethernet RJ45
- dimensions: 73 x 53 x 30 mm [8].

Les principales fonctionnalités devaient être :

- le stockage des données sur carte mémoire SD.
- l'envoi des données vers le site web.

II .2. 1 . La connexion du Ethernet Shield avec le router

On doit réserver une adresse ip statique pour le Ethernet Shield ,et activer DHCP du router pour ne pas avoir un conflit d'adresse IP.



The screenshot shows a web browser window with the URL `192.168.8.1/html/ethernetsettings.html`. The page title is "Paramètres Ethernet". On the left, there is a navigation menu with options: "Configuration rapide", "Accès par ligne commutée", "Ethernet" (selected), "Statut Ethernet", "Clone MAC", "WLAN", "Sécurité", and "Système". The main content area is titled "Paramètres Ethernet" and contains the following settings:

- Remarque : si vos produits n'ont pas de port Ethernet, vous devez utiliser un adaptateur réseau local USB.
- Mode de connexion:
- Le système utilise l'adresse IP délivrée par votre FAI pour se connecter au réseau.
- Adresse IP:
- Masque de sous-réseau:
- Passerelle par défaut:
- Serveur DNS principal: (Facultatif)
- Serveur DNS secondaire: (Facultatif)
- Taille MTU: Par défaut : 1500

At the bottom, there is a note: "Modifier uniquement en cas de nécessité absolue." and a "Clavier visuel" button.

-L'adresse MAC de Ethernet Shield doit unique dans notre LAN.

-L'adresse IP de Ethernet Shield doit unique dans notre LAN.

Chapitre III : Conception et réalisation du station météo

III .1.Introduction

Le besoin d'observer et de contrôler des phénomènes physiques tels que la température, la pression, l'humidité ou encore la vitesse du vent est essentiel pour de nombreuses applications industrielles et scientifiques, cette nécessité nous à diriger de réfléchir à la réalisation d'une nouvelle station météo qui peut mesurer ces grandeurs et l'envoyer vers un site web par Ethernet Shield à base d'une carte Arduino UNO.

On va essayer de faire une petite application sur l'internet des objets pour utiliser le site ThingSpeak, qui va nous permettre de connecter notre système par Ethernet Shield. On va présenter à la fin de ce chapitre les résultats obtenu après la réalisation de cette station.

III .2.Schéma en bloc du système à réaliser

Ce système est divisé en trois parties, la première partie c'est l es capteurs qui reçoivent les phénomènes physiques (température, humidité, pression, vitesse du vent et la présence de pluie) et la deuxième partie l'unité de traitement des données (Arduino UNO) et la dernière partie est l'unité de transmission des données (Ethernet Shield).

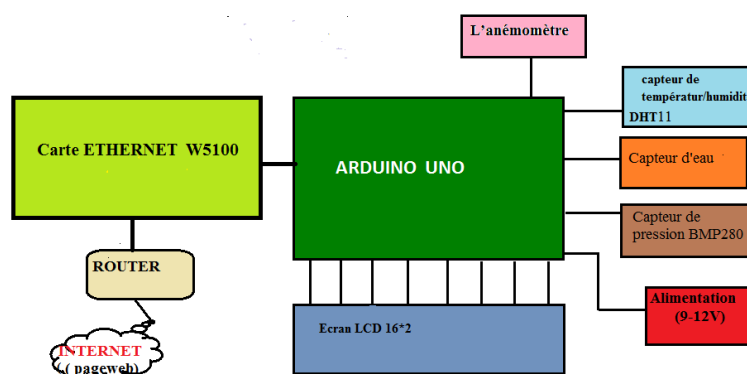


Figure 15 :Schéma synoptique de système à réaliser.

III .2.1. Schéma électrique du montage sur Fritzing

Tous d'abord on à tracé le schéma électrique du système sur le logiciel Fritzing. Le logiciel Fritzing est un outil de création des projets et des circuits

électroniques, il permet aussi l'édition de circuit imprimé, il est disponible gratuitement sur internet. Il a notamment pour vocation de favoriser l'échange de circuits électroniques libres et d'accompagner l'apprentissage de la conception de circuits.[9]

