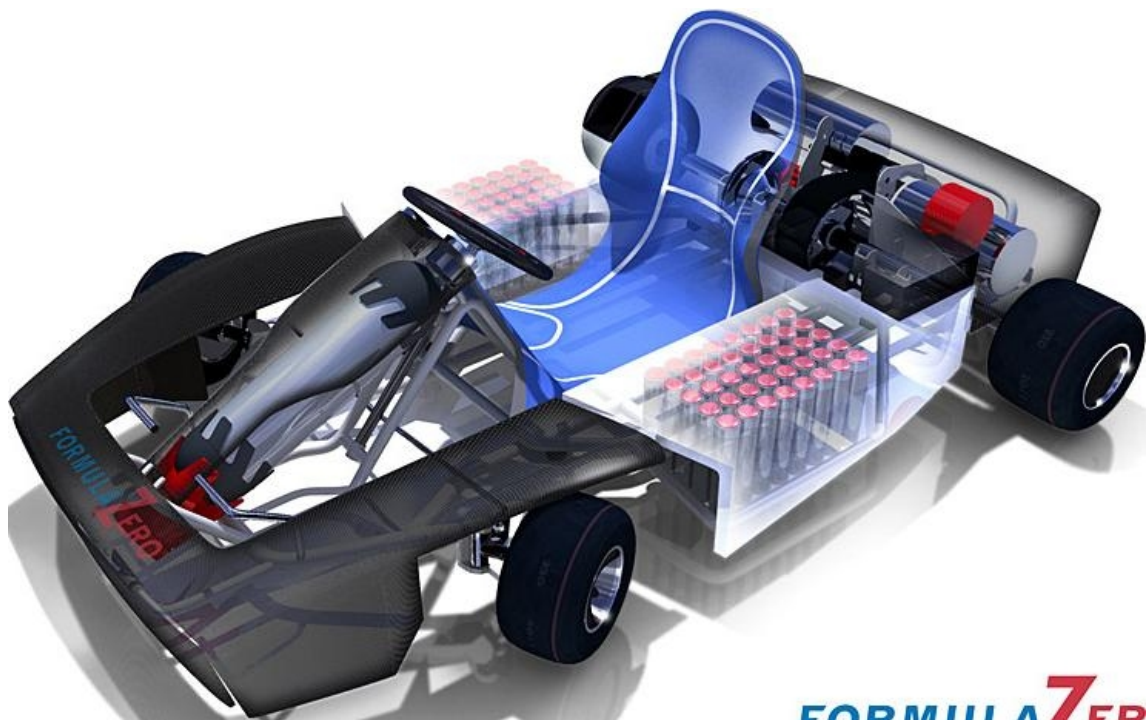


## Gestion des feux de signalisation pour une voiture électrique



Cheikh abba DIEME & Boura MOUZAIR  
2er Année  
Groupe K3B  
200-200

Enseignants

LEQUEU Thierry  
GLIKSOHN Charles

Université François-Rabelais de Tours  
Institut Universitaire de Technologie de Tours  
Département Génie Électrique et Informatique Industrielle



## **Gestion des feux de signalisation pour une voiture électrique**

Cheikh abba DIEME & Boura MOUZAOIR  
2er Année  
Groupe K3B  
200-200

### Enseignants

LEQUEU Thierry  
GLIKSOHN Charles

# Sommaire

INTRODUCTION.....	4
1. Présentation du projet.....	5
1.1. Cahier des charges.....	5
1.2. Schémas fonctionnels.....	6
1.3. Planning Prévisionnel.....	8
2. Étude théorique.....	8
2.1. Les lampes.....	8
2.2. L'ATMega8535.....	9
2.3. Les transistors.....	10
2.4. L'alimentation 5 volts.....	10
2.5. Les feux de stop et la photodiode.....	11
2.6. Liste des composants.....	13
3. Réalisation et tests.....	13
3.1. Schéma de la carte de contrôle.....	14
3.1.1. Logiciel Orcad Capture.....	14
3.1.2. L'ATMega8535.....	15
3.1.3. L'alimentation à découpage.....	15
3.1.4. Les entrées.....	16
3.1.5. Les sorties.....	18
3.2. Le typon.....	19
3.2.1. Le logiciel Orcad Layout plus.....	19
3.2.2. Les contraintes.....	19
3.2.3. Le résultat final.....	20
3.2.4. Le test d'alimentation.....	21
3.3. La programmation.....	22
3.3.1. Le programme de test.....	22
3.3.2. Programme implanté sur les ports entrées et sorties.....	22
3.3.3. Programme de la fonction MLI.....	23
3.3.4. Planning finale.....	26
Résumé.....	28
Index des illustrations.....	29
Bibliographie.....	30
Annexes :.....	31

