

Projet ETML-ES - Cahier des charges

Système de régulation thermique

Projet N° 2226

Entreprise/Client:		Département:	
Demandé par (Prénom, Nom):		Date:	

Auteur (ETML-ES):	Taulant Neziri	Filière:	SLO2
		Date:	16.11.2022

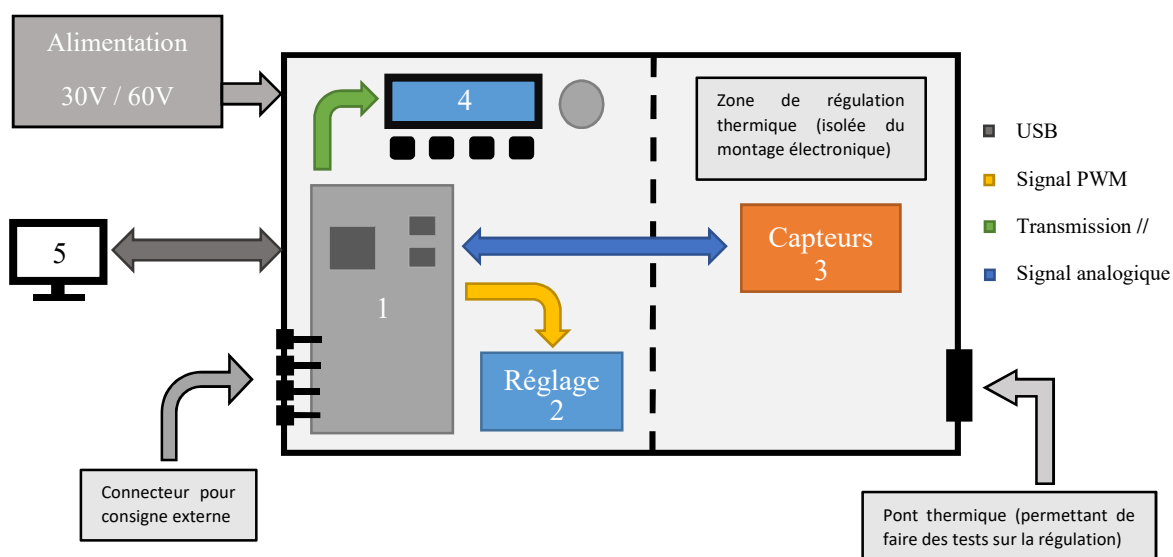
1 But du projet

Concevoir un système de régulation thermique dans un boîtier fermé pour les cours de REGL. L'utilisateur aura deux possibilités de réglages. La première pourra être effectuée via un générateur externe et des touches physiques (bouton, potentiomètre). La deuxième, se fera via USB.

2 Spécifications du projet

Voici les fonctions du projet représentées sous forme d'un schéma bloc.

2.1 Schéma bloc





A l'aide d'un câble USB, l'utilisateur pourra relier le module de régulation à un PC. Ce dernier pourra ensuite configurer les paramètres de régulations à l'aide d'une interface Web graphique.

V : Vacances
R : Remise de documentation/dossier/fichiers. Peut être remis plus tôt.
P : Présentation.
Remise d'un rapport chaque semaine précédant une présentation.

2.3 PCB (bloc 1 et 2)

Mode 1 : L'utilisateur pourra s'il le désire fournir une consigne via à un générateur de fonction externe ou directement paramétrer la période, la forme et la valeur crête de la consigne générée en interne par le PIC32. La régulation se fera dans ce même PIC32 en fonction de la valeur mesurée par l'un des capteurs (voir bloc 2), l'utilisateur pourra paramétrer le gain des trois composantes PID (K_p , K_i , K_d) via le bloc 4 afin de modifier la commande en sortie du μC .

Mode 2 : La consigne et le processus de régulation se feront via un script en python. Le PC recevra la valeur mesurée par l'un des capteurs (voir bloc 2) via USB et fournira la commande au système. Dans ce mode, le PIC32 est utilisé uniquement pour de l'affichage (voir bloc 4).

