

TISSOT Elise

TIRABY Céline



Rapport de Projet de Fin d'études

Thermostat connecté et intelligent

IMA5

A decorative graphic at the bottom of the page consisting of several overlapping, semi-transparent geometric shapes in shades of blue and grey, creating a layered, architectural effect.

2015-2016

Remerciements

Tout d'abord, nous souhaitons remercier nos tuteurs école : M. Xavier Redon et M. Alexandre Boé, qui par leurs conseils et leur aide apportés, ainsi que leur écoute, nous ont permis d'avancer dans notre projet lorsque nous rencontrions des difficultés.

Nous remercions également notre tuteur industriel M. Guillaume Renault, qui par sa présence hebdomadaire et son suivi quotidien du projet, nous apportait son aide, son avis et ses conseils avisés pour que nous puissions mener à bien le projet.

Nous remercions M. Thierry Flamen, qui a nous a rendues un grand service en soudant par trois fois un composant CMS extrêmement petit et en réalisant nos cartes électroniques. Ses conseils et sa patience nous ont également été d'une très grande aide.

Puis, nous souhaitons remercier M. Abdelkader Benabou, qui nous a apporté son aide, reçues au laboratoire d'électrotechnique et présentées aux chercheurs y travaillant afin qu'ils nous conseillent au mieux.

Nous remercions M. Thierry Duquesne, qui a nous lui-aussi beaucoup aidées en soudant par une fois le petit composant. Ses conseils et son aide apportés nous ont également été d'une grande utilité.

Enfin, nous souhaitons remercier M. Olivier Ferla, qui par sa patience, son expertise, ses conseils et sa gentillesse nous a permis de réaliser au mieux notre projet de fin d'études. Son aide nous aura été très précieuse.

Table des matières

Remerciements	1
Introduction	3
I Cahier des charges.....	4
A) Objectif	4
a. Le thermostat d’ambiance et étude.....	4
b. Fonctionnalités du thermostat.....	5
c. Fonctionnalités “avancées” du thermostat à implémenter	5
B) Les différentes étapes du projet	6
C) Schéma général du projet.....	6
D) Choix technologiques.....	7
a. Utilisation de panStamps	7
b. Composants CMS	7
II Réalisation	8
A) Partie Électronique	8
a. Recherches bibliographiques afin de réaliser les fonctions souhaitées	8
b. Choix des composants	9
c. Réalisation des schémas Altium et routage des cartes	10
d. Routage et réalisation des cartes électroniques.....	11
B) Partie Informatique	17
a. Chaîne de communication des panStamps	17
b. Interface graphique.....	19
C) Conception du boîtier	21
D) Évolutions futures du projet	21
a. Implémentation de la carte “chaudière” en 220V	21
b. Implémentation pour un radiateur	23
III Résultats finaux et perspectives d’amélioration	25
A) Résultats finaux.....	25
B) Problèmes rencontrés et bilan personnel.....	26
C) Comparatif avancée prévisionnelle/avancée réelle	26
a. Avancée prévisionnelle	26
b. Avancée réelle	27
D) Perspectives d’amélioration	27
Conclusion	28
Annexe 1 : consommation du circuit carte 1.....	29
Annexe 2 : liste des composants commandés.....	30
Annexe 3 : Calcul de la résistance thermique du dissipateur pour le triac.....	31

Introduction

Les objets connectés prenant de plus en plus de place dans la vie de tous les jours, notre projet de fin d'études nous permet de rester dans l'ère du temps. En effet, son but est de concevoir et de réaliser un thermostat connecté intelligent.

Ce thermostat devra s'adapter à la tendance mondiale qui est d'automatiser un maximum d'objets au quotidien ainsi que de réduire les coûts énergétiques liés en partie à la consommation des différents composants des cartes électroniques.

Ce projet nous permettra de mettre en pratique des connaissances acquises lors des cours que nous avons suivis durant les trois dernières années, mais est également pour nous une bonne occasion de développer nos compétences et nos connaissances.

Il se composera en deux parties principales qui consistent respectivement en la conception de deux cartes électroniques et la réalisation du programme permettant la communication entre ces dernières.

De plus, nous effectuerons une interface web, ainsi que la conception d'un boîtier à l'aide d'une imprimante 3D qui contiendra l'une des deux cartes électroniques.

I Cahier des charges

A) Objectif

Comme énoncé précédemment, l'objectif général de notre projet est de réaliser un thermostat d'ambiance connecté possédant différentes fonctionnalités.

a. Le thermostat d'ambiance et étude

Un thermostat d'ambiance permet de réguler la température ambiante d'une pièce. Il peut être incorporé dans l'appareil de chauffe comme par exemple une chaudière, ou déporté dans la pièce et être installé en position murale. La plupart du temps de conception numérique, le thermostat d'ambiance peut également être équipé de fonctions de programmation mode occupation ou confort, mode inoccupation ou mode hors-gel.



Cependant, nous pouvons nous demander quel est l'intérêt de concevoir un thermostat connecté. Après avoir fait quelques recherches, nous avons trouvé que 85% des personnes interrogées pour un sondage (sur un échantillon de 1148 personnes) s'intéressaient aux objets connectés. De plus, 76% les trouvent utiles, ce qui est un bon point pour la conception d'un tel projet.

Lorsque la question de l'objet connecté qui semble le plus utile dans une maison est posée, 54,4% citent le thermostat connecté qui se place donc derrière les alarmes (59,7%) et les caméras de surveillance (56,4%). Plus d'une personne sur deux est donc susceptible de s'intéresser à un thermostat connecté puisqu'il est jugé utile.

De plus, lorsqu'on demande quel objet connecté est en possession des personnes interrogées pour le sondage, seulement 7,4% déclarent avoir un thermostat connecté à la maison. Nous pouvons donc penser qu'il existe un marché important pour les thermostats connectés.

Enfin pour terminer cette étude, nous pouvons noter que 79% des Français cherchent à réduire leur consommation d'énergie. Il est donc intéressant d'inclure dans les fonctionnalités du thermostat un moyen permettant la réduction de la consommation d'énergie. En effet, cela pourrait être un facteur qui peut pousser les clients à en acquérir un.

Lien du sondage :

<http://www.lesnumeriques.com/objet-connecte/resultat-sondage-que-pensez-objets-connectes-n46135.html>

b. Fonctionnalités du thermostat

Le thermostat connecté devra donc posséder plusieurs fonctionnalités.

Dans un premier temps, il devra avoir les mêmes fonctionnalités qu'un thermostat classique. Il sera donc capable de capter la température de la pièce ainsi que de la modifier selon les souhaits de l'utilisateur. Une fonctionnalité essentielle du thermostat sera donc l'augmentation ou la diminution d'un certain nombre de degrés de la température par rapport à celle ressentie par le capteur de température. Une autre fonctionnalité du thermostat sera de faire passer la chaudière en mode confort, éco ou hors-gel à partir de l'interface graphique.

De plus, le thermostat connecté sera contrôlable à partir d'une interface web afin de faciliter son utilisation pour le client.

Ensuite, il devra être capable de mesurer les importants écarts de température dus par exemple à l'ouverture d'une fenêtre. Ces anomalies seront détectées par le calcul statistique du temps de chauffe.

Enfin, il sera installé dans un boîtier modélisé fabriqué au Fablab de l'école afin de, cette fois encore, améliorer sa prise en main et donc le confort de l'utilisateur.

Le thermostat devait permettre, en outre, d'effectuer des mesures de la consommation énergétique en Wh et en Euro. Nous avons cependant abandonné cette idée car cela est trop dépendant des caractéristiques de la chaudière ou du radiateur.

Au début du projet, nous avons également l'ambition d'auto-alimenter le thermostat à l'aide d'une énergie lumineuse par exemple. En effet, cela aurait permis de réduire fortement l'usage de la batterie et aurait été une fonctionnalité intéressante. Le prototype doit avoir une consommation minimale qui permet de changer les piles moins souvent pour permettre une charge de maintenance raisonnable. Un thermostat où la nécessité de changer les piles est moins récurrente que la normale attirera plus facilement qu'un thermostat qui nécessite un changement de piles plus fréquent. Malheureusement, nous avons dû abandonner cette idée car il s'est avéré que l'auto-alimentation du thermostat aurait pu être un sujet de projet à lui seul et nous n'aurions donc pas suffisamment de temps pour réaliser cette fonctionnalité supplémentaire.

c. Fonctionnalités "avancées" du thermostat à implémenter

En plus des fonctionnalités énoncées ci-dessus, nous allons par la suite rajouter d'autres fonctions qui valoriseront le projet.

Il pourrait être intéressant de permettre l'affichage de la température actuelle de la pièce sur l'interface graphique ainsi qu'un suivi de la température à l'aide de courbes qui retracent son évolution dans le temps.

Enfin, nous pourrions également inclure des modes de fonctionnement différents selon la période à laquelle on se situe (vacances, période hivernale, etc).

B) Les différentes étapes du projet

Pour réaliser notre thermostat, nous avons découpé le travail en plusieurs parties.



La première consiste en la construction des cartes électroniques, qui sont au nombre de deux. En effet, la première carte correspond au thermostat en lui-même, c'est-à-dire qu'il capte la température de la pièce et envoie les instructions permettant de commander la chaudière à la seconde carte. Cette dernière quant à elle, réalise la commande de la chaudière.

La seconde étape de notre projet consiste à l'établissement d'une communication entre les différentes "parties" du projet. Pour ce faire, nous utilisons les panStamps en RF (radio-fréquences).

La troisième étape réalise l'implémentation des différentes fonctions sur chaque panStamp (par exemple, il peut s'agir de récupérer la température de la pièce)

La quatrième étape consiste en l'élaboration d'une interface web graphique permettant à l'utilisateur de contrôler à distance le thermostat connecté. Cette interface doit être simple d'utilisation, intuitive et réalisée avec un design attractif. Elle sera réalisée à l'aide des langages PHP, HTML, CSS et JavaScript.

Enfin, la dernière étape du projet est la fabrication d'un boîtier pour le thermostat (carte portable).

