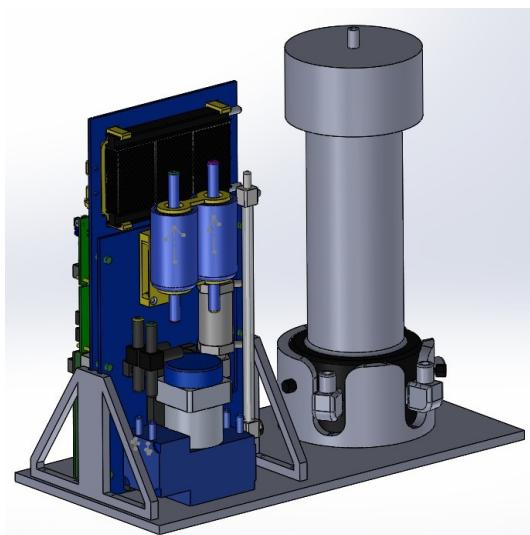
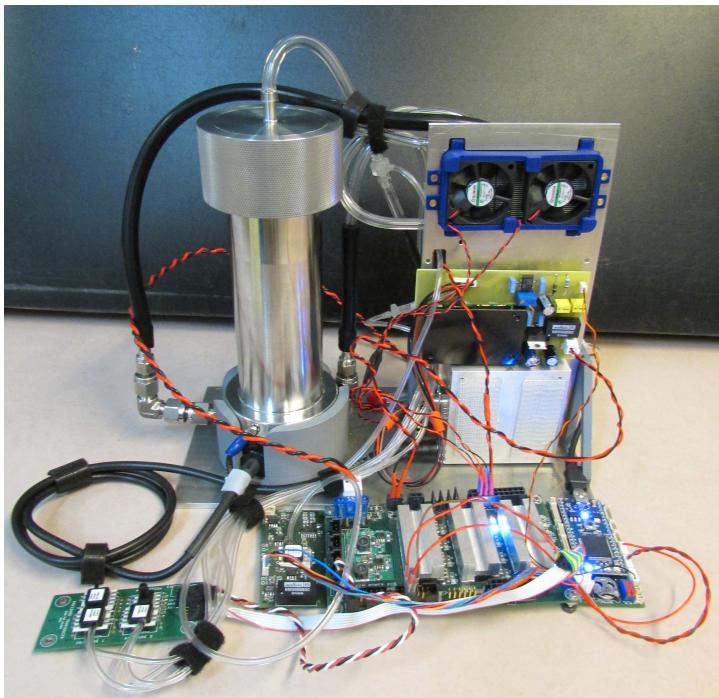


Stage effectué au Centre National de Recherche Météorologique – Météo France

Miniaturisation d'un Scanning Mobility Particle Sizer (SMPS)



DATO BRUNO

Stagiaire, Licence Parcours Spéciaux Physique

Encadrant : Dr Greg ROBERTS

Stage réalisé du 10 février 2014 au 24 avril 2014

Remerciements

Je souhaite tout d'abord remercier mon encadrant de stage Greg ROBERTS pour m'avoir proposé ce stage et m'avoir guidé tout au long de son déroulement.

Je tiens aussi à remercier toute l'équipe MNPCA qui a su se montrer disponible chaque jour, notamment Thierry BOURRIANNE qui a toujours été là pour répondre à mes questions de météorologie comme de programmation et qui m'a dirigé vers ce stage. Je pense aussi à Jean-Michel ETCHEBERRY qui m'a beaucoup aidé pour le côté électronique, Brigitte Gaillard qui m'a initié au fonctionnement du SMPS et Géraud MONBOISSE qui a toujours été présent pour me conseiller. Mais aussi José NICOLAS, Marie MAZOYER et Marine CLAEYS qui étaient là pour répondre à mes interrogations, Laurent GUSTAVE et Sébastien BARRAU pour le côté administratif de mon stage et Frédéric BURNET et Grégoire CAYES pour m'avoir permis de participer aux activités pratiques du lundi midi.

Je voudrais aussi remercier Alain FRAPPIER et Denis SUBRA du LEM (Laboratoire d'électronique et de micro-processeurs) avec qui j'ai travaillé pendant mon stage.

Enfin, je souhaite remercier l'Université Paul Sabatier ainsi que le corps enseignant des Parcours Spéciaux qui nous ont donné la chance d'effectuer ce stage et de découvrir le monde de la recherche, je pense notamment à Jesse GROENEN, Xavier BRESSAUD et Florence PETTINARI-STURMEL.

Sommaire

Remerciements	p. 1
Table des figures	p. 3
Introduction	p. 4
1. Présentation du laboratoire de recherche	p. 5
1.1. Centre National de Recherches Météorologiques - Groupe d'études de l'Atmosphère Météorologique (CNRM-GAME)	
1.2. Micropysique des Nuages et Physico-Chimie de l'Atmosphère (MNPCA)	
2. Aérosols	p. 6
3. SMPS (Scanning Mobility Particle Sizer)	p. 7
3.1. Fonctionnement général	
3.2. Differential Mobility Analyzer (DMA)	
3.3. Flux	
3.3.1. Sheath air	
3.3.2. Aérosols	
3.4. Représentations graphiques	
3.5. Colonne radiale	
4. Processus de miniaturisation	p. 10
4.1. Conception assistée par ordinateur (CAO)	
4.2. Fonctionnement électronique	
4.2.1. Carte d'acquisition	
4.2.2. Cartes électroniques analogiques	
4.2.3. Ports série	
4.2.4. Schéma de l'instrument	
4.3. Programmation du mbed	
4.3.1. Contrôle de la Haute tension	
4.3.2. Contrôle de la pompe	
4.3.3. Programme de contrôle général	
4.4. Montage de l'instrument	
4.5. Tests	
Conclusion	p. 18
Bibliographie	p. 19
Annexes	p. 20

Table des figures

- Figure 1** : Schéma du Scanning Mobility Particle Sizer (SMPS) p. 7
- Figure 2** : Schéma du Differential Mobility Analyzer (DMA) p. 8
- Figure 3** : Graphe représentant les diamètres des particules en nanomètres que l'on peut sélectionner avec un SMPS en fonction de la tension en volts appliquée à sa colonne pour un débit de 3 lpm p. 9
- Figure 4** : Graphe représentant les diamètres des particules en nanomètres que l'on peut sélectionner avec un SMPS en fonction de la tension en volts appliquée à sa colonne pour un débit de 10 lpm p. 9
- Figure 5** : Coupe de la colonne radiale d'un classificateur électrostatique p. 9
- Figure 6** : Graphe représentant le débit de sheath flow (en lpm) par rapport aux dimensions d'une colonne radiale possédant sensiblement les mêmes capacités de sélection que la colonne axiale du SMPS p. 10
- Figure 7** : Capture d'écran d'une partie du prototype de mini-SMPS étudié pour être embarqué dans un drone p. 11
- Figure 8** : Capture d'écran d'une partie du prototype de mini-SMPS étudié pour faire des tests en laboratoire p. 11
- Figure 9** : Photo du micro-contrôleur Mbed LPC1768 p. 11
- Figure 10** : Schéma d'un capteur de pression absolue p. 12
- Figure 11** : Schéma d'un capteur de pression différentielle p. 12
- Figure 12** : Schéma d'une trame de caractères transmis par un port série p. 13
- Figure 13** : Schéma synoptique du mini-SMPS p. 14
- Figure 14** : Schéma du montage effectué pour tester le fonctionnement du mini-SMPS p. 17

Introduction

Dans le cadre de notre troisième année de Licence de Physique à l'Université Paul Sabatier dans la filière Parcours Spéciaux, nous avons la chance de faire un stage de trois mois dans un laboratoire de recherche. Ce stage a pour but de nous familiariser au monde de la recherche tout en nous faisant appliquer les connaissances que nous avons acquises pendant nos trois années de Licence.

J'ai choisi de faire mon stage au CNRM par intérêt pour la météo et afin d'en apprendre un peu plus sur le déroulement des recherches dans ce domaine. J'ai plus particulièrement choisi un service instrumental car si je poursuit ma carrière dans la recherche, je préférerais qu'il y ait un côté pratique dans mon travail.

Mon stage consistait à miniaturiser l'instrument de mesure SMPS en me servant de travaux qui avaient déjà été faits auparavant. Le SMPS étant un instrument assez encombrant, on ne peut l'utiliser qu'à l'aide d'un avion pour faire des mesures dans l'atmosphère, ce qui rend les campagnes de mesures plus difficiles à entreprendre pour des raisons de coûts, d'autorisations et de sécurité si l'on vole dans la basse atmosphère. La miniaturisation aurait donc pour but d'installer l'instrument de mesure dans un drone qui serait alors transportable dans presque toutes les régions du globe et permettrait de faire des mesures plus aisément à basse altitude.

Mon travail de miniaturisation s'est déroulé suivant diverses étapes qui seront décrites dans mon rapport (conception, fabrication, montage, tests) et qui font notamment appel à des notions de dessin sur ordinateur et de programmation.

Je commencerai tout d'abord par présenter le Centre National de Recherches Météorologiques et le groupe de recherche dans lequel j'ai travaillé. Puis, je définirai brièvement ce que sont les aérosols car c'est essentiellement le sujet d'étude de l'équipe dans laquelle je me trouvais. Ensuite, je décrirai le principe de fonctionnement d'un SMPS et je finirai par le processus de miniaturisation.

1. Présentation du laboratoire de recherche

1.1. Centre National de Recherches Météorologiques - Groupe d'études de l'Atmosphère Météorologique (CNRS-GAME)

Le CRNM-GAME est constitué de chercheurs CNRS et Météo-France, il regroupe les recherches effectuées à Grenoble et à Toulouse et il travaille en collaboration avec les laboratoires de l'Observatoire Midi-Pyrénées (OMP).

Les projets de recherche du CNRM-GAME portent sur la prévisibilité des phénomènes atmosphériques, l'étude du climat et du changement climatique, le cycle de l'eau, l'étude des échanges océan-atmosphère, la physico-chimie atmosphérique et météorologie urbaine, l'assimilation et la modélisation pour la prévision numérique du temps, les développements instrumentaux et la micro-structure du manteau neigeux.

Il regroupe huit unités de recherche :

- GMAP : Groupe de Modélisation et d'Assimilation pour la Prévision, il est chargé de développer de nouveaux modèles numériques de prévision météorologique.
- GMME : Groupe de Météorologie de Moyenne Échelle, il étudie par voie expérimentale et de simulation numérique la dynamique de phénomènes de petite et moyenne échelle pour pouvoir mieux les paramétriser et les inclure dans les modèles de prévision.
- GMGEC : Groupe de Météorologie de Grande Échelle et Climat, il étudie l'évolution du climat de façon globale (évolution ozone, prévision saisonnière, interactions air/mer ...).
- GMEI : Groupe de Météorologie Expérimentale et Instrumentale, il développe et entretient les systèmes d'instrumentation notamment un réseau moyenne échelle de stations météorologiques automatiques. De plus, il organise des campagnes de mesures pour une large communauté scientifique (les unités de recherche en modélisation du CNRM, la communauté nationale en sciences de l'atmosphère, et des consortium internationaux dans un cadre spécifiquement européen) et il traite et valide les données collectées.
- CAM : Centre d'Aviation Météorologique, ce centre est chargé de l'organisation des campagnes de mesure en vol à l'aide de trois avions instrumentés (ATR 42, Falcon 20 et Piper Aztec).
- CMM : Centre de Météorologie Marine, il s'occupe de la mise en œuvre des bouées océaniques et météorologiques pour l'étude des interactions air/mer.
- CEN : Centre d'Études de la Neige, il étudie le manteau neigeux (propriétés, interactions neige/atmosphère, météorologie de montagnes, risques d'avalanches).
- RETIC : Réseau d'Études et de Transfert Interne des Connaissances, il permet le transfert de connaissances des équipes de recherche du CNRM vers les services opérationnels de Météo-France.

1.2. Microphysique des Nuages et Physico-Chimie de l'Atmosphère (MNPCA)

L'équipe MNPCA est l'une des équipes du GMEI, elle s'occupe de l'étude expérimentale de la microphysique nuageuse et de l'aérosol, plus particulièrement de la capacité des aérosols à former des gouttelettes. On parle d'hygroscopicité des aérosols, c'est à dire la capacité des aérosols à devenir des noyaux de condensation nuageuse (Cloud Condensation Nuclei : CCN). L'équipe s'intéresse aussi à la possibilité que ces gouttelettes grossissent et forment du brouillard ou alors forment un nuage s'accompagnant peut être de précipitations.

Elle possède un dispositif expérimental sophistiqué (granulomètres : mesure de la taille et de la concentration en particules, compteur CCN, veine de prélèvement des aérosols en vol...) .

Cette équipe participe à de nombreux projets dont :

-VOLTIGE (Vecteurs d'Observation de La Troposphère pour l'Investigation et la Gestion de l'Environnement) : un projet de développement de drones ultra-légers instrumentés pour la recherche météorologique. C'est une collaboration entre le CNRM, l'ENAC (École Nationale de l'Aviation Civile) et l'ENM (École Nationale de la Météorologie).

-CHARMEX (Chemistry-Aerosol Mediterranean Experiment) : un projet qui vise à évaluer l'état et l'évolution de l'atmosphère méditerranéenne et son impact sur le bassin méditerranéen.

-PARISFOG : Le brouillard a un impact important sur notre vie quotidienne (transports aériens, maritimes et terrestre) et est étroitement relié à la qualité de l'air (la pollution a une influence sur certains brouillards). Un des objectifs de ce projet est d'étudier les processus microphysiques dans les brouillards et leur influence sur le cycle de vie de ces derniers, afin d'en savoir plus sur l'impact de la distribution de particules d'aérosols sur la distribution de gouttelettes constituant le brouillard.

