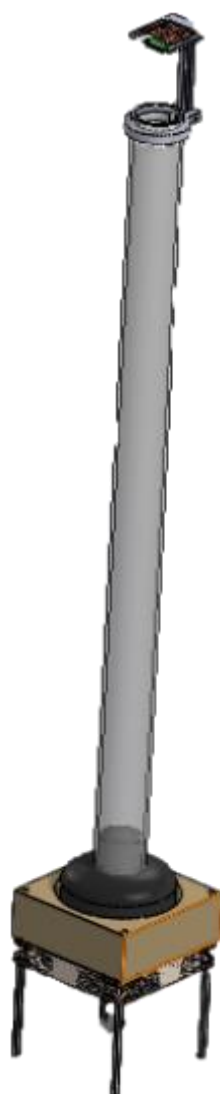


ETML

# Pre-TPI Démonstrateur PID

Rapport R&amp;D



Thomas Blatti  
06/03/2023



## Table des matières

1	Cahier des charges .....	3
2	Planification .....	3
3	Analyse.....	3
3.1	Analyse de l'existant.....	3
3.2	Prise de décisions .....	4
4	Choix des composants principaux.....	4
5	Mécanique .....	5
6	Schémas.....	7
6.1	Dimensionnement de composants .....	11
7	Programmation .....	12
7.1.1	Paramétrages du $\mu$ C .....	12
7.1.2	Astuces de codage.....	12
8	Évaluation du projet.....	13
8.1	État d'avancement du projet .....	13
8.2	Travaux restants à effectuer .....	13
9	Conclusion.....	13
Annexe A	Planification.....	14
Annexe B	Documents de production .....	14
	Schémas→.....	14
Annexe C	Data Sheets .....	14
	Data Sheets .....	14

## Versions

Version	Date	Auteur	Remarques
00	2023-03-06	TBI	Version initiale

## 1 Cahier des charges

Dans le cadre du cours d'électronique analogique, les élèves sont amenés à étudier la régulation en boucle fermée d'un système asservi. Par exemple, une résistance de chauffage maintient une certaine température (consigne) dans une cuve grâce à la mesure de la température (mesure). La dissipation thermique refroidit la cuve (erreur) et le système régule le courant dans la résistance afin de maintenir la température.

Dans le cadre de ce projet (N°1391), l'idée est de maintenir une balle à une hauteur définie par la consigne dans un tube placé verticalement. La position de la balle sera réglée par un clapet permettant d'amener un flux d'air depuis un ventilateur.

Ce rapport contiendra l'ensemble du travail du pre-TPI afin de documenter les recherches et réalisations,

## 2 Planification

Liste des tâches	Durée Planifiée	Durée Réalisée
Planifier	2h	1h30
Décider	11h	13h50
Réalisation 3D	16h	5h30
Schéma bloc	3h	→
Schéma électrique	5h	12h50
Total →	37h	34h
Erreur Relative→	-8%	

Planification et journalisation complète, voir Planification

## 3 Analyse

### 3.1 Analyse de l'existant

Documents :

- Cahier des charges → TPI 2023 - Blatti - CdC pré-TPI.pdf
- Calendrier approximatif → TPI 2023 - Blatti - Calendrier approximatif.pdf
- Coordonnées experts → Coordonnées experts.pdf
- Ensemble de fichiers du projet 1138 qui est semblable a mon projet.

Logiciel :

- Altium designer
- Solidworks
- Suite Office

### **3.2 Prise de décisions**

Concernant ce TPI j'ai abordé la chose de la manière suivante :

Ce projet ressemble de près à ce que pourrait réaliser un travail d'ingénieur. En effet partir de « rien » et de concevoir le système de A à Z. En se référant à la planification la première partie consiste en décider des composants principaux du système. Une fois les composants principaux choisis je ferai la partie mécanique ou du moins la réalisation 3D Du montage final. Et en dernier une fois que toutes les composants principaux et la partie mécanique finie, je réaliserai les schéma de la carte de commande.

## **4 Choix des composants principaux**

Le premier composant est le ventilateur. Il responsable de fournir une ventilation assez puissante pour soulever un objet assez léger certes mais avec un poids non négligeable. Pour le choix de ce dernier, il s'agissait d'un « saut dans le vide ». En effet n'étant pas un connaisseur j'ai pris un premier ventilateur qui me semblait faire l'affaire, le « Noctua NF-R8 redux 1200 3-PIN » comme précise dans son nom il atteint un maximum de 1200 tours minutes pour un ventilateur de 80mm de diamètre.