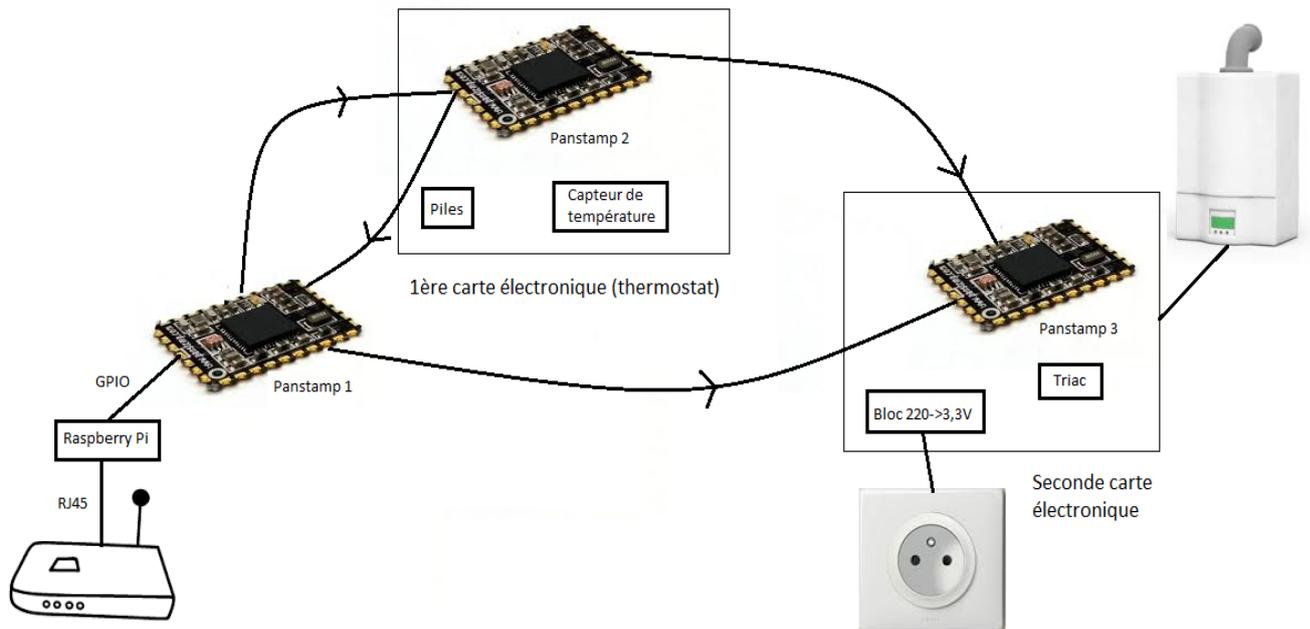
Ce boîtier sera réalisé au Fablab à l'aide d'une imprimante 3D. Le logiciel permettant de modéliser le prototype est Solid Works.

Ce dernier aura pour but de rendre le thermostat plus esthétique et devra ne pas être trop imposant afin de permettre une meilleure prise en main du thermostat pour l'utilisateur.

Une étape supplémentaire consistant à la réalisation d'un prototype permettant de commander un radiateur à la place d'une chaudière peut être ajoutée au projet par la suite.

C) Schéma général du projet

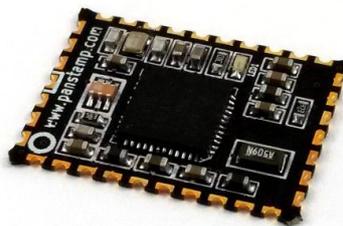
Afin d'avoir une vue plus générale de notre projet, voici un schéma récapitulant les différents éléments ainsi que les communications entre chacun d'entre eux :



D) Choix technologiques

a. Utilisation de panStamps

L'une des particularités de notre projet était l'utilisation obligatoire de panStamps, qui sont des modules autonomes à basse consommation conçus pour des applications de télémétrie et de contrôle. Ils nous permettront d'établir une communication en RF (Radio Fréquence) entre nos différentes cartes. En effet, de par leur ergonomie, il est facile d'implanter un panStamp sur une carte électronique, ce qui justifie ici le choix de cet équipement pour notre projet. Enfin, le fait qu'ils utilisent l'environnement Arduino permet de les prendre en main plus rapidement et ainsi d'optimiser le temps de réalisation du projet.



b. Composants CMS

Nous devons également utiliser des composants CMS (en anglais : SMD qui signifie Surface Mounted Device ou Composant Monté en Surface) dont les avantages seront présentés dans la suite de ce rapport.

II Réalisation

A) Partie Électronique

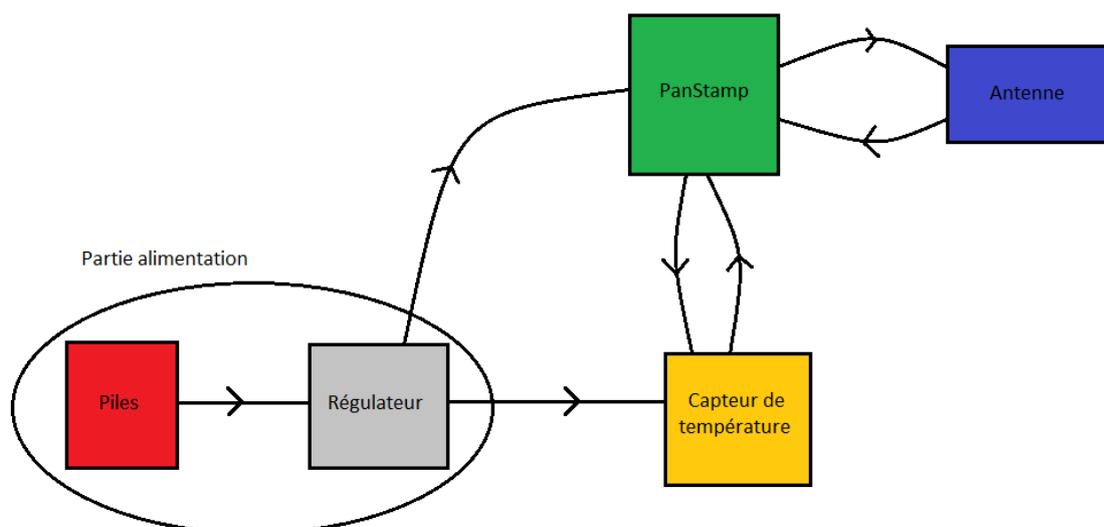
La partie électronique consiste, comme cela est présenté dans le cahier des charges, en la réalisation de deux cartes électroniques, ayant respectivement pour fonction de capter la température et d'actionner la chaudière. La particularité de la première carte est qu'elle est portable, l'utilisateur peut donc placer le thermostat où il le souhaite dans la pièce de son choix, tandis que la seconde est directement reliée à la chaudière.

a. Recherches bibliographiques afin de réaliser les fonctions souhaitées

Afin de réaliser la partie mobile du thermostat, nous avons choisi de nous munir d'un capteur de température, d'un panStamp (fonctionnant de 2VDC à 3.6VDC) et de piles pour l'alimenter (2 piles 1.2V). Néanmoins, afin d'optimiser la consommation du panStamp, et dans le cas où les piles ne fourniraient plus une tension suffisante (lorsqu'elles sont partiellement déchargées par exemple) il était nécessaire d'implanter également un régulateur. Ce composant permet en effet de réguler la tension de sortie à la valeur souhaitée. Nous avons ici choisi 2.4V. En effet, nous pouvons voir sur la datasheet du CC430F5137 (microcontrôleur du panStamp) que la tension minimale pour utiliser le module radio est 2,2V. Nous remarquons aussi que la consommation en courant est plus faible pour une tension d'entrée de 2,2V que pour une tension d'entrée de 3V. Nous pouvons ainsi réduire la consommation du circuit en ne l'alimentant pas en 3.3V par exemple. Ces consommations seront mesurées dans la suite du projet.

Pour synthétiser la carte 1 possède :

- Un panStamp NRG 2
- Une antenne
- Un boîtier de deux piles 1.2V rechargeables
- Un régulateur de tension
- Un capteur de température

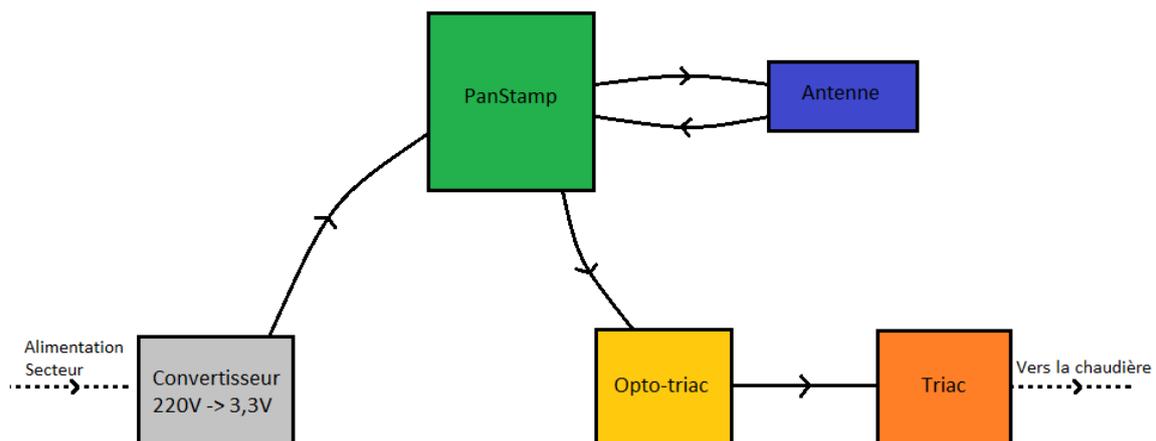


Concernant la carte directement implantée sur la chaudière, nos recherches bibliographiques nous ont tout d'abord amenées à l'utilisation d'un relais, afin de piloter la chaudière en ouvrant et en fermant le circuit électrique. Malgré tout, la plupart des chaudières délivrent une tension et un courant élevés (220 V AC et 16A). Nous devons donc trouver un relais 3,3V/220V-16A, mais ce type de composant est cher et prend beaucoup de place. La solution était donc de prendre un triac et un opto-coupleur, beaucoup moins contraignants. Suite à une discussion avec M.Benabou et M.Flamen, nous avons opté pour un opto-triac suivi d'un triac, configuration qui réduit fortement les parasites.

La carte ne nécessitant pas d'être déplacée, nous avons décidé de l'alimenter grâce au secteur. L'emploi d'un bloc convertisseur 220V/3,3V devenait alors indispensable afin de ne pas endommager le panStamp et les autres composants.

Pour synthétiser la carte 2 possède :

- Un bloc convertisseur 220V/3.3V
- Un panStamp NRG 2
- Une antenne
- Un opto-triac
- Un triac



b. Choix des composants

Il est judicieux d'utiliser des composants CMS pour plusieurs raisons :

→ Leurs avantages mécaniques (Réduction du volume des composants, réduction de la surface du circuit imprimé, gain de poids, etc.)

→ Leurs avantages électriques (Réduction des parasites, plus grande fiabilité, etc)

Le seul inconvénient est la difficulté à souder les petits composants.

Nous avons utilisé des critères comme le prix, la disponibilité et bien sur la fonctionnalité pour établir notre liste de composants principaux (Régulateur, capteur de température, triac..). Nous avons ensuite étudié les datasheets pour acheter les composants annexes (résistances, capacités etc).

Pour le triac et l'opto-triac, nous avons dû effectuer un travail de recherche un peu plus important étant donné que ces deux composants devaient être compatibles. Nous avons donc concentré nos recherches dans un premier temps sur l'opto-triac afin de trouver un composant permettant le passage d'une tension supérieure ou égale à 220V. Ensuite, en examinant la datasheet, nous avons trouvé un schéma électrique correspondant à une application de l'opto-triac avec notamment la référence du triac utilisé. Ce dernier étant à présent obsolète, nous avons alors recherché un triac avec des caractéristiques similaires et surtout qui supporte un courant de minimum 16A.

La liste des composants se trouve en annexe 2 à la fin du rapport.

