

Semestre 2 de l'année universitaire 2020-2021
L1 CMI Mécanique

Capteurs de CO₂, température, humidité

Jessy AZIZI, Mickael YE, Meili HELBERT
Encadré par José Maria Fullana

Remerciements

Dans un premier temps, nous tenons très sincèrement à remercier M.Fullana, qui a encadré ce projet. Il nous a beaucoup appris et guidés sur l'aspect théorique comme pratique de la chute d'une particule. Nous lui sommes très reconnaissants pour sa confiance, sa patience et le temps qu'il nous a consacré.

Nous tenons ensuite à remercier M. Berthaud, ainsi que Mme Dumontet, pour nous avoir donné l'opportunité de faire ce projet. Ils ont fait preuve de gentillesse, se sont rendus disponibles et surtout ils nous ont accordé leur confiance.

Nous aimerais enfin remercier Sorbonne Université, pour nous avoir permis de mener à bien ce projet malgré la crise sanitaire.

Table des matières

1 Résumé	4
2 Abstract	4
3 Introduction	5
4 Étude théorique	5
4.1 Particule en suspension dans l'air sans flux extérieur	5
4.2 Validation du modèle	7
4.2.1 Objectif	7
4.2.2 Protocole	8
4.2.3 Résultats	8
5 Influence d'un apport en flux	9
6 Élaboration d'un système de prise de mesures	11
6.1 Carte Arduino et capteur de CO ₂	11
6.2 Carte Raspberry Pi	14
6.3 I2C DEV	16
7 Mesures et interprétations	21
7.1 Amphithéâtre A1 durant un examen	21
7.2 Transports en commun	23
7.3 Bibliothèque des L1	24
8 Conclusion	25
A Annexe : Programme Arduino	26
B Références	29
C Bibliographie	30

1 Résumé

Dans le contexte actuel de pandémie, nous avons étudié le comportement des particules en suspension pour comprendre comment des particules de virus peuvent rester en suspension et en quoi l'aération des espaces fermés est importante non seulement pour réduire les chances d'être contaminé par le virus mais aussi pour assainir la qualité de l'air. A l'aide d'une modélisation, nous avons montré que la petite taille d'une particule, dont le diamètre varie entre 0.005 et 100 micromètres, demande de prendre en compte une nouvelle force s'opposant au poids de la particule et définie par la loi de Stokes. La présence de cette force explique alors le fait qu'une particule puisse rester en suspension dans l'air durant plusieurs heures. De plus, nous avons constaté en étudiant le nombre de Stokes qu'une particule a tendance à suivre l'écoulement du fluide dans lequel elle se trouve, d'où l'importance de l'aération des espaces fermés pour permettre le renouvellement de l'air. Ainsi, à l'aide de capteurs de CO₂, nous avons mené une campagne de mesures nous permettant de mesurer la concentration en CO₂ de certains espaces fermés de notre quotidien et d'évaluer la qualité de l'air.

2 Abstract

In the current pandemic context, we studied the behaviour of airborne particles with the aim of understanding how virus particles remain in suspension. Afterwards, we showed why ventilation of enclosed spaces is important, not only to reduce the chances of being contaminated by the virus but also to purify the air quality of a room. By means of modelling, we found that the small size of a particle, whose diameter varies between 0.005 and 100 micrometers, requires us to take into account a new force opposing the weight of the particle and defined by Stokes' law. The presence of this force explains why a particle can remain suspended in the air for several hours. Furthermore, we observed by studying the Stokes number that a particle tends to follow the flow of the fluid in which it is found, hence the importance of ventilating closed spaces to allow air renewal to avoid the accumulation of particles. With the help of CO₂ sensors, we conducted a measurement campaign enabling us to measure the CO₂ concentration and to assess the quality of the air of some closed spaces in our daily lives.

3 Introduction

Le sujet des particules fines entre directement dans le contexte actuel de la pandémie. En effet, la transmission du virus du COVID-19 se fait majoritairement par voie aérienne : celui-ci peut être porté par des particules fines présentes un peu partout dans l'air (pouvant être aussi appelées aérosols). L'étude des particules fines peut donc nous aider à mieux comprendre ce qui nous entoure. Tout d'abord, définissons notre sujet : les particules fines sont une catégorie de particules en suspension, présentes dans l'air ambiant. Généralement, elles ont un diamètre inférieur à 2,5 micromètres (soit $2,5 \cdot 10^{-6}$ m). Les particules en suspension sont un ensemble de particules portées par les liquides, les solides ou l'air. Elles peuvent rester dans l'atmosphère de plusieurs heures à plusieurs jours. Un taux trop élevé de particules fines dans l'air est un facteur de risque sanitaire pouvant provoquer des maladies cardiovasculaires, difficultés respiratoires, notamment des cancers des poumons, entraînant une nette diminution de l'espérance de vie. Une exposition chronique peut aussi provoquer voire amplifier le risque de maladie. Selon l'OMS (Organisation Mondiale de la Santé), les particules fines tuent plus de 4 millions de personnes par an. Elles ont deux origines, la première est naturelle avec notamment les éruptions volcaniques, feux de forêt, pollens etc. ; la deuxième est due aux activités humaines comme le tabagisme, le chauffage au bois, la combustion dans les centrales thermiques, les navires marchands ou navires de guerre, la pollution automobile, avec une nette augmentation de production d'aérosols ces 2 derniers siècles du fait du développement de plusieurs pays. Plusieurs politiques environnementales ont été menées pour contrer cela en durcissant les normes dans le secteur industriel. Dans la lutte contre le COVID-19, le gouvernement français a donné différentes recommandations, notamment celle d'ouvrir les portes et les fenêtres des établissements recevant du public afin de renouveler l'air ambiant et ainsi diminuer la transmission virale. Pour vérifier l'importance de ces décisions et pour éviter la propagation du virus, il faut donc contrôler la qualité de l'air. Le Haut Conseil de la santé publique indique que le taux de renouvellement de l'air peut être approché facilement par la mesure de la concentration en dioxyde de carbone dans l'air. Si la concentration en CO₂ est supérieure à un seuil de 800 ppm (Partie Par Million : Unité de mesure indiquant combien de molécules (ici de CO₂) on peut trouver sur un million de molécules d'air) alors une aération du lieu est nécessaire. L'objectif de notre projet est de comprendre comment une particule peut rester plusieurs heures en suspension dans l'air, puis, nous montrerons la nécessité d'une aération régulière pour améliorer la qualité de l'air d'un espace. Pour cela, nous étudierons, dans un premier temps, les lois physiques qui s'appliquent à la particule. Ensuite, nous traiterons de son mouvement en présence d'un écoulement dans le fluide où elle se trouve. Enfin, nous présenterons les résultats obtenus suite à la campagne de mesures que nous avons menée.

4 Étude théorique

4.1 Particule en suspension dans l'air sans flux extérieur

Dans cette partie, nous allons essayer de comprendre pourquoi la particule reste aussi longtemps en suspension dans l'air. Pour cela, nous étudierons le mouvement d'une particule M de masse m, que l'on suppose sphérique, dans un fluide au repos. Ainsi, nous considérerons le système masse situé dans le référentiel terrestre supposé galiléen. Les forces s'appliquant sur le système :

- Le poids : C'est une force de la pesanteur exercée par la Terre sur un corps massique, dirigée vers le bas. Elle est définie par :

$$\vec{P} = m \cdot \vec{g} = mg \vec{u}_z$$

avec :

\vec{g} est l'accélération de la pesanteur (exprimée en m/s^2)
 \vec{u}_z le vecteur unitaire vertical dirigé vers le bas.

- La poussée d'Archimède : C'est une force subie par tout corps plongé entièrement ou partiellement dans un fluide et soumis à un champ de gravité. Elle est définie par :

$$\vec{P}_A = -\rho_f \cdot V \cdot \vec{g} = -\rho_f \cdot V_p \cdot g \cdot \vec{u}_z$$

avec :

ρ_f : masse volumique du fluide dans lequel évolue la particule (exprimée en $kg.m^{-3}$)

\vec{g} : l'accélération de la pesanteur (exprimée en m/s^2)

V : volume de la particule (exprimé en m^3)

- Force de frottement : Le frottement est une interaction qui s'oppose au mouvement entre 2 systèmes au contact. Elle peut s'exprimer en tant que force. Hypothèse : On supposera ici que le nombre de Reynolds Re est très petit devant 1 (que l'on vérifiera) et donc en appliquant la loi de Stokes :

$$\vec{F} = -6\pi\eta r v \vec{u}_z$$

avec :

v : vitesse de la particule (exprimée en m/s)

r : rayon de la particule (exprimé en m)

η : viscosité dynamique (exprimée en Pa.s)

Le nombre de Reynolds est un nombre sans dimension dans le domaine de la mécanique des fluides. Il caractérise un écoulement et en particulier la nature de son écoulement (laminaire, turbulent, transitoire). Il est défini comme le rapport entre les forces d'inertie et les forces visqueuses.

$$Re = \frac{\rho v d}{\eta}$$

avec :

ρ : la masse volumique du fluide,

v : la vitesse de la particule,

d : la longueur caractéristique,

η : la viscosité dynamique.

La Loi de Stokes est une loi donnant la force de frottement d'un fluide sur une sphère en déplacement dans un fluide.