Mode 1 et mode 2 : Les blocs 1 et 2 feront varier la puissance dissipée (via la commande) au travers de la résistance de puissance pour modifier la température ambiante. La résistance de puissance sera dissipée dans la partie isolée du boîtier. Le débit d'air pourra être modifié par l'utilisateur via le bloc 4 en mode 1.

2.4 Capteur (bloc 3)

Mode 1 et mode 2 : Le bloc capteur viendra mesurer la température produit par la résistance de puissance via deux sondes de température. L'une se trouvera à l'extrémité gauche du tube et l'autre sera près de la résistance de puissance. La valeur mesurée par une des sondes sera transmise soit au myRIO soit au PIC32 dépendant du mode.

2.5 Affichage + Réglage (bloc 4)

Mode 1 : L'écran pourra afficher différents menus gérés par des boutons et un encodeur incrémental. Ils permettront de régler le débit d'air du ventilateur, la consigne fournie par le PIC32 (forme, valeur crête et période du signal), les valeurs K_p / K_i / K_d et la source de la consigne (PIC32 ou générateur de fonction).

Mode 2 : La consigne, la commande et la mesure seront affichées sur un écran.

2.6 E/S

HMI (Human Machine Interface) :

Une sortie analogique pour envoyer la consigne (pour affichage uniquement).

Une sortie PWM pour envoyer la commande.

Une entrée analogique pour recevoir la mesure.

Un port USB pour communiquer avec le PIC32 en USB/UART.

PIC32 :

Une entrée analogique pour recevoir la consigne,

Une entrée analogique pour recevoir la commande.

Une (ou 2 ?) entrée analogique pour recevoir la mesure

Une sortie PWM pour modifier le débit d'air du ventilateur,

Une sortie PWM pour la commande de puissance du corps de chauffe,

Des entrées numériques pour gérer les boutons et le commutateur rotatif.

E/S numériques pour communiquer avec l'écran.

Deux E/S numériques pour communiquer avec le PC en UART -> USB.

3 Tâches à réaliser

1. Concevoir le cahier des charges avec le supérieur technique.
2. Déterminer les composants électroniques et mécaniques du projet
3. Faire un rapport de pré-étude.
4. Concevoir un schéma électrique.
5. Faire les plan 3D sur Solidworks des pièces mécaniques.
6. Router le PCB sur Altium via le schéma électrique préalablement conçu.
7. Concevoir un montage mécanique du projet, usiner ou faire usiner les pièces
8. Commander le PCB, les composants électroniques et les composants mécaniques.
9. Monter les composants électroniques et les pièces mécaniques indispensables.
10. Programmer le PIC32.
11. Programmer l'interface WEB.
12. Tester la carte, mise en service.
13. Monter le reste des composants mécaniques.
14. Tester le montage final dans les 2 modes.

4 Jalons principaux

Voir le planning excel.

5 Livrables

- Les fichiers sources de CAO électronique des PCB réalisés
- Tout le nécessaire à fabriquer un exemplaire hardware de chaque :
- Fichiers de fabrication (GERBER) / liste de pièces avec références pour commande / implantation (prototype) / modifications / dessins mécaniques, etc
- Les fichiers sources de programmation microcontrôleur (.c / .h)
- Tout le nécessaire pour programmer les microcontrôleurs (logiciel ou fichier .hex)
- Le cas échéant, les fichiers sources de programmation PC/Windows/Linux.
- Le cas échéant, tout le nécessaire à l'installation de programmes sur PC/Windows/Linux.
- Un mode d'emploi du système.
- Un calcul / estimation des coûts.
- Un rapport contenant les calculs - dimensionnement de composants - structogramme, etc.

6 Convention de nommage et liens

Le nom de ce fichier doit être unique et doit donc contenir le nom du projet avec le format suivant :

aaii_nomProjet-CDC_Vn.docx

Avec :

- CDC : pour Cahier des charges
- aaii : numéro de projet, exemple 1708 pour projet de 2017 no 08
- nomProjet : comme son nom l'indique.
- Vn: ou « n » indique la version du document.

Exemple :

- **0910x_PICEthernet-CDC_V1.docx**

6.1 Stockage du fichier

Ce fichier sera stocké à la racine du dossier **/doc** d'un projet.

Ainsi, tous les fichiers de documentation faisant partie du projet sont centralisés dans le même répertoire.