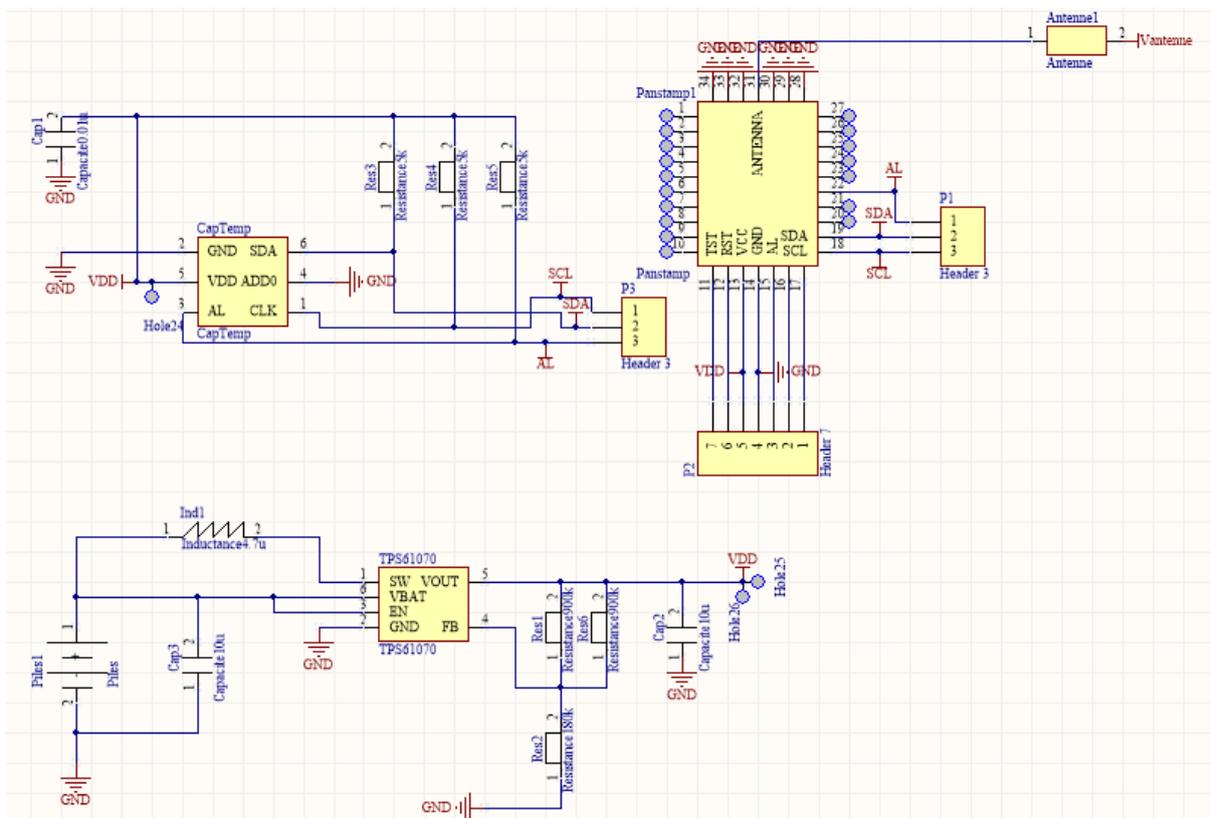
c. Réalisation des schémas Altium et routage des cartes

La première étape fut la réalisation des schématiques Altium sur la base des schémas présentés en section en II.A.a.

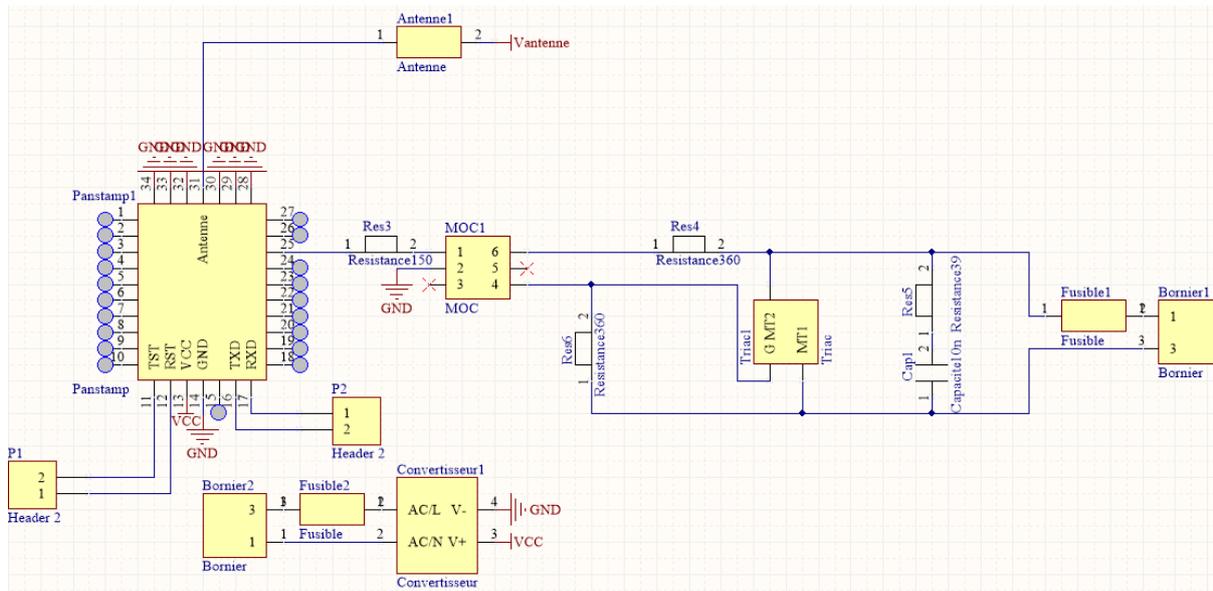
Pour cela nous avons attentivement étudié les datasheets et les circuits basiques de fonctionnement préconisés par les constructeurs.

A la fin du projet, les schémas définitifs sont les suivants :

- Schéma de la première carte :



- Schéma de la seconde carte :



d. Routage et réalisation des cartes électroniques

Après avoir réalisé les schémas électriques, il nous restait donc à créer les PCB correspondants à chacune des cartes afin de tirer et de souder ces dernières.

Dans un premier temps, nous avons créé les footprints des différents composants en utilisant notamment l'outil de création automatique d'Altium et en se référant encore une fois aux datasheets des composants.

La bibliothèque des composants de nos cartes associés à leur empreinte respective créée, nous avons ensuite généré les PCB.

1. La carte "thermostat"

Nous avons regroupé les composants correspondant à la partie "alimentation", c'est-à-dire le boîtier de piles, le régulateur et les composants tels que les résistances, bobines et condensateurs associés au régulateur. Nous avons fait de même avec la partie permettant de capter la température ainsi que l'antenne.

Nous avons également ajouté un plan de masse sur la quasi-totalité du PCB. Cependant, le plan de masse ne doit pas être sous l'antenne sous peine d'empêcher son fonctionnement.

De plus, certaines pistes ne pouvaient être placées sur la face « top » du PCB. Pour contourner ce problème et éviter de graver une carte en double face, la solution est d'utiliser des straps passant à l'arrière de la carte électronique.

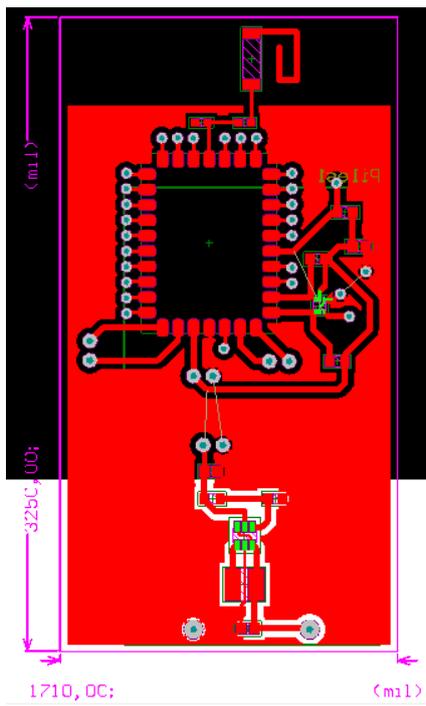
- Routage et réalisation de la deuxième version

La carte 1 présentée avant la mi-soutenance n'étant pas fonctionnelle, nous l'avons réalisée une nouvelle fois en veillant à ne pas commettre les erreurs effectuées précédemment.

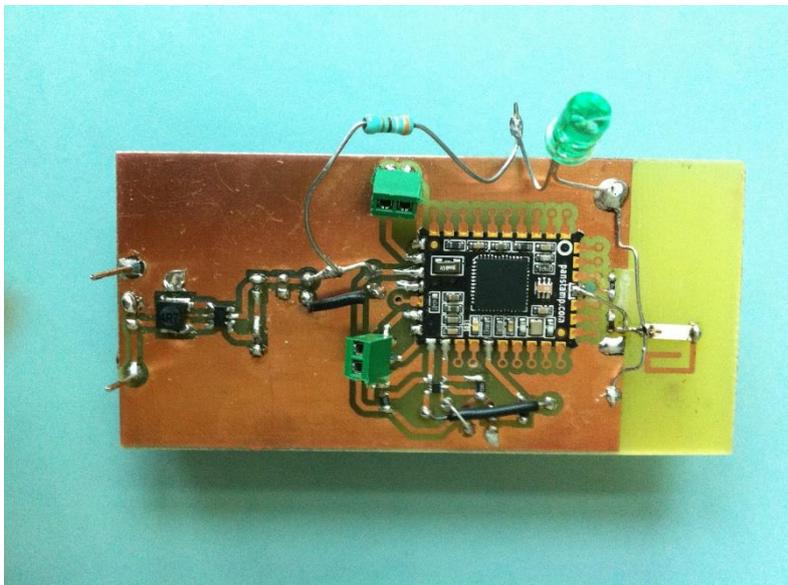
En effet, les pistes ont été agrandies et espacées du plan de masse afin d'éviter les courts-circuits et faciliter le soudage des composants. Les parties "alimentation" et "capteur de température" ont été isolées afin de tester l'alimentation indépendamment du capteur de température.

Enfin, nous avons optimisé la taille de la carte en plaçant le boîtier de piles sous les autres composants situés en face "top" du PCB.

Voici le nouveau PCB de la carte 1 :

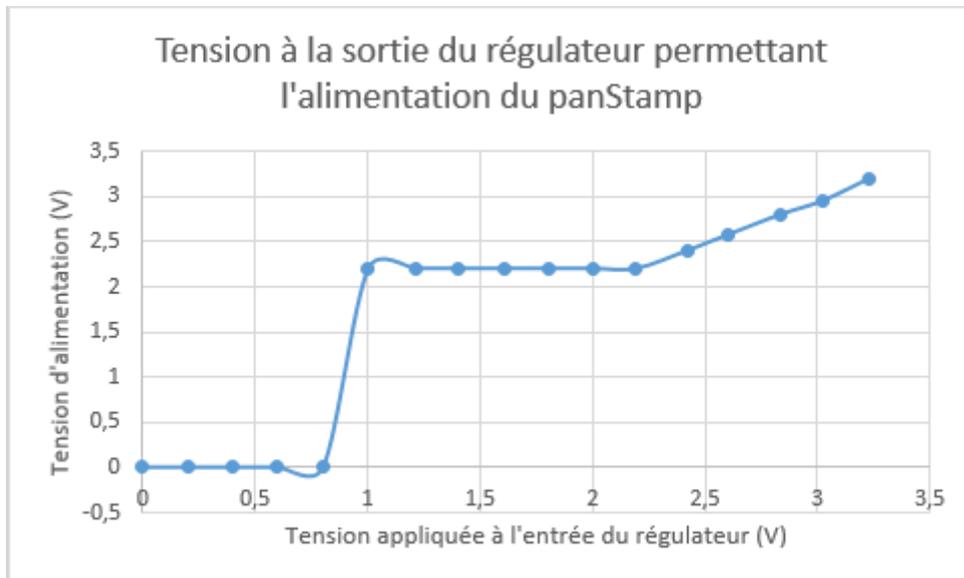


Après soudage des différents composants, le résultat est le suivant :



- Tests de la seconde version de la carte

Tout d'abord, nous avons effectué des tests d'alimentation afin de voir si le régulateur était fonctionnel, et nous obtenons les résultats suivants :



Nous remarquons alors que la tension régulée en sortie est égale à 2,2V, ce qui est la valeur souhaitée, lorsque la tension d'entrée est située entre 1V et 2,2V. Au-delà de 2,2V, la tension de sortie du régulateur est égale à la tension d'entrée, et en-dessous de 1V, elle est égale à 0V. Les résultats obtenus sont cohérents avec les données de la datasheet du régulateur.