Malheureusement son flux d'air d'environ 35m<sup>3</sup>/h ne suffisait pas du tout. Le but de ce ventilateur est de la piloter en PWM afin de régler la hauteur de l'objet dans le tube, si le ventilateur ne peut même pas soulever l'objet à pleine puissance il est clairement écarté des choix possibles. Le ventilateur qui sera utilisé est un autre ventilateur de chez Noctua ayant un flux d'air et un diamètre plus grand.

Le deuxième composant essentiel est le capteur de distance. Le composant sélectionné est le suivant : Le capteur IR distance Click de chez Mikroe. Ce capteur nous permet de faire une mesure précise entre 0 et 80 cm. Grâce à ce capteur je peux déterminer la hauteur de l'objet dans le tube. Bien que le capteur présent sur le petit PCB soit obsolète étant donné sa taille et sa précision je ne me voyais pas changer de capteur.

Le troisième composant nécessaire est le servomoteur. Ce composant permettra d'ouvrir une portion du tube pour contrôler l'échappement du flux d'air. Le servo moteur utilisé pour mes tests et pour la version finale du projet sera sûrement un SG92R, un Servo moteur basique permettant de contrôler précisément l'ouverture du tube.

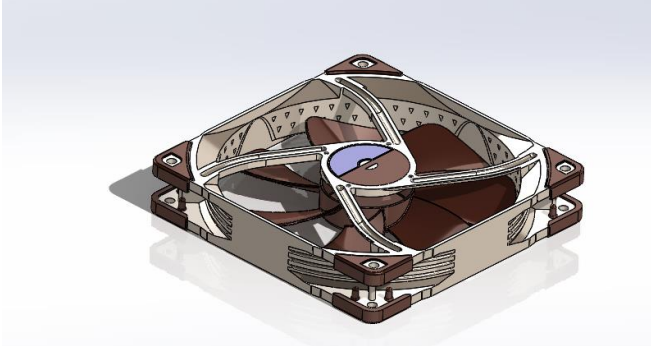
Tous ces composants ont été testés avec de petits bouts de code afin de me familiariser à ces derniers. Cette partie programmation est décrite plus loin dans le rapport.

## 5 Mécanique

Une fois les composants principaux testé et choisis il me fallait travailler un peu sur la partie mécanique et la conception du projet. J'ai donc décider de réaliser en 3D le projet afin de me donner une idée de comment allait être fixer tous les composants

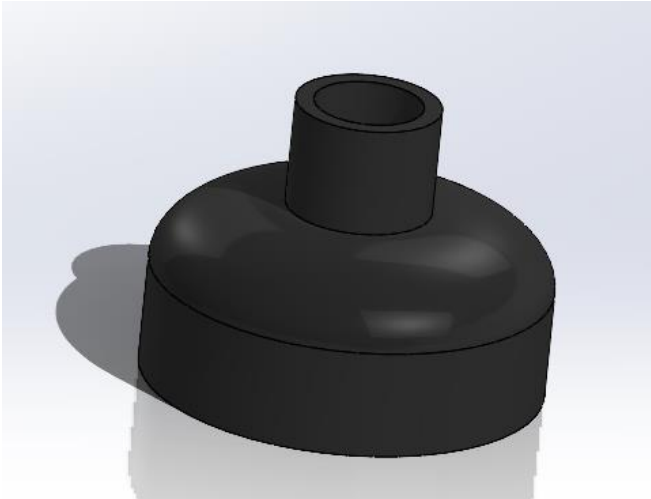
Principaux entre eux. Le design 3D effectuer sur SolidWorks est le suivant :

Ventilateur (design non-effectué par mes soins)