# INTRODUCTION

Au cours de ce troisième semestre nous avons choisi en binôme un projet lié au karting électrique dans le cadre du cours d'étude et réalisation. Ce rapport sera suivi d'une présentation orale. Ce projet a pour thème la gestion de l'éclairage des feux de signalisation d'une voiture électrique. Il consiste à réaliser des feux de signalisation avant et arrière pour un kart électrique. Pour cela, nous allons réaliser des feux de STOP progressive en fonction de la pression exercée sur la pédale de frein, les feux avant progressif en fonction de la lumière ambiante et enfin les clignotants et feux de détresses. Nous travaillerons à partir de recherches menées précédemment par d'anciens élèves.

Ce rapport présentera tout d'abord le projet, une étude théorique et enfin la réalisation et les tests effectués.

# 1. Présentation du projet

Le projet consiste à réaliser des feux de signalisation avant et arrière du Kart électrique. Pour cela, nous allons réaliser les feux de STOP progressif en fonction de la pédale de frein, les feux avant progressif en fonction de la lumière ambiante et enfin les clignotants et feux de détresses. Pour cela nous essayerons de renouveler la carte de fonctionnement ( prototype existant) et nous vérifierons son bon fonctionnement. Pendant les premiers tests nous prendrons l'ancienne programmation de l'ATMega8535 pour essayer de faire fonctionner le nouveau prototype. Si la programmation est non compatible nous en formulerons une nouvelle qui nous permettra de faire valider le fonctionnement de notre prototype. Pour mieux réaliser ce projet nous avons créé un cahier des charges dans lequel nous marquerons toutes les étapes et les fonctionnalités qui serviront à bien mener nos recherches, puis un planning qui montrera l'évolution de notre projet du début à la fin du projet.