Cependant, le régulateur que nous avons choisi n'est donc pas optimal car il effectue son rôle seulement pour des tensions inférieures à celle choisie en sortie (ici 2,2V). Il aurait fallu en choisir un qui régule également les tensions d'entrée supérieures à la tension de sortie souhaitée.

La partie "alimentation" est donc fonctionnelle mais pourrait être optimisée.

Ensuite, nous avons testé la partie "récupération de température" en téléversant un programme sur le panStamp permettant la récupération, et nous avons réussi à obtenir la température ambiante de la salle où nous trouvons. Nous avons ensuite vérifié ces valeurs à l'aide d'un thermomètre, et la température affichée étant identique à celle récupérée par le capteur, nous en avons conclu que cette partie fonctionnait également.

Cependant, lorsque nous avons essayé d'effectuer une communication entre le panStamp NRG de la carte et un panStamp AVR, nous n'avons pas eu de résultats satisfaisants. Lors du test de communication entre deux panStamps NRG, nous n'obtenons pas non plus le résultat escompté.

Le programme téléversé fonctionnant parfaitement lorsqu'il a été testé entre deux panStamps AVR, nous avons alors supposé une défaillance matérielle.

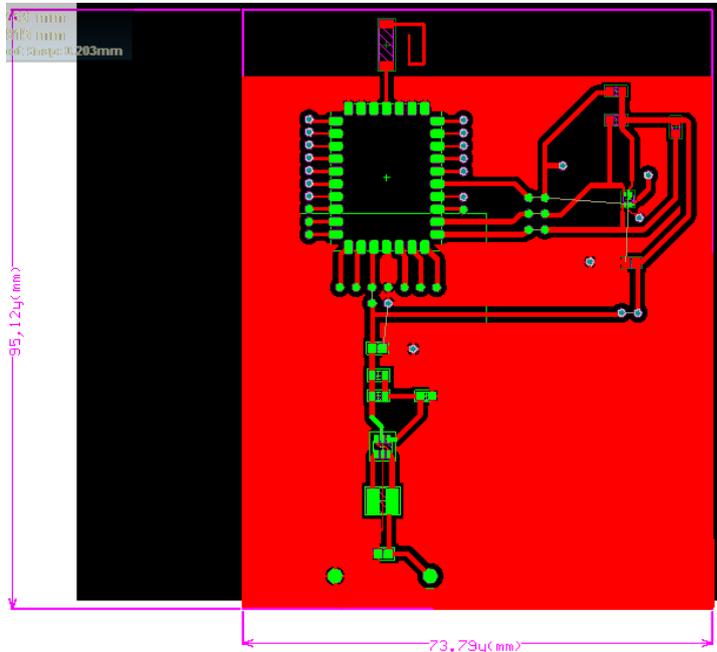
Nous pensons donc que soit l'antenne, soit le panStamp est défectueux(se), ce qui entraînerait une communication RF non fonctionnelle. En effet, l'antenne est l'élément qui permet la réception et la transmission des données, tandis que le panStamp a possiblement été grillé suite à une mise dans le mauvais sens des piles.

- Routage et réalisation de la troisième version

Suite aux tests non concluants sur la communication de la seconde version, nous avons décidé de réaliser un nouveau PCB. Ce dernier a pour but de séparer totalement les

différentes parties de la carte à l'aide de plots et de jumpers, ce qui permet de tester chaque partie indépendamment. La carte a encore une fois été agrandie en conséquence. De plus, les dimensions de l'antenne ont été modifiées pour correspondre parfaitement à celles données dans la datasheet de l'antenne et éviter un éventuel problème de communication avec les autres panStamps.

Voici le PCB de la troisième version de la carte "thermostat" :



- **Tests de la troisième version de la carte**

La carte a été réalisée deux fois, ce qui permet de prévenir des problèmes qui peuvent survenir lors du soudage de la première carte.

Comme pour la version précédente, nous avons commencé par souder et tester la partie "alimentation", avec des résultats identiques à ceux rencontrés pour la version précédente.

En revanche, lors du test concernant le capteur de température, nous n'avons pas réussi à obtenir des résultats satisfaisants. En effet, une erreur était indiquée lorsque l'on essayait d'écrire ou de lire une valeur du capteur. Nous avons alors supposé une défaillance du capteur de température (peut être trop chauffé lors du soudage).

Suite aux tests sur cette carte, nous avons décidé de souder et de tester la partie "capteur de température" en premier sur la seconde carte tirée, car c'est cette dernière qui poserait certainement le plus de problèmes.

Malheureusement, nous avons obtenu le même problème que pour la carte précédente, c'est-à-dire une incapacité à communiquer avec le capteur de température.

A ce jour, nous ne savons toujours pas quel est le problème du non fonctionnement de cette carte (capteur de température trop chauffé ou problème de conception dans le PCB).

- **Retour sur la deuxième version**

Par la suite, nous avons choisi de remplacer le panStamp probablement défectueux situé sur la carte par un nouveau.

Les résultats sont immédiats : la récupération de la température fonctionne toujours, mais cette fois-ci la communication entre les panStamps est également opérationnelle.

Nous avons donc une carte fonctionnelle permettant de poursuivre le projet.

2. La carte "chaudière"

- Routage et réalisation de la carte

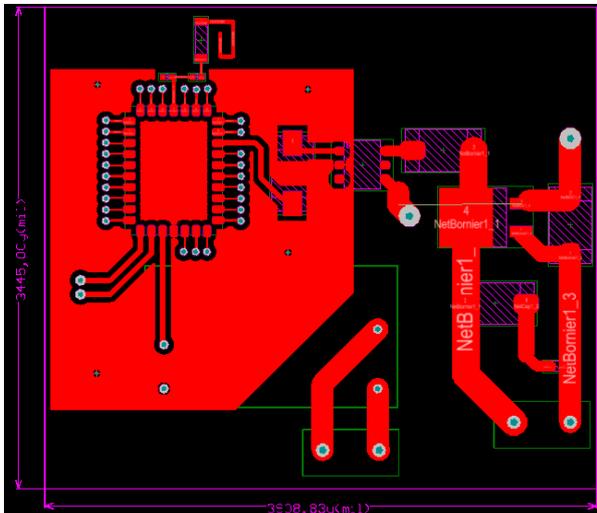
La deuxième carte étant moins complexe que la première (moins de composants) et ayant déjà effectué le PCB de la carte 1 auparavant, la réalisation de ce PCB fut plus rapide que le précédent.

Comme pour la carte 1, nous avons séparé les différentes parties de la carte qui correspondent ici aux parties "commande" et "puissance".

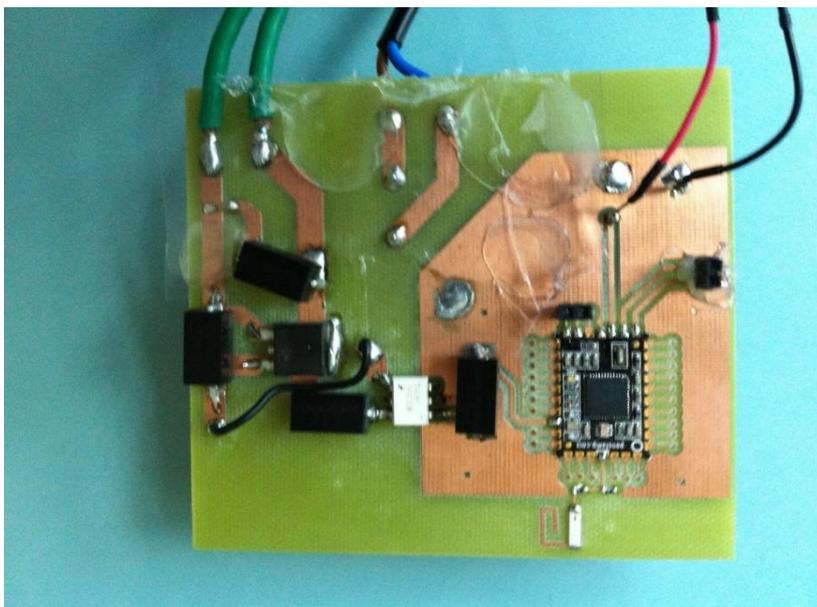
La difficulté de cette carte résidait dans la précaution quant aux largeurs des pistes et à l'espacement entre ces dernières. En effet, la tension pouvant être de 220V maximum, il nous fallait espacer les pistes de 3mm minimum. De plus, la carte doit pouvoir supporter un courant maximal de 16A. Pour cela, les pistes devaient être élargies d'au moins 4mm afin de répondre à cette contrainte.

Nous avons également ajouté un plan de masse sur ce PCB, qui ne doit ni interférer avec l'antenne, ni avec la partie "puissance".

Voici le PCB de la carte 2 :



Après soudage des différents éléments, voici le résultat obtenu :



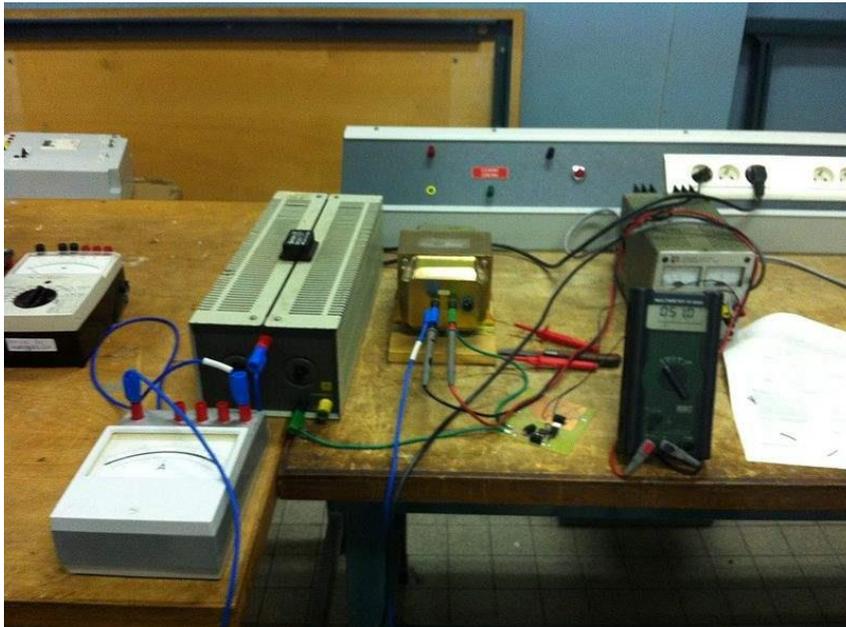
- Tests effectués sur la carte

Tout d'abord, nous avons testé la partie "puissance" permettant d'actionner la chaudière indépendamment de la partie "commande" (donc sans solliciter le panStamp).

