



Dossier d'études et réalisations:

Affichage de données sur un volant de karting électrique et gestion de la fonction « Boost »

Etudiant:

JACQUIN Julien Gr. Q1

Enseignants:

Mme DA COSTA I.

Mr LEQUEU T.



Dossier d'études et réalisations:

Affichage de données sur un volant de karting électrique et gestion de la fonction « Boost »

Etudiant:

JACQUIN Julien Gr. Q1

Enseignants:

Mme DA COSTA I.

Mr LEQUEU T.

Sommaire:

Introduction

I. Cahier des charges

II. Présentation des solutions envisagées et retenues

II.1. L'acquisition de la température

II.2. Affichage LCD

III. Programmation d'un PIC18f4620 avec MPLAB et C18

IV. Réalisation des cartes électroniques

IV.1. La carte microcontrôleur

IV.2. La carte alimentation

IV.3. Le capteur à fourche *EE-SX1070*

IV.4. Le module "Tension de batterie"

IV.5. Le module "Capteur de courant"

V. Conclusion

Introduction

La technologie embarquée dans les nouveaux véhicules roulants est en perpétuelle évolution et met en œuvre électronique et informatique. Dans le cadre de la formation professionnalisante du DUT Génie Electrique et Informatique Industrielle (GEII) de l'Université François Rabelais de Tours, nous avons effectué un projet dans ce domaine d'activité.

Notre projet, consiste à la mise en œuvre du traitement de données informatiques, numériques et analogiques, afin de piloter un Karting électrique et afficher plusieurs de ces données sur un écran LCD placé sur le volant.

Quelles sont les difficultés rencontrées ? Quelles sont les limites des moyens mise en oeuvre? Ce sont autant de questions auxquelles nous tenterons de répondre de manière claire et simple, en vous expliquant notre démarche, les difficultés rencontrées ainsi que les compromis qui ont du en découler. Pour ce faire, nous décrirons tout d'abord les solutions retenues par rapport aux autres, ensuite, nous évoquerons les outils de la conception, suivra la description du modèle réalisé et enfin nous présenterons les résultats obtenus.

I. Cahier des charges

Notre projet consiste à réaliser une carte électronique d'acquisition de signaux. Ces signaux sont pour la grande partie analogiques. On devra traiter ces signaux en numérique via un microcontrôleur afin de contrôler l'accélération du karting et l'affichage de diverses données sur un écran LCD.

Cette carte, destinée à être intégrée au volant d'un karting électrique, devra être de taille réduite (Nous utiliserons peut être la technologie CMS).

Elle acquerra:

- 4 signaux analogiques différents qui sont :
 - la tension image de la température du moteur,
 - la tension aux bornes de la batterie,
 - la charge en courant de la batterie,
 - la tension image de l'accélération.

- 2 signaux numériques :
 - les informations envoyés par le capteur inductif,
 - l'interrupteur de boost.

Pour le microcontrôleur, notre choix s'est porté sur un PIC18F4620 (microcontrôleur élaboré par Microchip). Le choix s'est effectué après une étude de coût, des fonctionnalités du microcontrôleur et de sa simplicité de mise en œuvre.

L'image de la température sera acquise par le biais d'une thermistance CTN monté en pont diviseur de tension.

La charge en courant de la batterie sera acquise par le biais d'un capteur à effet Hall.

La tension image de l'accélération sera acquise par le biais d'un potentiomètre déjà installé sur la pédale d'accélérateur (signal compris entre 0V et 1V).

L'acquisition de la vitesse se fera via le capteur à fourche EE-SX1070.

L'interrupteur de boost est directement connecté au microcontrôleur sur une entrée numérique.

La tension image de la tension de la batterie sera acquise par le biais de deux résistances montées en diviseur de tension.

Nous aurons en sortie de microcontrôleur:

- 1 sortie analogique : l'accélération du karting.
- 7 sorties numériques : (3 de contrôle et 4 de donnée) qui serviront à l'affichage sur l'écran LCD de la vitesse, la température moteur, la charge et la tension de batterie ainsi que le temps de boost restant.

Pour la partie software, nous avons décidé de travailler pour la partie CAO avec le logiciel Orcad. Pour la partie programmation, nous allons utiliser le logiciel gratuitement fourni par Microchip auquel on ajoutera le module de programmation en C « C18 », module spécifique aux microcontrôleurs PIC18.