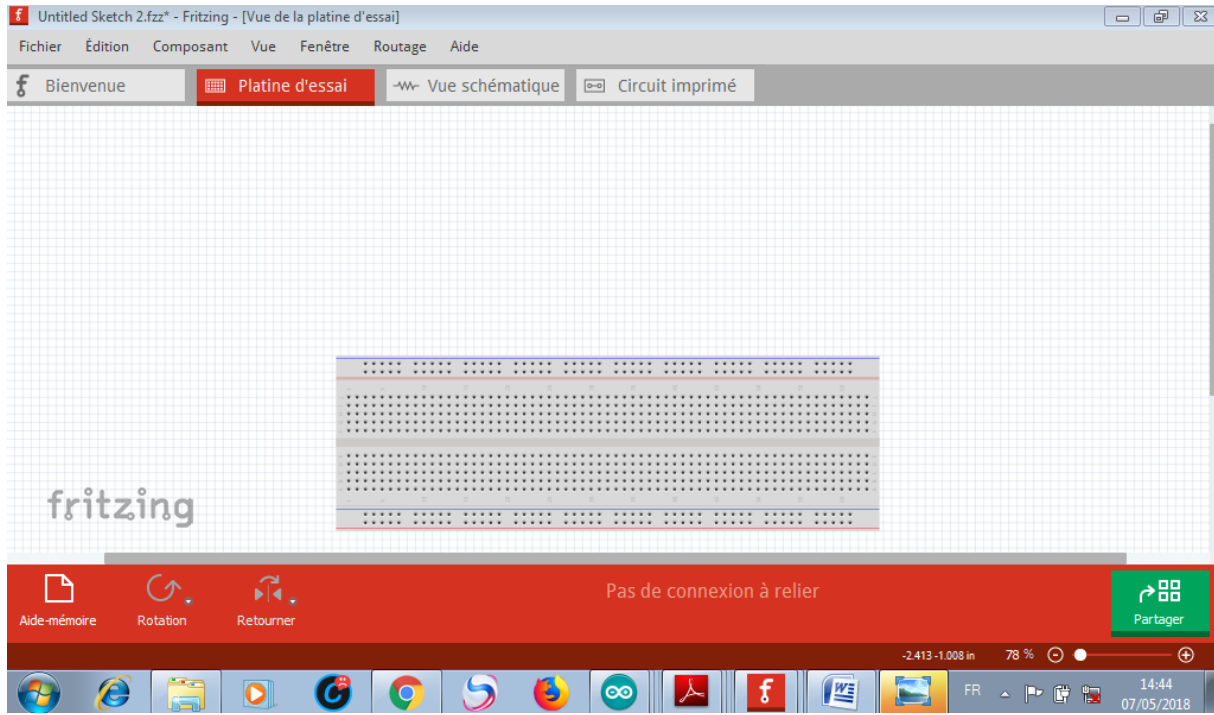


Figure 16 : Interfaçage de logiciel de simulation fritzing .

Le logiciel comporte trois vues principales:

- La « Platine d'essai », où l'on voit les composants tels qu'ils sont dans la réalité et où l'on construit le montage.
- La « Vue schématique », représentant le schéma fonctionnel du circuit.
- Le « Circuit imprimé », représentant la vue du circuit imprimé tel qu'il sera sorti en PDF pour être imprimé [10].

Dans les figure 17 et 18 on présente, respectivement, le schéma électrique tracé sur Fritzing et le montage de notre système sur plaque d'essai obtenu aussi par Fritzing.

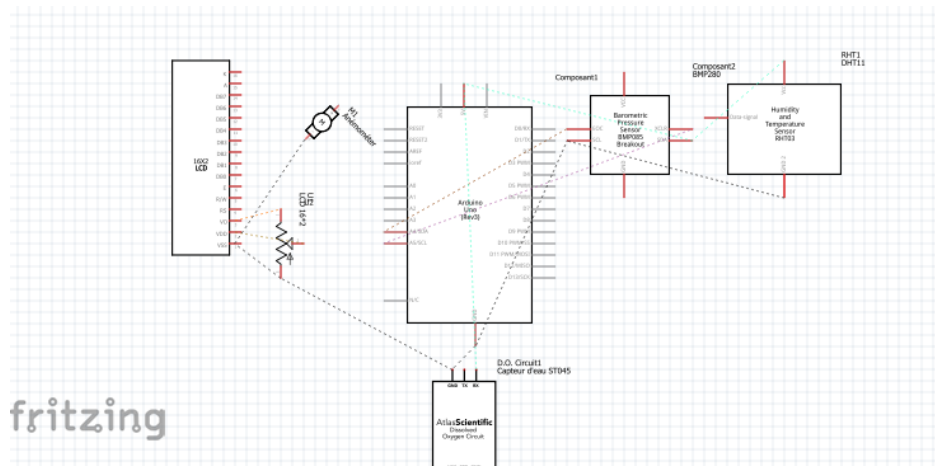


Figure 17: Schéma électrique.

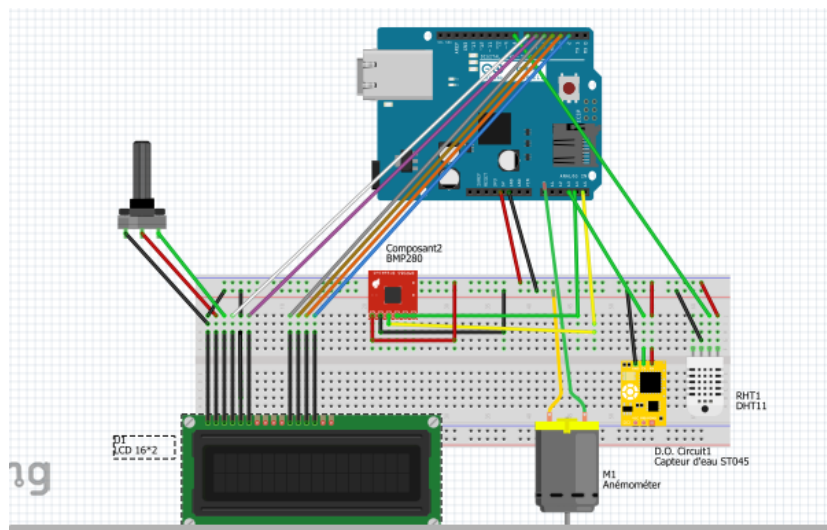


Figure 18: Montage en plaque d'essai

III .3. Présentation de l'interfaçage Arduino-PC par câble USB et écran LCD

Premièrement pour tester le fonctionnement des capteurs utilisés: (DHT11, BMP280, ST045, et l'anémomètre), On a essayé de faire un câblage entre ces capteurs et la carte Arduino UNO. La station météo réalisée est reliée de façon simple par un câble USB. Une image réelle de la station météo réalisée sur une plaque d'essai et connectée par un câble USB est présentée à la figure 19.