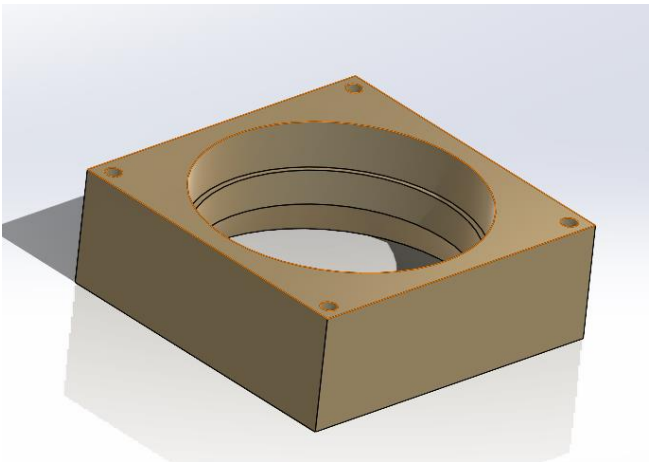
Ce ventilateur le Noctua NF-A14 PPC 3000 est un ventilateur de 140 mm de diamètre nous permettant d'avoir un flux d'air assez conséquent pour soulever l'objet à travers le tube.

Guide d'air :



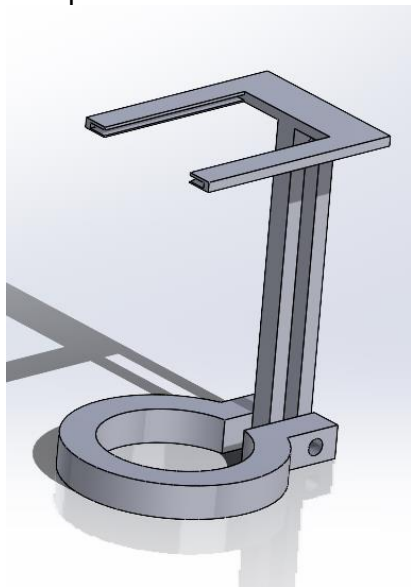
Piece permettant de guider l'air et de le concentrer a travers le tube. Il se fixe a la pièce de raccord.

Piece de raccord :



Piece convertissant le diamètre du ventilateur au diamètre du guide d'air. De 140mm a 120mm.

Pièce de support pour  
le capteur de distance :



Cette pièce permet de tenir le capteur de distance en hauteur afin de viser l'intérieur du tube. La bague viendra se serrer autour du tube et l'oblong dans la colonne permet de régler la hauteur du capteur.

Une fois la modélisation 3D elle ressemble à la photo présente en couverture de ce rapport.

## 6 Schémas

La première feuille ci-dessous est celle du schéma bloc. Elle comporte la listes des feuilles du schéma et des interactions entres elles. Dans ce schéma bloc se trouve également les symboles de normes respectée pour la production de la carte et de son entretiens, RoHS, ESD, WEEE. Il y a 4 points test de GND pour pouvoir faire plusieurs mesures en simultané ainsi que 4 trous non-métallisés pour fixer la carte sur son support.