II. Présentation des solutions envisagées et retenues

Nous avons, pour ce projet pensé à plusieurs solutions technologiques concernant l'acquisition de la vitesse du Karting, l'acquisition de la température, le choix de l'alimentation etc,,,. Nous allons voir les principaux avantages et inconvénients de chaque technologie utilisable et voir les solutions choisies.

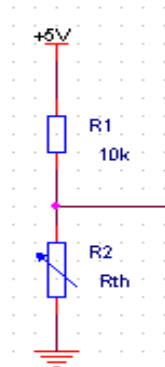
II.1. L'acquisition de la température.

Afin d'acquérir la température du moteur, nous avons eu le choix entre deux technologies complètement différentes :

- une solution numérique
- une solution analogique

La solution analogique.

Le PIC18F4620 contient treize convertisseurs analogiques/numériques (CAN). Afin de calculer la température, nous utilisons une thermistance (composant passif dont la résistance (R_{th}) varie selon la température) ainsi qu'une résistance de $10k\Omega$ (valeur choisie pour avoir un maximum de précision, la valeur de notre thermistance à température ambiante étant de $10k\Omega$) avec lesquelles nous faisons un pont diviseur de tension



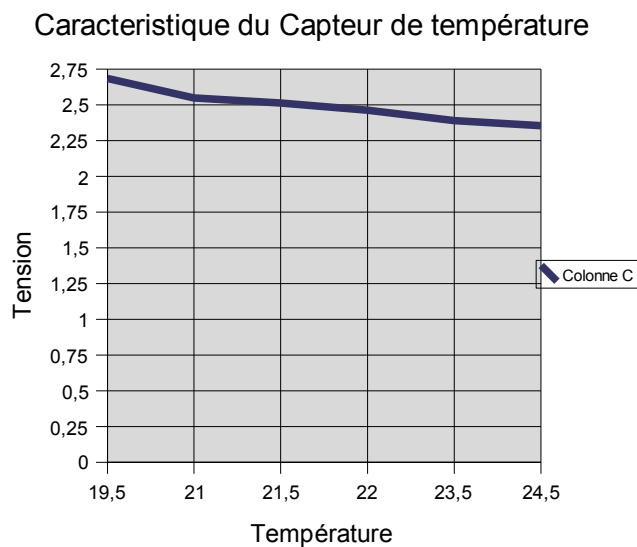
Grâce à ce pont diviseur de tension, je récupère sur l'entrée du CAN RA2 de mon microcontrôleur une tension (U) image de la température selon l'équation suivante :

$$V = 5 * R_{th} / (R_{th} + 10000) (V)$$

Je récupère donc une tension proportionnelle à la température que je convertie en donnée numérique. Une fois la conversion effectuée, je calcule la température extérieure grâce à la courbe caractéristique de ma thermistance (figure n°)

En prenant en compte cette solution, j'ai réalisé un test du montage afin d'obtenir une idée de la tension récupérée ainsi que de la valeur de la thermistance.

T (°C)	Rth(ohm)	V(V)
19,5	11600	2,69
21	10400	2,55
21,5	10100	2,51
22	9700	2,46
23,5	9150	2,39
24,5	8900	2,35



La solution numérique.

La seconde solution technologique envisagée consiste à utiliser une puce programmer qui acquière automatique la température lorsqu'on lui demande via le microcontrôleur. Cette puce communique avec le microcontrôleur via un protocole de communication SPI (communication synchrone).

II.2.Affichage LCD

Avant de prévoir une carte avec le connecteur nécessaire pour brancher notre afficheur LCD, nous avons testé mon programme d'affichage sur la carte de démonstration PICDEM 2 PLUS.

Cette carte de démonstration de chez Microchip est munie d'un afficheur LCD et d'un socle pour microcontrôleur PIC 40 broches nécessaire pour programmer notre PIC18f4620.



Figure 1 : Carte de démonstration PICDEM 2 PLUS avec le programmeur ICD2

Après plusieurs recherches sur internet, j'ai trouvé un programme en langage C qui gère l'affichage sur cette carte de démonstration. En effet, les codes sources fournis par Microchip sont tous en langage assembleur, langage différent de celui spécifié dans mon cahier des charges.