2. Aérosols

L'atmosphère terrestre est composée de beaucoup d'éléments chimiques. Majoritairement, on y trouve de l'azote (78%) et du dioxygène (21%), ces quantités étant calculées par rapport à l'air sec. On trouve donc aussi beaucoup de vapeur d'eau (0,33 % en moyenne) ce qui en fait le troisième constituant majeur de l'air atmosphérique. Les gaz rares (argon, néon, hélium), étant principalement des résidus de l'atmosphère originelle, n'interviennent pas dans le cycle de l'atmosphère. Le dioxyde de carbone, lui, joue un rôle non négligeable car il intervient dans le phénomène de photosynthèse et dans le réchauffement climatique comme gaz à effet de serre.

Ensuite, on y trouve aussi les constituants mineurs dont la concentration varie de quelques ppm (10^{-6} mole/mole d'air) à quelques dizaines de ppt (10^{-12} mole/mole d'air). Ces concentrations sont infimes comparées aux autres constituants mais leur rôle dans l'environnement terrestre est important :

- Bilan radiatif terrestre (gaz à effet de serre, aérosols) : C'est la différence entre le rayonnement reçu par la terre et le rayonnement émis.
- Qualité chimique de l'air : les constituants mineurs ont un impact important sur la qualité de l'air atmosphérique et des précipitations.
- En trop grande quantité, ils deviennent toxiques pour les êtres vivants.

On définit alors les aérosols comme un ensemble de particules, solides ou liquides, d'une substance chimique donnée en suspension dans un milieu gazeux. On peut aussi voir les aérosols comme un groupement stable de molécules dans un gaz.

Dans la communauté scientifique, par abus de langage, on nomme souvent aérosols les particules en suspension. C'est pourquoi, dans la suite de ce rapport, on utilisera aussi le terme d'aérosol lorsque l'on parlera de ces particules bien que ce ne soit pas la définition exacte. La taille des aérosols varie entre une dizaine d'Angströms et une centaine de micro-mètres. On distingue deux types d'aérosols :

- Les aérosols primaires : ils sont émis dans l'atmosphère sous forme solide ou liquide de façon naturelle comme les poussières minérales désertiques (érosion des sols), les sels de mer (embruns marins) et les cendres volcaniques. Mais aussi par la main de l'homme comme la suie (combustion thermique), le ciment (production industrielle) et les particules issues de l'usure des pneumatiques.
- Les aérosols secondaires: ils sont créés par des processus de conversion gaz-particule, condensation et coagulation. Une bonne partie des sulfates et des nitrates mais aussi la majorité des particules organiques sont des aérosols secondaires.

Ainsi, les aérosols peuvent, d'une part, avoir une influence sur le climat grâce à leur capacité hygroscopique qui peut les amener à former des nuages. Et d'autre part, ils peuvent directement interagir avec le rayonnement solaire par diffusion ou absorption. Il est donc intéressant de pouvoir accéder à leurs dimensions et concentrations ainsi que de connaître leurs compositions chimiques et origines pour mieux comprendre comment ils interagissent avec l'atmosphère. Enfin, il est aussi utile de connaître la taille des aérosols qui se trouvent dans l'atmosphère car elle a un impact sur les problèmes de santé (plus les particules sont petites, plus elles se fixent profondément dans les poumons).

3. SMPS (Scanning Mobility Particle Sizer)

Parmi tous les instruments dont dispose le laboratoire, j'ai donc travaillé sur le Scanning Mobility Particle Sizer (SMPS).

3.1. Fonctionnement général

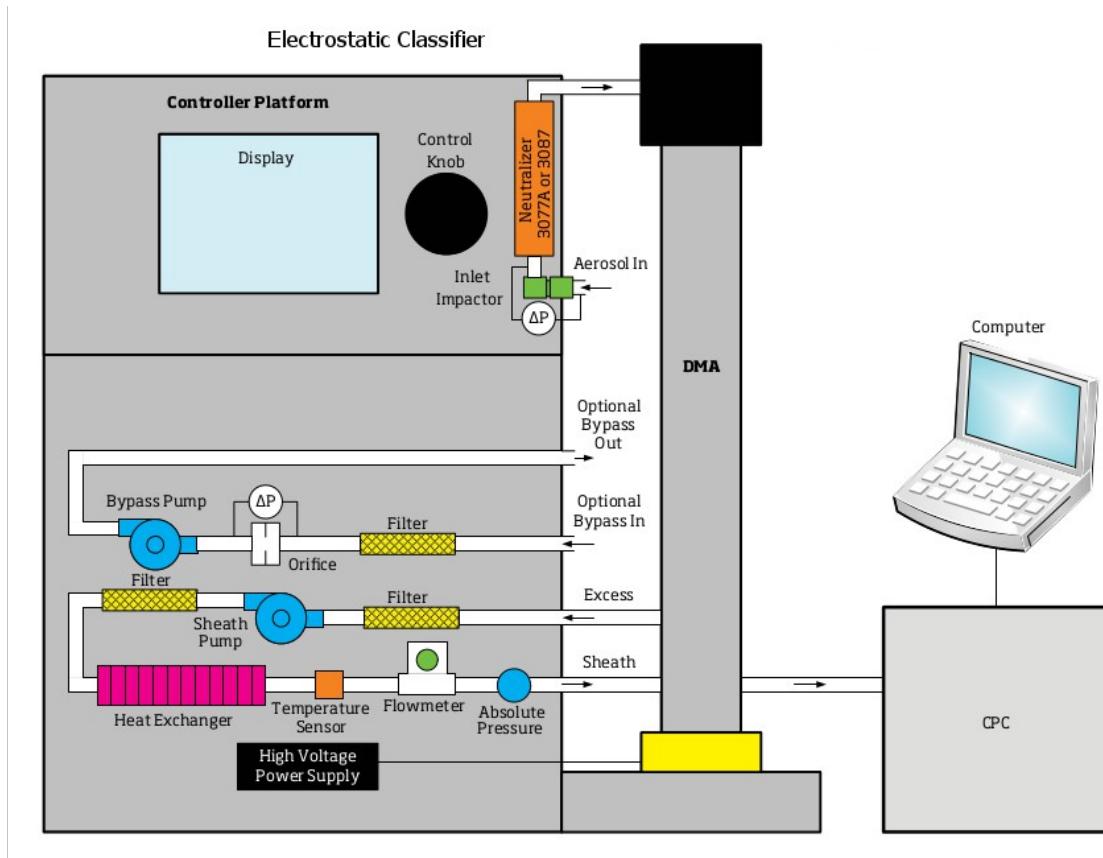


Figure 1

Un SMPS est un granulomètre ; il permet de faire des spectres de dimension des aérosols se trouvant dans l'air. Il est constitué d'un classificateur électrostatique, d'un compteur de particules (CPC : Condensation Particle Counter) et d'un ordinateur pour faire l'acquisition. Grâce à sa colonne (DMA : Differential Mobility Analyzer), le classificateur permet de sélectionner des aérosols d'une certaine taille et le CPC, lui, fait grossir les particules par condensation de butanol ou d'eau selon les instruments pour pouvoir compter optiquement ces particules. Ainsi, on obtient les concentrations en aérosols en fonction de leur taille : ce que l'on appelle le spectre dimensionnel. Il est aussi possible de coupler le classificateur avec un Cloud Condensation Nuclei (CCN) ; c'est un instrument qui compte les noyaux de condensation nuageuse en faisant grossir les aérosols suivant le même processus que dans l'atmosphère.

3.2. Differential Mobility Analyzer (DMA)

Pour trier les aérosols, cet instrument se sert de la mobilité électrique Z que possède une particule chargée. En effet, on a :

$$Z = \frac{neC_u}{3\pi\mu D_p} \quad \text{et} \quad C_u = 1 + \frac{2\lambda}{D_p} [1,142 + 0,558 + e^{-\frac{0,999 D_p}{2\lambda}}]$$

avec n le nombre de charges, e la charge élémentaire, μ la viscosité dynamique, D_p le diamètre d'une particule, C_u le coefficient de correction de Cunningham et λ le libre parcours moyen de la particule.

On applique au cylindre central de la colonne une tension et on connecte le cylindre extérieur à la masse afin de créer un champ électrique à l'intérieur de la colonne, on peut donc écrire la mobilité d'une autre façon :

$$Z = \frac{Q}{4\pi CV} \quad \text{et} \quad C = \frac{L}{2 \ln(R_2/R_1)}$$

avec Q le débit du fluide qui transporte les particules (sheath air), V la tension de la colonne de potentiel non nul et C la capacité d'un condensateur formé par deux cylindres de rayon R_1 (cylindre intérieur) et R_2 (cylindre extérieur) de longueur L .

Ainsi, on a :

$$D_p = C_u \frac{2neVL}{3\mu Q \ln(R_2/R_1)}$$

Pour chaque tension on a un certain diamètre de particule. Cependant, les aérosols ne sont pas naturellement chargés, c'est pour cela que l'instrument est muni d'une source radioactive (Neutralizer) qui charge les particules selon une loi aléatoire (**annexe I.4**). Il faut donc en tenir compte lors du calcul de la concentration des particules.

3.3. Flux

Alors que certains aérosols de même taille sont expulsés (Monodisperse Aerosol Out), le reste du flux est récupéré dans le Excess Air Out (ou Sheat air) pour être purifié des particules de diverses tailles qu'il contient et être renvoyé dans la colonne à un débit que l'on contrôle. Le excess air passe par deux filtres pour supprimer les particules, par la pompe qui régule le débit puis par un échangeur de chaleur qui permet de refroidir le fluide réchauffé par la pompe. Le flux d'air passe aussi par un capteur de température, d'humidité, de pression absolue et différentielle pour mieux contrôler le débit et pour connaître les paramètres du facteur de correction de Cunningham que nous avons vu précédemment. Pour ce qui est du débit du flux d'aérosols, il faut qu'il soit dix fois inférieur au débit du sheath pour limiter le temps de résidence dans la colonne et améliorer la résolution.

3.4. Représentations graphiques

A l'aide de scilab, on peut alors créer un programme itératif pour avoir la gamme complète de diamètres que l'on peut détecter avec cet instrument. En effet, on a vu précédemment que le diamètre des particules que l'on sélectionne avec le DMA dépend d'un terme de correction qui dépend lui-même du diamètre, on fait donc plusieurs itérations en se fixant un seuil de convergence pour définir ces diamètres (**annexe II.1**). Pour deux débits différents de sheath flow, on a :

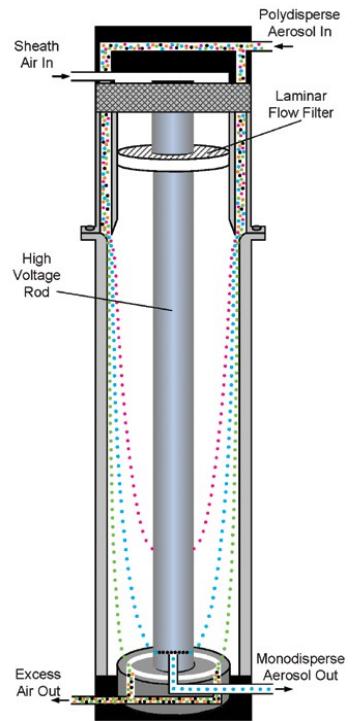


Figure 2

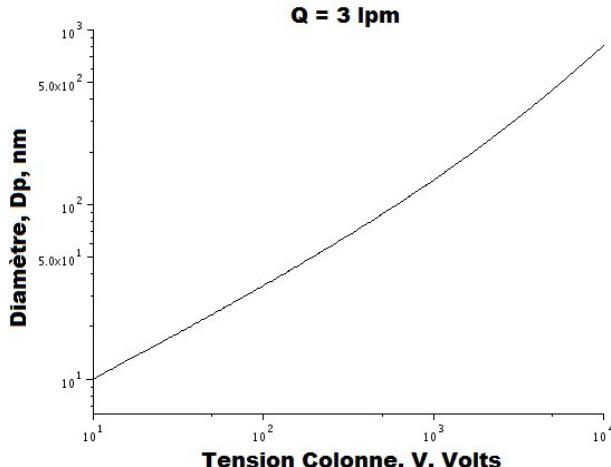


Figure 3

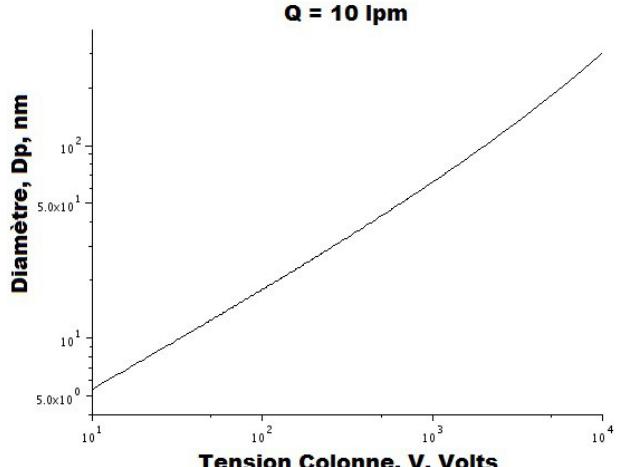


Figure 4

On peut donc voir qu'avec un débit de 3 lpm (litre par minute), on a accès à des diamètres entre 10 nm et 800 nm alors qu'avec 10 lpm, les diamètres varient de 5 nm à 300 nm. On choisit alors de travailler avec un débit de 3 lpm pour avoir une plus large fourchette de diamètres. Cependant, en faisant cela, on perd un peu de résolution, en effet les particules d'aérosol mettent un certain temps à traverser la colonne (**annexe II.2**). Il existe alors une incertitude sur la mesure faite par ce CPC qui dépend du temps que mettent les particules à traverser la colonne et donc du débit du fluide dans la colonne. Cette incertitude augmente lorsque le débit dans la colonne diminue.