Pour cela, nous avons utilisé une résistance variable comme charge (pour remplacer la chaudière, nous nous sommes ici limitées à une résistance 50 Ohm) et envoyé une tension de 3,3V en commande. Nous avons également utilisé un transformateur 220V / 48V pour limiter les risques dus aux essais. Nous devons donc obtenir un courant d'environ 1A.

Au final, nous avons réussi à commander le courant de sortie du triac. Le résultat est donc satisfaisant et cette partie de la carte fonctionnelle pour un courant égal à 1A.

En revanche, nous ignorons si cela fonctionne pour un courant de 16A mais nous ne pouvons pas effectuer ce test avec la carte en l'état actuel (manquent les fusibles, un dissipateur ainsi que le boîtier adéquat).



Ensuite, nous avons testé la partie "commande". Pour ce faire, nous avons téléversé un programme dans le panStamp permettant la mise à l'état "HIGH" pendant cinq secondes, puis la remise à l'état "LOW" pendant cinq secondes également de la pin reliée à l'opto-triac (et donc au triac permettant d'actionner et d'arrêter la chaudière). Nous avons utilisé le même matériel que pour le test précédent.

Le résultat suite à ce test fut également concluant : en effet, lors du passage à l'état "HIGH", nous obtenions un courant de quelques mA entre la carte et la résistance simulant la chaudière.

Enfin, nous avons remarqué que la carte ne fonctionnait pas lorsque nous voulions effectuer uniquement une réception. Nous avons d'abord pensé à un problème d'antenne, que nous avons changée, mais nous avons remarqué qu'un des codes basiques de panStamp permettait une communication. Nous avons donc cherché plus en profondeur et nous sommes rendues compte que c'était la transmission dans le code qui permettait la réception. Après affinage des recherches, nous sommes arrivées à la conclusion qu'il était nécessaire de redémarrer le module radio du panStamp (en le passant en mode « idle/inactif » puis « rx/réception ») et que cela était donc un problème du à panStamp.

B) Partie Informatique

La partie informatique représente une partie importante de notre projet. En effet, la communication entre les différents panStamps (comme vu ci-dessus) est indispensable pour réaliser un thermostat fonctionnel. Il nous était imposé d'effectuer les échanges en RF (Radio Fréquence) en utilisant des panStamps.

a. Chaîne de communication des panStamps

Nous utiliserons les panStamps **NRG 2**, dernière version de panStamps mise sur le marché, afin de pouvoir bénéficier des améliorations les plus récentes en termes de consommation, etc. Par exemple, les coûts de consommation pour une réception sont réduits de 25%. De plus, leur cœur est un MSP430 (contre un Atmega pour les AVR), ce qui leur permet d'être le module sans fil le plus puissant que produit actuellement panStamp. Le NRG 2 permet également de crypter les données grâce à un algorithme de chiffrement symétrique.

Les panStamps utilisent une plage de fréquence de communication comprise entre 868 - 915 MHz. Nous avons choisi d'utiliser la fréquence **868 MHz**, la plupart des autres étant plus réglementées en France (certaines nécessitent notamment des licences).

Le module de communication RF des panStamps nécessite l'envoi de certaines données dans les paquets dont la forme peut être caractérisée de la façon suivante :

Mots de sync 4 octets	Longueur 1 octet	Adr de dest 1 octet	Adr source 1 octet	Numéro de commande 1 octet	Données	CRC 1 octet
--------------------------	---------------------	------------------------	-----------------------	-------------------------------	---------	----------------

- 4 octets réservés aux **mots de synchronisation**, qui permettent de percevoir plus facilement le paquet dans le bruit
- 1 octet pour la **longueur du paquet**
- 1 octet pour l'**adresse de destination**
- 1 octet pour le **CRC** (*Cyclic Redundancy Check*) calculé automatiquement

Nous avons également choisi de décomposer les données en sous-parties afin de réaliser des paquets plus clairs pour une future lecture. Ainsi l'espace des données est divisé de la façon suivante :

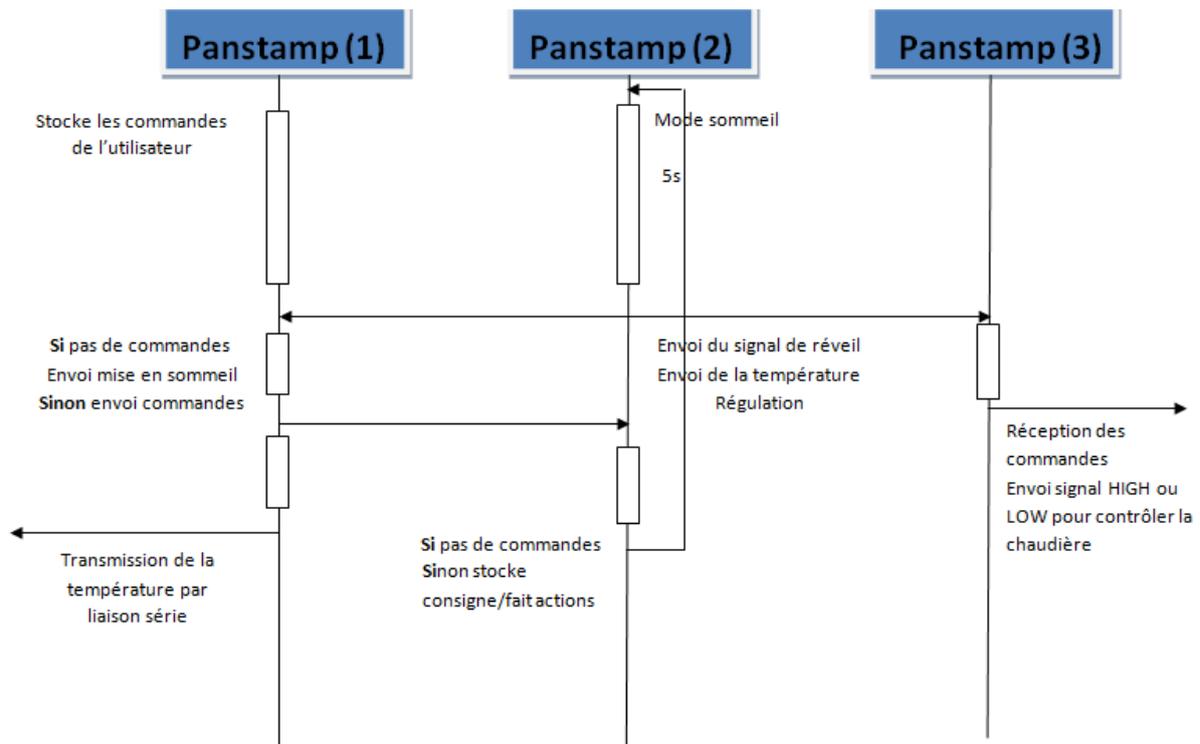
- 1 octet pour l'**adresse source**
- 1 octet pour le **numéro de commande**
- Les octets restants sont dédiés aux **données**

1. Avec des panStamp AVRs

Après avoir réalisé des premiers échanges avec les AVRs, nous avons pu adapter le code afin qu'il réponde au mieux à notre application.

Pour cela nous avons réalisé les trois codes qui seront implantés sur les différents panStamps.

Voici un diagramme de séquence qui expose leur fonctionnement :



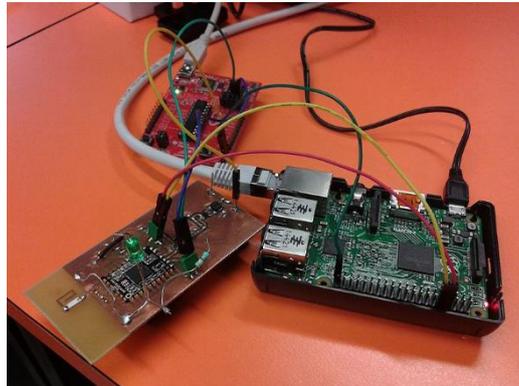
- Le code du panStamp (1) [Relié à l'interface graphique] permet :
 - De stocker toutes les commandes reçues de l'utilisateur et de les envoyer lorsque le panStamp (2) n'est pas en sommeil.
 - De recevoir la température réelle en RF et de la transmettre au code C contenant le serveur Websocket par liaison série.
 - D'identifier s'il n'y a pas eu de commandes utilisateur et d'envoyer une commande "sleep" au panStamp(2)
- Le code du panStamp (2) [Carte thermostat portable] permet :
 - De récupérer et d'envoyer à chaque réveil du panStamp la température réelle au panStamp (3)
 - De réguler la température en envoyant des commandes de marche/arrêt au panStamp (3), à chaque réveil également.
 - De réceptionner les commandes de l'utilisateur et notamment la température de consigne.
 - D'informer le panStamp (1) du réveil de la carte.
 - De mettre en sommeil le panStamp pour une durée déterminée.
- Le code du panStamp (3) [Carte implantée sur la chaudière] permet :
 - D'activer ou non la chaudière en fonction des commandes reçues du panStamp (2)

2. Mise en application sur des panStamps NRGs

Le micro-contrôleur des panStamps NRG étant différent de celui des AVR, la première étape fut de comprendre comment upload notre code sur nos cartes. Pour cela nous avons utilisé un launchpad msp430, l'IDE arduino, ainsi qu'une Raspberry.

Dans un premier temps, nous avons installé le patch Panstamp permettant la compilation pour des panStamps NRG grâce à l'IDE arduino. Mais il était impossible de directement

téléverser notre programme sur les NRG grâce à l'IDE. Nous avons donc uniquement récupéré le fichier de compilation créé par le logiciel et effectué l'upload via le launchpad. Enfin, afin de visualiser le port série, nous utilisons une Raspberry (voir photo ci-dessous).



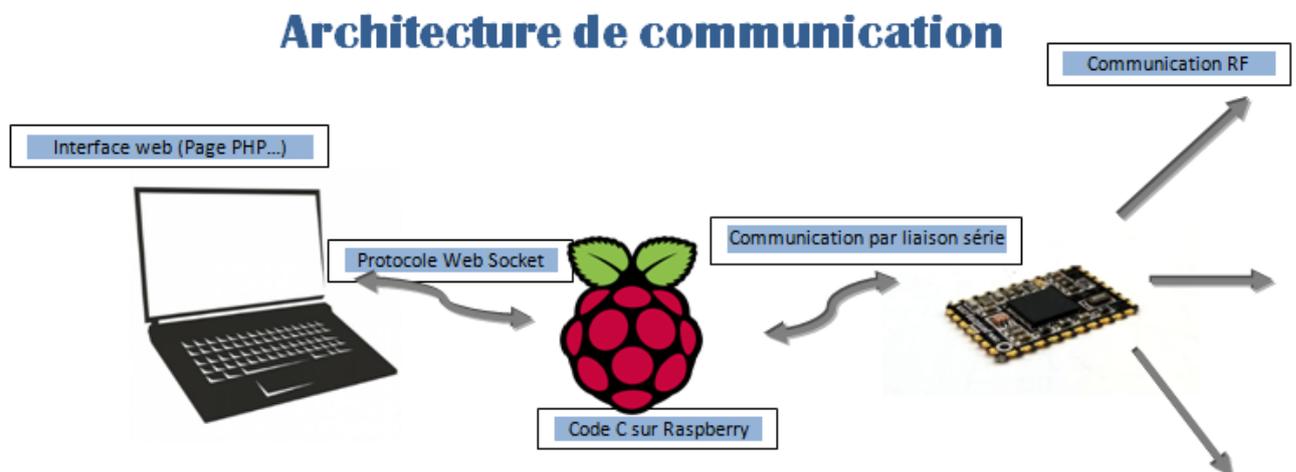
Les changements dans le code entre les AVR et les NRG sont peu nombreux. Nous avons du uniquement adapter les initialisations du TIMER.

b. Interface graphique

1. Architecture de communication

L'une des étapes du projet était la réalisation d'une interface graphique afin que l'utilisateur puisse contrôler le thermostat d'ambiance à distance. Cette interface devait être une interface web, réalisée grâce au langage HTML et PHP.