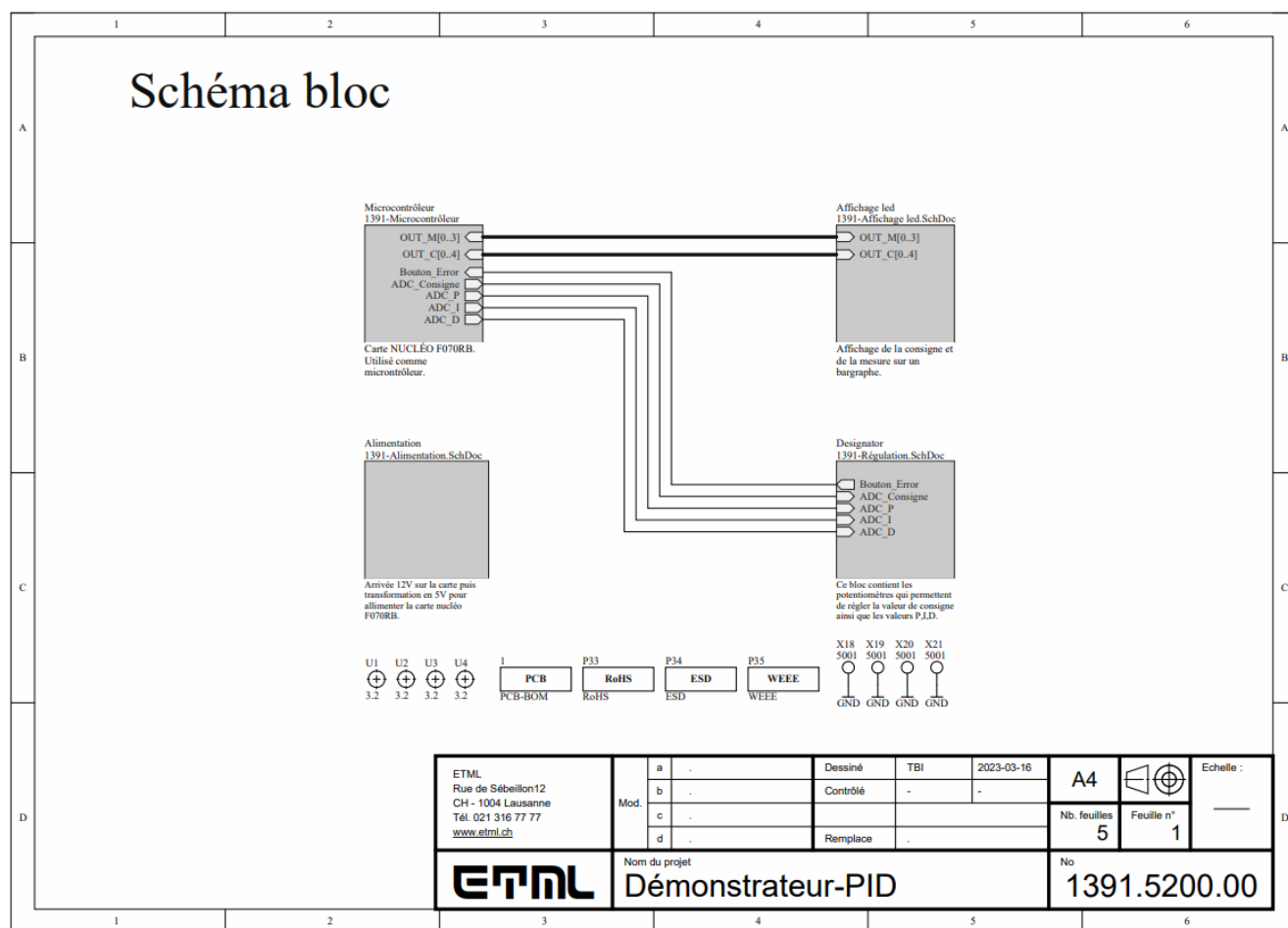