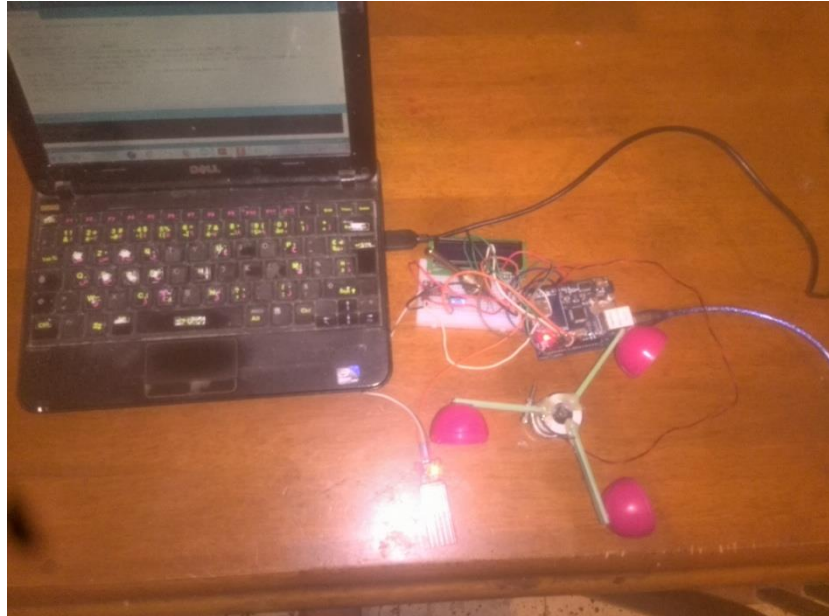


Figure 19: Image réel du montage réalisé connecté par USB.

Le programme de gestion des données permet d'afficher toutes les valeurs en temps réel et chaque deux secondes sur une fenêtre du moniteur sérié (Voir figure 20). Dans les annexes vous allez trouver le programme exécuté (voir l'annexe programme 01).

```
COM8
Station météorologique faite par MAAFA ABDELKADER
vitesse du vent:194.37Km/h
pas de pluie
Tempurature:24.0C
Humidite:35.0%
pression :950.85hPa
vitesse du vent:176.32Km/h
pas de pluie
Tempurature:24.0C
Humidite:35.0%
pression :950.81hPa
```

Figure 20: Fenêtre d'affichage des résultats de mesures des capteurs sur le moniteur sérié.

Après le test, nous avons remarqué que tous les capteurs fonctionnent correctement, avec de bons résultats. Par USB est instable. La deuxième étape est l'envoi de ces données vers un site web.

III .4. Choix du support d'affichage

Il y a plusieurs méthodes qu'on peut utiliser pour afficher les résultats des mesures, dans notre cas nous avons opté pour l'Internet des Objets en anglais : «Internet of thing (IoT)».

L'Internet des Objets appelé la 3^{ème} révolution industrielle, va profondément modifier le quotidien des personnes avec la domotique, la santé et loisirs, l'énergie, la distribution et notre environnement avec la cité intelligente ou les transports connectés.

Durant ces dernières années, le monde a connu un développement impressionnant de l'univers multimédia. Cela s'explique par le progrès technique et technologique ainsi que des innovations majeures qui ont bouleversé le monde de télécommunication : Informatique des nuages (Cloud Computing), social media, Internet des Objets...

Ce dernier représente l'extension d'Internet à des choses et à des lieux du monde physique. Alors qu'Internet ne se prolonge habituellement pas au-delà du monde électronique, l'Internet des Objets représente les échanges d'informations et de données provenant de dispositifs présents dans le monde réel vers le réseau Internet. L'Internet des Objets est considéré comme la troisième évolution de l'Internet, baptisée Web 3.0. Les Objets constituant « l'Internet des Objets », qualifiés de « connectés », « communicants » ou « intelligents » [11].

Actuellement les Objets connectés ou intelligents sont partout, ils ont envahi le monde et impacté notre vie personnelle et professionnelle. Ils génèrent des milliards d'informations qu'il faut traiter et analyser pour les rendre utilisables. Selon les statistiques de Cisco comme on peut voir sur la figure 21, 50 à 80 milliards d'appareils connectés seront en circulation dans le monde en 2020. Bien qu'il ne s'agisse que d'une estimation, une chose est sûre c'est qu'actuellement, en moyenne dans le monde une personne dispose de plus de trois Objets connectés. Une réalité qui a vu le jour et qui s'est développée depuis l'année 2003.[12]