Une fois le choix de l'affichage sur la carte de démonstration testé sur le PICDEM 2 PLUS et ayant reçu l'afficheur LCD que j'allais utiliser, une problématique

s'est posée. En effet, il est possible de configurer l'envoi de données à un écran LCD de deux manières différentes :

- soit le mode parallèle 8 bits,
- soit par le mode parallèle 4 bits.

Le mode huit bits signifie que les données sont envoyées à l'afficheur LCD octet par octet en parallèle. Cela signifie également qu'il est nécessaire d'utiliser huit broches du microcontrôleur, uniquement pour l'envoi de données, auxquelles s'ajoutent quatre broches pour le contrôle du LCD, soit douze broches au total. Or le nombre de broches libres sur les ports de sortie de la carte mère est très réduit.

La solution est de gérer l'envoi de données en mode quatre bits. Le fonctionnement n'est pas plus compliqué que le mode parallèle huit bits. En effet, au lieu d'envoyer directement les données par octets, le programme les envoie par quartets. Les quatre bits de poids forts d'abord, puis les 4 bits de poids faible. Cette méthode me permet d'économiser quatre broches sur mon microcontrôleur et de ne dédier qu'un seul port (le port B) pour l'affichage sur le LCD.

Le mode de transmission déterminé, j'ai codé mon programme et testé l'affichage de la température sur plaque d'essai selon le schéma de test suivant.

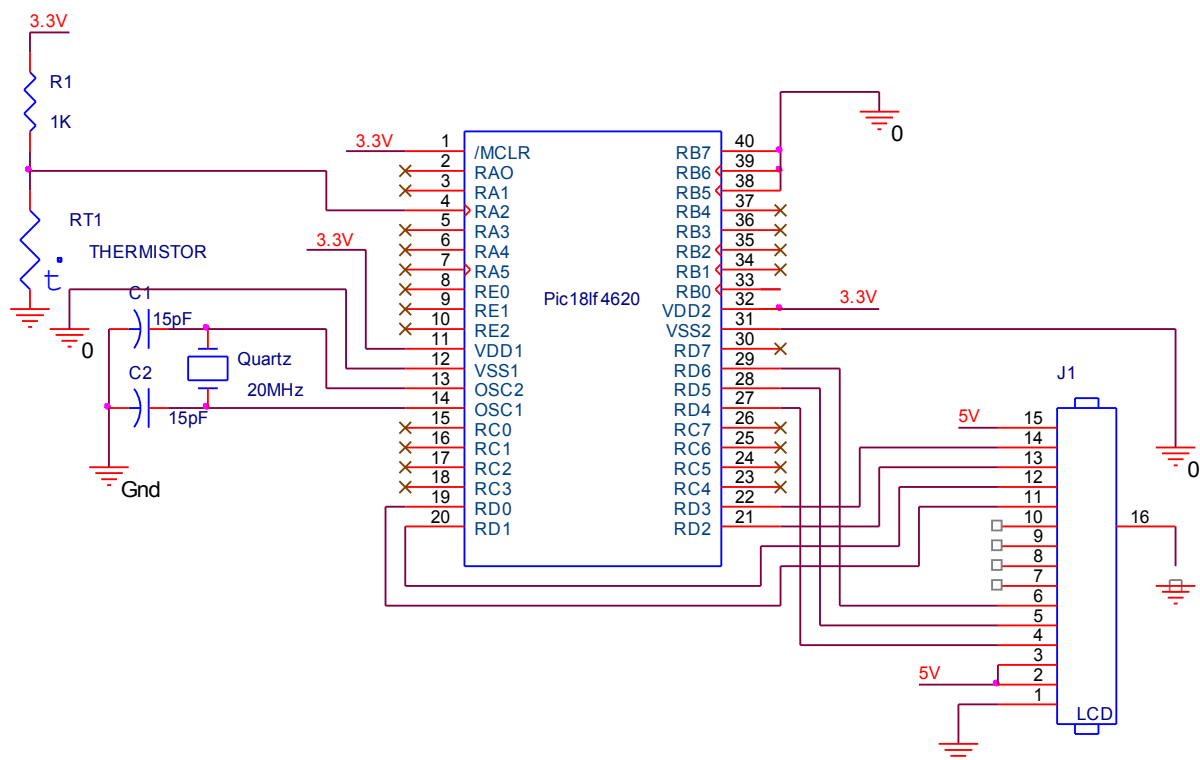


Figure 2 : Schéma Orcad du branchement de l'écran LCD

III. Programmation d'un PIC18f4620 avec MPLAB et C18

Ce projet a commencé par la familiarisation avec la programmation du PIC18f4620 à partir du logiciel de programmation MPLab et du module C18⁶. En effet, la mise en œuvre du microcontrôleur nous a demandé une étude des documents constructeurs afin d'utiliser les diverses fonctionnalités du microcontrôleur.