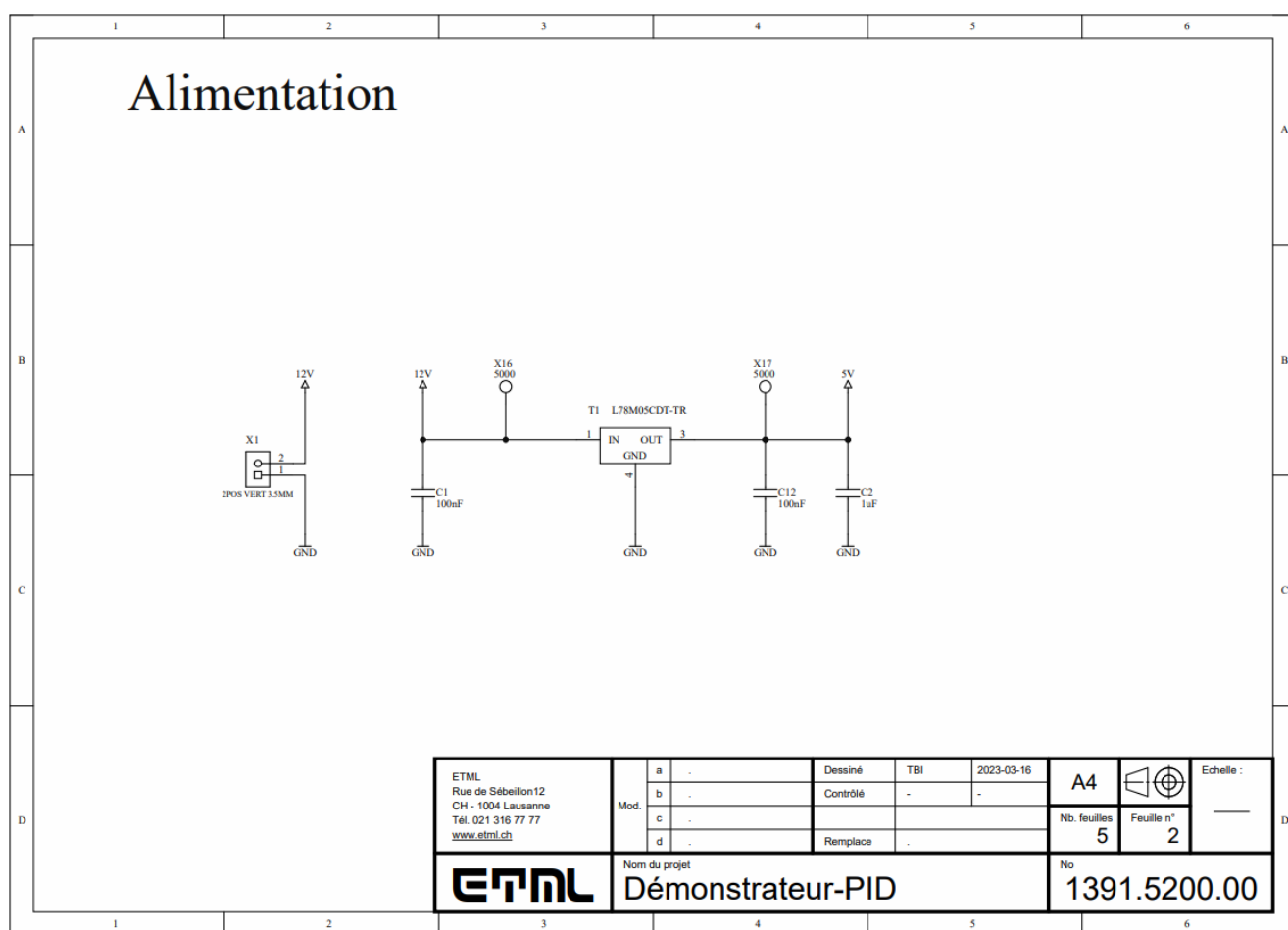