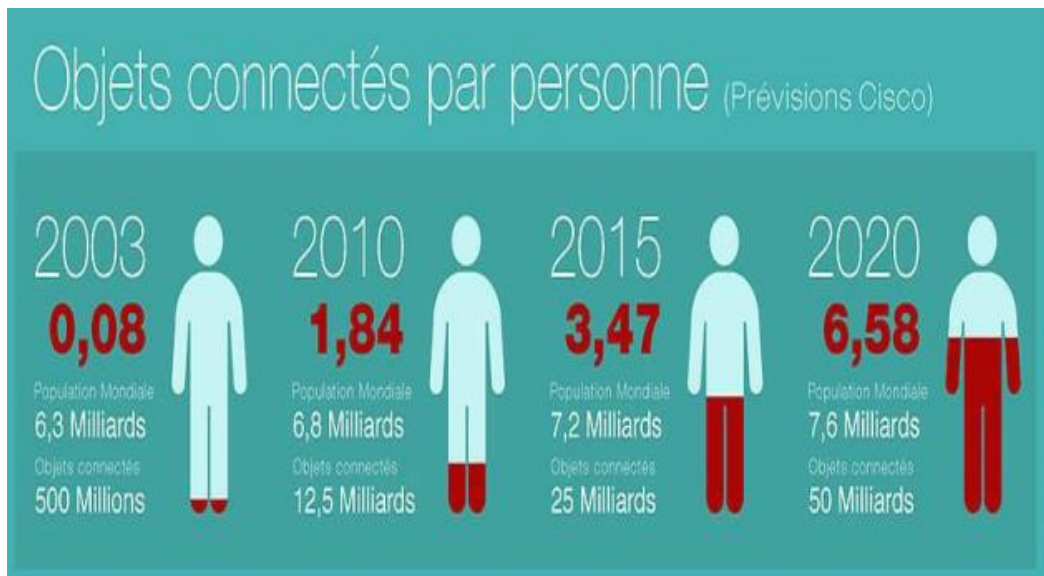


Figure 21 : Nombre des objets connectés à l'horizon 2020.

III .4.1 Avantages de l'Internet des objets

- Gagné beaucoup temps.
- Perfection.
- Accessibilité et mobilité : à tout moment, en tout lieu, tout appareil».
- Les objets connectés feront partis intégrante de notre quotidien dans tous les domaines (Santé, voiture, Life style, météo).

III .4.2 Choix du site d'affichage

Après le choix de l'internet des objets pour afficher les résultats, la dernière étape est de choisi un site d'affichage permet d'afficher les résultats en temps réel, nous avons opté le site "**ThingSpeak**".

Le site ThingSpeak est une entreprise qui propose différents services exclusivement destinés à la construction d'applications IoT.



Figure 22 : Logo de site ThingSpeak

Le site propose d'être un support d'IOT en permettant de:

- collecter les données en temps réel (fréquence supérieure ou égale à 15 secondes)
- visualiser les données collectées sous forme de graphes
- créer des plug-ins et des applications pour collaborer avec des web services, des réseaux sociaux et d'autres APIs.
-

Maintenant on doit créer un projet qui envoie des informations de notre station de météo vers ThingSpeak, en suivre les étapes suivantes:

- s'inscrire
- créer un Channel
- créer cinq "Fields" dans le Channel
- récupérer la clé de mise à jour (API Key ; Write KEY)

III .5. Installation de la carte à l'extérieur de Laboratoire - les résultats Obtenus

Nous avons installé la carte à l'environnement afin d'effectuer des mesures de tous les phénomènes physiques à différents moments de la journée pour savoir comment changer la température, l'humidité, la pression la présence du pluie tout au long de la journée, notez que il est nécessaire la connexion du station au LAN (router) dans le site choisi, en fin de journée nous avons acquis les mesures suivantes (figure 23 , 24 , 25 ,26 ,27).

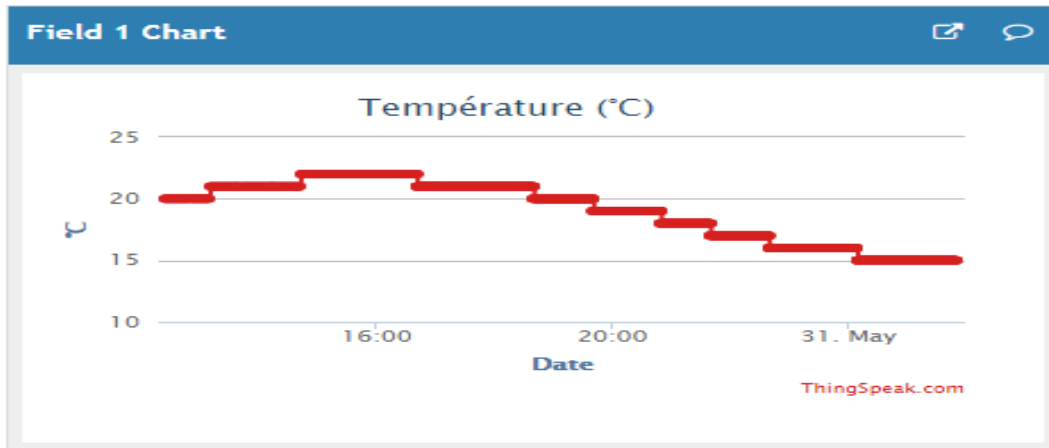


Figure 23: Courbe graphique exprime le changement de la Température.