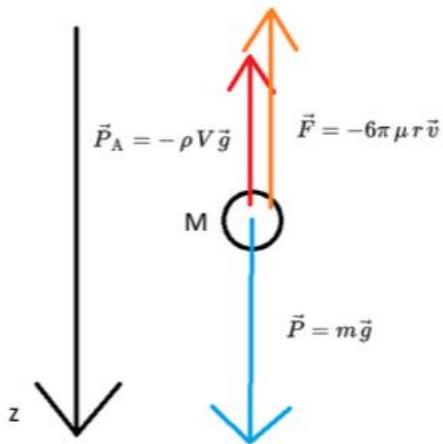


Schéma de la modélisation

En utilisant la seconde loi de Newton, on obtient :

$$m.\vec{a} = \text{Somme des forces}$$

$$m.\vec{a} = \vec{F} + \vec{P}_A + \vec{P}$$

En projetant la relation sur l'axe u_z :

$$m.\frac{d^2z}{dt^2} = m.g - \rho_f.V.g - 6\pi.\eta.r.v$$

On peut donc déterminer la vitesse terminale de la particule dans le fluide : son accélération étant nulle, on a $m.\frac{d^2z}{dt^2} = 0$ et donc on obtient la relation :

$$m.g - \rho_f.V - 6\pi.\eta.r.v_{terminale} = 0$$

Et en posant $m = \rho_{particule}.V_{particule}$ (avec $\rho_{particule}$ la masse volumique de la particule), on a :

$$v_{terminale} = \frac{g.\Delta\rho.d^2}{18.\eta}$$

avec :

$\Delta\rho$: la différence de masse volumique (exprimée en kg/m^3)

d : diamètre de la particule (en m)

On vérifie à présent si ce modèle nous donne une bonne vision de la réalité.

4.2 Validation du modèle

4.2.1 Objectif

L'objectif de cette partie est de valider les résultats théoriques en vérifiant si la formule obtenue pour la vitesse terminale rend bien compte de la réalité. Pour cela, nous utilisons des billes de masse volumique et de diamètre différents. Nous les plongeons dans une éprouvette remplie de

glycérine, dont on pourra modifier la viscosité en ajoutant un certain volume d'eau. On utilise de la glycérine et non de l'eau car l'écoulement de l'eau autour de la bille est turbulent, tandis qu'avec la glycérine l'écoulement est laminaire. On dit qu'un écoulement est laminaire si l'ensemble du fluide s'écoule approximativement dans la même direction, sans changement local. Un écoulement est dit turbulent si, en tout point, la vitesse possède un caractère tourbillonnaire.

À $t=0$, on lâche la bille sans vitesse initiale dans l'éprouvette. Après un certain temps τ , considéré comme très court, la bille atteindra une vitesse constante : la vitesse terminale. En traçant 2 traits sur l'éprouvette séparés d'une distance d , et en mesurant , à l'aide d'un chronomètre, le temps Δt que met la bille pour parcourir la distance d , on peut déduire la vitesse terminale de la bille avec la formule : $v = \frac{d}{\Delta t}$ et la comparer avec la valeur théorique obtenue.

4.2.2 Protocole

Matériel : Éprouvette, glycérine, marqueur, chronomètre, billes de tailles et masses volumiques différentes , balance, pied à coulisse et règle.

- Faire 2 traits sur l'éprouvette séparés d'une distance d .
- Remplir l'éprouvette de glycérine.
- Mesurer le diamètre d'une bille à l'aide du pied à coulisse, puis sa masse avec la balance, en séduire la masse volumique de la bille.
- Lâcher la bille à la surface de la glycérine.
- Lorsque la bille passe le premier trait, lancer le chronomètre.
- Lorsque la bille passe le deuxième trait, arrêter le chronomètre.

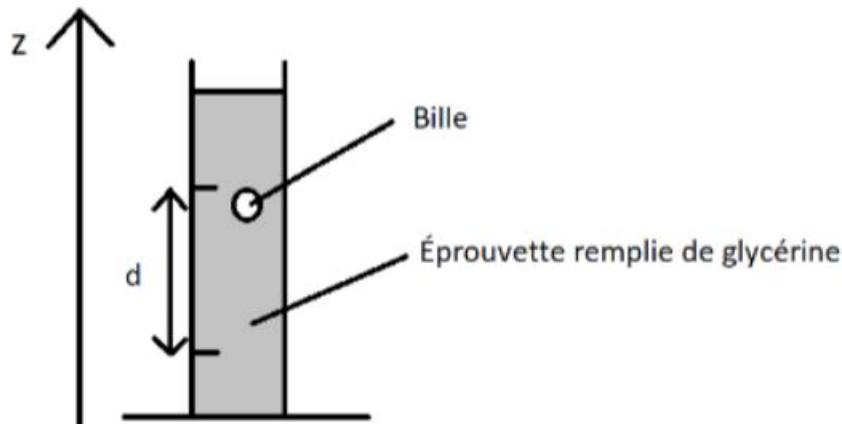
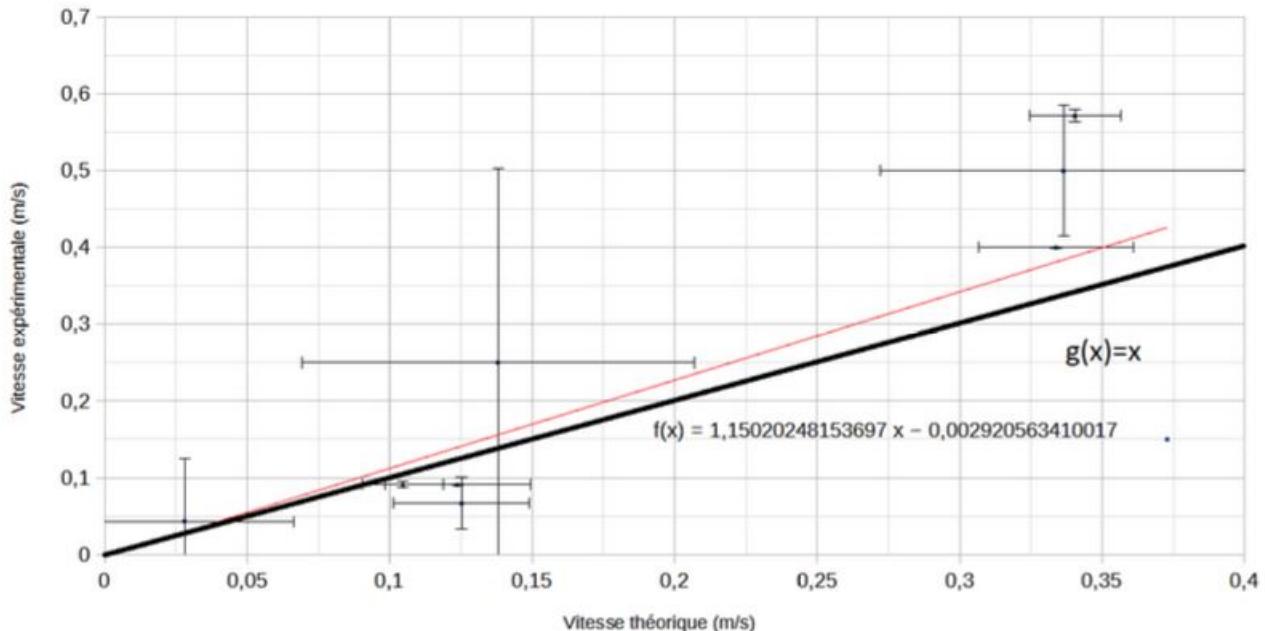


Schéma de l'expérience

4.2.3 Résultats

Nous avons utilisé une balance Roberval pour mesurer la masse des billes, filmé au téléphone la chute de la bille pour minimiser l'incertitude sur le temps en enlevant le temps de réaction et un pied à coulisse pour les mesures de distance. Nous obtenons les résultats suivants :

Graphique de la vitesse théorique en fonction de la vitesse expérimentale



Nous pouvons remarquer des différences notables entre les vitesses théoriques et les vitesses obtenues expérimentalement. Nous pouvons justifier ces différences tout d'abord par l'accélération de la particule. En effet, au moment où la particule passe par le premier trait, celle-ci n'a pas encore atteint sa vitesse finale et continue à accélérer jusqu'à atteindre sa vitesse maximale. Ceci pourrait contredire notre hypothèse de vitesse constante et de temps d'accélération négligeable. Cela est illustré par une plus grande différence de vitesse quand celle-ci devient de plus en plus grande. En lâchant avec une vitesse initiale nulle, si la vitesse maximale est trop grande, le temps pour que la particule atteigne cette vitesse, celle-ci dépasse déjà le premier trait et n'a toujours pas atteint sa vitesse finale. De plus, nous avons supposé que la particule est sphérique mais les billes ne sont pas des sphères parfaites et possèdent des défauts, ce qui peut influencer la vitesse à cause des frottements. Cependant, une partie des valeurs obtenues font partie de la droite et en prenant en compte les erreurs dans le protocole, le modèle peut être considéré comme validé.

Avec cette expérience, nous avons montré que le modèle choisi est valide. Nous pouvons supposer que ce modèle est applicable dans notre domaine. En prenant une particule dans l'air de diamètre d'environ 10 micromètres de masse volumique $\rho_p = 1000 \text{ kg/m}^3$, à environ 10 mètres du sol, dans l'air. Nous obtenons une vitesse terminale égale à 2,9 mm/s. Ce qui nous donne un temps t de $t = 3449 \text{ s}$ pour tomber, soit environ 58 min. Nous calculons le nombre de Reynolds $Re = 0.0019 < 1$. Le modèle utilisé est correct.