Le but était de faire communiquer le serveur web avec le panStamp présent sur la Raspberry. Pour cela, nous avons choisi d'utiliser le protocole réseau WebSocket. Ce protocole permet une communication bidirectionnelle entre un navigateur et une application distante. Ainsi nous obtenons une structure de communication se présentant de la façon suivante :



Afin d'implanter les websockets, nous avons utilisé Javascript et JQuery qui est très utilisé. Ainsi, le code C permet de créer un serveur WebSocket sur la Raspberry, auquel notre page

HTML/PHP peut se connecter. Les chaînes tapées dans les champs de texte de la page sont alors envoyées au serveur WebSocket, et donc à la Raspberry, qui assure la transmission des informations entre le panStamp et l'interface. La page web peut également recevoir des informations grâce à ce protocole bidirectionnel. Enfin, afin de transmettre ou de recevoir des informations du panStamp nous utilisons une simple communication série.

2. Modélisation de l'interface grâce à HTML, PHP et CSS

Afin de modéliser l'interface, et comme cela était demandé dans le cahier des charges, nous avons utilisé les langages HTML, PHP et CSS. Nous devons créer une interface simple d'utilisation pour l'utilisateur qui restait tout de même assez esthétique. Pour gagner du temps, nous avons utilisé le framework CSS Bootstrap, qui permet de réaliser des pages esthétiques à partir de modèles, ou en utilisant ses diverses fonctionnalités avancées.

Voici l'interface web telle qu'elle se présente aujourd'hui :



Interface de commande iThermostat

Menu de commandes Retour accueil

Contrôlez votre thermostat de n'importe quel endroit !

Avec iThermostat, c'est possible. Entrez dans les champs ci-dessous la valeur de consigne de la température de votre maison, ou encore arrêtez tout simplement le thermostat ou votre chaudière !



Température actuelle

Température de consigne Mode du thermostat Marche/Arrêt chaudière

Température (*°C) Mode actif / Mode économie Marche chaudière / Arrêt chaudière

Envoyer

© 2015 Company, Inc.

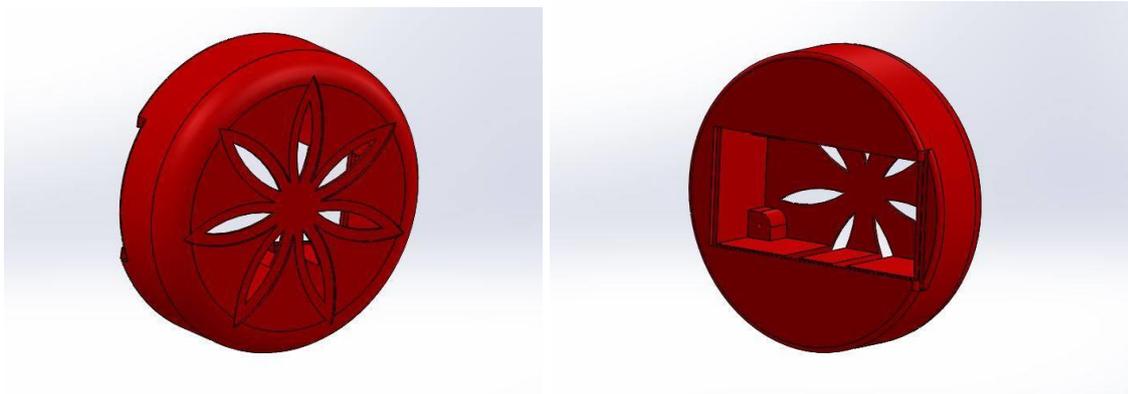
C) Conception du boîtier

Afin d'obtenir un boîtier esthétique, nous avons choisi de le réaliser grâce à l'imprimante 3D et au logiciel de Conception Assistée par Ordinateur SolidWorks.

Il était nécessaire de pouvoir fixer la carte au boîtier afin qu'elle ne bouge pas. De plus, une des contraintes était que l'utilisateur puisse accéder aux piles afin de pouvoir les changer.

Nous avons donc implanté un système de fermeture coulissante qui permet d'ouvrir et de fermer le boîtier facilement grâce à un système de CLIPS. Enfin, notre boîtier devait permettre une aération suffisante de la carte afin que la valeur de la température ne soit pas erronée.

Voici les vues du boîtier que l'on peut obtenir sur SolidWorks :



D) Évolutions futures du projet

a. Implémentation de la carte "chaudière" en 220V

La carte de puissance devant être reliée au secteur (220V), nous avons pensé à des améliorations possibles concernant la carte.

Tout d'abord au niveau protection : en effet, la carte conçue lors du projet ne contenait pas d'emplacement pour des fusibles nécessaires afin de prévenir des éventuels courts-circuits et donc de protéger l'ensemble des éléments de la carte. Ils ont donc été ajoutés en conséquence.

Le premier se situe à l'entrée de la carte au niveau de la phase, avant le bloc convertisseur 220V-3,3V. D'après la datasheet du convertisseur, son courant d'entrée est de 80mA pour une tension de 230V. Il est donc préférable de choisir un fusible 80mA temporisé. En effet, un fusible temporisé permet de laisser passer une pointe de courant sur un certain temps, par exemple à la mise sous tension.

Le second se situe en sortie de carte, entre cette dernière et la chaudière. Nous choisirons ici un fusible rapide étant donné que l'élément à protéger, c'est-à-dire le triac, est un semi-conducteur. Comme le courant maximal pouvant passer dans ce dernier est de 16A, nous optons pour un fusible d'intensité 16A, que nous placerons côté phase comme le fusible précédent.

Ensuite, le triac étant un composant dissipant de la chaleur, il est préférable d'ajouter un dissipateur sur ce dernier afin qu'il ne chauffe pas.

Pour cela, nous avons calculé la résistance thermique que le dissipateur doit posséder afin de remplir correctement son rôle. En effet, d'après la datasheet du triac et des formules permettant d'effectuer les calculs nécessaires, nous avons déterminé que la résistance thermique du dissipateur devait être égale à $3,3 \text{ }^\circ\text{C/W}$ (voir annexe n°3).

Afin de ne pas agrandir la carte en longueur et largeur, nous avons choisi de surélever le dissipateur à l'aide d'un bout de cuivre (conducteur), étant donné que le triac est moins haut que les résistances situées autour. De plus, le cuivre serait fixé au triac et au dissipateur à l'aide d'une pâte thermique.

L'autre solution aurait été d'espacer les éléments se trouvant autour du triac afin de ne pas placer que le triac sous le dissipateur. La carte aurait alors été agrandie en conséquence de ce choix.

Le dissipateur serait maintenu sur la carte à l'aide de vis, nous avons alors placé les emplacements adéquats sur le PCB.

Nous avons donc choisi le dissipateur de référence 567-547-24AB disponible sur le site de Mouser.

Enfin, il nous restait un boîtier à trouver permettant la protection de la carte et surtout des utilisateurs en faisant usage.

Pour cela, nous avons cherché un boîtier d'une matière qui n'interfererait pas avec la communication RF du panStamp de la carte, comme la plastique par exemple. Il devait évidemment respecter les dimensions de la carte.

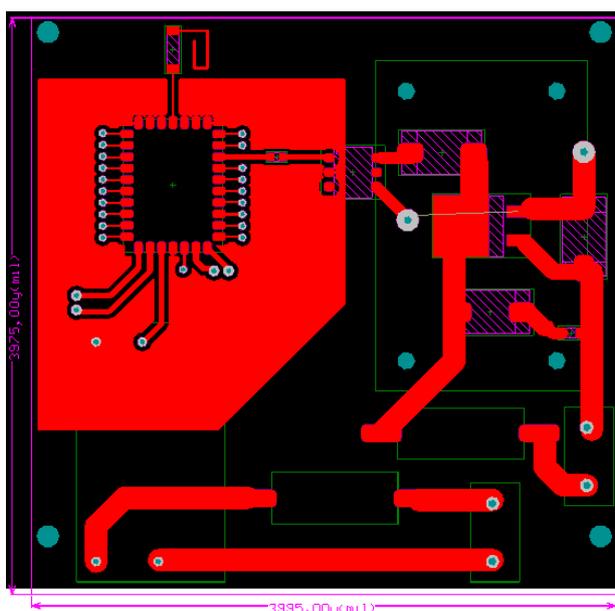
Nous avons pensé à un boîtier où la carte serait fixée à l'intérieur, à l'aide de vis. Des emplacements sont alors prévus sur le PCB à cet effet.

De plus, nous avons songé à trouver un boîtier pouvant être fixé au mur. Un trou devrait certainement être fait dans ce dernier afin de laisser passer les câbles permettant de relier la carte au secteur.

Suite à ces contraintes, nous avons sélectionné le boîtier de référence 546-1591EFLBK disponible également sur le site de Mouser.

Nous avons également effectué quelques modifications mineures comme la modification de la taille des pads des résistances se trouvant dans la partie "puissance", ainsi que le changement d'empreinte de la résistance de 150 Ohm située dans la partie "commande".

Le schéma électronique n'étant pas différent de celui "présenté" dans la partie "réalisation électronique", voici le PCB réalisé suite aux exigences ci-dessus :



b. Implémentation pour un radiateur

Pour notre projet, nous devons, en plus de réaliser une carte permettant d'actionner une chaudière, réfléchir à une solution pouvant être implantée sur un radiateur.

Après quelques recherches, nous avons trouvé une solution existante s'appliquant aux radiateurs électriques contrôlables par fils pilotes. Le principe est d'envoyer un signal alternatif modulé sur une entrée spécifique du thermostat du radiateur. La modulation utilisée permet donc de spécifier le mode de fonctionnement souhaité. Le but est de moduler le signal alternatif pour contrôler le radiateur.

On se concentrera uniquement sur les 4 ordres qui utilisent des alternances complètes du signal, qui sont : éteint, confort, hors-gel et éco. Moduler le signal alternatif pour ces 4 cas revient à pouvoir bloquer ou laisser passer des alternances du signal.