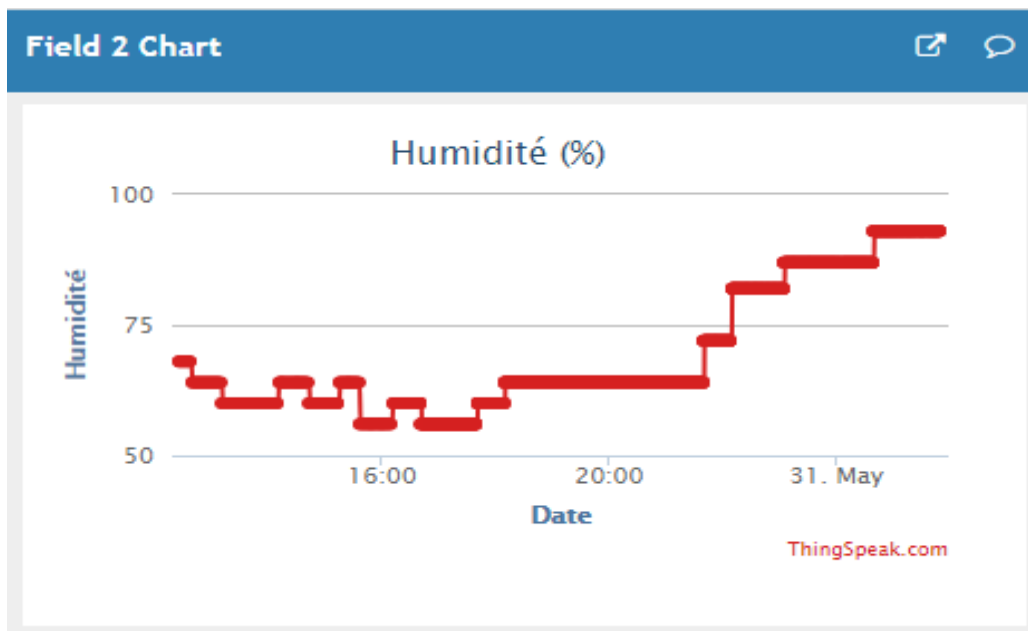


Figure 24: Courbe graphique exprime le changement de l'Humidité.

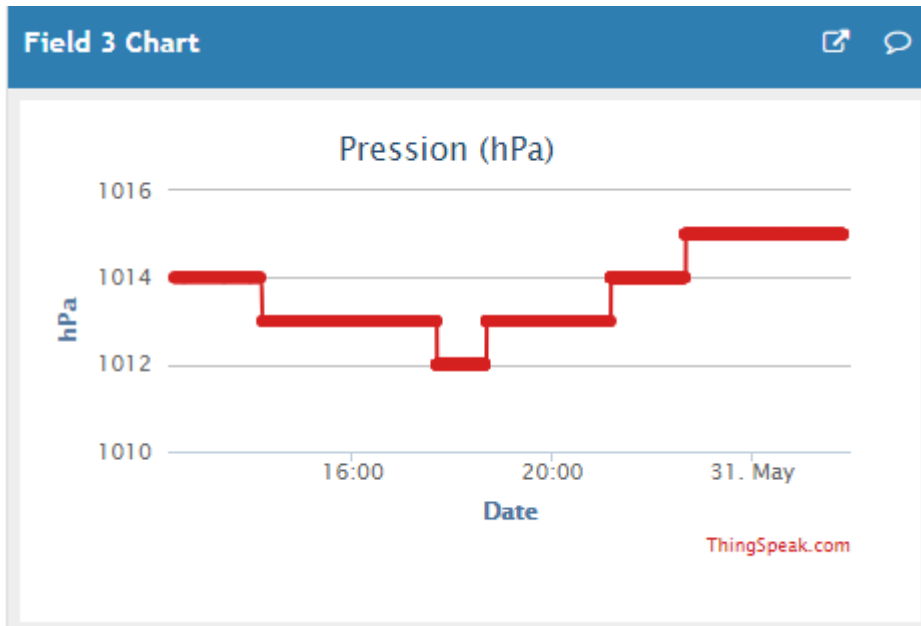


Figure 25 : Courbe graphique exprime le changement de la Pression.