5 Influence d'un apport en flux

Dans cette partie, on cherche à connaître l'importance de l'ouverture des fenêtres et des portes pour le renouvellement de l'air. Nous considérons donc une particule dans une salle avec tous les accès fermés. A un instant t, nous ouvrons par exemple une fenêtre. La différence de pression entre l'intérieur et l'extérieur va provoquer un débit d'air de l'extérieur vers l'intérieur et aussi de l'intérieur vers l'extérieur. Généralement, la pression extérieure étant inférieure à celle de l'intérieur due à une plus grande température, nous pourrons donc considérer l'écoulement de l'intérieur vers

l'extérieur. Pour savoir si la particule va suivre l'écoulement, on introduit le nombre de Stokes : S_t . Le nombre de Stokes est un nombre sans dimension permettant de comprendre le comportement d'une particule dans un fluide. Il est défini par :

$$S_t = \frac{\rho_p \cdot d_p^2 \cdot v}{18 \cdot \mu \cdot L_c}$$

avec :

ρ_p : Masse volumique de la particule (exprimée en kg/m^3)

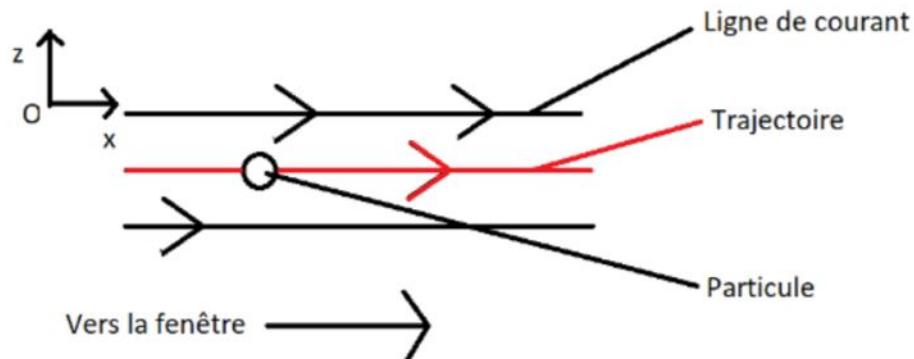
d_p : Longueur caractéristique de la particule (exprimée en m)

v : vitesse du fluide (exprimée en m/s)

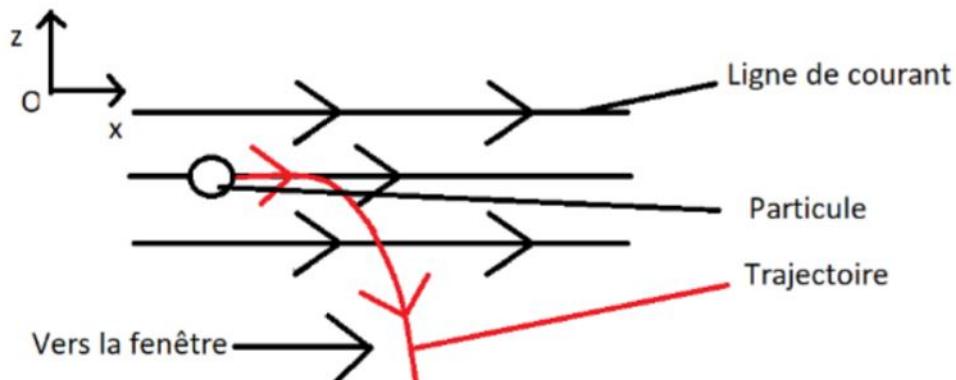
μ : Viscosité dynamique du fluide (exprimée en Pa.s)

L_c : Longueur caractéristique (exprimée en m)

Si ce nombre est très grand devant 1, la trajectoire de la particule est très peu affectée par l'écoulement et on dit qu'elle est en régime inertiel. Au contraire, si ce nombre est très petit devant 1, on dit que la goutte est un traceur de l'écoulement, c'est-à-dire qu'elle est directement emportée par celui-ci.



Dans le cas $S_t << 1$: La particule suit l'écoulement



Dans le cas $S_t >> 1$: La particule ne suit pas l'écoulement

Faisons une estimation du nombre de Stokes de la particule : prenons comme longueur caractéristique de la particule (d_p) son diamètre de 10 micromètres, sa masse volumique est de $1000 kg/m^3$, la viscosité dynamique de l'air, $18,5 \cdot 10^{-6} \text{ Pa.s}$ et une longueur caractéristique correspondant à la

longueur de la salle soit environ 10 mètres, ce qui donne un nombre de Stokes en fonction de la vitesse :

$$S_t(v) = 3,00 \cdot 10^{-5} \cdot v$$

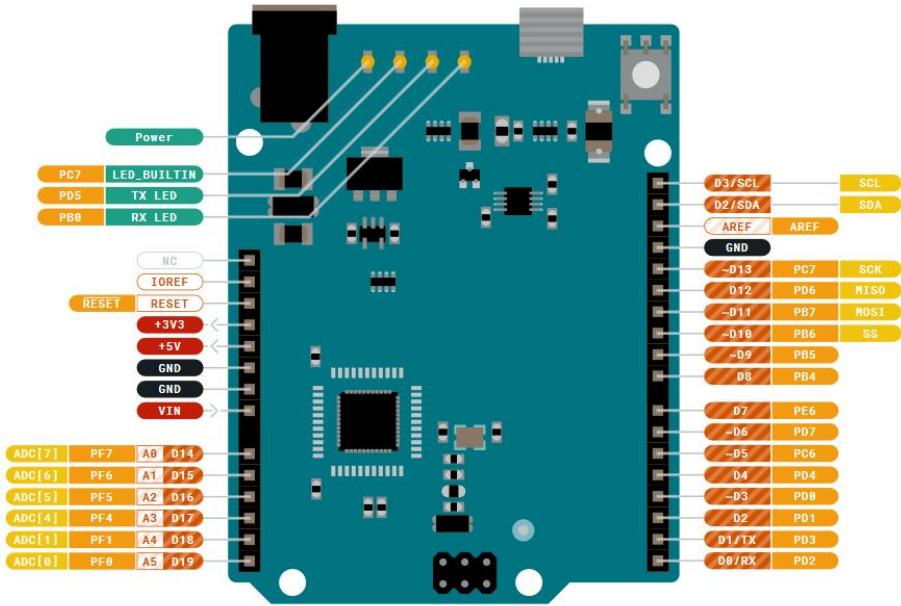
On remarque que le nombre de Stokes reste petit devant 1. Nous en déduisons donc que la particule va suivre l'écoulement et sortir de la salle lorsque l'on ouvre la fenêtre ou la porte. Le renouvellement de l'air pourra être vérifié avec les mesures.

6 Élaboration d'un système de prise de mesures

Dans la partie précédente, nous avons mis en évidence le fait qu'une particule entre en vitesse de sustentation dans les conditions où celle-ci ne serait pas engendrée par la diffusion d'autres particules de son environnement. D'après notre modèle d'étude, en considérant une particule stationnaire, nous pouvons conjecturer le fait que les particules sont présentes dans un champ homogène autour du capteur.

6.1 Carte Arduino et capteur de CO₂

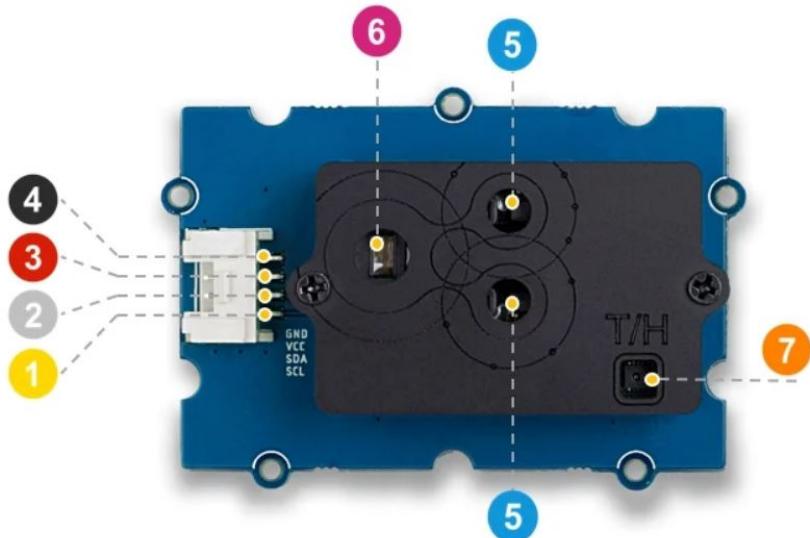
Pour cela nous avons utilisé une carte Arduino et un capteur de co₂. Une carte Arduino est une carte électronique équipée d'un microcontrôleur. Celui-ci récupère les informations des capteurs auxquels la carte est reliée pour les transmettre en suivant le programme que nous lui aurons téléversé. Ici, nous avons utilisé la carte Leonardo basée sur l'ATmega32u4 composée de 20 broches d'entrées et sorties. Pour la faire fonctionner, il faut simplement la brancher à un ordinateur ou à une batterie et y téléverser un programme que l'on aura écrit avec le logiciel Arduino (cf Annexe). La programmation Arduino fonctionne avec seulement deux fonctions : setup et loop. Ainsi, cela garantit une certaine simplicité qui la rend accessible au grand public. La fonction setup permet la configuration des paramètres que l'on utilise dans la fonction loop. Ensuite, les instructions de la fonction loop sont exécutées en boucle par la carte Arduino avec une période que l'on définit. Dans notre cas, nous pouvons ainsi récupérer une mesure à chaque tour de boucle.



Arduino Leonardo utilisé dans le montage [1]

Pour effectuer nos mesures, nous avons relié la carte Arduino à un capteur SCD30 qui permet de mesurer la concentration en CO₂ avec une précision pouvant atteindre $\pm(30 \text{ ppm} + 3\%)$ pour des concentrations comprises entre 400 ppm et 10 000 ppm. Ce capteur mesure également la température et l'humidité du lieu où il se trouve.

Pour connecter le capteur et la carte Arduino, nous avons relié les 4 broches (numéros 1 à 4 de la figure) aux 4 broches correspondantes sur la carte. Ensuite, une fois allumé, le capteur émet une source de lumière infrarouge (numéro 6 de la figure) qui permet de mesurer la concentration en CO₂ (numéro 5). En effet, le dioxyde de carbone absorbe dans l'infrarouge à une longueur d'onde de 4,2 micromètre. Ainsi, la quantité de rayonnement traversant le gaz dépend directement de la concentration en CO₂ et permet au capteur de faire une mesure. La température et l'humidité sont, quant à elles, obtenues au niveau du numéro 7.