Cependant, la partie programmation n'a pu être traitée par faute de programmeur et de quelques soucis de paramétrages du logiciel.

IV. Réalisation des cartes électroniques

Pour des soucis de gain de place au niveau du volant, j'ai décidé, plutôt que de placer tous les éléments sur une seule et unique carte, de créer plusieurs petites cartes afin que la carte principale, intégrée au volant, ne soit pas surchargée. Ce procédé par modules interposés m'a permis de router ma carte microcontrôleur et LCD sur une seule face, un routage sur deux faces aurait rendu le système inutilisable tel quel en cas de pluie car les gouttelettes d'eau auraient créé des interconnexions entre les pistes.

IV.1. La carte microcontrôleur

La carte microcontrôleur peut-être vu comme le cerveau du système, c'est grâce à cette carte que les données vont être traitées puis affichées au pilote via l'écran LCD.

Cette carte est constituée de deux « modules » distincts que sont le microcontrôleur (partie traitement des signaux reçus et envoi des informations) et l'écran LCD (lecture des données relevées).

⁶ Compilateur C spécifique au microcontrôleur PIC18.

Comme on peut le voir sur le schéma, la carte est seulement composée du microcontrôleur, du LCD, d'un quartz (élément générant un signal carré de fréquence 2MHz afin de cadencer le microcontrôleur) et de borniers.

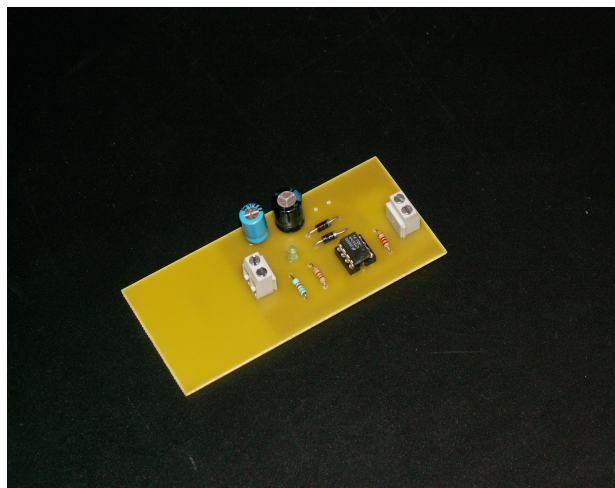
Le bornier deux points est relié à la carte alimentation afin de fournir la masse ainsi que la tension d'alimentation du microcontrôleur et du LCD (+5V).

Le bornier points va recevoir les fils comprenant toutes les informations nécessaires pour le traitement par le microcontrôleur, cela comprend:

- la tension image de l'accélération
- le signal de l'interrupteur de boost
- la tension image du courant dans la batterie
- la tension aux bornes de la batterie
- la tension image de la température du moteur
- le signal image de la vitesse du kart

IV.2.La carte alimentation

Cette carte va nous permettre d'alimenter le système en convertissant du 24V venant de la batterie en 5V par le biais d'une alimentation à découpage de type LM2574N-ADJ.



Le principe de cette alimentation à découpage a été inspiré d'un schéma d'alimentation d'internet, puis nous avons dû dimensionner nous même les éléments afin d'adapter l'alimentation pour notre application.

Nous pouvons remarquer qu'en sortie de ce régulateur à découpage, nous avons disposé une bobine en série avec un condensateur, cet ensemble d'éléments constitue un filtre afin d'obtenir une tension de sortie lissée.

On envoie sur l'entrée Feedback (littéralement Rétroaction) une tension choisie par l'utilisateur en agissant sur le rapport des résistances formant le pont diviseur, cette tension appliquée à l'entrée Feedback permet de choisir la tension souhaitée en sortie du régulateur.

Nous avons décidé de placer une diode électroluminescente en série avec une résistance afin de pouvoir voir rapidement, au cas où un problème surviendrait, si cela venait de l'alimentation ou non.