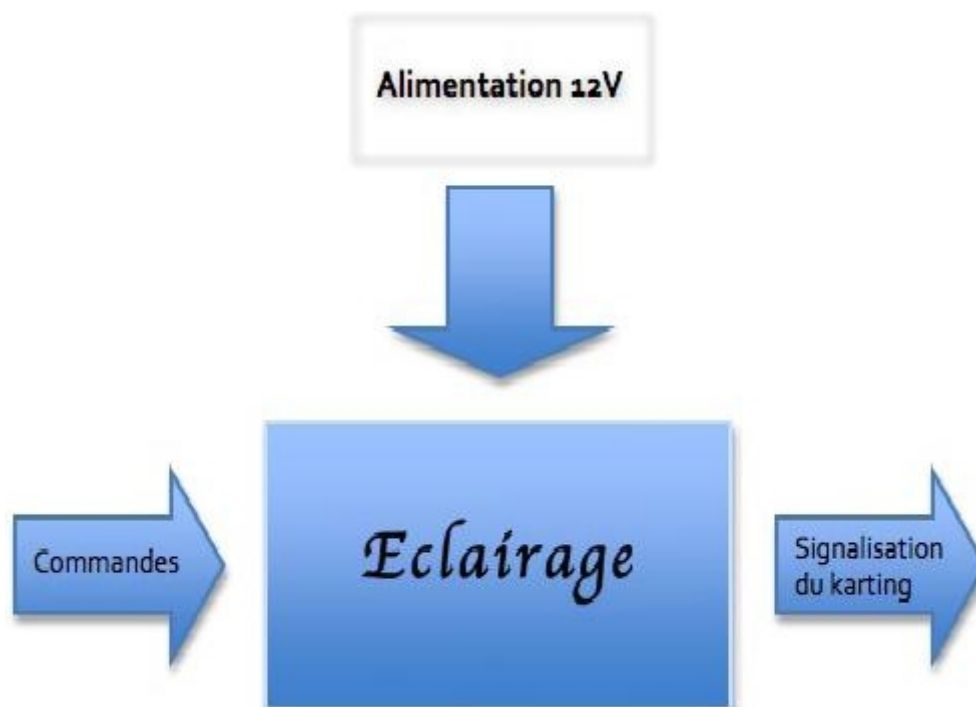
## 1.1. *Cahier des charges*

### ■ Contraintes

- Alimentation continue 0/12V grâce aux batteries du Kart.
- Micro contrôleur ATMega8535
- Étude de l'ancienne carte
- Réalisation d'un circuit de commande
- Interrupteur deux positions (manuel, automatique)
- Interrupteur trois positions (feux de croisement, feux de position, repos)
- Interrupteur trois positions (clignotant gauche, clignotant droite, repos)
- Interrupteur deux positions ( warning, repos)
- Interrupteur deux positions ( marche, arrêt)
- Signalisation des commandes par des LEDS
- Programmation du micro contrôleur
- Feux de position

- Feux de croisement
- Clignotants
- Feux avant progressif en fonction de la lumière
- Feux de recul
- Feux de stop
- Warnings
- Feux de détresses
- Utilisation du logiciel Orcad Layout
- Code vision AVR programmation Atega8535
- Coût et encombrement limités.
- Le prix du système doit rester dans le budget accordé par l'IUT