- ④ GND: connect this module to the system GND
- ③ VCC: you can use 5V or 3.3V for this module
- ② SDA: I²C serial data
- ① SCL: I²C serial clock

- ⑤ CO₂ Sensor Opening
- ⑥ Infrared Light Source
- ⑦ Temperature & Humidity Sensor Opening

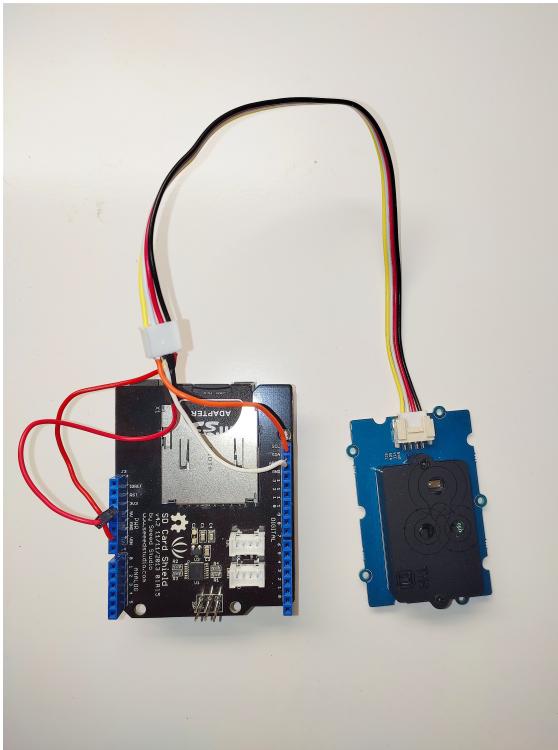
Capteur SCD30 [2]

Par ailleurs, afin de récupérer les données collectées par le capteur, nous avons utilisé un shield hébergeant une carte SD. Le shield possède le même nombre de broches d'entrées et sorties que la carte Arduino, ce qui nous permet de le superposer à la carte à laquelle il est alors relié. Il nous est alors possible d'écrire les mesures obtenues sur un fichier stocké dans la carte SD.



Shield Arduino

Nous obtenons alors le montage suivant :



Montage Arduino + Capteur de CO2

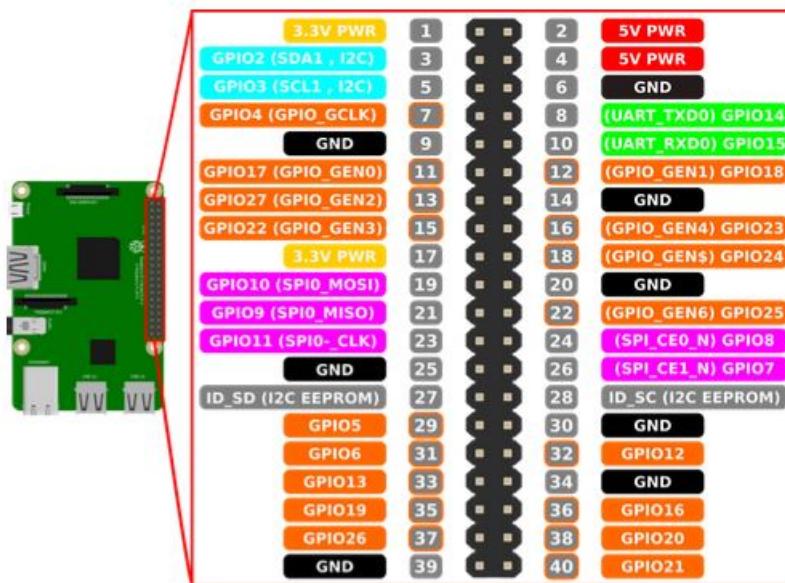
Lors de l'élaboration de ce système, nous avons rencontré quelques problèmes. Avant de pouvoir utiliser les trois capteurs de CO₂ à notre disposition, il est recommandé d'exposer les capteurs à l'air libre au moins 1 heure/jour durant 7 jours afin de l'étalonner. Or, nous avons étalonné les trois capteurs à trois endroits différents, ce qui a engendré une différence d'environ 200 ppm entre un capteur et les deux autres sur les mesures de concentration en CO₂ à un même endroit. Ensuite, en ce qui concerne la carte Arduino, elle ne nous permet pas de traiter les données en traçant un graphique. Il faut alors récupérer les mesures collectées sur la carte SD et réaliser des graphiques manuellement. De plus, nous n'avons pas pu récupérer les données en fonction du temps avec la carte Arduino. Ici aussi, nous avons ajouté le temps sachant qu'une mesure est réalisée toutes les 2 secondes. Afin de prendre des mesures dans différents lieux de façon autonome, et sans ordinateur, il nous fallait élaborer un système alimenté par une batterie, enregistrant chaque donnée ainsi que son heure d'enregistrement. Nous avons alors opté pour une carte raspberry pi.

6.2 Carte Raspberry Pi

Un raspberry Pi est un micro-ordinateur, basé sur une architecture ARM, embarquant un système d'exploitation. Il embarque une multitude de capteurs et d'antennes ainsi que des ports I/O, ces ports dits GPIO peuvent être pilotés via une librairie officielle, par exemple en langage Python. Plus précisément, ces ports peuvent être soit en position INPUT, en entrée, alors ils attendent de recevoir un signal pour le retranscrire du monde "réel" au monde "électronique"; ou en position OUTPUT, sortie, et ils attendent alors une instruction soit, pour émettre un 1, soit un 0 (respectivement 3.3v ou 0v). Cependant, il est également doté de plusieurs types de port, nous ne détaillerons que celui qui nous concerne.



Carte Raspberry Pi



Composants carte Raspberry Pi [3]

On remarquera 7 types de broches différentes :

- Une broche d'alimentation, afin d'alimenter des modules ajoutés au raspberry.
- Une broche de masse, appelée Ground, étant complémentaire à la broche d'alimentation.
- Ports GPIO.

Parmi les ports GPIO, 4 en particulier nous intéressent :

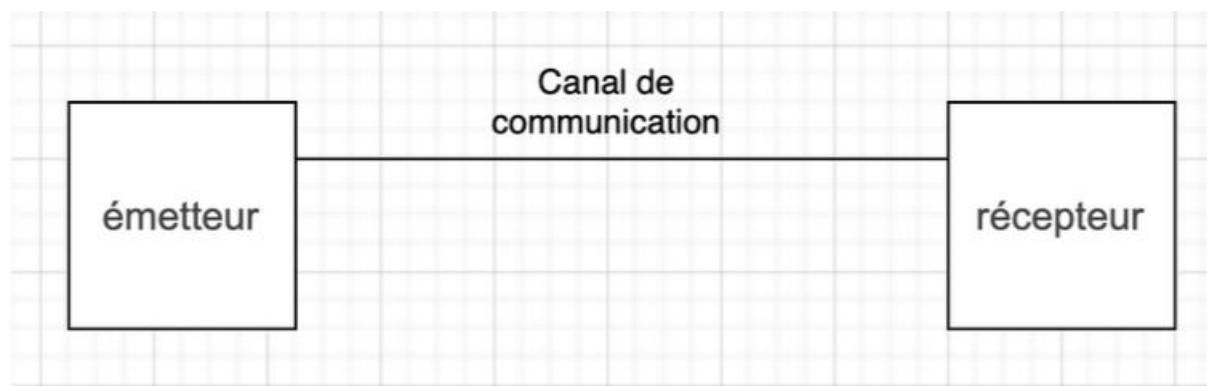
- Ports GPIO SPI, pour Serial Peripheral Interface.
- Ports GPIO PWM, pour Pulse-width modulation, celui-ci est particulièrement intéressant puisqu'il permet d'émettre des signaux analogiques.
- Ports GPIO UART, pour Universal Asynchronous Receiver Transmitter.
- Ports GPIO I2C, pour Inter Integrated Circuit.

Ce dernier va particulièrement nous intéresser. puisqu'il nous permettra de connecter notre capteur, en série, sur un bus. Pour être plus précis, grâce à cela, nous devrions être capable de connecter une multitude de capteurs, sur une seule broche d'information.

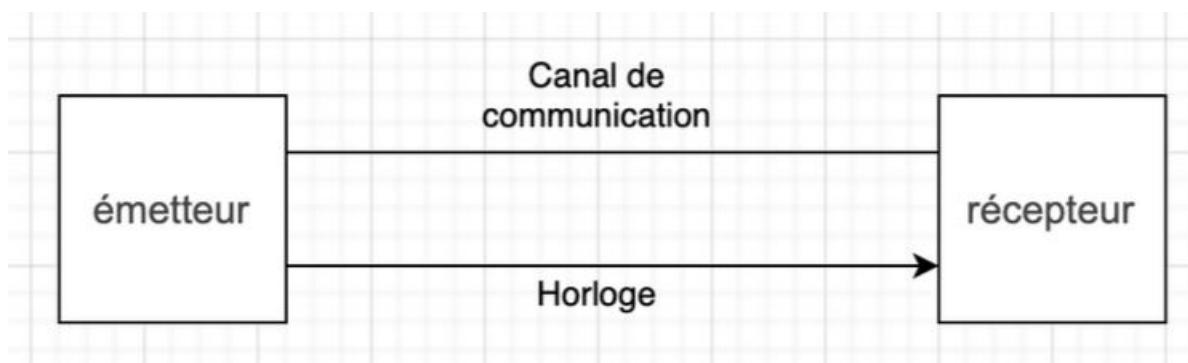
6.3 I2C DEV

I2C, pour IIC, dit Inter Integrated Circuit, est un protocole introduit en 1982 par la société MXP. Il est activement développé par Philips lors de la “Guerre des standards”, dans les années 1990 jusqu’à nos jours. Cette “guerre” avait pour but de standardiser au maximum les différents périphériques. Par exemple, Philips voulait que leurs clients aient la possibilité d’utiliser la même télécommande pour leur tuner ainsi que leur télévision.