Pour une raison que l'on verra plus tard, lorsque l'on fait un spectre de dimension, on fait varier la tension de la colonne de façon exponentielle, ainsi, si on trace le diamètre en fonction du temps (**annexes I.1.a et II.3**). On peut voir que l'incertitude en bleu est supérieure pour un débit de 3 lpm plutôt que pour un débit de 10 lpm mais que le diamètre atteint est bien supérieur pour 3 lpm.

3.5. Colonne radiale

Nous avons vu que l'instrument fonctionne avec un séparateur axial (colonne), il existe aussi un séparateur radial qui a la forme de deux disques, on peut en voir une coupe figure 5.

Le fonctionnement est le même sauf que c'est un disque qui est chargé et l'autre qui est portée à la masse. On a alors une capacité différente pour le condensateur formé par ces deux disques :

$C = a^2 / 8h$ avec a le rayon des disques et $2h$ la distance entre les disques. On peut alors écrire le diamètre en utilisant la mobilité électrique comme précédemment :

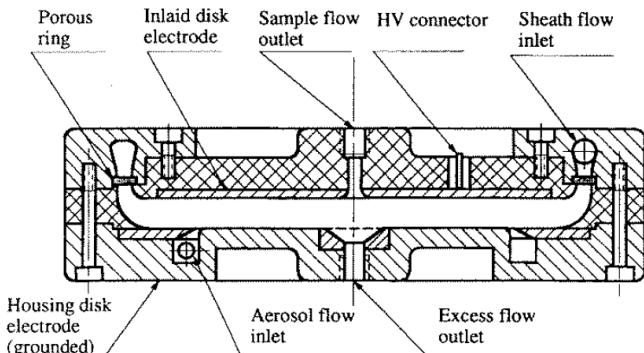


Figure 5

$$D_p = C_u \frac{neVa^2}{3\mu Q 2h}$$

Si l'on souhaite travailler dans la même gamme de diamètres qu'avec la colonne radiale, il suffit de jouer sur les paramètres a , h et Q (**annexe II.4**). On peut alors tracer toutes les possibilités de dimensions du disque (figure 6). Cependant, il faut veiller à ce que la hauteur ne soit pas trop faible non plus pour ne pas créer d'arc électrique entre les deux disques.

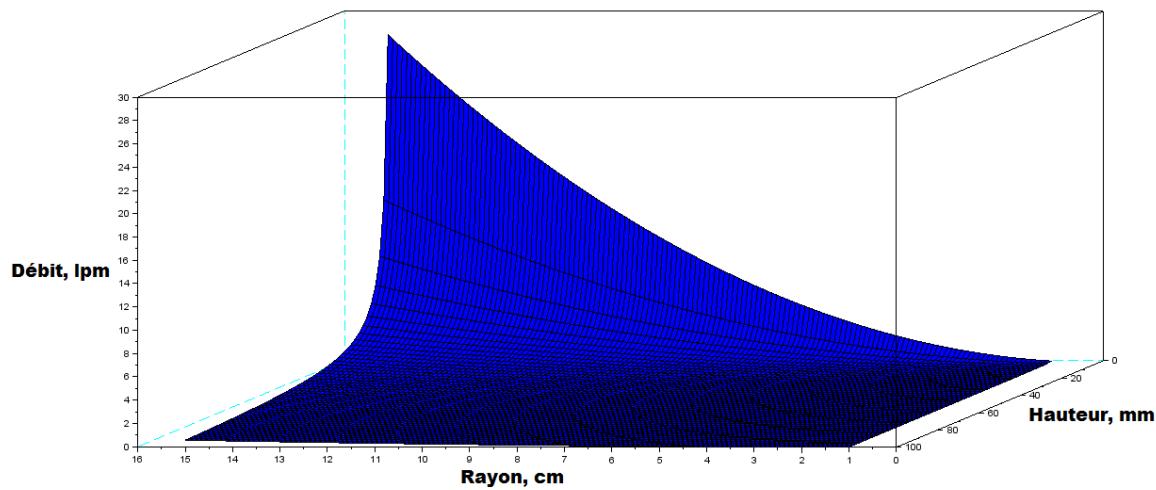


Figure 6

4. Processus de miniaturisation

L'équipe MNPCA du CNRM utilise, pour l'instant, les mesures SMPS sur des sites sols fixes ce qui permet de caractériser un site ou d'étudier, en associant d'autres instruments, le vieillissement des aérosols. Le SMSP est aussi utilisé sur avion ce qui permet d'explorer de vastes zones géographiques et d'étudier la distribution sur la verticale des aérosols. De nombreux phénomènes prennent naissance dans la couche limite atmosphériques (entre 0 et 1500 m), l'avion permet d'explorer la partie haute de la couche limite, mais en dessous de 500 m les règles de l'aéronautique et la sécurité des scientifiques restreignent le domaine de mesure. Seuls les drones permettront d'étudier de manière précise cette zone de l'atmosphère. C'est pour cette raison que l'on essaie de miniaturiser les instruments de mesure.

4.1. Conception assistée par ordinateur (CAO)

J'ai dû tout d'abord me servir d'un logiciel de CAO (SolidWorks) afin d'optimiser la disposition des différentes pièces qui composent l'instrument (pompe, filtre, échangeur, certaines cartes électroniques, alimentation haute tension...). J'ai procédé à plusieurs itérations pour voir quelle disposition était la plus intéressante (**annexe I.2**).

J'ai aussi utilisé SolidWorks pour dessiner toutes les pièces que j'ai fait construire au CEMES par impression 3D (**annexe I.5.a**) ou à l'atelier de météo-France (**annexe I.5.b**).

Ci dessous les deux dispositions que nous avons retenues. La figure 7 serait pour une partie de l'instrument (il manque la colonne) qui irait dans un drone et la figure 8 possède un socle pour que l'on puisse faire des tests sur l'instrument en laboratoire.

La colonne que l'on peut voir ici n'est que provisoire, en effet, elle est bien trop lourde pour pouvoir dans l'avenir être embarquée sur un drone de mesure. La colonne radiale que nous avons vu précédemment est en cours de développement et serait plus intéressante pour cette tâche. Grâce aux nouvelles technologies d'impression 3D, la colonne radiale serait beaucoup plus légère et sa construction en serait facilitée.

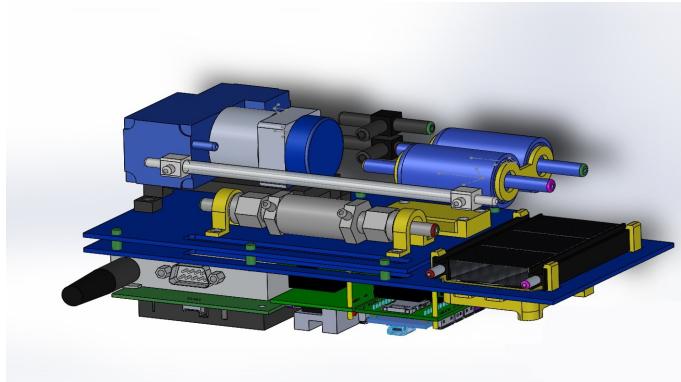


Figure 7

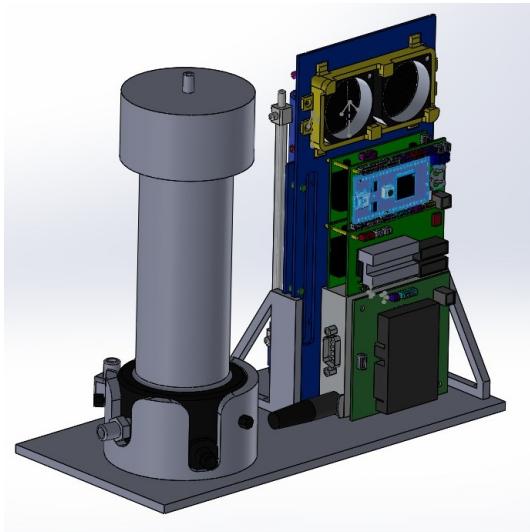


Figure 8

4.2. Fonctionnement électronique

D'un point de vue électronique, il faut une carte d'acquisition qui puissent faire le lien entre les cartes gérant les différents capteurs (Carte Pression Température Humidité : PTU et Carte Pression Différentielle : $2\Delta P$) et contrôlant les cartes d'alimentations (Haute Tension et PWM).

4.2.1. Carte d'acquisition

La carte d'acquisition est composée d'un Mbed LPC1768 soudé sur une autre carte pour pouvoir l'alimenter. Le Mbed LPC1768 (Figure 9) est un micro-contrôleur, c'est un circuit intégré qui regroupe tous les éléments principaux d'un ordinateur : processeur, mémoire (morte pour le programme, mémoire vive pour les données), plusieurs périphériques et diverses interfaces d'entrée et de sortie (**annexe I.3**). Il dispose en effet de :

- 6 entrées analogiques avec une des six qui peut basculer en entrée analogique
- 6 sorties PWM (Pulse Width Modulation) qui peuvent toutes basculer en entrée
- 2 bus SPI (Serial Peripheral Interface)
- 3 ports série
- 2 bus I²C (Inter Integrated Circuit)
- 1 port USB (Universal Serial Bus)
- 1 bus Ethernet

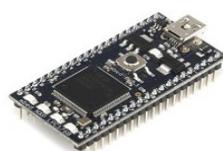


Figure 9

Cette carte permet de contrôler l'instrument (pompe, ventilateurs, haute tension) tout en lisant des informations sur l'instrument qui sont recueillies à l'aide des capteurs et des autres cartes électroniques. On peut alors analyser et stocker les données qui nous intéressent sur une carte µSD que l'on peut mettre sur le Mbed :

- Pression absolue sur le circuit du sheath flow
- Pressions différentielles dans le cycle du sheath flow et dans le flux d'aérosols pour pouvoir mesurer au débit et le contrôler
- Température et humidité du sheath flow
- Tension et courant circulant dans la colonne

Pour ce contrôle, il faut écrire un programme en C à l'aide de l'environnement de développement en ligne sur le site web du mbed. Le compilateur proposé offre une bibliothèque (mbed.h) qui permet de programmer toutes les interfaces du micro-contrôleur. Pour savoir si le programme à l'intérieur de la carte fonctionne, on utilise les quatre LED dont il dispose pour envoyer des signaux lumineux lors du fonctionnement. On peut aussi lire les signaux sur les pattes du mbed à l'aide d'un oscilloscope.

4.2.2. Cartes électroniques analogiques

Tout d'abord, pour que les cartes avec des capteurs fonctionnent, il faut qu'elles soient alimentées en 3.3V et 5V. La carte d'acquisition, la carte du contrôle PWM et la carte du contrôle de la haute tension doivent être alimentées par du +12V.

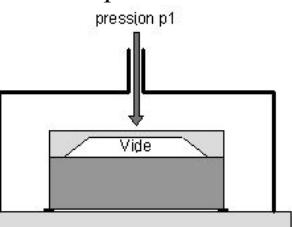


Figure 10

La carte PTU est donc reliée à un capteur de pression absolue (figure 10, il renvoie une tension), un capteur de température (c'est une résistance variable en fonction de la température qui fera varier la tension à ses bornes) et un capteur d'humidité (c'est un condensateur intégré dans un circuit oscillant qui renvoie une fréquence proportionnelle à l'humidité). Un étalonnage de tout ces capteurs en amont est nécessaire pour que les grandeurs mesurées par le mbed aient une signification. La carte $2\Delta P$ est sensiblement la même que la carte PTU, elle est équipée de deux capteurs de pression différentielle (figure 11) qui renvoient une tension proportionnelle à la différence de pression entre les deux points de mesure, elle doit être étalonnée avec un débitmètre car c'est la grandeur qui nous intéresse.

Les capteurs pressions fonctionnent à l'aide d'une membrane piézorésistive qui fait varier la tension à leurs bornes suivant sa déformation. Les capteurs de pression différentielle sont reliés au tube capillaire et au frit alors que le capteur de pression absolue peut être relié à n'importe quel tuyau du circuit comme on peut le voir sur la figure 13. Le frit et le tube capillaire créent une variation de pression en diminuant l'orifice de passage du fluide ce qui permet de remonter jusqu'au débit comme on l'a vu en mécanique des fluides.

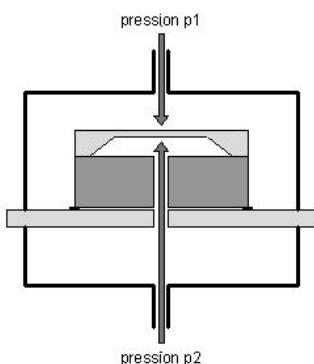


Figure 11

La carte du contrôle PWM permet d'amplifier le signal 0-3.3 volt (en sortie du mbed) en un signal 0-12 volt et de délivrer une plus forte intensité que ne le permet le mbed, c'est un étage de puissance. Le principe d'un signal PWM est que l'on envoie un signal rectangulaire d'une fréquence donnée d'amplitude 3,3V crête à crête (pour le mbed, 12V avec la carte) et que l'on joue avec le rapport cyclique pour avoir une tension moyenne entre 0 et 3,3V. Un rapport cyclique de 50 % donnera une tension moyenne de 1,75V. De plus, c'est cette carte qui reçoit l'alimentation générale de l'instrument (0-12V) et qui la distribue à la carte d'acquisition et à la carte haute tension.

La carte du contrôle de la haute tension permet d'amplifier le signal envoyé par le mbed à l'aide du bus SPI car l'alimentation haute tension qui fournit entre 0 et 10000V en sortie à besoin d'une tension de commande variant entre 0 et 10V en entrée. Cette carte alimente aussi l'alimentation HT en 24V pour qu'elle puisse fonctionner.

4.2.3. Ports série

En instrumentation mais aussi dans l'industrie, les instruments communiquent et s'envoient des commandes à l'aide de ports série qui sont souvent des ports RS-232 (le port série standard). Aujourd'hui, les ports série ont tendance à disparaître sur les ordinateurs personnels pour laisser la place aux ports USB ce qui pose pas mal de problèmes car les instruments disposent encore de port RS-232, il faut donc utiliser des adaptateurs pour faire des échanges d'information entre les ordinateurs et les instruments de mesure.