Schéma Bloc

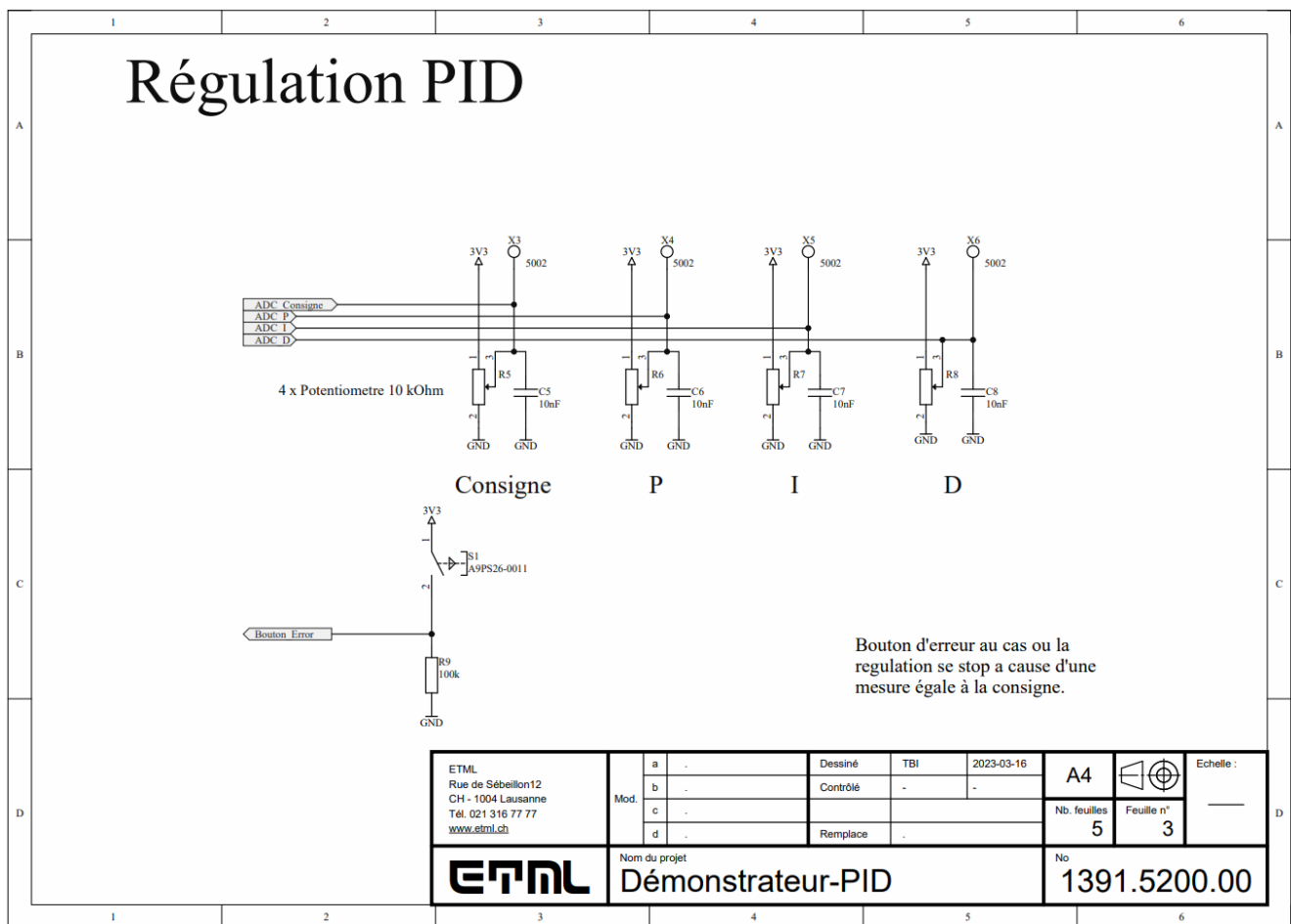


Ensuite le bloc le plus simple du projet et celui de l'alimentation, en effet la carte est alimentée en 12V par un transformateur 230V-12v de chez tracopower pour ensuite être transformée en 5V par un régulateur. Le 12V sera utilisé pour commander le ventilateur, le 5V pour alimenter une bonne partie de la carte ainsi que le NUCLEO F070RB qui lui fournira notre tension 3V3. Les condensateurs sont de valeur normalisée pour ce genre de régulateur.