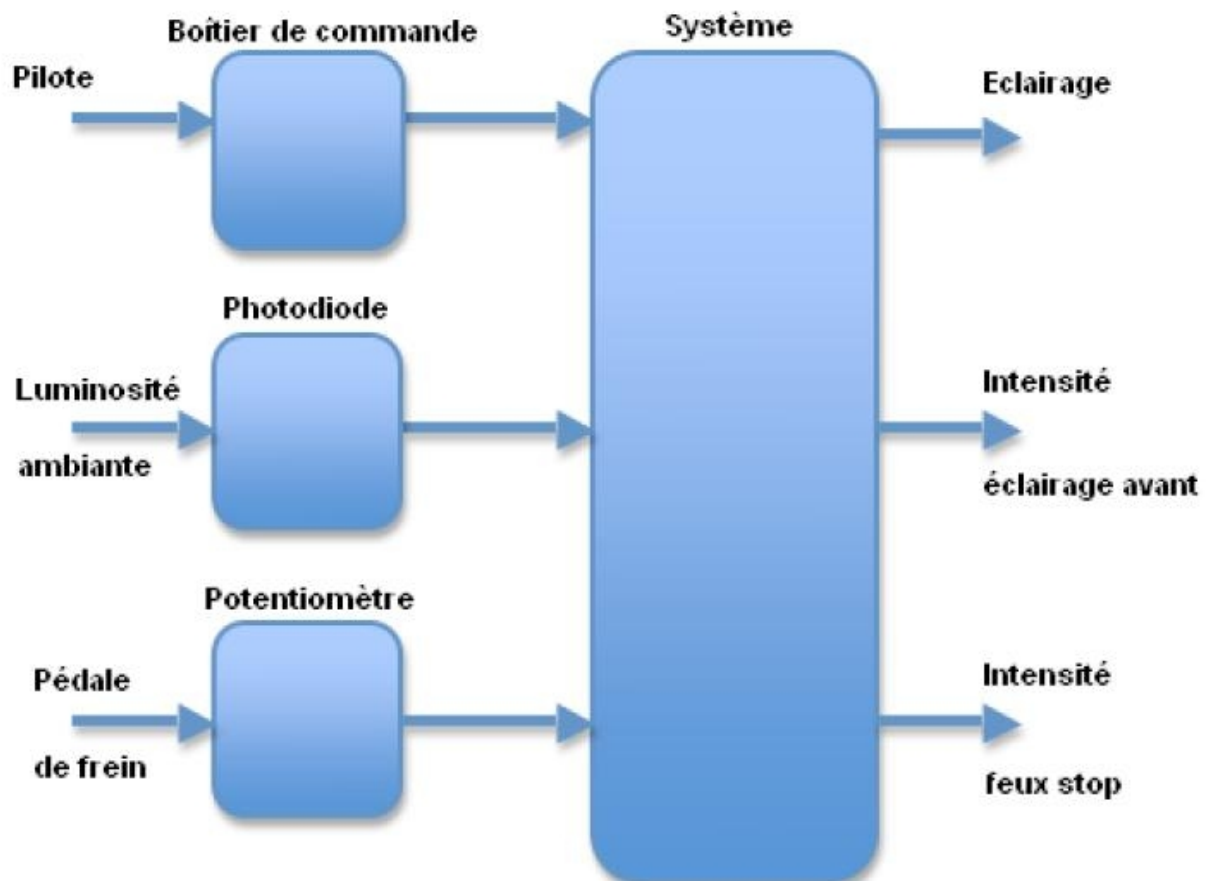
## 1.2. Schémas fonctionnels



*Illustration 1: schéma fonctionnel de niveau 1*

Dans ce schéma on a un système nommé éclairage, et ce système pour fonctionner a besoin d'une alimentation de 12V fournie par les batteries du kart électrique et d'un autre

système, associés aux commandes. L'ensemble de ces trois systèmes fournissent la signalisation du karting.



*Illustration 2: schéma fonctionnel de niveau 2*

### 1.3. Planning Prévisionnel

Afin de garder une bonne organisation de l'évolution de notre projet, nous avons pu faire un planning prévisionnel de répartition des tâches.

Planning Prévisionnel																								
Taches/Semaines	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24							
Prise de connaissance du sujet	P																							
Recherche d'information	R																							
Élaboration du cahier des charges et du planning	P	P																						
Formation Orcad	R	R																						
Recherche de solution		P	P	P																				
Réalisation du typon		R	R	R					P															
Test et vérification										P	P													
Rédaction du document de synthèse													P	P	P									
Essai du prototype	R	R	R	R	R	R			R	R	R		R	R	R	R	R							
Remise des dossiers																		P						
Soutenance orale																		R						
<b>P: prévisionnelle</b>							<b>R: réelle</b>																	

Illustration 3: planning prévisionnel

## 2. Étude théorique

### 2.1. Les lampes

L'éclairage de la voiture doit fournir une visibilité maximale au conducteur de la voiture électrique. Pour cela nous nous sommes basés sur l'étude faite sur 3 différents types de lampes faites par d'anciens étudiants. Cette étude a permis d'établir un comparatif sur un tableau les capacités de chacune de ces lampes.



Type de lampe	Tension d'alimentation	Puissance	Luminosité
Lampe à LED	12V	1W	11 Klux
Lampe allogène	12V	20W	40 Klux
Prototype	30V	1W	7 Klux

*Illustration 4: comparatif des lampes disponibles*

Le choix des lampes se fera donc en fonction de l'usage et de l'intensité lumineuse nécessaire. Nous avons donc fait le choix suivant. Cette étude a permis de démontrer que les lampes à LED délivrent une faible luminosité. Ceci n'est pas en accord avec notre cahier des charges car les lampes doivent être visibles des utilisateurs à une distance suffisante. Pour cela nous avons choisis les lampes de types allogènes qui nous fournissent un courant de 1,6A . Pour avoir ce résultat on a fait le calcul suivant :

$$I = P/U = 20W/12V = 1,6A.$$

Ces lampes nous fournissent plus de luminosité que les deux autres types de lampes. Donc ce modèle de lampes allogènes pourrait convenir à peu près à ce que l'on recherche. En raison des différentes luminosités obtenues lors des tests nous utiliserons les LED pour les clignotants, les feux de reculs et feux arrières, alors que les lampes allogènes seront plus adaptées pour les feux de croisements, les feux avant progressif. Voici le calcul fait pour savoir quelles intensités doivent supporter les transistors :