Contrairement au port parallèle avec lequel on envoie chaque bit d'information sur un fil différent, le principe du port série est de transmettre l'information *bit à bit* sur un fil en envoyant des trames de caractères. Il y a un bit de départ qui définit le début de la trame, entre 5 et 9 bits d'information suivant les modes de transmission, un bit de parité qui permet de détecter des erreurs de transmission et un bit d'arrêt. Il faut aussi choisir la vitesse de transmission qui est en bauds : le nombre de symboles transmissibles par seconde. En effet, ce mode de transmission est asynchrone, c'est pourquoi les deux instruments qui communiquent doivent connaître la vitesse de transmission et les bits de départ et d'arrêt peuvent aussi être vus comme des bits de synchronisation.

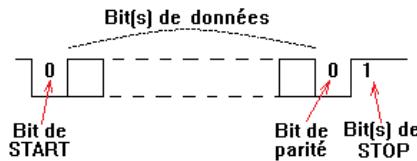


Figure 12

Les ports série du mbed possèdent deux canaux : un pour transmettre des données et un autre pour les recevoir. Comme on l'a vu précédemment, le mbed possède d'autres ports série avec un fonctionnement quelque peu différent :

- Port SPI : il utilise quatre broches, une pour émettre des trames d'information, une autre pour les lire, une horloge (signal carré) qui permet la synchronisation de la transmission et un dernier signal qui permet de sélectionner avec quel instrument on communique car les ports SPI peuvent être connectés à plusieurs appareils.
- Bus I²C : il utilise deux connections, une pour émettre et lire des trames d'information (ligne de données bidirectionnelle) et une horloge de synchronisation.
- Port USB : il utilise quatre connections, deux pour recevoir et émettre des paquets de données mais en plus grande quantité que précédemment et deux broches pour l'alimentation.

4.2.4. Schéma de l'instrument

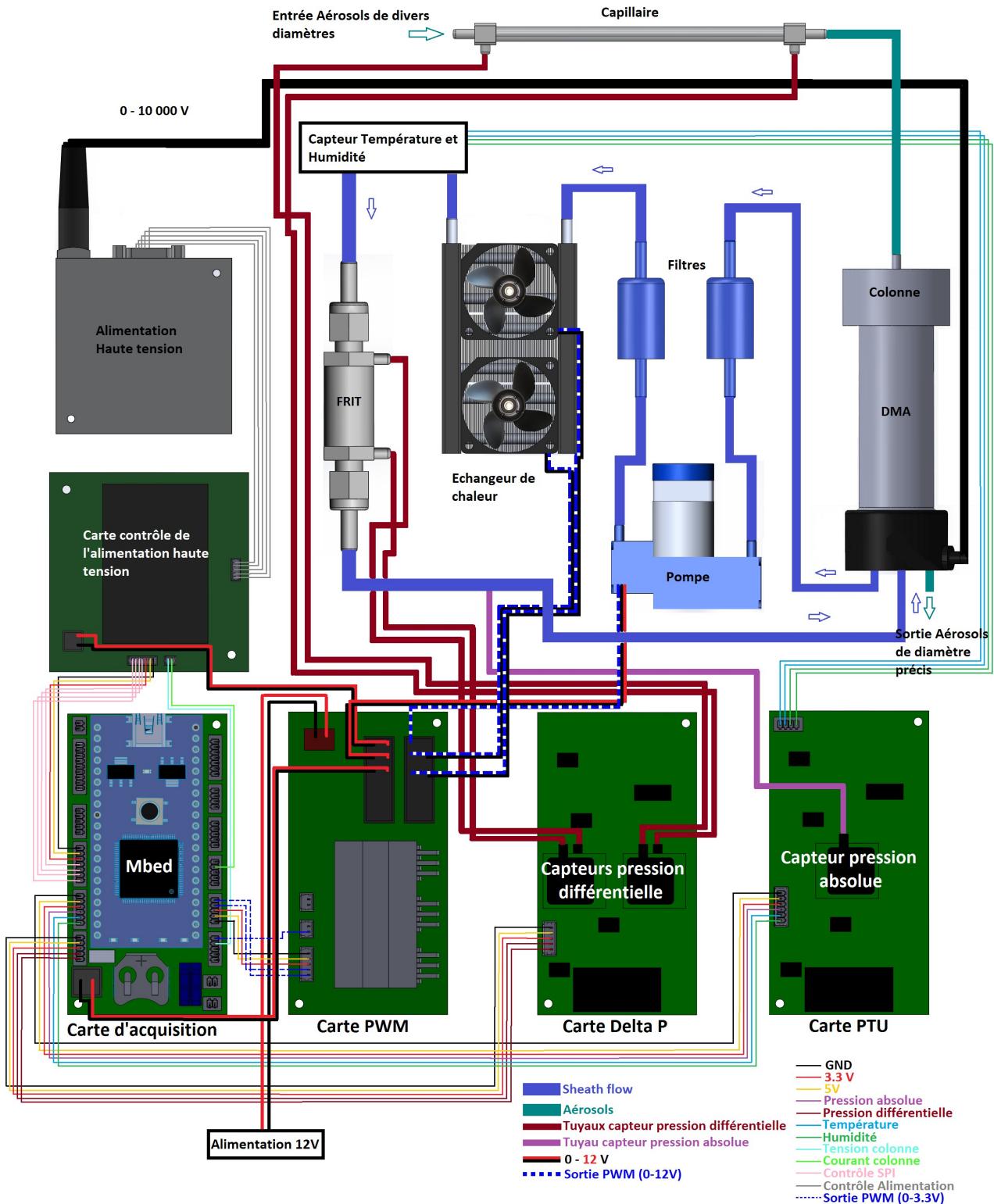


Figure 13

4.3. Programmation du mbed

4.3.1. Contrôle de la Haute tension

On représente généralement le spectre dimensionnel des aérosols comme une distribution log-normale, en effet nous avons vu précédemment que leurs tailles vont d'une dizaine de nanomètres à une dizaine de micromètres. Il est alors plus judicieux de faire varier la tension de la colonne de façon exponentielle pour couvrir toute la gamme de tailles. Ainsi, pour tracer un spectre, la tension suit une loi exponentielle croissante jusqu'à 10000V puis une exponentielle décroissante jusqu'à 0V pour pouvoir redémarrer un spectre par la suite. La tension de la colonne suit donc la loi suivante qui fut déterminée lors d'un précédent stage :

$$V_{HT}(t) = 10 e^{0,0575646t} \quad \text{avec } t \text{ le temps}$$

Il faut 120 secondes pour aller de 0 à 10000V. Pour obtenir ceci en sortie de l'alimentation haute tension il faut commander la carte du contrôle de la haute tension avec un programme (**annexe III.1**) qui envoie des données numériques suivant cette loi exponentielle. Des tests, effectués lors d'un stage précédent, ont montré que le convertisseur analogique numérique du mbed n'avait pas une résolution suffisante pour couvrir la dynamique attendue tout en conservant une bonne résolution, j'ai donc utilisé un composant supplémentaire que l'on commande par le bus SPI. Ce composant électronique (DAC1220) qui permet de faire la conversion numérique-analogique peut être commandé par 20 bits d'information, cependant le bus SPI n'envoie que des octets (8 bits), ainsi, le programme doit donc calculer la valeur de la loi exponentielle en numérique puis répartir cette valeur sur 3 octets pour les envoyer. Pour pouvoir communiquer et configurer la carte électronique, je me suis servi de la documentation (voir références) du DAC1220 qui permet en partie de faire la conversion numérique-analogique.

La carte électronique dont j'avais besoin n'ayant pas été construite à temps, j'ai utilisé la sortie analogique dont dispose le mbed pour contrôler l'alimentation haute tension en écrivant un autre programme (**annexe III.2**).

4.3.2. Contrôle de la pompe

Afin de conserver un débit stable dans le sheath flow, il faut en permanence réguler la pompe. Pour cela, on procède à un asservissement en utilisant la méthode de régulation PID (Proportionnel-Intégral-Dérivé). On fixe la consigne que l'on souhaite (dans notre cas, le débit du sheath flow qui correspond à une tension donnée par un capteur que l'on a étalonné pour qu'elle ait un sens), ensuite on compare la valeur mesurée par le capteur à la valeur de la consigne et on applique la commande de tension suivante :

$$V_{comande} = V_{mesuré} + K_p (V_{consigne} - V_{mesuré}) + K_i \int (V_{consigne} - V_{mesuré}) dt$$

avec $V_{mesuré}$ la tension mesurée par le capteur de pression différentielle qui est proportionnelle au débit, $V_{consigne}$ la tension de consigne, K_p la constante de proportionnalité qui détermine la vitesse avec laquelle on tend vers la tension de consigne et K_i la constante d'intégration qui influe sur les oscillations autour de la tension de consigne. Le terme d'intégration est là pour rendre l'asservissement plus efficace et moins brutal. Pour choisir les constantes K_p et K_i j'ai d'abord effectué quelques simulations sur scilab puis j'ai alors fait plusieurs tests de programmes (**annexe III.3**) pour mon instrument. En général, il y a aussi un terme de dérivation pour le PID mais il n'était pas indispensable pour cet asservissement.

4.3.3. Programme de contrôle général

On ne peut mettre qu'un seul programme qui s'occupe de tout contrôler à fois sur le mbed. Je me suis donc aidé du programme qui existait déjà pour contrôler l'instrumentation des drones construits à ce jour pour faire le mien (**annexe III.4**). Mon programme contrôle la pompe, les ventilateurs et l'alimentation haute tension à la fois. Bien que je n'ai pas eu le temps de m'y consacrer durant mon stage, il faudrait par la suite programmer le stockage des données utiles sur la carte µSD pour les traiter puis la communication avec le CPC et CCN avec lesquels le classificateur est prévu de fonctionner. Mon programme comporte presque toutes les variables analogiques qui seraient utiles au programme complet même si je ne les utilise pas toutes. Il manque une des deux pressions différentielles car j'utilise une sortie analogique pour commander l'alimentation haute tension comme je l'ai expliqué précédemment.

4.4. Montage de l'instrument

Au fur et à mesure que les pièces et les cartes électroniques de l'instrument étaient disponibles, je les montais ensembles et je faisais le câblage. Les cartes électroniques pour le contrôle de la haute tension avec bus SPI et le contrôle par PWM n'ont pas pu être construites pendant la durée de mon stage, j'ai donc utilisé une carte amplifiant un signal analogique pour la commande de la haute tension et une carte un peu différente de celle prévue pour le contrôle par PWM (elle possède plus de sorties et d'entrées qu'il n'est nécessaire). Je n'ai donc pas pu monter l'instrument exactement comme c'était prévu car les cartes de "secours" étaient plus grandes (**annexe I.5.c**).

De plus, j'ai eu un problème avec la pompe qui était prévue car elle s'est endommagée, j'ai du alors utiliser une pompe plus puissante avec une électrovanne pour réguler le débit (le programme pour contrôler l'électrovanne est exactement le même que celui décrit pour le contrôle de la pompe, cette fois ci la pompe plus puissante fonctionne à plein régime et l'électrovanne est contrôlée par PWM).

Pour ce qui est de la source radioactive, il est prévu qu'elle soit remplacée par une autre technologie utilisant l'effet corona pour charger les particules (les particules sont chargées à l'aide d'une décharge électrique) mais l'équipe n'a pas encore pu la se procurer. Pour les tests, j'ai utilisé la source radioactive du SMPS

Ensuite, les pompes que nous utilisons créent des oscillations dans le débit (les valves créent des variations de pression dans le circuit du sheath). Ces oscillations ce répercutent sur les mesures des capteurs et la régulation par PID devient moins précise et même chaotique. Les deux filtres utilisés dans le sheath flow n'étant pas suffisants pour diminuer ces oscillations, nous avons dû rajouter deux autres petits filtres dans le circuit. Enfin, en utilisant l'électrovanne, j'ai veillé à la mettre avant la pompe et non après pour éviter des problèmes de surpression entre la pompe et l'électrovanne.

4.5. Tests

Afin de tester l'instrument que j'ai construit pendant trois mois, je me suis servi du montage Classificateur + CPC + PC.

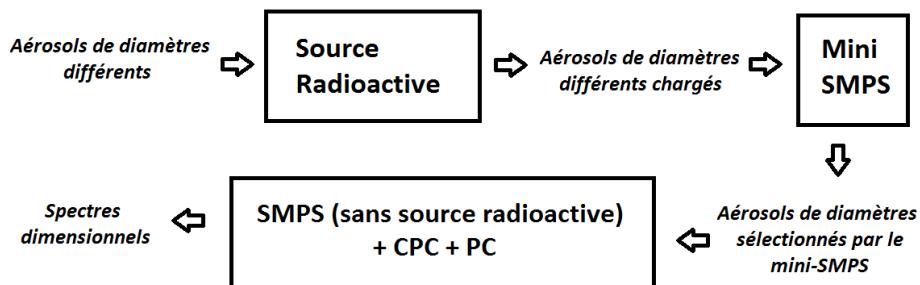


Figure 14

J'ai tout d'abord tracé un spectre des particules que le SMPS seul pouvait détecter (**annexe I.1.b**) pour le comparer ensuite aux particules sélectionnées par mon instrument. Ensuite, j'ai programmé mon instrument de façon à ce que la tension dans la colonne soit fixée pour que les particules qui sortent du mini-SMPS soient de diamètre constant dans le but tester comment le mini-SMPS sélectionne les particules. Afin que les particules soient chargées pendant le test, j'ai enlevé la source radioactive du SMPS pour la mettre avant le mini-SMPS. J'ai alors tracé un spectre pour chaque tension différente qui était dans la colonne (**annexe I.1.c**). Puis j'ai calculé les diamètres théoriques qui devraient correspondre aux tensions que j'ai testées et j'ai obtenu les résultats suivants :

Tension (V)	10	50	100	300	800	1300	2000	2500	4000	6000
Dp_exp (nm)	40 (±7)	45 (±7)	60 (±10)	80 (±15)	130 (±25)	180 (±40)	250 (±50)	300 (±50)	350 (±50)	520 (±50)
Dp_théo (nm)	7,00	16,0	23,3	42,9	76,3	103	136	159	221	300

Les diamètres expérimentaux sont approximatifs car comme on peut le voir sur les annexes, les diamètres des particules détectées s'étaisent en forme de pyramide. On remarque qu'une sélection sur les diamètres est faite mais il y a un certain décalage entre les diamètres théoriques et expérimentaux.