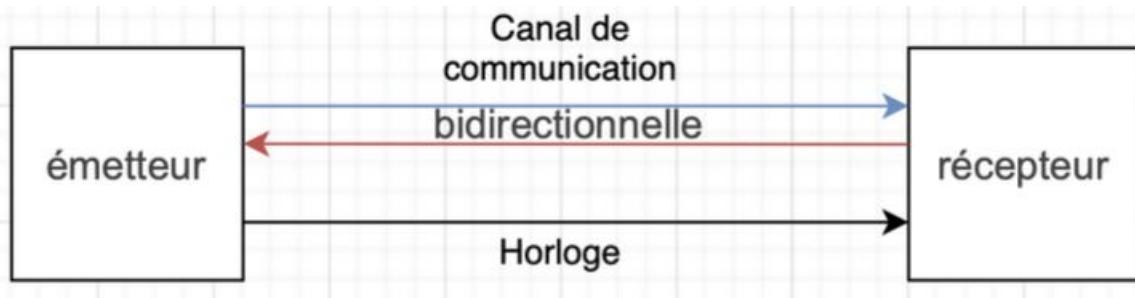
A la fin des années 2000, Arduino et plus tard Raspberry PI, se sont dotés de cette technologie et l’ont intégrée à leurs micro-contrôleurs. Aujourd’hui I2C est utilisé pour la plupart des instruments que nous utilisons tels que les téléphones portables ou encore les oscilloscopes. Mais il sera encore probablement très utile dans le futur puisque les nouvelles technologies telles que la robotique dans la médecine à distance, ou encore les avions de ligne, utilisent I2C. En effet, ce protocole est très réputé pour sa grande fiabilité, sa robustesse, et sa simplicité mais aussi, en grande partie pour son coût de déploiement relativement faible par rapport à son large spectre de possibilités. Le protocole I2C est un protocole dit Serie, synchrone, et half-duplex. Série veut dire que la communication se fait entre un émetteur et un récepteur de manière séquentielle via un canal de communication.



Synchrone car la communication est guidée par une horloge système.



Half-duplex car la communication est faite de manière alternée sur le même canal. Ce canal de communication possède une capacité de “va et vient”, il peut communiquer dans un sens, puis dans l’autre, autrement dit il est bidirectionnel.



Le principe du fonctionnement de I2C repose donc sur deux fils de données : la ligne de transport données, dite SDA pour Serial Data Line, ainsi que la ligne SCL pour Serial Clock Line, c'est la ligne d'horloge de synchronisation. L'I2C possède également une ligne d'alimentation qui est généralement d'un potentiel de 5V, mais également une ligne de masse de 0V, permettant de synchroniser l'état bas des signaux. Cependant une petite subtilité en plus, l'émetteur est dit "maître" et le récepteur est dit "esclave". Ce qui dans la réalité permet d'avoir une multitude de récepteurs pilotés par le même maître et sur la seule et même ligne de communication. Le nombre de périphériques par ligne est donné par le nombre de bits d'adressage du système I2C utilisé. Pour nous, chaque composant est adressé selon une adresse de 7 bits. On peut donc connecter jusqu'à 2^7 , soit 128 appareils au réseau I2C. Sous réserve de ne pas dépasser la charge maximale de la ligne d'alimentation.

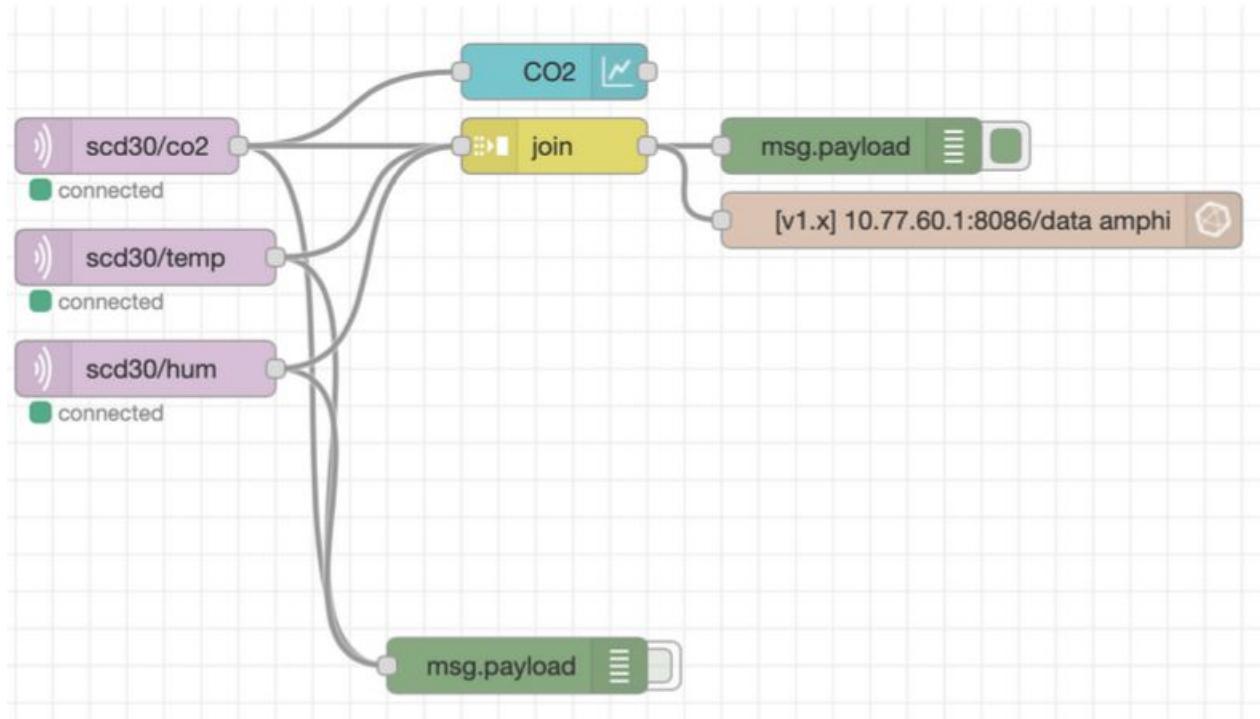
Pour en revenir au raspberry PI, comme dit précédemment, nous avions donc opté pour un raspberry pi pour prendre nos mesures. Il a la capacité de faire fonctionner des applications légères, étant donné qu'il possède assez de mémoire vive (RAM) ainsi qu'un système d'exploitation. Utiliser des applications pour permettre l'analyse et le traitement des données, tel était notre objectif. Afin d'optimiser au mieux notre système nous avons choisi d'utiliser Docker.



[4]

Docker, développé par un franco-américain et rendu open source en 2013, est un moteur de conteneurisation. Il permet le lancement de systèmes d'application en conteneur. Un conteneur est ce qu'on peut qualifier de zone virtuelle sur l'ordinateur, docker applique une configuration à une application puis la lance virtuellement sur le système, ce qui permet aussitôt de la fermer en quelques secondes et d'en laisser aucune trace. Si nous avons choisi cet outil, c'est pour sa haute portabilité, ses performances ainsi que sa fiabilité sur une carte Raspberry Pi.

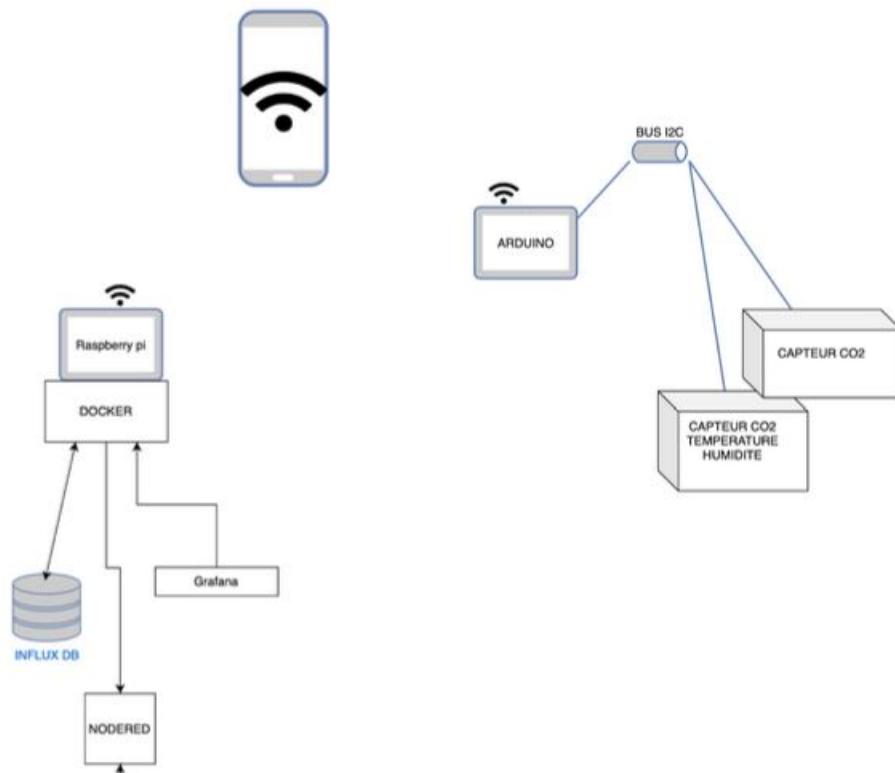
Pour les applications, nous avons opté pour Grafana, logiciel open source de visualisation des données, simple et rapide. Nous avons également utilisé influxDB afin d'enregistrer les données dans une base de données. InfluxDB est particulièrement intéressant pour notre utilisation car les données sont stockées selon une structure temporelle. Nous avons enfin choisi d'utiliser NodeRED, la dernière création d'IBM. Une simplicité qui nous a permis d'obtenir un système extrêmement flexible, notamment dans la possibilité d'ajouter des modules tels que d'autres capteurs. NodeRED, ainsi que notre système par conséquent, fonctionne avec du flow code, piloté par du JavaScript.