Cette carte comprend deux borniers, l'un permettant de « recevoir » la tension de batterie (un fil de masse et un fil de +24V), l'autre permet d'« envoyer » la tension d'alimentation vers les différents modules et la carte principale (un fil de masse et un fil +5V).

IV.3. Le capteur à fourche EE-SX1070

L'acquisition de la vitesse du kart va se faire par le biais d'un capteur à fourche, au départ, nous voulions utiliser un interrupteur à lame souple cependant l'usage du capteur à fourche permet entre autre d'augmenter la précision de la vitesse en fonction du nombre de parties claires et obscures. Le principe de ce capteur est décrit ci-après.



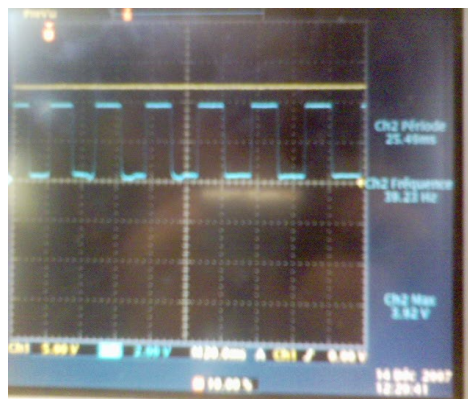
Comme on peut le voir sur la photo, le capteur est en forme de U. D'un côté, une diode électroluminescente va émettre de la lumière en continu, de l'autre côté, il y a un optocoupleur (transistor qui sature lorsqu'il « voit » la lumière émise par la diode), c'est cet optocoupleur qui va permettre de générer un signal rectangulaire de fréquence plus ou moins élevée selon la vitesse du kart.

En plaçant un disque entre les parties émission et réception du capteur, on va pouvoir « cacher » la lumière et bloquer le transistor.



Pour effectuer les tests, nous avons utiliser un disque en plastique, lorsque nous voudrons adapter le système au kart, nous devons alors confectionner un disque en métal afin d'offrir une plus grande résistance aux intempéries, le capteur sera quant à lui protégé par une boite afin d'éviter toute possibilité de projection d'eau ou de terre pouvant nuire à son bon fonctionnement. Nous avons prévu de monter le disque soit sur l'axe de la roue, soit de mettre un pignon supplémentaire et de mettre le disque sur l'axe du pignon seul.

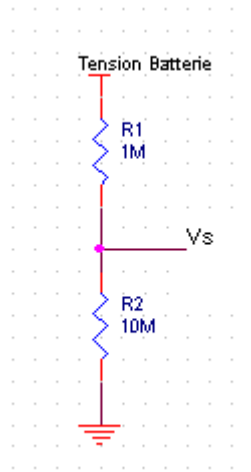
N'ayant pas eu le temps de réaliser un typon pour ce module, nous avons dû réaliser des tests sur plaque d'essai, nous n'avons donc pas de schéma mais des oscillogrammes ont été relevés.



Sur cet oscillogramme, nous pouvons observer le signal généré par le capteur à fourche, c'est à partir de ce signal et d'un calcul effectué par le microcontrôleur que nous pourrons déterminer la vitesse du kart. Ainsi, le programme inséré dans le microcontrôleur permettra de convertir la fréquence de ce signal en vitesse.

IV.4. Le module "Tension de batterie"

La mise en place de ce module d'acquisition est relativement simple, grâce à deux résistances montées en série, nous pouvons obtenir une tension "image" de la tension de batterie, tension "image" car cette tension pourra être traitée par le microcontrôleur, contrairement à la tension directe d'environ 48V.



Grâce à la relation du pont diviseur de tension: $V_s = V \cdot R_2 / (R_1 + R_2)$, on obtient la tension image de la tension batterie, de plus, en choisissant des résistances élevées (1M Ohm), on limitera le courant de perte dû à la mesure.

IV.5. Le module "Capteur de courant"

La détermination du niveau de charge de la batterie est réalisée à partir d'un capteur de courant qui est déjà présent sur le karting électrique. Ainsi, cela nous permet de récupérer directement la donnée envoyée par le capteur afin de la traiter par le microcontrôleur.