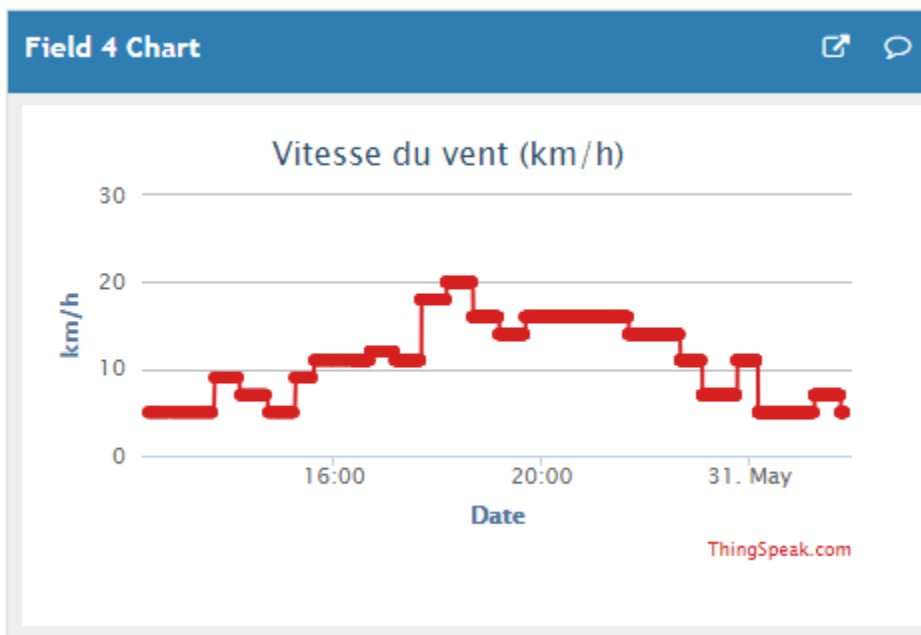


Figure 26 : Courbe graphique exprime le changement de la Vitesse du vent.

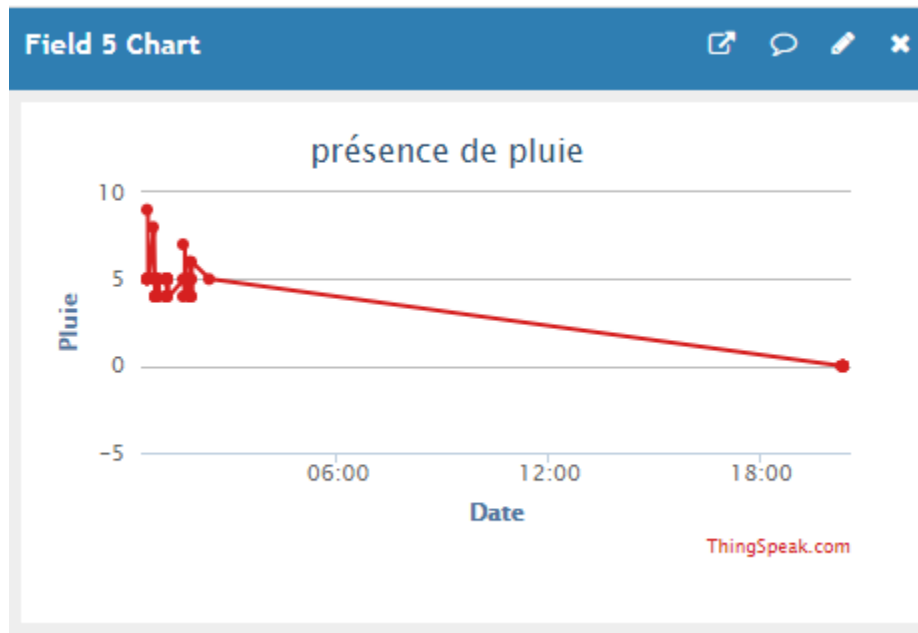


Figure 27 : Courbe graphique exprime la présence de la pluie.

III .6. Conclusion

Au début de ce chapitre, nous avons fourni une explication détaillée de la méthode de conception et réalisation de notre station météo et la méthode suivie pour envoier les données vers un site web.

Nous avons fait des mesures expérimentales en dehors de laboratoire pour voir comment les phénomènes physiques variés (la température, la pression, l'humidité, la vitesse du vent et la présence de pluie) les résultats des mesures en une journée sont montrés sous forme de graphes en fonction du temps.

Conclusion générale

Dans ce projet nous nous sommes concentrés à la conception et la réalisation d'une station météo pour l'acquisition des données grâce aux quatre capteurs: de température /humidité, de pression, capteur d'eau et capteur de vitesse du vent.

Nous avons réalisé un système de mesure en temps réel de l'ensemble des phénomènes physiques à base d'une carte Arduino UNO comme une station météo, le rôle de la carte Arduino UNO est de traiter les données délivrées par les capteurs utilisées. Au début on a essayé de relier le système de mesure au support d'affichage (PC) par un câble USB pour assurés le bon fonctionnement des capteurs. Le programme écrit sur IDE Arduino permet d'afficher les résultats sur le moniteur série.

La deuxième partie de ce travail consiste à faire une connexion filaire (Ethernet) à travers le module Ethernet Shield , nous avons créés un autre programme capable d'afficher les résultats en temps réel sur le site internet ThingSpeak sous forme des graphes.