Système NODERED

Entre autres, on peut remarquer des noeuds, avec marquer scd30/co2, ou encore scd30/temp. Il s'agit du noeud du capteur de co2. L'arduino publie en fait sur un serveur mosquito dit MQTT. Mosquitto est un broker de messages, fonctionnant sur un principe de file de messages, où le serveur fonctionne sous forme de tunnel plus communément appelé Pipeline. Par exemple, un client s'abonne à un pipeline appelé /scd30/temp et un autre client publie, sur ce même pipeline, le chiffre 30. Le client A étant “abonné” à celui-ci reçoit alors tout message passant dans le pipeline “/scd30/temp”, c'est-à-dire que A reçoit forcément le message s'il est abonné. On l'utilise donc afin de transmettre des messages au travers un réseau wifi que nous aborderons plus bas.

Finalement, nous obtenons ce système :



Architecture de notre système



Montage de notre système

Notre montage est composé d'une boîte de raccordement électrique, avec à l'intérieur un simple câble USB, branché à l'arduino, ainsi que les 4 fils permettant la communication I2C vers le capteur CO₂. Le câble USB sert à la fois d'alimentation, et d'accès à l'interface afin de reprogrammer

l'arduino si besoin, le tout, alimenté grâce à une batterie portable de téléphone. Au branchement du raspberry, docker est automatiquement lancé et reçoit pour ordre de démarrer tous les conteneurs un à un. Une fois tous ses conteneurs lancés, le raspberry lance un réseau wifi. Ce réseau wifi permet la connection entre l'arduino et le raspberry. L'arduino ne commencera la prise de mesure que lorsqu'il sera connecté.



Intérieur de la boîte de raccordement électrique



Montage de notre système en fonctionnement

Afin de vérifier que les mesures sont bien effectuées et que, nous pouvons nous connecter à ce réseau avec n'importe quel appareil électronique tel qu'un téléphone ou un ordinateur. Une fois la connexion établie, nous nous connectons à l'adresse ip du raspberry (la passerelle réseau), suivi d'un port donné en fonction du service auquel nous voulons accéder, 1880 pour nodeRED ou 3000 pour grafana. L'accès à ces deux applications est particulièrement intéressant puisque via nodeRED nous avons accès à un graphique de mesures en continu et instantané. Mais il n'est pas possible de récupérer les mesures passées. Tandis que sur Grafana nous pouvons accéder à toutes nos mesures, enregistrées dans la base de donnée, et ainsi les ressortir sur des graphes tout en pouvant y appliquer des opérations mathématiques pour par exemple, disproportionner le graphe afin d'avoir un meilleur visuel comparatif entre les courbes de grandeur différente (température et taux de CO₂).

7 Mesures et interprétations

Durant notre campagne de mesures, nous avons eu l'occasion de récupérer des données à de multiples endroits extrêmement intéressants. Ces mesures sont parfois assez impressionnantes.

7.1 Amphithéâtre A1 durant un examen

Lors de notre examen d'informatique, nous avons déployé notre système dans l'amphithéâtre A1 du campus Jussieu. Nous l'avons posé sur la table du professeur. D'emblée, nous remarquons un taux de CO₂ parfaitement acceptable avec une première mesure d'environ 800 PPM. On remarque que le taux de CO₂ augmente très légèrement en quelques minutes, cela est probablement dû au mouvement d'air dû à l'entrée des étudiants dans l'amphithéâtre. Seulement, une fois l'examen commencé nous remarquons que le taux de CO₂ ne fait que diminuer. On note que seulement une porte sur les trois était ouverte. En vert ci-dessous, la courbe de taux CO₂ en ppm.



Concentration en CO₂ en fonction du temps en amphi A1

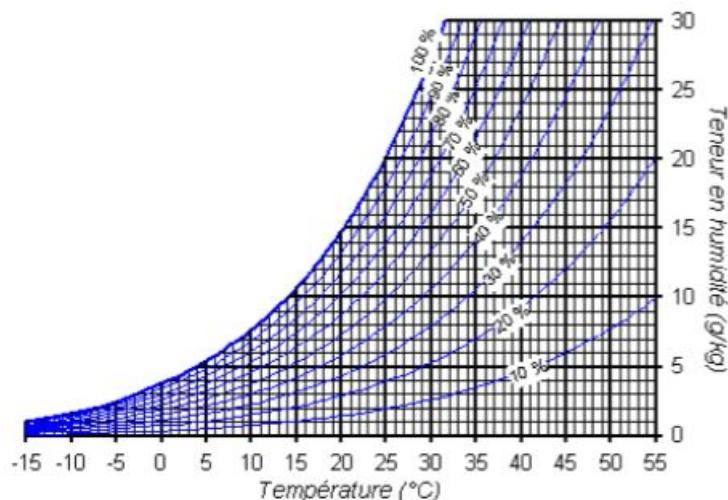
Nos capteurs nous permettent d'obtenir également la température de l'air ainsi qu'à son humidité relative. Nous nous sommes alors demandé si, température, humidité et taux de CO₂ avaient

un lien. Afin de comparer ces courbes, nous avons amplifié les courbes de température pour les rendre comparables à l'échelle du taux de CO₂. On observe ci-dessous, en vert, le taux de CO₂, en rouge, le taux d'humidité relative, et en orange, la température de l'air.



Concentration en CO₂, température et humidité en fonction du temps en amphi A1

Nous remarquons que la température est montée, ce qui confirme bien la présence des étudiants dans l'amphithéâtre. Nous constatons également, que l'humidité mesurée est bien l'humidité relative, dû au fait qu'elle diminue lorsque que la température augmente. Elle suit ainsi le diagramme ci-dessous. Ce diagramme est celui de la teneur d'eau dans l'air en humidité absolue (g/kg) en fonction de la température en degré Celsius (C°). Il donne également le pourcentage de l'humidité dite relative. C'est le rapport de la pression partielle de la vapeur d'eau contenue dans l'air sur la pression de la vapeur saturante d'eau à la même température.

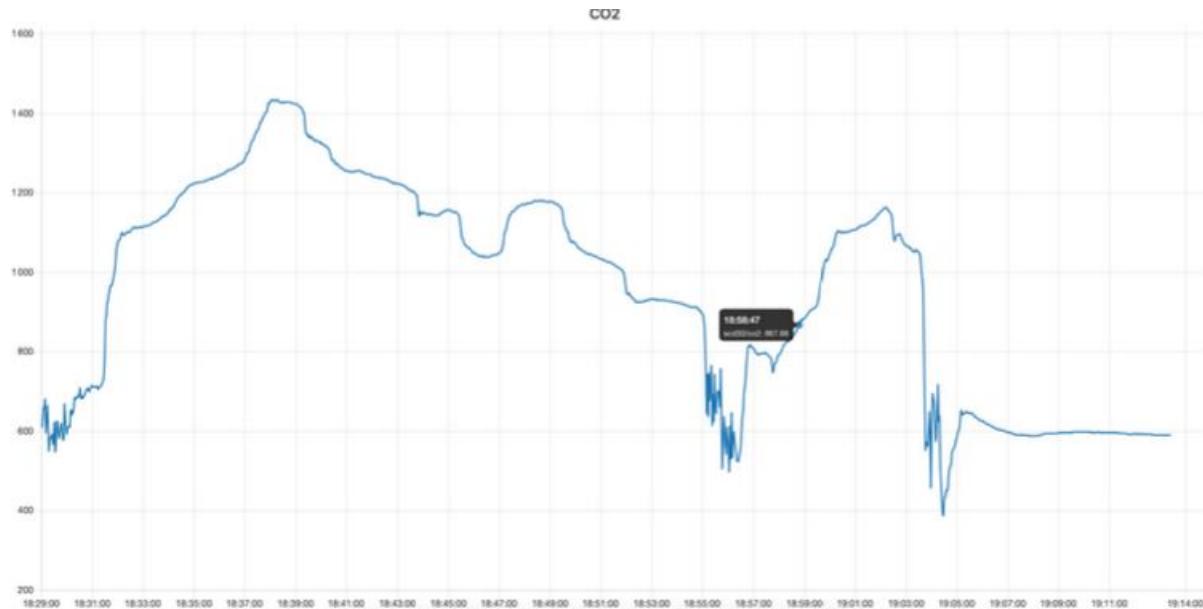


Teneur en humidité en fonction de la température

On remarque alors que la baisse de CO₂ n'est pas due à une chute de la température, elle est donc forcément dû au renouvellement de l'air, par de l'air frais à faible taux de CO₂ (400 ppm).

7.2 Transports en commun

Nous avons décidé d'embarquer notre système dans le RER B afin de vérifier si les normes "covid" du gouvernement sont respectées durant le trajet en transport en commun des voyageurs. Les mesures ont été prises dans le RER B, les fenêtres ouvertes. L'échelle des ordonnées correspond au taux de CO₂ en PPM. L'échelle des abscisses n'est autre que l'heure des mesures.