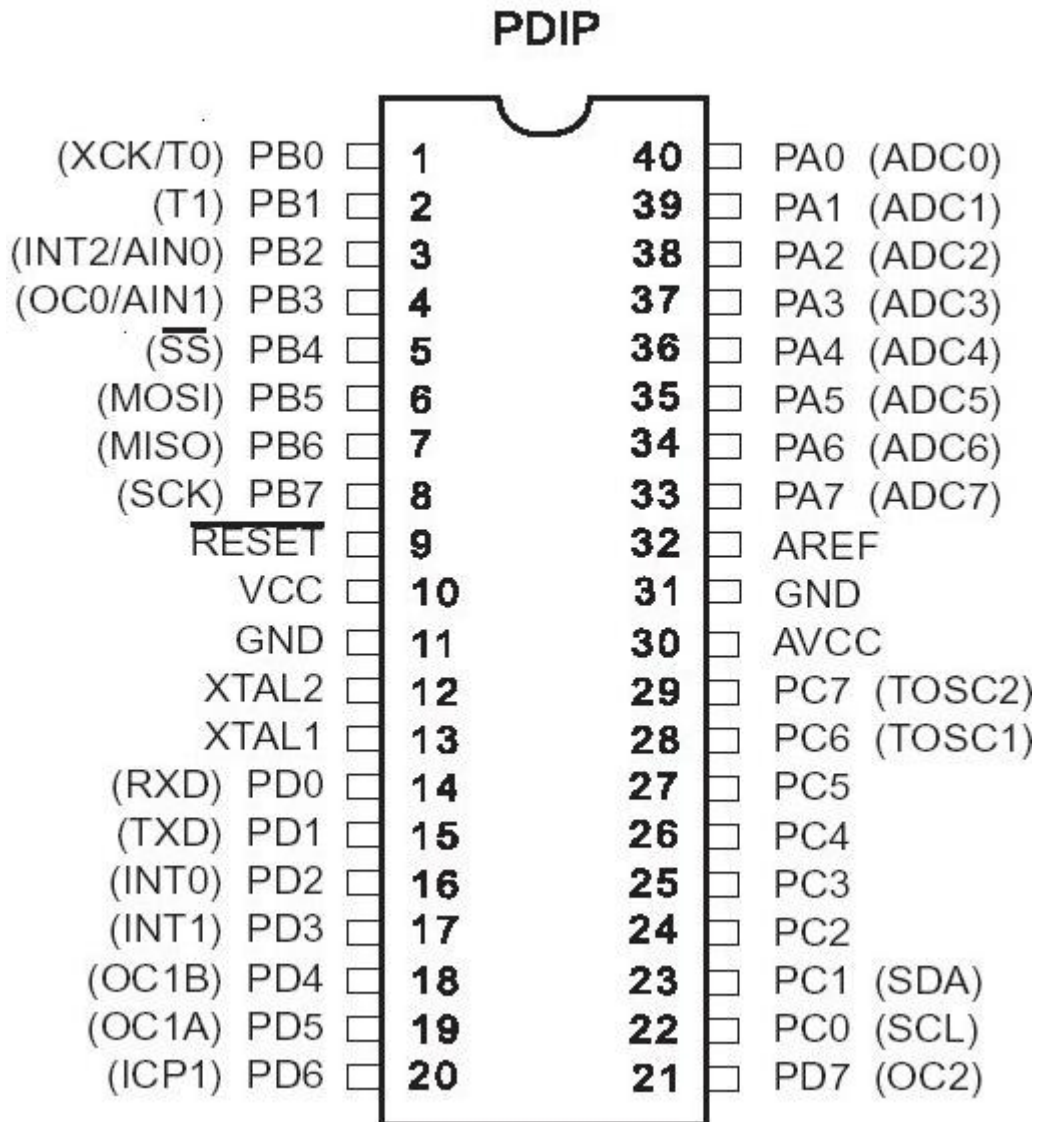
$$I = P/U = 1W/12V = 20mA.$$

Ainsi les LEDS de 1W nous permettrons de préserver l'autonomie des batteries grâce à leur faible consommation en courant.

## **2.2. L'ATMega8535**

Pour pouvoir mieux gérer la commande du projet, nous avons besoin de définir ses étapes aux différentes entrées. Pour cela nous avons choisi un micro-contrôleur de 8 bits offrant, 4 ports de 8 entrées/sorties chacun. Les entrées/sorties sont paramétrables selon

nos besoins. En plus il possède 3 sorties pouvant être programmées en modulation. Ce micro-contrôleur représentera le centre de notre étude et c'est à l'intérieur de lui que sera implanté le programme.