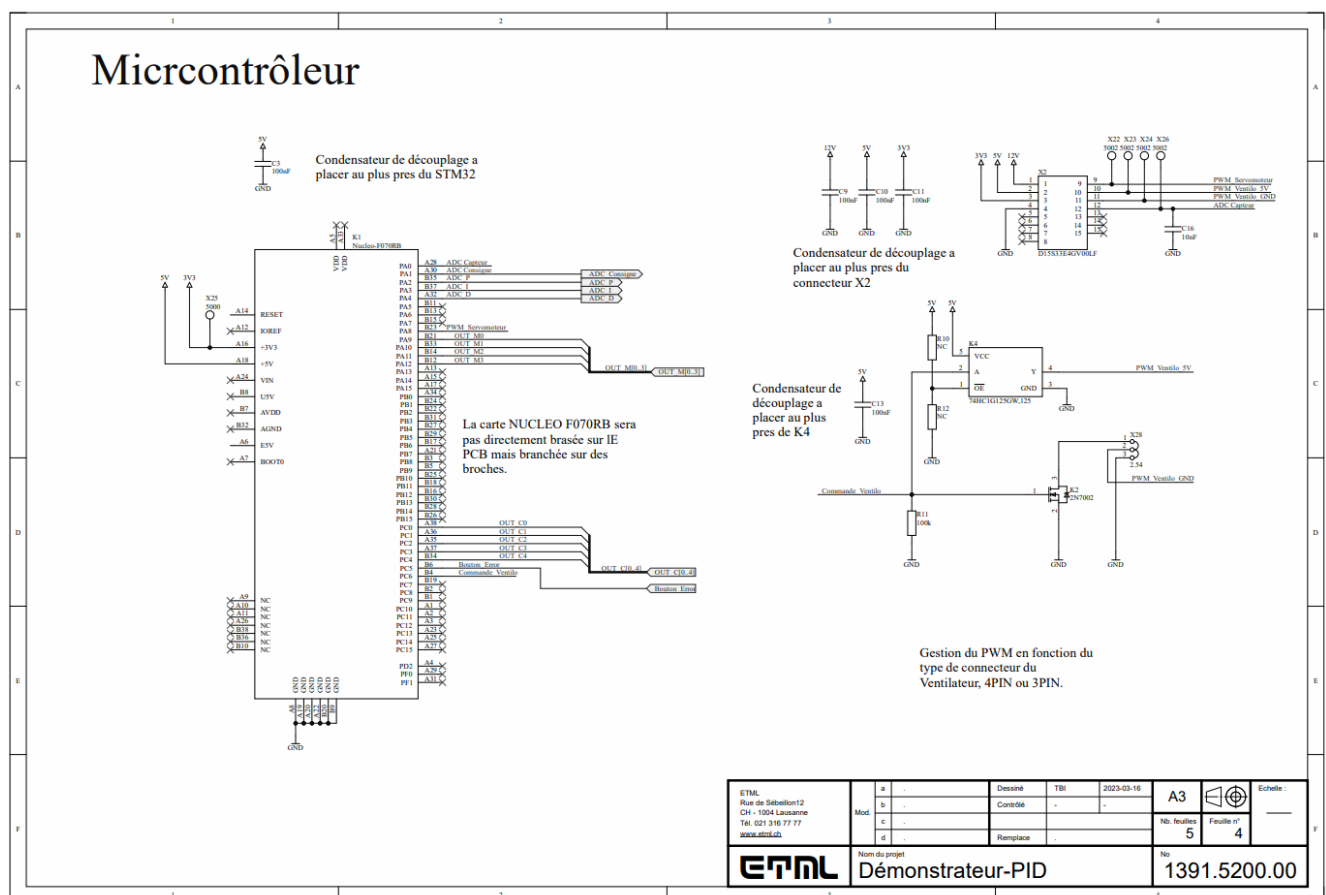
Alimentation

Le schéma de régulation est composé de 4 potentiomètres chacun relié sur un ADC du microcontrôleur afin de mesurer le réglage de la consigne ou des réglages PID. Les condensateurs sont des condensateurs de déparasitage afin de nettoyer le signal arrivant sur l'ADC. Il y a également un bouton au cas ou une erreur se produis qui pourrait générer une interruption.



Régulation PID

La partie microcontrôleur est le cerveau de la carte. Il contrôle les ADC pour la régulation ainsi que le capteur de distance, les sorties pour l'affichage ainsi que le PWM du servomoteur et du ventilateur. Pour la commande du ventilateur j'ai laissé la possibilité de choisir entre deux types de ventilateur, 3PIN ou 4PIN de chez Noctua. Pour cela il me fallait mettre un jumper pour la sélection ainsi qu'un buffer en open collector afin de pouvoir piloter le ventilateur de la bonne manière comme le recommande Noctua. Le connecteur pour aller jusqu'au différents périphériques est un connecteur SUB-D25.



**Affichage LEDs**

Représentation graphique de la consigne. LED RD x 16

Représentation graphique de la mesure. LED GN x 16

Condensateur de découplage à placer au plus près du connecteur X2

### Affichage LEDs

Le dimensionnement de composant a été plutôt simple en effet il ma suffit de dimensionner seulement les résistances de limitations de courant pour les 32 leds de l'affichage.

$$R_{LED} = \frac{U_{Rled}}{I_{LED}} = \frac{U_{REF} - U_{LED}}{I_{LED}} = \frac{5 - 2}{15 * 10^{-3}} = 200 \text{ Ohms} \xrightarrow{E12} 200 \text{ Ohms}$$

## 7 Programmation

Dans ce rapport la partie programmation ne concerne pas la carte de commande mais les tests réalisés sur breadboard avec une carte NUCLÉO F070RB (La même qu'utilisé sur la carte commande).

### 7.1.1 Paramétrages du $\mu$ C

Pour les paramètres du microcontrôleur j'ai utilisé une entrée ADC pour avoir les données de mon capteur de distance et deux sorties PWM, une pour contrôler le ventilateur et une autre pour contrôler le servo moteur.