Concentration en CO₂ en fonction du temps sur un trajet en transports en commun

Le plan du trajet est le suivant :

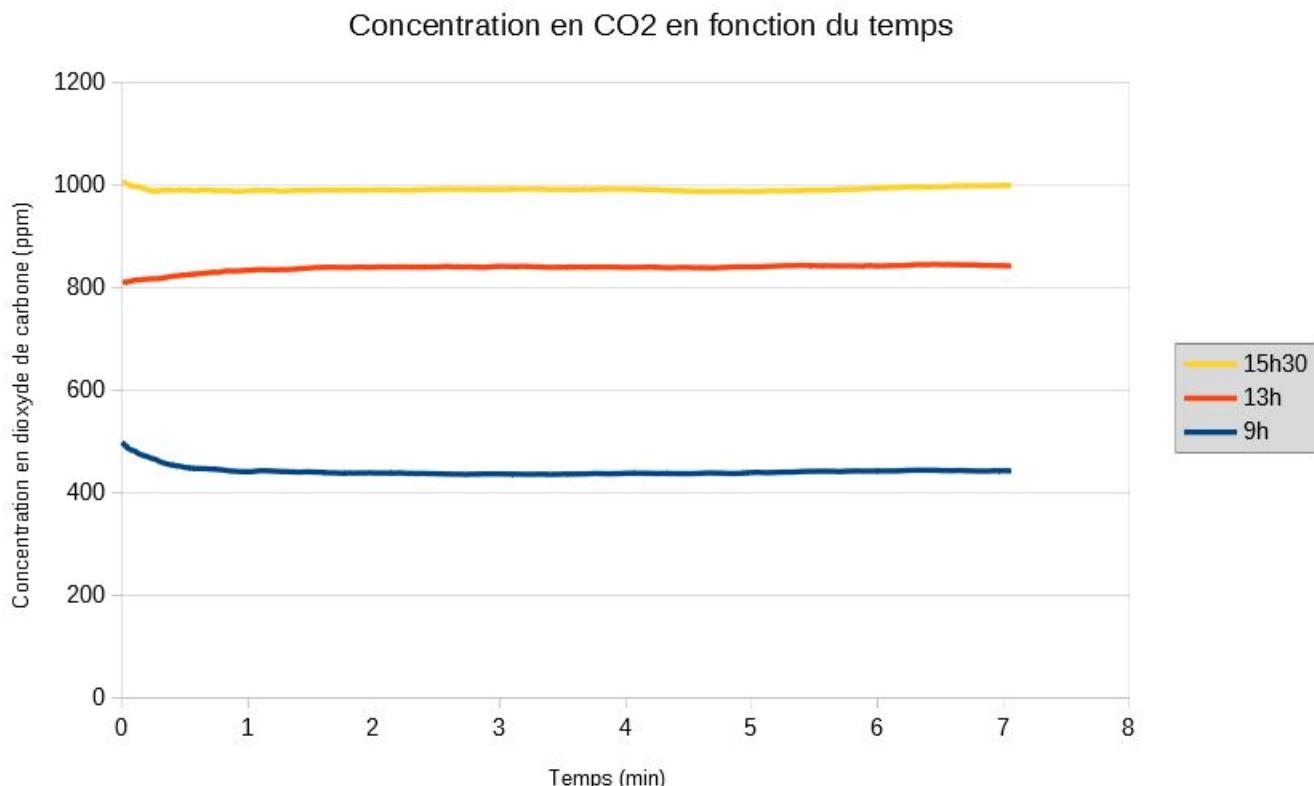
- à 18h31, nous sommes entrés dans le RER B à la gare Saint-Michel-Notre-Dame. Nous avons dès notre entrée, pris la liberté d'ouvrir les fenêtres à côté de nous dans notre wagon. Beaucoup de passagers étaient présents dans la rame, à titre indicatif, nous avions un espace d'environ 50 cm entre chaque personne.
- à 18h33, nous arrivons à Châtelet-les-halles. Nous remarquons que lorsque les portes se sont ouvertes, le taux de CO₂ s'est brutalement mis à monter. Nous pensons que cela provient du mouvement des personnes dans le wagon, ou peut-être du fait que la gare de Châtelet est à quelques dizaines de mètres en dessous du sol parisien, et que le CO₂, étant plus lourd que l'air tendance à être présent dans de plus grande quantité en sous-sol.
- à 18h37, nous arrivons à la gare du Nord. Nous remarquons que le taux de CO₂ poursuit sa montée.
- à 18h39, le point maximum enregistré est à 1430 PPM. Ce point correspond très exactement à la sortie du RER B du tunnel de gare du Nord.
- Jusque 18h54, le taux de CO₂ poursuit alors sa baisse au fur et à mesure que les voyageurs quittent le train.
- à 18h54 nous sortons à la gare d'Aulnay-sous-Bois afin de prendre notre correspondance. En extérieur nous relevons un taux de CO₂ d'environ 450 PPM.

- On remarque que le taux de CO₂ remonte à partir de 18h56, cela correspond à notre entrée dans le tram T4 reliant Aulnay-sous-bois à Bondy.
- Durant les 6 premières minutes correspondant à 3 arrêts, le taux de CO₂ poursuit une montée, jusqu'à notre sortie de celui-ci.
- Nous entrons ensuite dans une maison, puis nous posons le capteur sur le sol. Nous remarquons que le taux de CO₂ atteint une constante au bout de quelque minute.

On constate alors que la concentration de CO₂ dans les tunnels du RER est très élevée par rapport à l'air extérieur. Cette qualité de l'air est considérée comme mauvaise par le Haut Conseil de la Santé Publique (HCSP). Afin d'apporter une solution viable pour les voyageurs quotidiens, il faudrait repenser le système de renouvellement d'air des tunnels du RER parisien.

7.3 Bibliothèque des L1

Au cours d'une journée particulièrement chargée pour les bibliothèques de l'université, nous nous sommes rendu compte que les bibliothèques, et en particulier celle des L1, étaient très peu aérées malgré l'affluence importante d'étudiants à l'approche des examens. Nous avons alors placé un capteur de CO₂ dans la "Zone calme" de la bibliothèque des L1 et relevé les mesures à trois moments de la journée : le matin à l'ouverture, en début d'après-midi et en milieu d'après-midi. Nous estimons le taux d'occupation à l'ouverture de la bibliothèque à 10% de sa capacité d'accueil (540 places), soit environ 50 étudiants. A partir du milieu de matinée, il est déjà difficile de trouver une place libre. On estime alors le taux d'occupation à 75% jusqu'en fin d'après-midi, soit environ 400 étudiants.



Concentration en CO₂ en fonction du temps dans la bibliothèque des L1

On constate qu'à 9h, à l'ouverture de la bibliothèque, la concentration en CO₂ est d'environ 450 ppm. Ce qui est assez faible. La qualité de l'air est donc bonne à l'arrivée des premiers étudiants. Toutefois, après 4 heures, avec un taux d'occupation qui ne cesse de croître pour atteindre 75%, on remarque que la concentration en CO₂ a quasiment doublé, atteignant 830 ppm. La qualité de l'air dans la bibliothèque s'est détériorée et nécessite une aération. Enfin, en milieu d'après-midi, la concentration en CO₂ atteint les 1000 ppm. Cette mesure témoigne d'un non renouvellement de l'air au cours de la journée dans la bibliothèque des L1, chose pourtant essentielle pour garantir une meilleure capacité de concentration des étudiants.

8 Conclusion

Notre projet sur l'étude des particules nous a permis de comprendre leur comportement, et en particulier comment elles peuvent rester en suspension pendant plusieurs heures. Nous avons pu observer grâce à une modélisation que, lors de sa chute, une particule rencontre une force de frottement s'opposant au poids et déterminée par la loi de Stokes. Nous avons alors tenté de la valider en faisant tomber des billes différentes dans de la glycérine, puis en traçant la courbe de la vitesse expérimentale en fonction de la vitesse théorique. Ensuite, en introduisant le nombre de Stokes, nous avons pu constater que la petite taille d'une particule, ainsi que sa faible masse volumique devant celle de l'air, lui permet de suivre l'écoulement du fluide dans lequel elle se trouve. En particulier, la création d'un courant d'air dans une pièce permet d'évacuer des particules de virus vers l'extérieur par un renouvellement de l'air.

Pour avoir une idée de la qualité de l'air d'un espace fermé, nous avons utilisé des capteurs de CO₂ connectés à une carte Arduino ou Raspberry Pi. En effet, si la mesure de la concentration en CO₂ d'une pièce est élevée, au-delà de 800 ppm environ, cela signifie que l'aération n'est pas suffisante. Notre campagne de mesures dans les transports en commun et au sein de l'université nous a permis d'évaluer la qualité de l'air de différents espaces de notre quotidien. Nous avons remarqué que les espaces très fréquentés avec peu de fenêtres sur l'extérieur (Bibliothèque des L1), ou sans accès direct avec l'extérieur (RER B), ne bénéficient pas de système d'aération suffisamment puissant. Dans le contexte actuel de pandémie, la mesure de la concentration en CO₂ est une méthode de plus en plus utilisée dans certains lieux publics pour garantir une bonne aération.

Ce projet nous a permis de développer différentes compétences scientifiques et personnelles. Les phases de recherches sur les phénomènes physiques et l'utilisation d'un matériel nouveau, qu'il a fallu apprendre progressivement à maîtriser de notre côté, nous a demandé une certaine autonomie. Par ailleurs, la rédaction de ce rapport a nécessité une capacité d'organisation dans la répartition du travail et de communication pour l'échange des idées de chacun. Sur le plan scientifique, nous avons acquis des compétences informatiques en reprenant nos connaissances du langage C pour programmer sur Arduino. Enfin nous avons développé notre capacité de traitement et d'analyse des données en récupérant les mesures des capteurs pour les interpréter.