Ce problème peut venir de plusieurs paramètres ; j'ai effectué ces tests le dernier jour de mon stage et je ne disposais pas des capteurs température et humidité dans le sheath flow alors j'ai fait mes calculs théoriques avec des valeurs arbitraires de température et d'humidité et non les valeurs exactes lors des mesures.

De plus, comme on peut le voir sur le spectre dimensionnel du SMPS (**annexe I.1.b**), les particules détectées mesurent entre 15 et 500 nm et les concentrations sont plutôt faibles sur les bords du spectre. On peut alors se douter que la précision des mesures n'est pas la même sur tout le spectre. Pour plus de précision, il aurait fallut augmenter artificiellement la concentration des particules pour avoir plus de chances d'en détecter.

Conclusion

Ce stage a été très enrichissant, il m'a permis de découvrir un peu mieux le milieu de la recherche mais aussi celui de l'entreprise. J'ai découvert comment se déroule le travail de recherche dans une équipe et j'ai appris à travailler en coopération avec ses membres.

Bien que je n'ai pu achever complètement mon projet, il m'a permis de faire des travaux variés tels que la conception assistée par ordinateur, le suivi de fabrication de pièces, la modélisation du fonctionnement d'un instrument de mesure, la programmation de ce dernier, l'intégration de l'instrument et ses tests.

L'instrument que j'ai continué à développer n'est qu'un prototype et il est encore loin d'être opérationnel pour être embarqué dans un drone mais le fait d'avoir travaillé dessus m'a permis de mettre en pratique les connaissances que j'avais acquises et cela m'a surtout apporté beaucoup d'expérience dans des domaines dont j'ignorais tout.

J'ai aussi constaté que trois mois dans un projet de recherche représente très peu de temps et il suffit qu'il y ait un petit imprévu pour que le projet n'avance plus.

Ce stage m'a aussi confirmé le fait que c'est le côté pratique et les tâches variées qui me plaisent dans la recherche et si j'envisage de poursuivre ma carrière dans ce domaine, ce serait dans une équipe comme celle là.

Bibliographie et références

Oeuvres :

Les aérosols : Physique et métrologie (André Renoux, Denis Boulaud ; Tec & Doc Lavoisier)

Physique et chimie de l'atmosphère (Robert Delmas, Gérard Mégie, Vincent-Henri Pleuch ; BELIN)

Aerosol Measurement :Principles Techniques et Applications (Paul A. Baron, Klaus Willeke ; Wiley)

Documentations et thèses :

Series 3080, Electrostatic Classifiers, Operation and Service Manual

(http://cires.colorado.edu/jimenez-group/Manuals/SMPS_3080_manual.pdf)

Documentation DAC1230 (<http://www.ti.com/lit/ds/sbas082g/sbas082g.pdf>)

Documentation Mbed LPC1768 (<https://mbed.org/platforms/mbed-LPC1768/>)

Scanning Mobility Particle Sizer™ Spectrometer (SMPS)

(http://www.google.fr/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&sqi=2&ved=0CEMFjAB&url=http%3A%2F%2Fwww.tsi.com%2FuploadedFiles%2F_Site_Root%2FProducts%2FLiterature%2FSpec_Sheets%2FSMPS3936-3034.pdf&ei=YPZMU876LOq20wWs54HoCw&usg=AFQjCNHXUOC7qU9UTcq-XTkh7YGulDTXAO&sig2=yvoOF_-6NMJ-CYOuTUI05g&bvm=bv.64764171,d.d2k&cad=rja)

Thèse : Etude des propriétés hygroscopiques des aérosols atmosphériques (http://www.google.fr/url?sa=t&rct=j&q=&esrc=s&source=web&cd=4&ved=0CEYQFjAD&url=http%3A%2F%2Ftel.archives-ouvertes.fr%2Fdocs%2F00%2F72%2F45%2F36%2FPDF%2F2009CLF21997.pdf&ei=INILU9u9Bq2a0QX1v4D4Ag&usg=AFQjCNF91SWtbL1Qk_6igA8G4GFmog0iA&sig2=EOSWe7qPSOtw6JgufjwI3Q&bvm=bv.64542518,d.d2k&cad=rja)

Sites Web :

Site du CNRM (<http://www.cnrm.meteo.fr>)

Le bilan radiatif de la Terre (<http://www.climatechallenge.be/fr/des-infos-en-mots-et-en-images/le-changement-climatique/leffet-de-serre/leffet-de-serre-naturel/le-bilan-radiatif-de-la-terre.aspx>)

Le port série des PC et compatibles (http://crazy.fred.free.fr/elec/port_serie.html)

Différents types de capteurs pression (<http://www.sensortronics.com/fr/produits/capteurs-et-transmetteurs-de-pression/types-de-pression.html>)

Port série, RS-232, connecteurs et connexions (<http://radio.pagesperso-orange.fr/RS232.htm>)

Projet VOLTIGE (<http://www.cnrm.meteo.fr/voltige/>)

Projet CHARMEX (<https://charmex.lsce.ipsl.fr/index.php/what-is-charmex-mainmenu-35.html>)

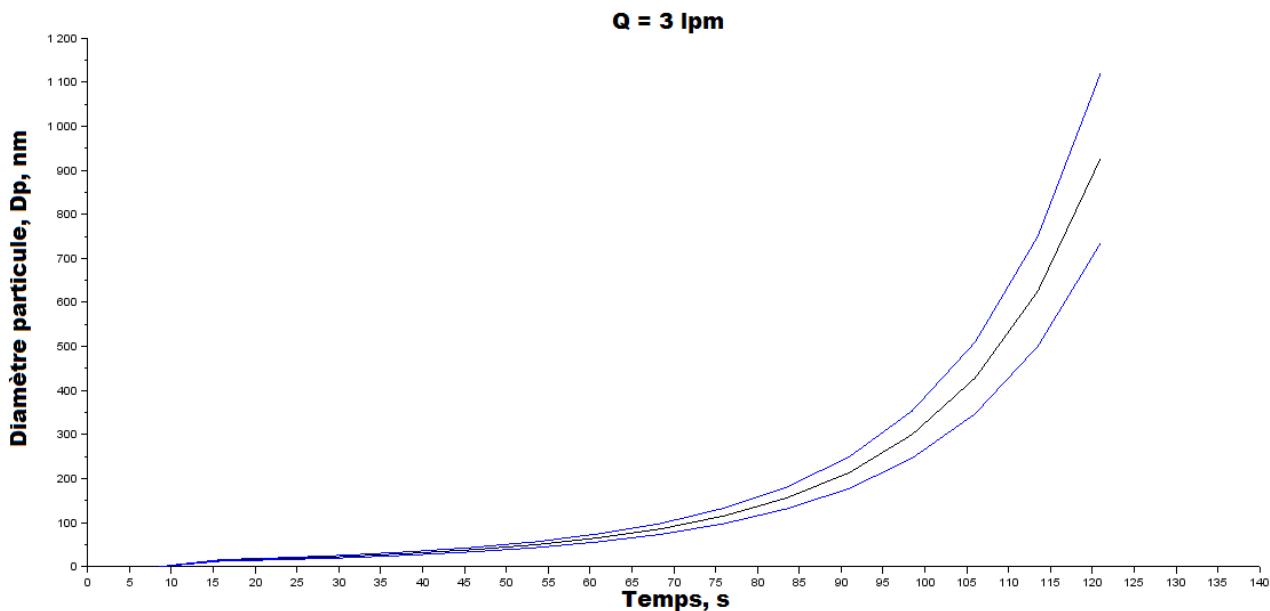
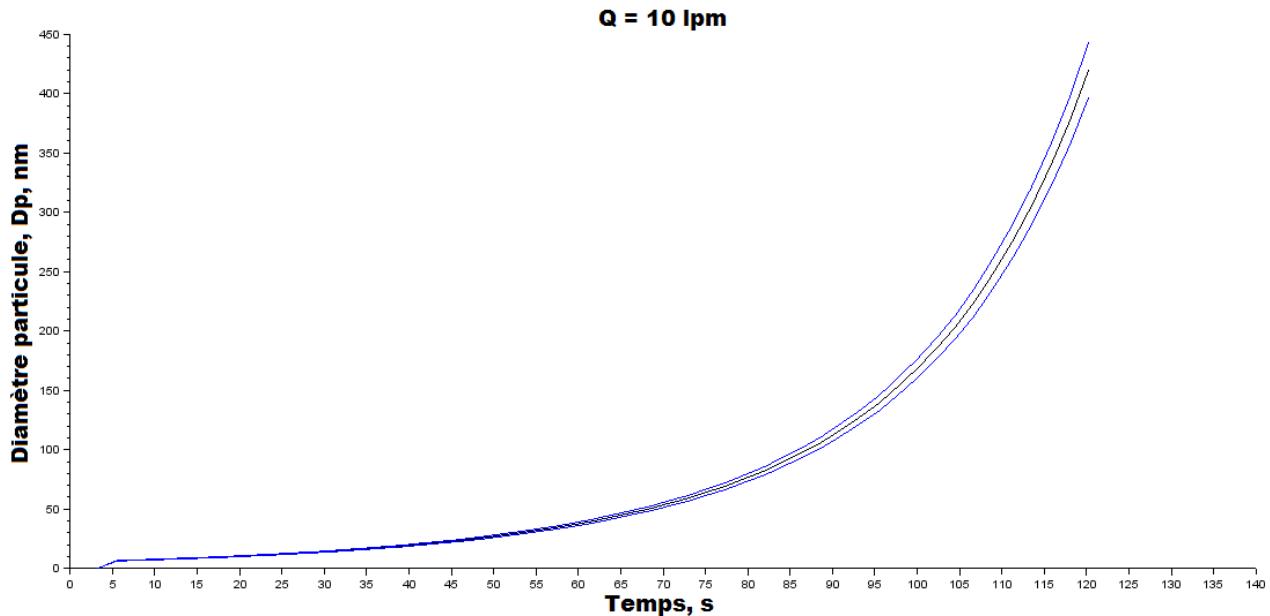
Projet PARISFOG (<http://sirta.ipsl.polytechnique.fr/parisfog/project-objectifs-scientifiques.html>)

ANNEXES

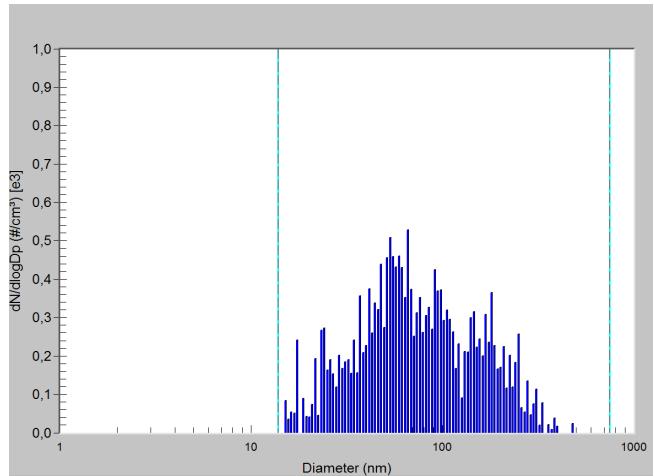
I. Courbes, Schémas et Photos

1. Courbes

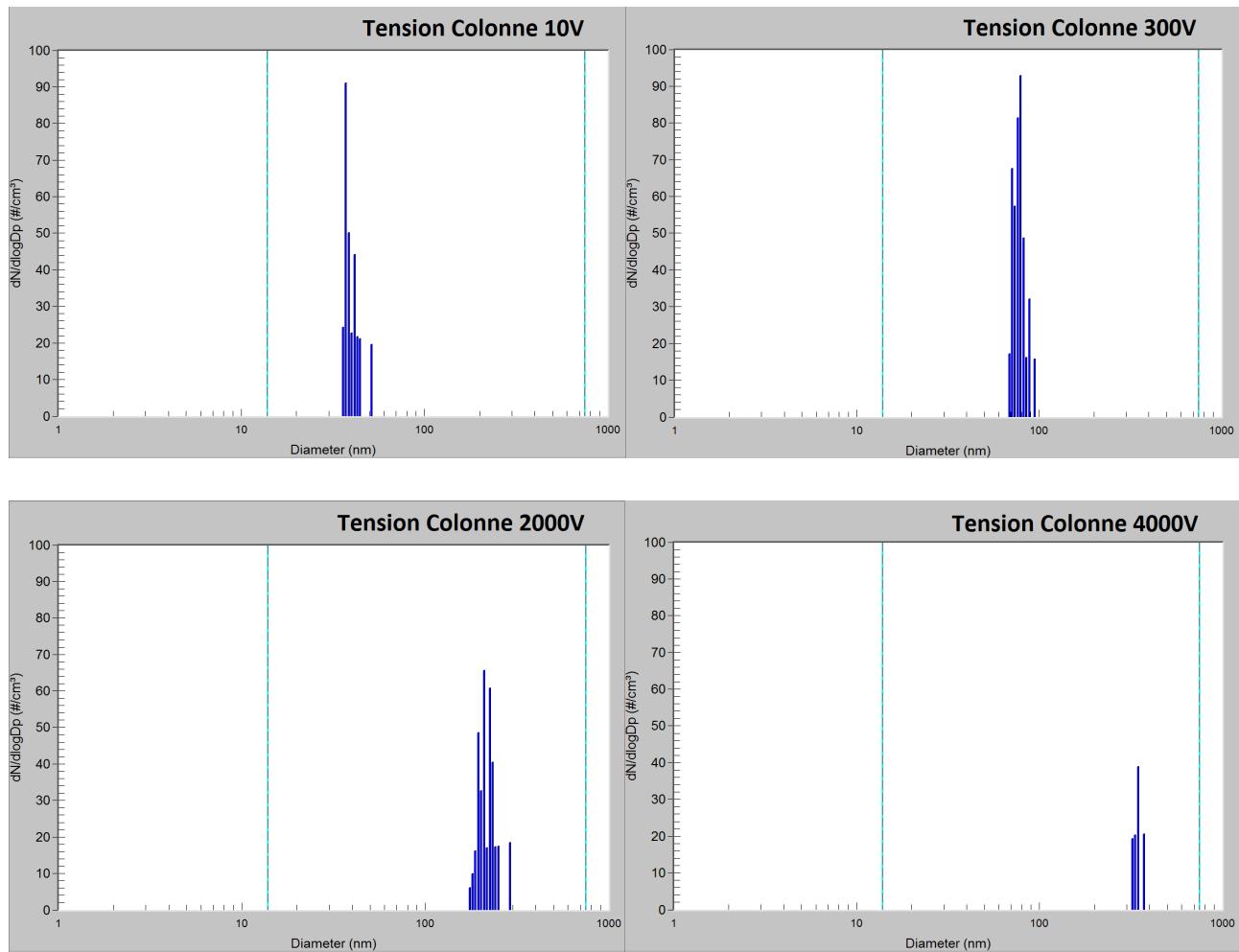
a. Courbes des diamètres dans la colonne axiale en fonction du temps avec les incertitudes