V. Conclusion

Le projet réalisé durant ce troisième semestre m'a permis de faire face à des problèmes et de tenter de les résoudre. En effet, à partir du cahier des charges établi, j'ai pu dresser une liste des tâches à effectuer et comment les effectuer et malgré l'apparente facilité qui apparaissait au début du projet, de nombreux problèmes ont surgi, ralentissant le développement du système. Je n'ai pas pu achever ce travail au cours du troisième semestre et l'idée d'utiliser un microcontrôleur PIC18F4620 afin de gérer cet affichage a été abandonnée, faute de connaissance sur le sujet.

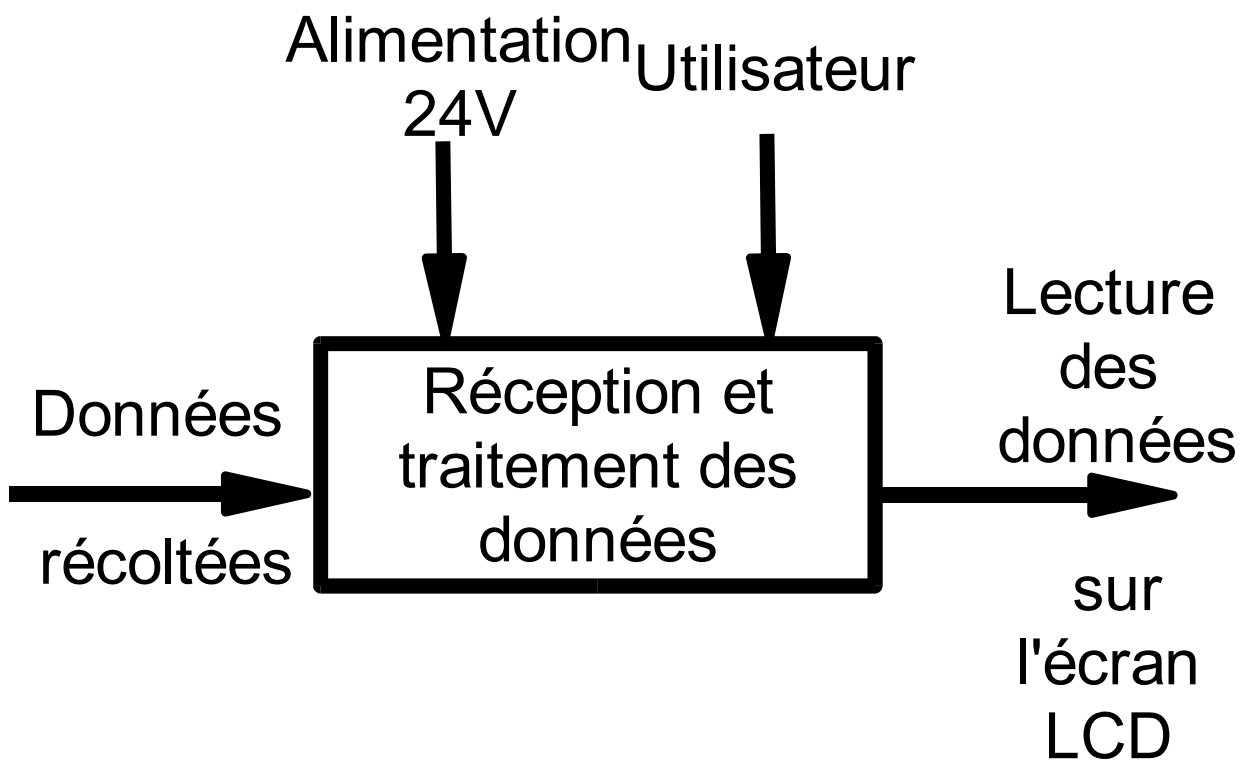
Bibliographie

Catalogue de composants électronique FARNELL

Catalogue de composants électronique RadioSpares

Site Web de Mr LEQUEU T.: <http://www.thierry-lequeu.fr>

Annexe 1:
Schéma Fonctionnel de niveau 1



Annexe 2:

Planning prévisionnel/effectué

	41	42	43	44	45	46	47	48	49	50	51
Choix du sujet											
Mise en place planning et cahier des charges											
Recherche de solution											
Etude du système											
Recherche des composant											
Début de la programmation											
Test des composants électrique											
Poursuite de la programmation											
Réalisation de la carte											
Finalisation du programme											
Phase de test											
Debbugage (si necessaire)											
Rédaction du dossier											

Planning prévisionnel

 Planning réel

Contrairement au planning, toute les phases de programmation n'ont pû être traitées.