Annexes

Programme (01) avec câble USB et affichage sur écran LCD

```
#include <Wire.h>
#include <LiquidCrystal.h>
#include <SPI.h>
//=====DHT11
#include <DHT.h>
#define DHTPIN 8 // ne pas connecter au 0 ou pin 1 connecter a pin 8
#define DHTTYPE DHT11 // Définir DHT11 module
DHT dht(DHTPIN, DHTTYPE);
LiquidCrystal lcd(7, 6, 5, 4, 3, 2);
//=====Anémomètre
int pin0 = A0; // analog pin0
int pin1 = A1; // analog pin1
int vent ;
int maxi = 198,mini = 0;
//=====capteur d'eau
const int waterSens = A3; //define water sensor
const int led =9; //define led to pin 9
int waterVal; //define the water sensor value
//=====BMP280
#include "SparkFunBME280.h"
BME280 capteur;
float p ;
//=====setup
void setup()
{
  Serial.begin(9600);
  Serial.println("Station météorologique faite par MAAFA ABDELKADER");
  lcd.begin(16,2);
  dht.begin();
  pinMode(led, OUTPUT); //set led as an output
  pinMode(waterSens, INPUT); //set water sensor as an input //=====BMP280
  while (!Serial) {

}
//configuration du capteur
capteur.settings.commInterface = I2C_MODE;
capteur.settings.I2CAddress = 0x76;
capteur.settings.runMode = 3;
capteur.settings.tStandby = 0;
capteur.settings.filter = 0;
capteur.settings.tempOverSample = 1 ;
capteur.settings.pressOverSample = 1;
capteur.settings.humidOverSample = 1;
delay(10); // attente de la mise en route du capteur. 2 ms minimum
// chargement de la configuration du capteur
capteur.begin();
}
void loop()
{
  delay(2000);
  //DHT11
float t = dht.readTemperature();
float h = dht.readHumidity();
//=====Anémomètre
int vent = ((abs(analogRead(pin0)-analogRead(pin1))) * 0.190 ); // 0,190 correspond à la pente de la courbe ici, ils doivent mettre le
nombre qu'ils ont calculé relation entre voltage et vitesse du vent
lcd.clear();
lcd.setCursor(0, 1); // Accédez à la ligne 1, position 0
lcd.print("vent:");
lcd.print(vent); // montre la vitesse du vent sur l'écran LCD
lcd.print("Km/h");
Serial.print("vitesse du vent:");
Serial.print(vent);
Serial.println("Km/h");
delay(2000);
lcd.clear();
//=====capteur d'eau
i = analogRead(eau);
Serial.print(i);
```



```

if (i<=0) {
lcd.setCursor(0,0);
lcd.print("pas de pluie");
Serial.print("pas de pluie");}
else if (i<=1) {
lcd.setCursor(0,0);
lcd.print("Il ya de pluie");
Serial.println("Il ya de pluie");}
delay(2000);
waterVal = analogRead(waterSens); //read the water sensor
if (waterVal > 0){
digitalWrite(led, HIGH);//if the water sensor senses water turn the led on
}
else{
digitalWrite(led, LOW);//if it doesn't sense anything turn the led off
}
//====================================================+DHT11
lcd.clear();
lcd.setCursor(0,0);
lcd.print("T:");
lcd.print(t,1);
Serial.print("Tempurature:");
Serial.print(t,1);
Serial.println("C");
lcd.print("C");
lcd.setCursor(9,0);
lcd.print("H:");
lcd.print(h,1);
Serial.print("Humidite:");
Serial.print(h,1);
Serial.println("%");
lcd.print("%");
delay(2000);
//====================================================BMP280
float p=captur.readFloatPressure()/100;
Serial.print(p);
Serial.println("hpa");
lcd.clear();
lcd.setCursor(0,1);
lcd.print("p:");
lcd.print(p);
lcd.println("hPa");
delay(2000);
}

```

Programme (02) Affichage sur site thingspeak

```

#include <SPL.h>
#include <Ethernet.h>
#include <Wire.h>
#include <LiquidCrystal.h>
LiquidCrystal lcd(7, 6, 5, 4, 3, 2);

//DHT11=====
#include <DHT.h>
#define DHTPIN 8 // ne pas connecter au 0 ou pin 1 connecter a pin 8
#define DHTTYPE DHT11 // Définir DHT11 module
DHT dht(DHTPIN, DHTTYPE);
//Anémometre=====
int pin0 = A0; // analog pin
int vent ;
int maxi = 198,mini = 0; // to start max/min temperature
//ST045=====
int pin3 = A3;
int pluie;
//====================================================BMP280
#include "SparkFunBME280.h"
BME280 capteur;
int pression ;

// Local Network Settings=====
byte mac[] = { 0xD4, 0xA8, 0xE2, 0xFE, 0xA0, 0xA1 }; // Must be unique on local network

```

```

byte ip[] = { 192,168,1,108 }; // Must be unique on local network
byte gateway[] = { 192,168,8,1};
byte subnet[] = { 255, 255, 255, 0 };

// ThingSpeak Settings
char thingSpeakAddress[] = "api.thingspeak.com";
String writeAPIKey = "JDWU34U8CC29B0FV"; // Write API Key for a ThingSpeak Channel
const int updateInterval = 10000; // Time interval in milliseconds to update ThingSpeak

// Variable Setup
long lastConnectionTime = 0;
boolean lastConnected = false;
int failedCounter = 0;

// Initialize Arduino Ethernet Client
EthernetClient client;

void setup()
{
    Serial.begin(9600);
    Ethernet.begin(mac, ip, gateway, subnet);
    delay(1000);
    Serial.print("ETHERNET SHIELD ip is :");
    Serial.println(Ethernet.localIP());
    // Start Ethernet on Arduino
    startEthernet();
    //=====280
    while (!Serial) {

    }
    //configuration du capteur
    capteur.settings.commInterface = I2C_MODE;
    capteur.settings.I2CAddress = 0x76;
    capteur.settings.runMode = 3;
    capteur.settings.tStandby = 0;
    capteur.settings.filter = 0;
    capteur.settings.tempOverSample = 1 ;
    capteur.settings.pressOverSample = 1;
    capteur.settings.humidOverSample = 1;
    delay(10); // attente de la mise en route du capteur. 2 ms minimum
    // chargement de la configuration du capteur
    capteur.begin();
}

void loop()
{
    //=====Anémometre
    int vent = ( analogRead(pin0) * 0.190 );
    String v = String( analogRead(pin0) * 0.190 );

    //=====pluie
    pluie = (analogRead(pin3)/1023);
    String p = String(pluie);
    //String pluie = String ( analogRead(pin3));

    //=====DHT11
    float t = dht.readTemperature();
    float h = dht.readHumidity();
    String temp=String(t);
    String humid=String(h);

    //=====BMP280
    int pression=capteur.readFloatPressure()/100;
    String pres=String(pression);
    //=====Affichage sur LCD
    //=====vitesse du vent
    lcd.clear();
    lcd.setCursor(0, 1); // Accédez à la ligne 1, position 0
    lcd.print("vent:");
    lcd.print(vent);// montre la vitesse du vent sur l'écran LCD
    lcd.print("Km/h");
    delay(2000);
    lcd.clear();
    //=====température et humidité
    lcd.setCursor(0,0);