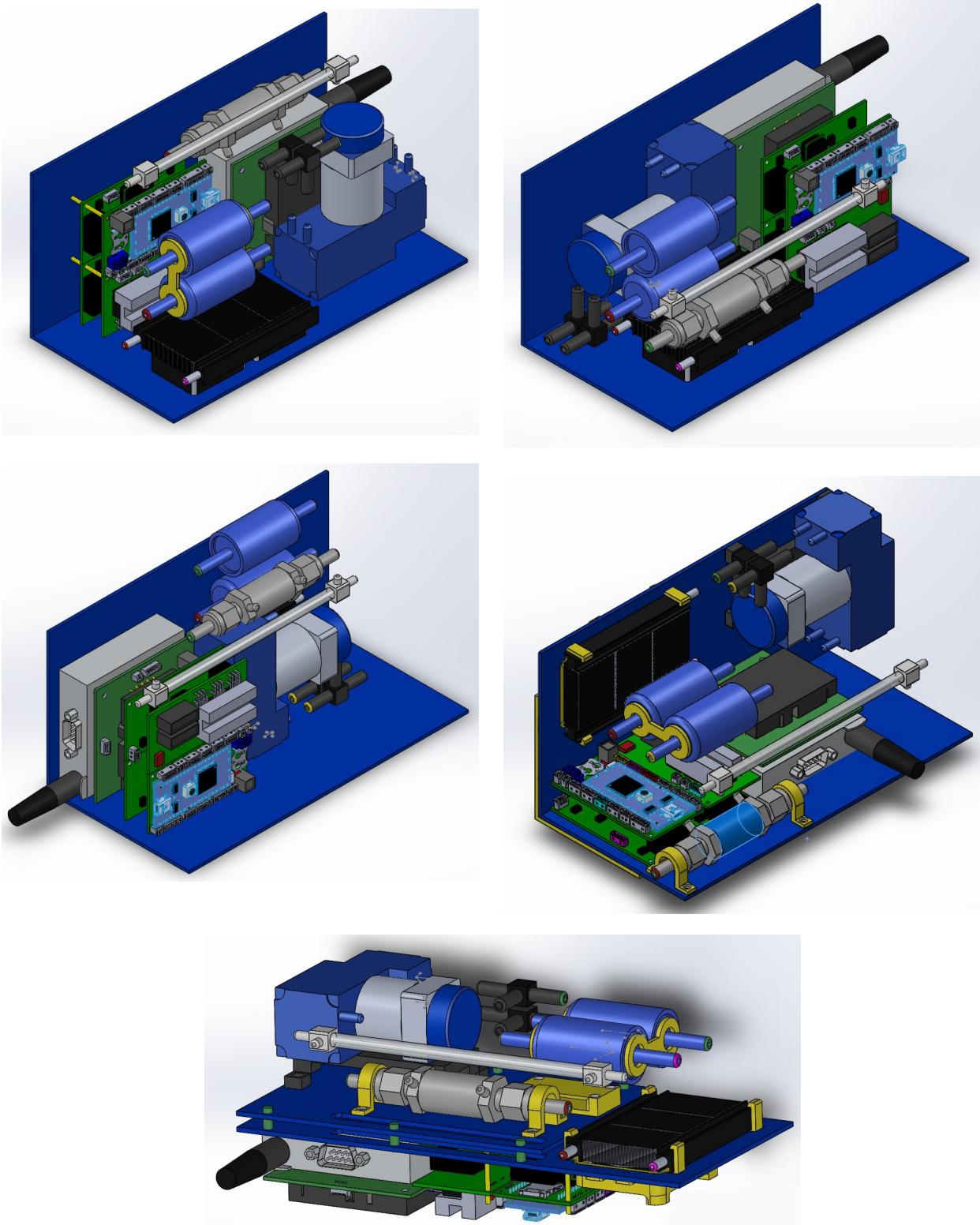
b. Spectre des particules d'aérosol du SMPS



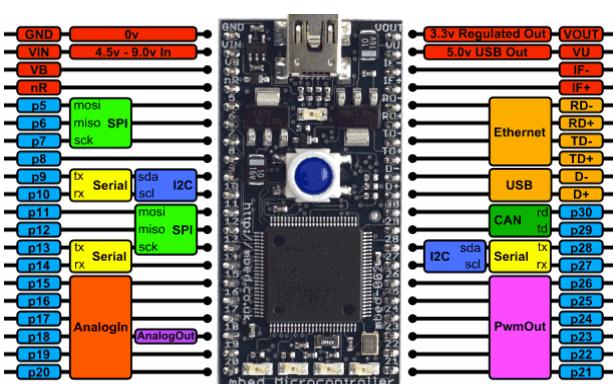
c. Exemples de spectres passant par le mini-SMPS



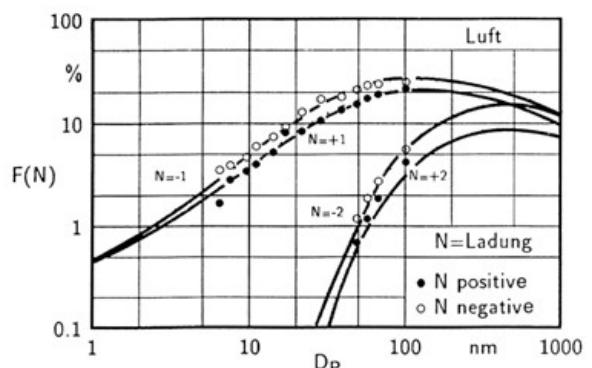
2. Itérations de SolidWorks



3. Sorties et entrées mbed

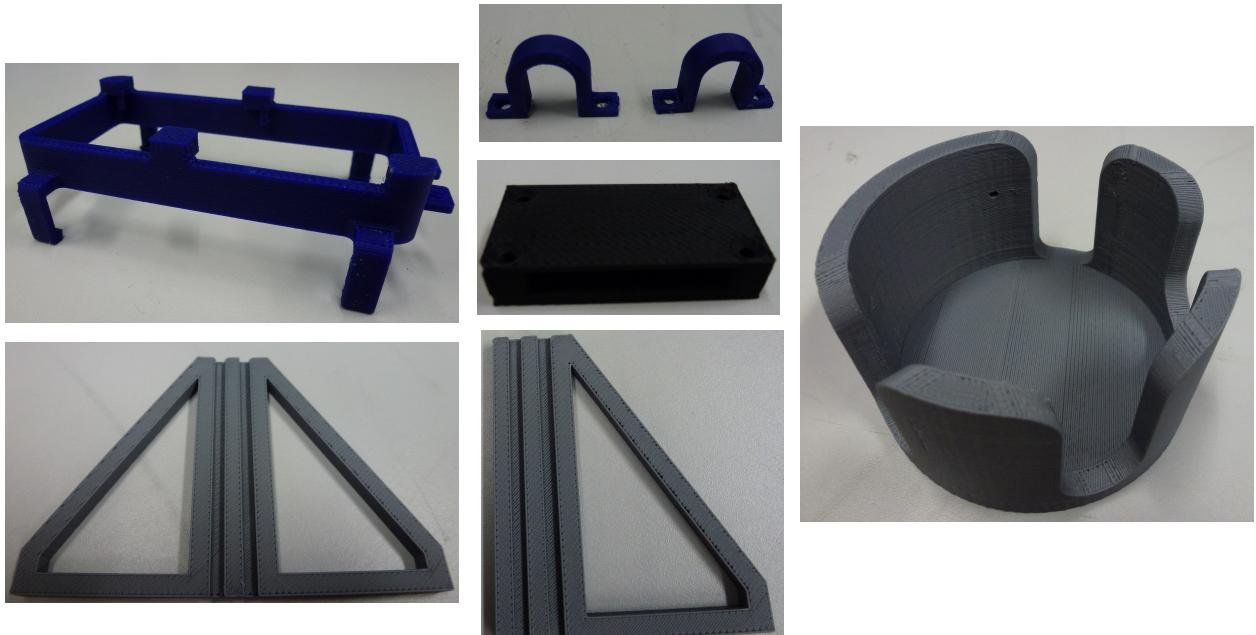


4. Loi aléatoire donnant la quantité de particules chargées en fonction de leur taille



5. Photos (elles ne sont pas à l'échelle)

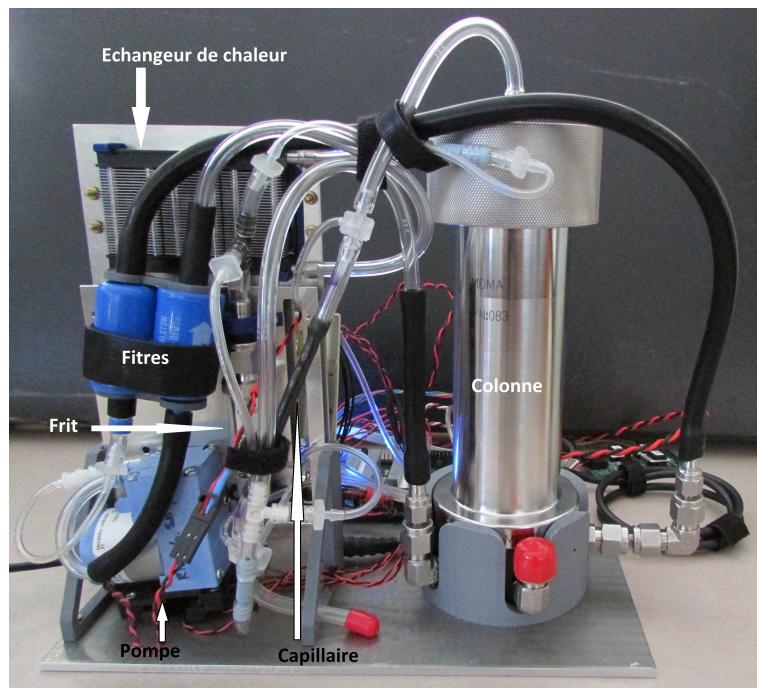
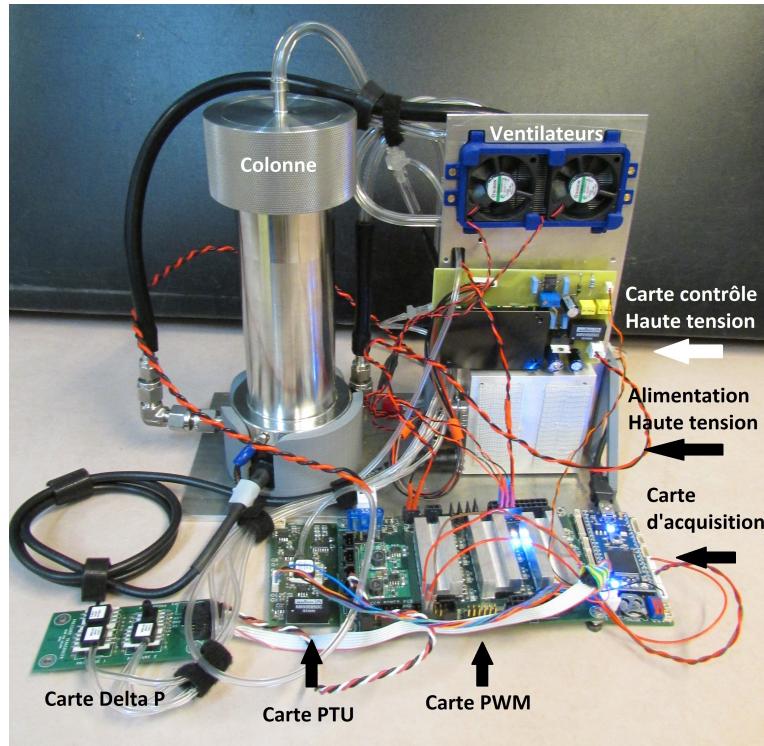
a. Pièces construites avec l'imprimante 3D du CEMES