A Annexe : Programme Arduino

Programme Arduino utilisé pour relever les mesures faites par le capteur

```
#include "SCD30.h"
#include <SPI.h>
#include <SD.h>

File myFile;

#if defined(ARDUINO_ARCH_AVR)
    #pragma message("Defined architecture for ARDUINO_ARCH_AVR.")
    #define SERIAL Serial
#elif defined(ARDUINO_ARCH_SAM)
    #pragma message("Defined architecture for ARDUINO_ARCH_SAM.")
    #define SERIAL SerialUSB
#elif defined(ARDUINO_ARCH_SAMD)
    #pragma message("Defined architecture for ARDUINO_ARCH_SAMD.")
    #define SERIAL SerialUSB
#elif defined(ARDUINO_ARCH_STM32F4)
    #pragma message("Defined architecture for ARDUINO_ARCH_STM32F4.")
    #define SERIAL SerialUSB
#else
    #pragma message("Not found any architecture.")
    #define SERIAL Serial
#endif

void setup() {

    // Open serial communications and wait for port to open:
    Serial.begin(9600);
    while (!Serial) {
        ; // wait for serial port to connect. Needed for native USB port only
    }
    Serial.print("Initializing SD card...");|
    if (!SD.begin(4)) {
        Serial.println("initialization failed!");
        while (1);
    }
}
```

```

Serial.println("initialization done.");

// open the file. note that only one file can be open at a time,
// so you have to close this one before opening another.
myFile = SD.open("test.txt", FILE_WRITE);

// if the file opened okay, write to it:
if (myFile) {
    Serial.print("Writing to test.txt...");
    myFile.println("testing 1, 2, 3.");
    // close the file:
    myFile.close();
    Serial.println("done.");
} else {
    // if the file didn't open, print an error:
    Serial.println("error opening test.txt");
}

// re-open the file for reading:
myFile = SD.open("test.txt");
if (myFile) {
    Serial.println("test.txt:");

    // read from the file until there's nothing else in it:
    while (myFile.available()) {
        Serial.write(myFile.read());
    }
    // close the file:
    myFile.close();
} else {
    // if the file didn't open, print an error:
    Serial.println("error opening test.txt");
}

```

```

Wire.begin();
SERIAL.begin(115200);
SERIAL.println("SCD30 Raw Data");
scd30.initialize();
}

void loop() {
    float result[3] = {0};

    if (scd30.isAvailable()) {
        scd30.getCarbonDioxideConcentration(result);

        myFile = SD.open("mesures.txt", FILE_WRITE);
        if (myFile) {
            Serial.print("Writing to mesures.txt...");
            Serial.println();

            myFile.print("Carbon Dioxide Concentration (ppm)");
            myFile.print(result[0]);
            myFile.print("\t");
            myFile.print("Temperature (°C)");
            myFile.print(result[1]);
            myFile.print("\t");
            myFile.print("Humidity (%) ");
            myFile.print(result[2]);
            myFile.println();

            myFile.close();
            Serial.println("done.");
        } else {
            Serial.println("error opening test.txt");
        }
    }
    delay(2000);
}

```

B Références

- [1] Arduino. Pinout-Leonardo latest. (17 juin 2020) [schéma] In : Arduino : ARDUINO LEO-NARDO WITH HEADERS . Disponible sur :
<https://store.arduino.cc/arduino-leonardo-with-headers>
- [2] SeedStudio. Grove - CO2 & Temperature & Humidity Sensor (SCD30) [En ligne]. (modifié le 25 septembre 2020) Disponible sur :
<https://wiki.seeedstudio.com/Grove-CO2TemperatureHumiditySensor-SCD30/>
- [3] Framboise314. Schéma du GPIO du Raspberry Pi (sans date) [shéma] In : GPIO - Framboise 314, le Raspberry Pi à la sauce française... Disponible sur :
<https://www.framboise314.fr/scratch-raspberry-pi-composants/gpio/>
- [4] SQX&BKI. DOCKER (12 novembre 2019) [Logo] In : Comment installer Docker sur Debian 10. Disponible sur :
<https://sqx-bki.fr/comment-installer-docker-sur-debian-10/>

C Bibliographie

Pour l'ensemble de nos recherches, nous avons eu très peu accès aux ouvrages des bibliothèques universitaires du fait de la situation sanitaire actuelle qui a entraîné la fermeture de certaines d'entre elles, notamment celle de la bibliothèque de géosciences de Sorbonne Université. Nous avons donc fait l'ensemble de notre rapport sur la base des explications apportées par notre encadrant, ainsi que les documents qu'il nous a partagés. Nous avons ensuite complété par des recherches menées sur les sites officiels des constructeurs du matériel électronique que nous avons utilisé. Pour toutes les explications sur les lois physiques, nous nous sommes principalement appuyés sur le site wikipédia qui permet un accès rapide et clair aux formules et définitions principales. Quant aux informations plus précises portant sur les particules de Covid en particuliers et à l'importance de l'aération, nous nous sommes référés au site du Haut Conseil de la Santé Publique.

SeedStudio. Grove - CO2 & Temperature & Humidity Sensor (SCD30) [En ligne]. (modifié le 25 septembre 2020) Disponible sur :

<https://wiki.seeedstudio.com/Grove-CO2TemperatureHumiditySensor-SCD30/>
(consulté le 18 mai 2021)

SeedStudio. SD Card Shield V4 [En ligne]. (sans date) Disponible sur :
<https://www.seeedstudio.com/SD-Card-Shield-V4-p-1381.html>
(consulté le 18 mai 2021)

SeedStudio. Arduino Leonardo with Headers [En ligne]. (sans date) Disponible sur :
<https://store.arduino.cc/arduino-leonardo-with-headers>
(consulté le 18 mai 2021)

Velleman group. CARTE DE DEVELOPPEMENT ATmega32u4 LEONARDO [En ligne]. (sans date) Disponible sur :
<https://www.velleman.eu/products/view/?id=460480>
(consulté le 16 mai 2021)

Wikipédia. Raspberry Pi [En ligne]. (modifié le 16 mai 2021) Disponible sur :
https://fr.wikipedia.org/wiki/Raspberry_Pi
(consulté le 11 mai 2021)

Olivier. GPIO - Entrée en matière [En ligne]. (modifié en septembre 2013) Disponible sur :
<https://www.magdiblog.fr/gpio/gpio-entree-en-matiere/>
(consulté le 11 mai 2021)

Wikipédia. Docker(logiciel) [En ligne]. (modifié le 13 avril 2021) Disponible sur :
[https://fr.wikipedia.org/wiki/Docker_\(logiciel\)](https://fr.wikipedia.org/wiki/Docker_(logiciel))
(consulté le 12 mai 2021)

Wikipédia. Grafana [En ligne]. (modifié le 6 mai 2021) Disponible sur :
<https://fr.wikipedia.org/wiki/Grafana>
(consulté le 12 mai 2021)

Wikipédia. InfluxDB [En ligne]. (modifié le 11 janvier 2021) Disponible sur :
<https://fr.wikipedia.org/wiki/InfluxDB>
(consulté le 12 mai 2021)

Wikipédia. Node-RED [En ligne], (modifié le 29 avril 2021) Disponible sur :
<https://en.wikipedia.org/wiki/Node-RED>
(consulté le 12 mai 2021)

Wikipédia. MQTT [En ligne]. (modifié le 11 janvier 2021) Disponible sur :
<https://fr.wikipedia.org/wiki/MQTT>
(consulté le 12 mai 2021)

Pr. GAGA Ahmed. Le bus I2C, Partie 1 :"INTRODUCTION".[Ajouté le 23 novembre 2019]
In : Youtube. Disponible sur :
<https://www.youtube.com/watch?v=tICJHYeqbAY>
(consulté le 18 mai 2021)

Pr. GAGA Ahmed. Le bus I2C, Partie 3 :"Application, Communication avec l'Horloge Temps Réel DS3231".[Ajouté le 21 janvier 2020] In : Youtube. Disponible sur :
<https://www.youtube.com/watch?v=TvTJuHX7cmk>

Wikipédia. Humidité relative [En ligne]. (modifié le 24 mars 2020) Disponible sur :
<https://fr.wikipedia.org/wiki/Humidit%e9relative>
(consulté le 18 mai 2021)

Wikipédia. I2C [En ligne]. (modifié le 12 mai 2020) Disponible sur :
<https://fr.wikipedia.org/wiki/I2C>
(consulté le 18 mai 2021)

Wikipedia. Poussée d'Archimède [En ligne]. (modifié le 7 mai 2021) Disponible sur :
https://fr.wikipedia.org/wiki/Poussee_d%27Archimde
(consulté le 18 mai 2021)

Wikipédia. Frottement [En ligne]. (modifié le 10 avril 2021) Disponible sur :
<https://fr.wikipedia.org/wiki/Frottement>
(consulté le 18 mai 2021)

Wikipédia. Nombre de Reynolds [En ligne]. (modifié le 10 mai 2021) Disponible sur :
https://fr.wikipedia.org/wiki/Nombre_de_Reynolds#text=Le%20nombre%20de%20Reynolds,laminaire%2C%20transitoire%2C%20turbulent
(consulté le 18 mai 2021)

Wikipédia. Loi de Stokes [En ligne]. (modifié le 11 décembre 2020) Disponible sur :
https://fr.wikipedia.org/wiki/Loi_de_Stokes
(consulté le 18 mai 2021)

Wikipédia. Nombre de Stokes [En ligne]. (modifié le 3 septembre 2020) Disponible sur :

https://fr.wikipedia.org/wiki/Nombre_d'Estokes

(consulté le 18 mai 2021)

Wikipédia. Particule en suspension [En ligne]. (modifié le 26 avril 2021) Disponible sur :

https://fr.wikipedia.org/wiki/Particules_en_suspension

(consulté le 18 mai 2021)

Youmatter. Particule fine : définition et impact sur la santé, source de pollution, mesures de lutte et protection [En ligne]. (modifié le 3 octobre 2019) Disponible sur :

<https://youmatter.world/fr/definition/particules-fines-definition-impact-sante/>

(consulté le 18 mai 2021)

HCSP. COVID-19 : aération, ventilation et mesure du CO2 dans les ERP [En ligne]. (modifié le 3 mai 2021) Disponible sur :

<https://www.hcsp.fr/explore.cgi/avisrapportsdomaine?clefr=1009>

(consulté le 18 mai 2021)

Wikipédia. Écoulement laminaire [En ligne]. (modifié le 10 octobre 2020) Disponible sur :

https://fr.wikipedia.org/wiki/écoulement_laminaire

(consulté le 18 mai 2021)

Wikipédia. Turbulence [En ligne]. (modifié le 20 janvier 2020) Disponible sur :

<https://fr.wikipedia.org/wiki/Turbulence>

(consulté le 18 mai 2021)