### 7.1.2 Astuces de codage

J'ai donc testé mes composants de la manière suivante :

Cette capture d'écran ci-dessous permet de voir comment est utilisé mon capteur de distance.

```
HAL_ADC_Start(&hadc); //ADC start
HAL_ADC_PollForConversion(&hadc, HAL_MAX_DELAY); //conversion
valeur_adc = HAL_ADC_GetValue(&hadc); //GET ADC VALUE
if(valeur_adc >= 2650) //définition des valeur min et max de distance
{
    distance = 10;
}
else
{
    //Calcul distance
    distance = (2e-12 * powf(valeur_adc, 4)) - (2e-08 * powf(valeur_adc, 3)) + (7e-05 * powf(valeur_adc, 2)) - (0.1096 * valeur_adc) + 80.802;
}

snprintf(uart_buffer, UART_BUFFER_SIZE, "La distance est %2.f cm\r\n", distance); //enregistrement message affichage distance
HAL_UART_Transmit(&huart2, (uint8_t*)uart_buffer, strlen(uart_buffer), HAL_MAX_DELAY); //affichage sur terminal serie
```

Cette capture d'écran ci-dessous permet de voir comment est utilisé le servo moteur.

```
//conversion de la valeur ADC en un rapport cyclique pour définir l'angle du servo moteur en fonction de la distance
D = -2E-08 * powf(valeur_adc, 3) + 1E-04 * powf(valeur_adc, 2) - 0.1601 * valeur_adc + 124.74;
if(D >= 125)
{
    D = 125;
}

if(D <= 25)
{
    D = 25;
}
htim1.Instance->CCR1 = D;
HAL_Delay(1000);
```

Pour la gestion du servo moteur la durée du rapport cyclique varie entre 0,5 et 2,5 ms il suffit de paramétrer la fréquence du PWM de la bonne manière et varier le rapport cyclique pour faire bouger le servo moteur.

Pour la gestion du ventilateur le PWM est utilisé pour faire varier le rapport cyclique d'un signal sur la grille d'un mosfet connecté au ventilateur pour faire varier sa vitesse de rotation.

## **8 Évaluation du projet**

### **8.1 État d'avancement du projet**

Les schémas sont terminés les composants ont été sélectionnés. Le cahier des charges du pré TPI est complété :

- ☒ 1) Imaginer à l'aide d'un schéma bloc les types de signaux et l'interface utilisateur ;
- ☒ 2) Déterminer la connectique (et les signaux) entre la carte commande et le démonstrateur (balle dans le tuyau) ;
- ☒ 3) Choisir un microcontrôleur répondant aux différents besoins ;
- ☒ 4) Choisir un ventilateur ainsi que son système de commande (PWM de préférence) ;
- ☒ 5) Choisir un servo-moteur pour le clapet régulant le flux d'air ;
- ☒ 6) Dessiner le schéma sur Altium.

### **8.2 Travaux restants à effectuer**

Le travail restant à effectuer est plutôt logique il s'agit du TPI en lui-même avec toutes les tâches suivantes :

- 1) Réalisation du routage et de la liste de pièce (BOM) ;
- 2) Commander le matériel avant les vacances de Pâques ;
- 3) Monter/braser le circuit imprimé ;
- 4) Mettre en service et tests à l'aide d'un programme de test (n'incluant pas la régulation !).
- 5) Réaliser une ébauche de mode d'emploi.

## **9 Conclusion**

Pour conclure même si le projet est loin d'être fini le cahier des charges pour le pré TPI est complété. J'ai pu me familiariser avec les composants que j'allais utiliser et préparer tout ce qu'il fallait pour le début du TPI, c'est-à-dire les footprints des composants et une bonne partie du design de la mécanique. Au niveau du temps, cela pris plus ou moins le temps estimé en tout, même si le temps estimé pour chaque tâche ne correspondait pas vraiment. Je vais donc faire une planification plus précise pour le TPI afin de mieux gérer mon temps. Ce début de projet c'est très bien passé.

**Annexe A Planification**

→C:\Users\thoblatti\Education Vaud\ETML\_XLO-19-23 - Blatti\Projets\1391-Demonstrateur\_de\_regulation\_PID\_TPI\3-Documents  
Journal de travail → Journal-Blatti-20230323\_1346.pdf  
Planification → Planification-Blatti-20230323\_1345.pdf

**Annexe B Documents de production**

**Schémas** C:\Users\thoblatti\Education Vaud\ETML\_XLO-19-23 - Blatti\Projets\1391-Demonstrateur\_de\_regulation\_PID\_TPI\5-Hardware\1391-Demonstrateur\_PID\Assembly\1391-Démonstrateur\_PID\_SCH.PDF

**Annexe C Data Sheets**

**Data Sheets** C:\Users\thoblatti\Education Vaud\ETML\_XLO-19-23 - Blatti\Projets\1391-Demonstrateur\_de\_regulation\_PID\_TPI\7-DataSheets