b. Pièces construites à l'atelier de météo France



c. Mini-SMPS



II. Programmes Scilab (étude théorique)

1. Programme qui trace le diamètre des aérosols sélectionnés par la colonne axiale en fonction de la tension appliquée à celle ci

```
clc
clear

E=1.602176565E-19// Charge élémentaire
L=0.44369// Longueur DMA
T=293.15// Température
S=110.4// Sutherland constant
Tr=293.15// Température de référence
mu_r=0.0001823*0.1// Viscosité de référence (à 293.15K et 1atm)
mu=mu_r*((T/Tr)^1.5)*((Tr+S)/(T+S))// Viscosité de l'air
Qsh=3/60000// Débit du Sheath air en m³/s
R2=0.01961// Rayon cylindre extérieur
R1=0.00937// Rayon cylindre intérieur
Pr=1.01325E5// Pression de référence (atmosphère)
P=1.01325E5// Pression
n=1// Nombre de charges
alpha=1.142
betaa=0.558
gammaa=0.999
lembda_r=0.0664E-6// libre parcours moyen de référence
lembda=lembda_r*(Pr/P)*(T/Tr)*((1+(S/Tr))/(1+(S/T)))// libre parcours moyen
printf("Début");

DP=[]; // Initialisation du vecteur diamètre des particules
V=[10:1:10000]; // Initialisation du vecteur des tensions appliqués à la colonne

for i=1:size(V,2)// Boucle sur le voltage de 10V à 10000V

    Dp=(2*n*E*V(i)*L)/(3*mu*Qsh*log(R2/R1)); // Calcul du diamètre sans le terme de correction
    Kn=2*lembda/Dp;
    C=1+Kn(alpha+(betaa*exp(-gammaa/Kn))); // Calcul du terme de correction

    Dp_old=Dp;
    Dp_new=C*(2*n*E*V(i)*L)/(3*mu*Qsh*log(R2/R1)); // Première itération du diamètre avec le terme de correction
    delta_Dp =abs((Dp_new-Dp_old)/Dp_new); // Calcul de l'écart entre l'ancien diamètre et le diamètre "corrigé"

    while delta_Dp>0.01// Suite des itérations jusqu'à obtenir un écart de 1% entre les Dp
        Dp_old=Dp_new;
        Kn=2*lembda/Dp_old;
        C=1+Kn(alpha+(betaa*exp(-gammaa/Kn)));
        Dp_new=C*(2*n*E*V(i)*L)/(3*mu*Qsh*log(R2/R1));
        delta_Dp =abs((Dp_new-Dp_old)/Dp_new);
    end

    DP(i)=Dp_new; // Assigmentation de diamètre final au vecteur des diamètres
```

```

end

scf();
plot2d("oll",V,DP.*1000000000); // on trace  $D_p = f(V)$  sur un graphe logarithmique
ylabel("Diamètre,Dp,nm");
xlabel("Tension colonne,V,Volts");

printf("Terminé")

```

2. Calcul de la durée du trajet des particules dans la colonne

```

clear

clc

L=0.44369; // Longueur LDMA
Qsh=10/60000; // Sheath air flow rate : 10 lpm
Qa=Qsh/10; // Débit du flux d'aérosols
R2=0.01961; // Rayon extérieur de la colonne
R1=0.00937; // Rayon intérieur de la colonne

V_colonne=L*(R2^2-R1^2)*%pi; // Calcul du volume de la colonne

Delta_t=(V_colonne/(Qsh+Qa)); // Calcul du temps de trajet des particules dans la colonne

disp(Delta_t); // affichage de la valeur

```

3. Programme qui trace le diamètre des particules sélectionnées par la colonne axiale en fonction du temps avec l'incertitude sur le diamètre

```

clc

clear

E=1.602176565E-19 // Charge élémentaire
L=0.44369 // Longueur LDMA
T=293.15 // Température
S=110.4 // Sutherland constant
Tr=293.15 // Température de référence
mu_r=0.0001823*0.1 // Viscosité de référence (293.15K et 1atm)
mu=mu_r*((T/Tr)^1.5)*((Tr+S)/(T+S)) // Viscosité de l'air
Qsh=10/60000 // Sheath air flow rate : 3 lpm
R2=0.01961 // Rayon extérieur colonne
R1=0.00937 // Rayon intérieur colonne
Pr=1.01325E5 // Pression de référence
P=1.01325E5 // Pression
n=1 // Quantité de charges
alpha=1.142
betaa=0.558
gammaa=0.999
lembda_r=0.0664E-6 // libre parcours moyen de référence
lembda=lembda_r*(Pr/P)*(T/Tr)*((1+(S/Tr))/(1+(S/T))) // libre parcours parcours moyen

```

```

printf("Début");

// Calcul du diamètre d'une particule
DP=[];// initialisation des diamètres
t=[1:2.25:122.50];// Le pas du temps est décidé en fonction du temps que mettent les particules à traverser la colonne (programme précédent)
//pour un débit de 10 lpm, ce temps de traversée est de 2.25 secondes
for i=1:size(t,2)//Boucle sur les différents voltages en fonction des temps que l'on a choisis

V=12.011*exp(0.0605*t(i))// Voltage exponentiel
Dp=(2*n*E*V*L)/(3*mu*Qsh*log(R2/R1));
Kn=2*lmbda/Dp;
C=1+Kn(alpha+(betaaa*exp(-gammaaa/Kn)));
Dp_old=Dp;
Dp_new=C*(2*n*E*V*L)/(3*mu*Qsh*log(R2/R1));
delta_Dp =abs((Dp_new-Dp_old)/Dp_new);

while delta_Dp>0.01//correction

    Dp_old=Dp_new;
    Kn=2*lmbda/Dp_old;
    C=1+Kn(alpha+(betaaa*exp(-gammaaa/Kn)));
    Dp_new=C*(2*n*E*V*L)/(3*mu*Qsh*log(R2/R1));
    delta_Dp =abs((Dp_new-Dp_old)/Dp_new);

end

DP(i)=Dp_new;

end

D_Dp=[];// initialisation des incertitudes

for i=2:size(t,2)-2// Calcul incertitude
    D_Dp(i)=abs(DP(i-1)-DP(i+1))/4;
end

DP_min=[];
DP_max [];

for i=2:size(t,2)-2// Calcul courbe inf et courbe max
    DP_min(i)=DP(i)-D_Dp(i);
    DP_max(i)=DP(i)+D_Dp(i);
end

scf();
tt=[3.25:2.25:120.25];// nouvelle fourchette de temps pour tracer l'incertitude et la courbe des diamètres
DP_= [];
for i=2:size(tt,2)
    DP_(i)=DP(i);
end

title("Q = 10 lpm")
plot2d(tt,[DP_.*1000000000 DP_min.*1000000000 DP_max.*1000000000],

```

```
[1,2],leg="Dp@Dp_min@Dp_max");
ylabel("Particule Diameter,Dp,nm")
xlabel("Temps,s")

printf("Terminé")
```

4. Programme qui trace le débit du sheath air nécessaire au fonctionnement de la colonne radiale en fonction de ses dimensions (rayon et hauteur) pour travailler dans la même gamme de diamètre que le DMA.

clc

```
clear
stacksize('max');// Réservation de mémoire supplémentaire
```

```
L=0.44369// Longueur DMA
Qa=3/60000// Débit du Sheath air : 3 lpm
R2=0.01961// Rayon extérieur
R1=0.00937// Rayon intérieur
```

```
a=[0.01:0.001:0.15];// On cherche le rayon entre 0.01m et 0.15m
h=[0.001:0.001:0.05];// On cherche la hauteur entre 0.001 et 0.05m
```

```
for j=1:size(a,2)
```

```
for i=1:size(h,2)
```

```
Qr(j,i)=(Qa*log(R2/R1)*(a(j))^2)/(4*L*h(i)) // Calcul des débits possibles en fonction des
dimensions pour avoir sensiblement les mêmes caractéristiques que le SMPS
```

end

end

```
scf();
plot3d(a.*100,2.*h.*1000,Qr.*60000); // Affichage de la courbe
xlabel("Rayon,cm");
ylabel("Hauteur,mm");
zlabel("Débit,lpm");
```

III. Programmes en C pour le mbed (application)

1. Contrôle haute tension avec le bus SPI

```
#include "mbed.h" // librairie mbed
#include "stdio.h"
#include "math.h"
```

```
SPI Spi_HT(p11, p12, p13);
DigitalOut cs(p14);
```

```
DigitalOut led1(LED1);
DigitalOut led2(LED2);
```

```

DigitalOut led3(LED3);
DigitalOut led4(LED4);

long V_analogique;
long Q_1,Q_2,Q_3,R_1,R_2,R_3;
long V_numerique;

// Commandes en hexadécimal et binaire (en commentaire) qui permettent de communiquer avec le composant

int CMD_RB = 0x24; // 0b00100100, écrire dans le registre map, 2 octets, commencer à l'adresse 4 du registre
int CMD_DIR = 0x40; // 0b01000000, écrire dans le registre map, 3 octets, commencer à l'adresse 0 du registre
int CMR_1 = 0x20; // 0b00100000, configurer les paramètre par défaut
int CMR_0 = 0xf0; // 0b11110000, résolution 20 bits et d'autre paramètres que je ne détaille pas

float V_max=10000; // Tension max
int V_min=10; // Tension min
float t=0; // Temps
float Div=256; // taille d'un octet plus 1 bit

void Voltage_exponentielle_croissante()
{
    V_analogique = (10*exp(0.0575646*t)-V_min)/(V_max-V_min); // Croissance exponentielle du voltage entre 0 et 1
    V_numerique = V_analogique*2^20; // Conversion en numérique
    t=t+1; // incrémentation du temps
}

void Voltage_exponentielle_decroissante()
{
    V_analogique = (10*exp(0.0575646*t)-V_min)/(V_max-V_min); // Croissance exponentielle du voltage entre 0 et 1
    V_numerique = V_analogique*2^20; // Conversion en numérique
    t=t-1; // décrémentation du temps
}

void Repartition_donnees_numeriques() // décomposition de V_numerique dans une base de 3 octets
{
    // Divisions euclidiennes

    Q_1 = V_numerique/Div;
    R_1 = V_numerique - (Q_1*Div); // reste : octet de poids faible

    Q_2 = Q_1/Div;
    R_2 = Q_1 - (Q_2*Div); // reste : octet de poids moyen

    Q_3 = Q_2/Div;
    R_3 = Q_2 - (Q_3*Div); // reste : octet de poids fort
}

```

```

void Envoi_donnees_CNA()
{
    Spi_HT.write(CMD_DIR); // on prévient le registre concernant le voltage qu'il va recevoir des commandes
    wait(0.001); // attente que l'information soit correctement envoyée
    Spi_HT.write(R_3); // envoi octet de poids fort
    wait(0.001); // attente que l'information soit correctement envoyée
    Spi_HT.write(R_2); // envoi octet de poids moyen
    wait(0.001); // attente que l'information soit correctement envoyée
    Spi_HT.write(R_1); // envoi octet de poids faible
    wait(0.001); // attente que l'information soit correctement envoyée

}

void Init_DAQ()
{
    cs=1; // désélection de l'appareil vers lequel on va communiquer pour configurer le mode de transition
    Spi_HT.frequency(1000000); // Configuration du signal horloge à 1MHz
    Spi_HT.format(20,3); // Configuration en transfert 20 bit de données, mode 3 : phase = 1 & polarité =1
    cs=0; // Sélection de l'appareil avec lequel on va communiquer
    Spi_HT.write(CMD_RB); // on prévient le registre concernant les commandes que l'on va les modifier
    wait(0.001); // attente que l'information soit correctement envoyée
    Spi_HT.write(CMR_1); // envoie commandes n°1
    wait(0.001); // attente que l'information soit correctement envoyée
    Spi_HT.write(CMR_0); // envoie commandes n°2
    wait(0.001); // attente que l'information soit correctement envoyée

}

int main()
{
    Init_DAQ();

    while(1) // boucle infinie
    {
        t=0;

        while(t<=120)
        {
            Voltage_exponentielle_croissante(); // calcul de valeur numérique à envoyer au CNA
            Repartition_donnees_numeriques(); // répartition de V_numérique en 3 paquets d'octets
            Envoi_donnees_CNA(); // envoi des 3 paquets au CNA
            led3= !led3; // changement d'état de la led3 pour vérifier le bon fonctionnement du programme
            wait(0.02);

        }

        wait(0.5) ;

        while(t>=1)
        {
            Voltage_exponentielle_decroissante(); // calcul de valeur numérique à envoyer au CNA

```

```

Repartition_donnees_numeriques(); // conversion de V_numérique en 3 paquets d'octets
Envoi_donnees_CNA(); // envoi des 3 paquets au CNA
led3= !led3; // changement d'état de la led3 pour vérifier le bon fonctionnement du programme
wait(0.02);

}
}

} // end main()

```

2. Contrôle haute tension avec la sortie analogout

```

#include "mbed.h"
#include "stdio.h"
#include "math.h"

DigitalOut led1(LED1);
DigitalOut led2(LED2);
DigitalOut led3(LED3);
DigitalOut led4(LED4);

AnalogOut V_HT(p18); // Sortie analogique
int V_max=10000; // Tension max
int V_min=10; // Tension min
int t=0; // Temps
float V_analogique=0; // Tension analogique

void Voltage_exponentielle_croissante()
{
    V_analogique = (10*exp(0.0575646*t)-V_min)/(V_max-V_min); // Croissance exponentielle du voltage
    V_HT.write(V_analogique); // envoie voltage sortie analogique
    t=t+1; // incrémentation du temps
}

void Voltage_exponentielle_decroissante()
{
    V_analogique = (10*exp(0.0575646*t)-V_min)/(V_max-V_min); // Croissance exponentielle du voltage
    V_HT.write(V_analogique); // envoie voltage sortie analogique
    t=t-1; // décrémentation du temps
}

int main() {

while(1)
{
    t=0; // initialisation du temps
    V_HT.write(0); // initialisation voltage nul
    wait(0.5);

    while(t<=120)
    {
        Voltage_exponentielle_croissante(); // calcul et envoie tension exponentielle
        led3= !led3; // changement d'état de la led3 pour vérifier le bon fonctionnement du programme
        wait(0.5);

    }
}