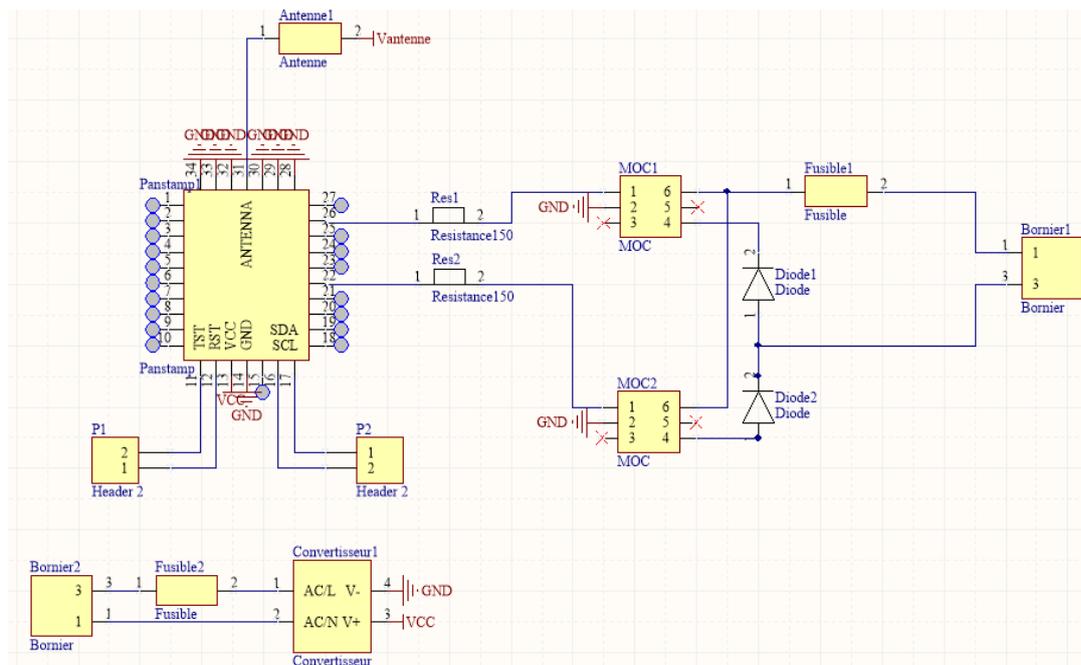
Pour bloquer les alternances, nous avons besoin de deux diodes (une par alternance). Avec ces diodes, nous pouvons donc bloquer ou laisser passer le courant dans un sens ou dans l'autre.

Dans la solution existante, la diode utilisée est la 1N4007, qui est un composant traversant. Après étude des caractéristiques de la diode à l'aide de la datasheet de cette dernière, une diode CMS qui peut tenir jusqu'à 1000V en tension inverse et jusqu'à 1A en courant serait appropriée dans notre cas.

Le but est de pouvoir contrôler par quelle diode on veut laisser passer le signal. Il faudrait alors utiliser un opto-triac capable de gérer des tensions de 220V. L'opto-triac aurait donc la fonction d'un interrupteur contrôlé en tension continue.

La solution trouvée utilise des MOC3041, et après une comparaison de la datasheet de ce dernier avec celle de l'opto-triac que nous avons utilisé pour la chaudière, il semblerait qu'aucune différence notable ne soit à déplorer et que nous pourrions donc l'utiliser au lieu du MOC3041.

Voici le schéma électronique réalisé :

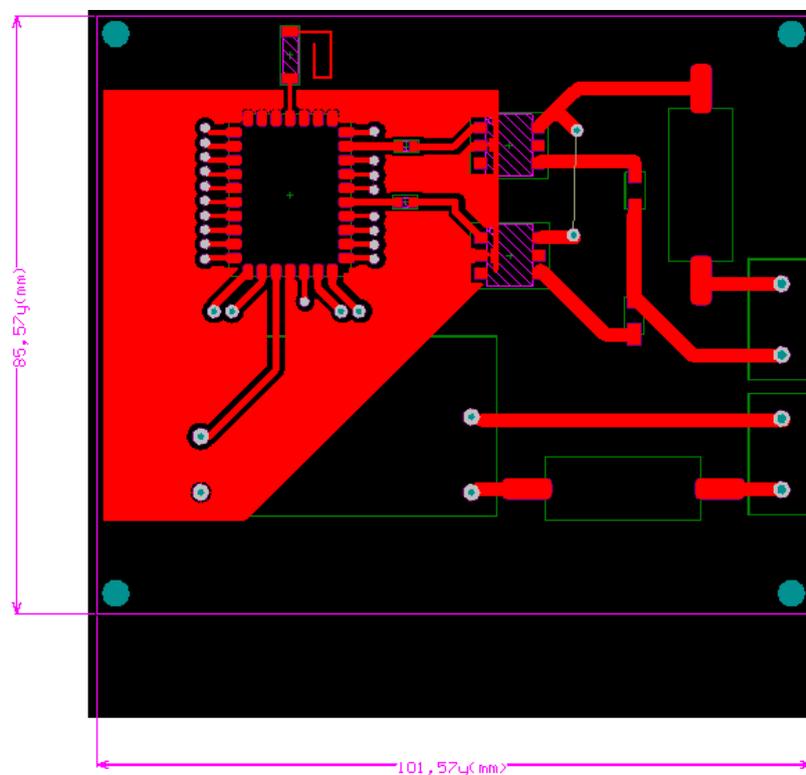


Nous obtenons donc une partie "commande" quasiment identique à celle présentée dans la carte reliée à la chaudière. En effet, nous retrouvons le panStamp, l'antenne, ainsi que la tension d'alimentation 3,3V provenant du bloc convertisseur. Le seul changement est que nous avons cette fois-ci deux résistances de 150 Ohm placées sur des sorties digitales du panStamp au lieu d'une seule, pour cause de deux opto-triacs au lieu d'un seul comme précédemment.

Les importants changements à noter par rapport à la précédente sont donc situés dans la partie "puissance" de la carte. En effet, comme expliqué ci-dessus, elle se compose de deux diodes, deux opto-triacs, et de deux fusibles.

Le premier fusible placé côté "commande" serait un 80mA temporisé, comme la chaudière, tandis que le second serait un 1A rapide, étant donné le courant maximum passant dans la partie "puissance" ne dépasse pas 1A.

Voici le PCB réalisé en conséquence :



Comme pour la carte reliée à la chaudière, nous avons tenté de séparer au mieux les parties "commande" et "puissance".

De plus, les pistes du côté "puissance" sont espacées entre elles d'au moins 3mm car la tension atteint 220V. En revanche, le courant ne dépassant pas 1A, la contrainte concernant les pistes agrandies à 4mm pour la carte de la chaudière n'est plus en vigueur cette fois-ci.

Enfin, nous avons choisi le même boîtier que pour la carte reliée à la chaudière et avons placé les trous pour les vis permettant de fixer la carte dans ce dernier sur le PCB.

III Résultats finaux et perspectives d'amélioration

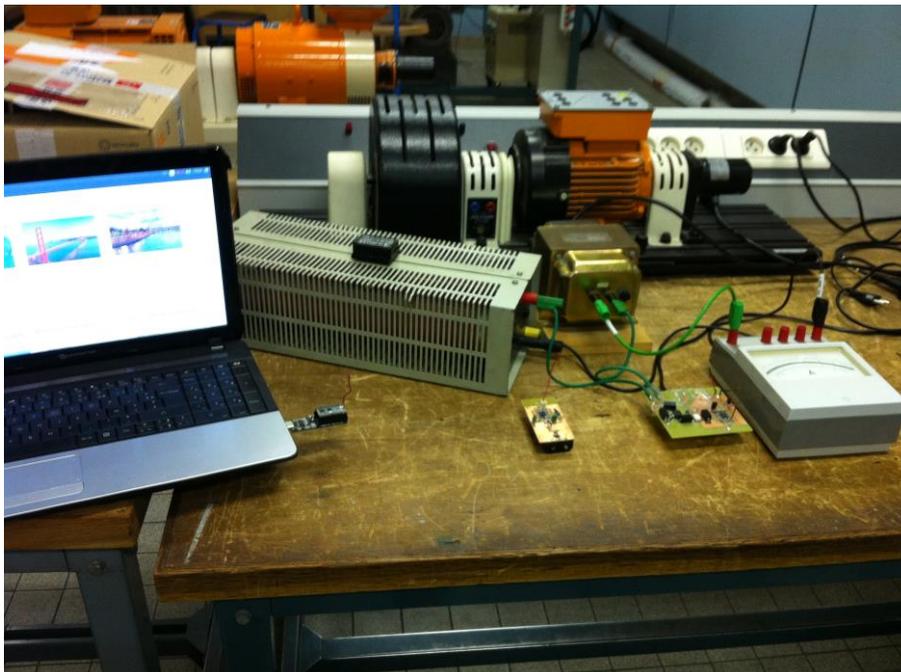
A) Résultats finaux

Nous avons réalisé le test final de notre projet au laboratoire d'électrotechnique afin d'avoir accès au matériel nécessaire afin de simuler la chaudière. Nous avons donc utilisé un transformateur 220V / 48V pour alimenter la partie "puissance" de la seconde carte (celle reliée à la chaudière), et une résistance de 50 Ohm permettant de remplacer la chaudière.

De plus, nous avons accès à l'interface graphique via l'un de nos ordinateurs personnels, et les programmes adéquats étaient téléversés sur chacun des trois panStamps nécessaires au bon fonctionnement de notre projet. La carte représentant le thermostat est donc alimentée par piles et celle reliée à la chaudière alimentée par le secteur.

Tout d'abord, la température relevée par le capteur de la carte "thermostat" est affichée en continu sur l'interface web. De plus, lorsque nous indiquons une valeur de consigne supérieure à la température actuelle de la pièce, nous remarquons que la valeur du courant de l'ampèremètre mis en série dans le montage augmente. En effet, le triac situé sur la carte "chaudière" est actionné afin d'allumer la chaudière. Dans le cas inverse, lorsque la consigne indiquée par l'utilisateur est inférieure à la température de la pièce, le courant diminue.

Nous avons donc réussi à tester la chaîne complète en obtenant des résultats satisfaisants. Il ne manquerait plus qu'à pouvoir effectuer le test en conditions réelles (avec une chaudière).



B) Problèmes rencontrés et bilan personnel

Bien que le projet se soit généralement bien déroulé, nous avons, comme cela a été expliqué précédemment, perdu beaucoup de temps lors de la réalisation des cartes, notamment car nous n'avons que peu d'expérience dans la soudure de composants CMS et que l'école ne disposait pas de four. Ainsi, nous pensons que l'utilisation de composants traversants pour réaliser un premier prototype puis la réalisation de la même carte en CMS aurait été plus simple pour obtenir un projet opérationnel. En revanche, la partie informatique a posé moins de problèmes.

Ce projet a été très enrichissant tant au niveau personnel qu'au niveau technique, puisque nous avons pu effectuer de la gestion de projet mais également travailler sur des sujets que nous ne connaissions pas forcément. En effet, nous avons pu apprendre à travailler en équipe (binôme), à gérer un planning et des réunions avec un client. L'autonomie accordée par le projet va nous permettre d'aborder au mieux notre stage et la gestion des nouveaux projets qui nous seront confiés.