```

```

lcd.print("T:");
lcd.print(t,1);
lcd.print("C");
lcd.setCursor(9,0);
lcd.print("H:");
lcd.print(h,1);
lcd.print("%");
delay(2000);
//=====pression
lcd.clear();
lcd.setCursor(0,1);
lcd.print("p:");
lcd.print(p);
lcd.println("hPa");
delay(2000);

// Print Update Response to Serial Monitor
if (client.available())
{
  char c = client.read();
  Serial.print(c);
}

// Disconnect from ThingSpeak
if (!client.connected() && lastConnected)
{
  Serial.println();
  Serial.println("...disconnected.");
  Serial.println();

  client.stop();
}

// Update ThingSpeak
if(client.connected() && (millis() - lastConnectionTime > updateInterval))
{
  updateThingSpeak("field1="+temp+"&field2="+humid+"&field3="+pres+"&field4="+v+"&field5="+p);
}

lastConnected = client.connected();
}

void updateThingSpeak(String tsData)
{
  if (client.connect(thingSpeakAddress, 80))
  {
    client.print("POST /update HTTP/1.1\n");
    client.print("Host: api.thingspeak.com\n");
    client.print("Connection: close\n");
    client.print("X-THINGSPEAKAPIKEY: "+writeAPIKey+"\n");
    client.print("Content-Type: application/x-www-form-urlencoded\n");
    client.print("Content-Length: ");
    client.print(tsData.length());
    client.print("\n\n");

    client.print(tsData);

    lastConnectionTime = millis();

    if (client.connected())
    {
      Serial.println("Connecting to ThingSpeak...");
      Serial.println();

      failedCounter = 0;
    }
    else
    {
      failedCounter++;

      Serial.println("Connection to ThingSpeak failed (" +String(failedCounter, DEC)+")");
      Serial.println();
    }
  }
}

```

```

}
else
{
  failedCounter++;

  Serial.println("Connection to ThingSpeak Failed (" +String(failedCounter, DEC)+")");
  Serial.println();

  lastConnectionTime = millis();
}
}

void startEthernet()
{

  client.stop();

  Serial.println("Connecting Arduino to network...");
  Serial.println();

  delay(1000);

  // Connect to network amd obtain an IP address using DHCP
  if (Ethernet.begin(mac) == 0)
  {
    Serial.println("DHCP Failed, reset Arduino to try again");
    Serial.println();
  }
  else {
    Serial.println("Arduino connected to network using DHCP");
    Serial.println();
    Serial.println("Data being uploaded to THINGSPEAK Server.....");
    Serial.println();
  }

  delay(1000);
}

```

Bibliographie

- [1 ,2] <https://akizukidenshi.com/download/ds/aosong/DHT11.pdf> (consulter le 05/05/2018).
- [3] <http://gilles.thebault.free.fr/spip.php?article47>(consulter le 05/05/2018).
- [4]https://www.gotronic.fr/art-module-capteur-de-niveau-d-eau-st045-26116.htm#compte_desc(consulter le 05/05/2018).
- [5] <https://www.youtube.com/watch?v=02e59n0ZmRE> (consulter le 05/04/2018).
- [6] <https://store.arduino.cc/usa/arduino-uno-rev3>
- [7] REGUIG BERRA Imadeddine , ALLAM Yassine,
“Etude et réalisation d’une station météo connectée par wifi “ ,Mémoire
présenté pour l’obtention Du diplôme de Master Académique ,
UNIVERSITE MOHAMED BOUDIAF - M’SILA 2016 /2017.
- [8] <https://www.gotronic.fr/art-module-ethernet-shield-2-arduino-23299.htm>
(consulter le 07/05/2018).
- [9, 10] <https://fr.wikipedia.org/wiki/Fritzing> (consulter le 07/05/2018).
- [11] Le Mag Web4. 10 Juillet 2015 <https://lemag.agenceweb4.ch/site/fr/lemag-web4?tag=1> (Consulter le 07/05/2018).
- [12] Evans. D, "L’internet des objets", Avril, 2011