```

```

while(t>=1)
{
    Voltage_exponentielle_decroissante(); // calcul et envoie tension exponentielle
    led3= !led3; // changement d'état de la led3 pour vérifier le bon fonctionnement du programme
    wait(0.5);
}

V_HT.write(0); // retour au voltage nul (précaution)
}

```

3. Contrôle de la pompe et des ventilateurs de l'échangeur de chaleur

```

#include "mbed.h" // Librairie mbed
#include "stdio.h"
#include "math.h"
#include <PwmOut.h> // Librairie pour l'utilisation d'une sortie PWM
#include <time.h> // Librairie pour les fonctions de temps
#include "rtc_time.h" // Librairie pour les fonctions de temps

#define MAXAD 30 // définition de la valeur MAXAD (nombre de valeurs du tableau d'intégration du PID)

float ad[MAXAD]; // initialisation du tableau ad[] de taille MAXAD

// Initialisation des variables LED

DigitalOut led1(LED1);
DigitalOut led2(LED2);
DigitalOut led3(LED3);
DigitalOut led4(LED4);

// Initialisation des variables PWM et Analogiques

PwmOut pwm_1(p23); // PWM régulation débit
PwmOut vent_1(p24); // PWM ventilateur 1
PwmOut vent_2(p26); // PWM ventilateur 2
AnalogIn ad4_in(p19); // Pdiff_2 (Débit Sheath flow)

float V_cons=1.48; // Tension consigne que l'on définit à l'aide de l'étalonnage
float V_mes; // Tension que l'on mesurera à l'aide du capteur
float D_err; // Erreur entre consigne et capteur
float D_err_int; // Terme d'intégration par mesure
float D_err_int_tot; // Terme d'intégration total
float Kp = 0.5; // Constante de proportionnalité
float Ki = 0.1; // Constante de d'intégration
int main()
{

    int i ; int j;

    pwm_1.period(0.001); // Choix de la période de PWM du débit de 1ms
}

```

```

vent_1.period(0.001); // Choix de la période de PWM ventilateur 1 de 1ms
vent_2.period(0.001); // Choix de la période de PWM ventilateur 2 de 1ms

vent_1=0.7; // Tension moyenne ventilateur 1 (8.4V)
vent_2=0.7; // Tension moyenne ventilateur 2 (8.4V)

for (i=1;i<= MAXAD;i++) ad[i]=0; //initialisation du tableau des valeurs du terme d'intégration

i=1;

while(1) // Boucle infinie
{
    if (i>MAXAD) {i=1;led2=!led2;} // Réinitialisation du tableau car on ne peut pas intégrer sur toute le
    temps de la mesure pour des raisons de mémoire et car 30 valeurs suffisent

    D_err_int_tot = 0 ; // Initialisation de l'erreur intégrale
    V_mes = ad4_in.read()*3.3; // Lecture de la tension du capteur sur la voie analogique 4
    D_err = V_cons - V_mes; // Calcul erreur proportionnelle
    D_err_int = D_err*2; // Calcul erreur intégrale, dt = 2s car on considère que le temps entre 2 mesure
    est de 2 secondes
    ad[i] = D_err_int; // On remplit le tableau contenant les erreur intégrales
    for (j=1;j<=MAXAD;j++) D_err_int_tot = D_err_int_tot + ad[i]; // Calcul erreur intégrale totale
    pwm_1 = (V_mes + (Kp*D_err) + (Ki*D_err_int_tot))/3.3; // Calcul duty cycle à émettre et envoie
    commande
    i = i + 1; // Incrémentation du compteur de valeur dans le tableau
    led1 = !led1; // Signal lumineux de fonctionnement
    wait(2);
}
}

```

4. Programme pour le contrôle de tout l'instrument

```

//DAQ SMPS
//Stage Bruno Dato 2014
//2014.03.19;

#include "mbed.h" // Librairie mbed
#include "stdio.h"
#include "math.h"
#include <time.h> // Librairie pour inclure les fonctions de temps
#include "rtc_time.h" // Librairie pour inclure les fonctions de temps
#include "PwmIn.h" // Librairie pour inclure les entrées PWM pour l'humidité

#define SAMPLEFREQ_PID 2 // période (en secondes) de régulation du débit
#define SAMPLEFREQ_HT 0.1 // période (en secondes) du contrôle Haute Tension
#define CONS_DEBIT 1.35 // Tension consigne pour la régulation du débit
#define MAXAD 30 // quantité de valeurs dans le tableau d'intégration du PID
#define T_VENT 0.7 // Tension moyenne ventilateurs
#define KP 0.5 // Constante Proportionnalité PID
#define KI 0.1 // Constante Intégration PID
#define V_MAX 10000 // Tension max colonne
#define V_MIN 10 // Tension min colonne

// Variables globales

```

```

Ticker daq_1; // variable d'interruption n°1
Ticker daq_2; // variable d'interruption n°2
Timer dt; // variable de temps (chronomètre)

float V_analogique; // Tension de commande que l'on calcule pendant le PID
float ad[MAXAD]; // tableau des valeurs du tableau d'intégration
float Kp = KP; // assignation constante de proportionnalité
float Ki = KI; // assignation constante d'intégration
int tab_int ; // Compteur tableau intégration PID
float t_exp_c; // Compteur temps croissant contrôle HT
float t_exp_d; // Compteur temps décroissant contrôle HT
int V_max=V_MAX; // assignation tension max
int V_min=V_MIN; // assignation tension min

```

//Initialisation des variables LED

```

DigitalOut led1(LED1);
DigitalOut led2(LED2);
DigitalOut led3(LED3);
DigitalOut led4(LED4);

```

// Initialisation des variables PWM

```

PwmIn pwm_hum(p21); // Humidité du sheath flow
PwmOut pwm_1(p23); // PWM régulation débit
PwmOut vent_1(p24); // PWM ventilateur 1
PwmOut vent_2(p26); // PWM ventilateur 2

```

// Initialisation des variables analogiques

```

AnalogIn ad0_in(p15); // Tension dans la colonne
AnalogIn ad1_in(p16); // Pression absolue dans le sheath flow
AnalogIn ad2_in(p17); // Température du sheath flow
AnalogOut V_HT(p18); // Contrôle Haute Tension
AnalogIn ad4_in(p19); // Pdiff_2 (Sheath flow)
AnalogIn ad5_in(p20); // Courant dans la colonne

```

```

void Init() // Vérification que le programme se lance
{
    int i;
    for (i=1; i<10; i++) {
        led1 = 1;
        wait(0.02);
        led1 = 0;
        led2 = 1;
        wait(0.02);
        led2 = 0;
        led3 = 1;
        wait(0.02);
        led3 = 0;
        led4 = 1;
        wait(0.02);
        led4 = 0;
        wait(0.02);
    }
}

```

```

led1=led2=led3=led4=0;
wait(0.5);

} //Fin

void Test() // Permet de faire un coupure avec de signaux lumineux pour savoir où l'on en est dans le programme
{
    int i;
    for (i=1; i<2; i++) {
        led1 = 1;
        wait(0.03);
        led1 = 0;
        led2 = 1;
        wait(0.03);
        led2 = 0;
        led3 = 1;
        wait(0.03);
        led3 = 0;
        led4 = 1;
        wait(0.03);
        led4 = 0;
        wait(0.03);
    }
}

led1=led2=led3=led4=0;
wait(0.1);

} //Fin

void Init_PWM_PID_HT() // Initialisation PWM et PID et Contrôle HT
{
    int i;

    pwm_1.period(0.001); // Choix de la période de PWM débit de 1ms
    vent_1.period(0.001); // Choix de la période de PWM ventilateur 1 de 1ms
    vent_2.period(0.001); // Choix de la période de PWM ventilateur 2 de 1ms

    vent_1=T_VENT; // assignation tension moyenne ventilateur 1
    vent_2=T_VENT; // assignation tension moyenne ventilateur 2
    pwm_1=1; // Ouverture électrovanne dans le cas où l'en utilise une dans le circuit du sheath flow avec une pompe que l'on ne commande pas

    for (i=1; i<= MAXAD; i++) ad[i]=0; //initialisation tableau erreur intégration
    tab_int=1; //Initialisation du compteur du tableau d'intégration
    t_exp_c=0; //Initialisation du compteur de temps de la croissante exponentielle de la tension
    t_exp_d=120; //Initialisation du compteur de temps de la décroissance exponentielle de la tension
}

} //Fin

void Control_PID() // Control du débit (sheath flow)
{
    float V_mes; // Tension mesuré par le capteur de pression différentielle (proportionnel au débit)
    float D_err; // Erreur entre consigne et capteur
}

```

```

float D_err_int; // Terme d'intégration par mesure
float D_err_int_tot; // Terme d'intégration comprenant toutes valeurs du tableau
int j;

// On réinitialise régulièrement le tableau du terme d'intégration car cela nécessiterai beaucoup trop de
// mémoire si on intégrait sur toute la durée de la mesure et 30 valeurs sont suffisantes

if(tab_int>MAXAD) {tab_int=1;}
D_err_int_tot = 0 ; // initialisation de l'erreur intégrale totale
V_mes = ad4_in.read()*3.3; // Lecture de la tension du capteur sur la voie analogique 4
D_err = CONS_DEBIT - V_mes; // Calcul de l'erreur
D_err_int = D_err*2; // Calcul erreur intégrale (le temps est de 2 seconde car on lance le programme de
// contrôle de débit toutes les 2 secondes)
ad[tab_int] = D_err_int; // On remplit le tableau contenant les erreur intégrales
for(j=1; j<=MAXAD; j++) D_err_int_tot = D_err_int_tot + ad[j]; // Calcul de erreur intégrale totale
pwm_1 = (V_mes + (Kp*D_err) + (Ki*D_err_int_tot))/3.3; // Calcul du duty cycle à émettre et envoie
// commande
tab_int = tab_int + 1; // Compteur de remplissage des valeurs du tableau d'intégration
led2 = !led2; // Signal lumineux pour montrer que le contrôle du débit est en fonctionnement

} // Fin

void Control_HT() // Contrôle de l'alimentation haute tension
{
    if ((t_exp_c>119)&&(t_exp_d<0)) // Réinitialisation des temps une fois qu'une croissance et une
// décroissances ont été réalisées
    {
        t_exp_c=0;
        t_exp_d=120;
    }

    if(t_exp_c<=119) { // Exponentielle croissante
        V_analogique = (10*exp(0.0575646*t_exp_c)-V_min)/(V_max-V_min); // Croissance exponentielle du
// voltage
        V_HT.write(V_analogique); // envoie voltage sur sortie analogique
        t_exp_c = t_exp_c + 0.1; // incrémentation du temps de croissance
        led3= !led3; // Signal lumineux de croissance exponentielle
    }

    if((t_exp_d>=0)&&(t_exp_c>119)) { // Exponentielle décroissante
        V_analogique = (10*exp(0.0575646*t_exp_d)-V_min)/(V_max-V_min); // Croissance exponentielle du
// voltage
        V_HT.write(V_analogique); // envoie voltage sur sortie analogique
        t_exp_d = t_exp_d - 0.1; // décrémentation du temps de décroissance
        led4= !led4; // Signal lumineux de décroissance exponentielle
    }
}

} // Fin

int main()
{
    Init(); // Lancement du programme

    Init_PWM_PID_HT(); // Initialisation des variables utiles aux sous programmes

```

```

Test(); // Signal lumineux attestant que les initialisations sont effectuées

dt.start(); // lancement du chronomètre

daq_1.attach(&Control_HT,SAMPLEFREQ_HT); // Interruption toutes les 0.1s pour lancer le contrôle
de la haute tension

daq_2.attach(&Control_PID,SAMPLEFREQ_PID); // Interruption toutes les 2s pour contrôler le débit du
sheath flow

while(1) // Boucle infinie qui envoie un signal lumineux toutes les 0.2s
{
    led1=!led1;
    wait(0.2);
}

} // END

```

Résumé

Dans l'atmosphère, de nombreux phénomènes naissent dans la couche limite (entre 0 et 1500m d'altitude). En effet, la formation et l'évolution de brouillards et le parcours d'aérosols dans l'atmosphère sont des thèmes qui intéressent beaucoup les chercheurs. C'est pourquoi de nombreuses mesures se font dans la basse atmosphère. Cependant, pour des raisons de sécurité et d'accès, les avions ne peuvent pas faire de mesure en dessous de 500m d'altitude, c'est ainsi qu'est né le projet VOLTIGE qui vise à développer des drones instrumentés pour la recherche météorologique afin de faire des mesures beaucoup plus précises dans la très basse atmosphère.

Ce rapport présente les travaux de miniaturisation d'un granulomètre, un instrument de mesure permettant de calculer la concentration en particules d'aérosols en fonction de leur taille. Il contient une présentation brève des travaux effectués au CNRM (Centre National de Recherches Météorologiques), une explication plus détaillée des intérêts de la recherche sur les aérosols, une description du fonctionnement de l'appareil de mesure SMPS (Scanning Mobility Particle Sizer) et le déroulement du processus de miniaturisation de ce dernier.

Abstract

In the atmosphere, many phenomena arise in the boundary layer (between 0 and 1,500m high). Indeed, the creation and the evolution of mists and the journey of aerosols in the atmosphere are big issues for the scientific community. That's why several measurements are made in the lower atmosphere. However, for access and safety reasons, planes can't make any measurements below 500m, thus, the project VOLTIGE came to life, its goal is to develop instrumented drones for the meteorological research in order to make more accurate measurements in the very low atmosphere.

This report presents the miniaturization work of a particle sizer, an instrument able to count the concentration in aerosols particles based on their size. It contains a brief presentation of the work done at the CNRM (Centre National de Recherches Météorologiques), a more detailed explanation of the interests of aerosol research, a description of how the SMPS (Scanning Mobility Particle Sizer) works and the process flow of the device miniaturization.