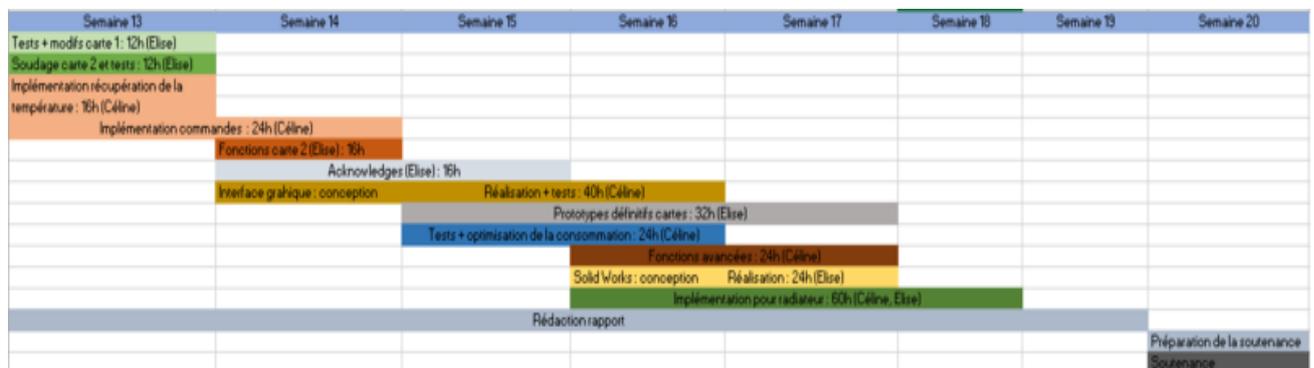
Au niveau des compétences, nous avons pu nous enrichir par rapport à la conception de cartes électroniques (prise en main du logiciel mais aussi du travail de recherches de composants). En informatique, nous avons pu approfondir nos connaissances du langage Arduino. De plus, la réalisation de l'interface nous a permis d'avoir des bases en programmation HTML, PHP, CSS et Javascript, ce qui est très intéressant.

Nous avons pu nous rendre compte que réaliser ce genre de projet permet de développer des compétences réelles et concrètes, que l'on pourra directement appliquer plus tard dans notre vie professionnelle. En effet, lorsque nous apprenons une notion en cours, nous n'avons pas forcément le temps de l'approfondir comme nous le voudrions, ce qui n'est pas le cas ici. Ainsi, nous pensons que le projet de fin d'études est important dans le début de carrière d'un jeune ingénieur.

C) Comparatif avancée prévisionnelle/avancée réelle

a. Avancée prévisionnelle

Lors de la soutenance de mi-projet, nous avons établi un planning prévisionnel pour les semaines futures que vous pouvez voir ci-dessous :



b. Avancée réelle

Au cours du projet, nous avons pu établir notre planning d'avancement réel qui se présente sous la forme suivante :

Semaine 13	Semaine 14	Semaine 15	Semaine 16	Semaine 17	Semaine 18	Semaine 19	Semaine 20
Soudage cartes 1&2							
Tests sur les deux cartes							
	Implémentation des commandes panStamp	Chaîne AVR's opérationnelle					
	Réflexion sur l'interface graphique	Finalisation de l'interface graphique					
		Transition sur les NRGs					
		Récupération de la température					
				Conception du boîtier			
				Recherches & implémentation radiateur			
				Modification des cartes			
					Réalisation de la chaîne complète		
					Rédaction rapport		
						Tournage vidéo	
							Préparation soutenance
							Soutenance

On peut voir que le planning prévisionnel n'a généralement pas été respecté, pour plusieurs raisons :

- Nous ne pouvions pas faire la récupération de température tant que les cartes n'étaient pas terminées et l'implémentation sur les panStamps NRGs opérationnelle. Cette étape a donc été décalée dans le temps.
- Les problèmes rencontrés avec la carte 1 nous ont fait perdre beaucoup de temps. Ainsi, nous n'avons pas pu réaliser l'étude sur la consommation du circuit car la carte n'a fonctionné que très tard.
- Enfin, on peut se rendre compte que nous avons mal estimé le temps de travail sur certaines parties (interface graphique, implémentation pour un radiateur, etc).

D) Perspectives d'amélioration

Comme nous avons rencontré plusieurs problèmes de carte électronique, nous n'avons pas pu effectuer les mesures et la réduction de consommation prévues. Ceci est donc un axe d'amélioration de notre projet. Il est en effet très intéressant que le thermostat portable puisse avoir l'autonomie la plus longue possible. De plus comme cela était prévu en début de projet, la réalisation d'une auto-alimentation pour cette carte serait une plus-value non négligeable.

De plus, nous pouvons améliorer le thermostat en lui rajoutant des fonctionnalités :

- Au niveau de l'interface graphique, il aurait été intéressant qu'elle puisse stocker les données météo afin de conseiller l'utilisateur sur le comportement à adopter au niveau de son chauffage.
- Au niveau de la carte thermostat, nous aurions pu implanter une fonction qui mesure le taux de déchargement des piles et qui envoie un message sur l'interface graphique en cas de piles trop faibles.
- Au niveau des cartes, nous pourrions optimiser la taille des cartes afin que le boîtier soit plus ergonomique.

De plus, la carte récupérant la température possédant des piles, il aurait été judicieux de réfléchir à un système de protection pour éviter que l'utilisateur insère les piles dans le mauvais sens. On aurait pu penser à une solution mécanique (par exemple, que le contact ne soit pas fait quand les piles sont à l'envers) ou électronique (à l'aide d'une diode et d'un fusible).

Enfin, souder des connecteurs afin de placer les panStamps sur les cartes nous aurait permis d'éviter de les abîmer et surtout de pouvoir les retirer lorsque des problèmes se présentent, comme ce fut le cas durant notre projet. Une idée serait de souder des connecteurs mâles sur le panStamp et des connecteurs femelles sur la carte, ce qui permettrait de pluggier le panStamp sur cette dernière.

Conclusion

La réalisation d'un thermostat connecté est un thème qui nous a tout de suite intéressés. Lors de ce projet, nous avons pu développer des connaissances que nous ne maîtrisions pas, notamment au niveau électronique (avec les créations de PCB) et au niveau informatique (avec la création d'une interface graphique). Les moyens mis en œuvre par l'école (matériel et aide des professeurs) nous ont permis de réaliser le projet dans de bonnes conditions, ce qui est très appréciable.

Malgré tout, les différents problèmes que nous avons rencontrés notamment au niveau de la communication des cartes et l'utilisation de composants CMS nous ont empêchées de mener le projet comme nous l'aurions souhaité. En effet, certains éléments n'ont pas pu être réalisés (étude de consommation).

Pourtant, il est important de rappeler que même si certaines cartes n'ont pas été tirées (comme par exemple la carte du radiateur ou celle de la chaudière avec le dissipateur), le travail de recherche a été effectué et elles pourront ainsi être reprises facilement par l'intervenant industriel. Le projet pourra ainsi être facilement amélioré dans les années à venir.

Annexe 1 : consommation du circuit carte 1

Afin de calculer la consommation du circuit, nous devons d'abord trouver le nombre de données échangées en kBps. Après avoir cherché dans des datasheets et sur le site de panStamp, nous avons trouvé que nos panStamps utilisaient un datarate de 38383,5 bps. Le minimum de bits échangés à chaque envoi de paquet étant de 104, nous obtenons facilement les temps de transmission/réception du panStamp :

$$104 \cdot 38383,4 = 0.002708333 \text{ secondes}$$

Nous avons supposé que le panStamp effectuait 2 transmissions et une réception.

Consommation	Nombre de mA	Temps en sec	Consommation sur le temps
Réception	18	0,002708333	0,04875
Transmission	36	0,005416667	0,195
Traitement	0	0	0
Sleep	0,002	60	0,12
		Temps total	Consommation totale sur le temps
		60,008125	0,36375
		Consommation ramenée à l'instantanée	0,006061679
		Capacité des piles en mA/h	2500
		Consommation du circuit en heures	412426,9759

Nous trouvons un résultat de 412426.9759 heures, ce qui correspond à environ 47 années lorsque le panStamp est mis en sommeil pendant 60 secondes.

Pour être plus précis, nous pouvons également rajouter les consommations du régulateur et du capteur de température, qui sont respectivement de 1.9×10^{-5} A et de 1×10^{-5} A.

Nous voyons donc que les résultats théoriques semblent très corrects. Néanmoins des tests de consommation seront effectués lorsque les cartes seront opérationnelles, ainsi qu'une étude plus détaillée.

Annexe 2 : liste des composants commandés

Boîtier à piles de référence fournisseur (Mouser) **534-2462**
Piles AA de référence fournisseur (Mouser) **547-NH1250AA**
Capteur de T°C de référence fournisseur (Mouser) **595-TMP112AIDRLT**
Bloc 220-3.3V de référence fournisseur (Mouser) **709-IRM05-3.3**
Fusible de référence fournisseur (Mouser) **576-083506.3MXEP**
Porte fusible de référence fournisseur (Mouser) **693-0031.8201**
Antenne plane de référence fournisseur (Mouser) **609-0868AT43A0020E**
Condensateur 10u de référence fournisseur (Mouser) **80-C0805L106K8P**
Condensateur 0.01u de référence fournisseur (Mouser) **80-C0805C103M5RAUTO**
Bobine 4,7u de référence fournisseur (Mouser) **710-744031004**
Régulateur Boost TEXAS de référence fournisseur (Mouser) **595-TPS61070DDCR**
LED de référence fournisseur (Mouser) **720-LHR974-LP-1**
Résistance 348kOhm de référence fournisseur (Mouser) **667-ERJ-6ENF3483V**
Résistance 1MOhm de référence fournisseur (Mouser) **71-CRCW08051M10FKEA**
Résistance 5kOhm de référence fournisseur (Mouser) **71-CRCW0805-5K**
Opto-triac de référence fournisseur (Mouser) **512-MOC3062SR2VM**
Triac de référence fournisseur (Mouser) **576-Q6016NH4RP**
Résistance 360Ohm de référence fournisseur (Farnell) **2331668**
Résistance 39Ohm de référence fournisseur (Mouser) **279-SMW539RJT**
Résistance 150Ohm de référence fournisseur (Mouser) **279-SMW5150RJT**
Condensateur 0,01uF de référence fournisseur (Mouser) **603-CC126KKX7RZBB103**
Inductance 12nH de référence fournisseur (Mouser) **609-L-07C12NJV6T**
Inductance 1,8nH de référence fournisseur (Mouser) **609-L-07C1N8SV6T**

Annexe 3 : Calcul de la résistance thermique du dissipateur pour le triac

Nous avons les formules suivantes :

$R_{thJA} = R_{thJC} + R_{thCR} + R_{thRA}$ avec :

- R_{thJA} : Résistance thermique entre la jonction et l'air
- R_{thJC} : Résistance thermique entre la jonction et le boîtier (case)
- R_{thCR} : Résistance thermique entre le boîtier et le radiateur (dissipateur)
- R_{thRA} : Résistance thermique entre le radiateur et l'air ambiant

$P_{diss} = \frac{T_j - T_a}{R_{thJA}}$ avec :

- T_j : température maximale de jonction
- T_a : température ambiante
- P_{diss} : puissance dissipée

La donnée que nous cherchons pour notre dissipateur est la résistance thermique entre le radiateur et l'air ambiant donc R_{thRA} .

D'après la datasheet du triac, nous savons que $T_{thJC} = 1,1^\circ\text{C}$, $P_{diss} = 17,5\text{W}$ et $T_j = 125^\circ\text{C}$.

De plus, comme de la pâte de résistance thermique $0,65^\circ\text{C/W}$ serait appliquée entre le triac et le cuivre et entre ce dernier et le dissipateur, nous avons $R_{thCR} = 1,3^\circ\text{C/W}$.

La température ambiante $T_a = 25^\circ\text{C}$.

Nous pouvons déterminer $R_{thJA} = \frac{125 - 25}{17,5} = 5,7^\circ\text{C}$.

Donc $R_{thRA} = 5,7 - 1,1 - 1,3 = 3,3^\circ\text{C/W}$.