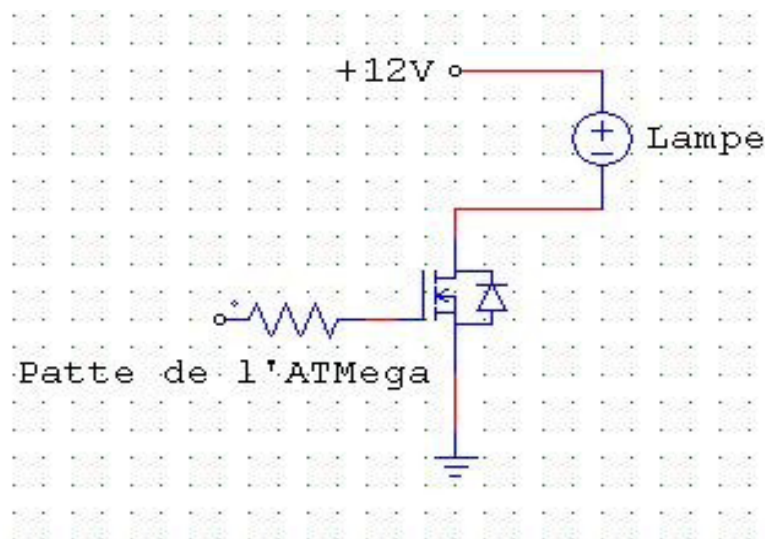


*Illustration 5: schéma du micro-contrôleur ATmega 8535*

### **2.3. Les transistors**

Nous avons choisi des transistors MOSFET connu sous le nom **IRL2203NPBF** pour la commande des lampes. Nous utiliserons ces transistors car ils peuvent supporter un

courant au niveau de la grille d'au moins 1,6 A. Ils sont branchés de la manière suivante :



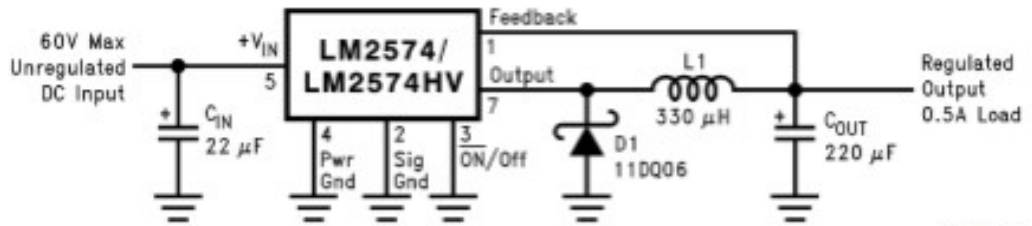
*Illustration 6: schéma des transistors MOSFET*

Les transistors jouent le rôle d'interrupteur lorsque la sortie du micro-contrôleur est activée. Il possède 3 pattes : la Grille(patte 1) est reliée ici à la résistance, le Drain(patte 2) est relié à la lampe puis à une tension de 12V, et la Source (patte 3) est relié à la masse. Ici la Grille sera reliée directement à L'ATMega8535 et recevra des impulsions. Ces impulsions au niveau de la grille vont occasionner le contact entre le Drain et la Source et ainsi permettre à la lampe d'être alimenté par une tension de 12 volts. Si la sortie du micro-contrôleur est à 0V, la grille du transistor ne recevra plus d'impulsions et cela occasionnerait l'ouverture du contact Drain Source, et l'extinction de la lampe .

## **2.4. L'alimentation 5 volts**

La carte est alimentée sur 12 volts fournis par les batteries de la voiture électrique. Cette tension continue sera convertie en une tension continue de 5 volts grâce à une alimentation à découpage de type BUCK. Cette alimentation à découpage sera reliée à un régulateur de tension à découpage LM2575. Avec l'apport du régulateur de tension on aura à sa sortie une tension continue de 5 volts et cette tension permettra d'alimenter le micro-contrôleur ATMega8535. Ce montage permet de réduire les pertes de puissance que nous pourrions avoir si nous avions utilisé un composant linéaire tel qu'un régulateur. Voici un type de montage LM2575 donné par le fabricant. Celui-ci va nous permettre d'obtenir une tension continue de 5 volts à partir d'une tension de 12 volts comme vue en

### Typical Application (Fixed Output Voltage Versions)



Note: Pin numbers are for 8-pin DIP package.

DS011394-1

Illustration 7: schéma du régulateur de tension

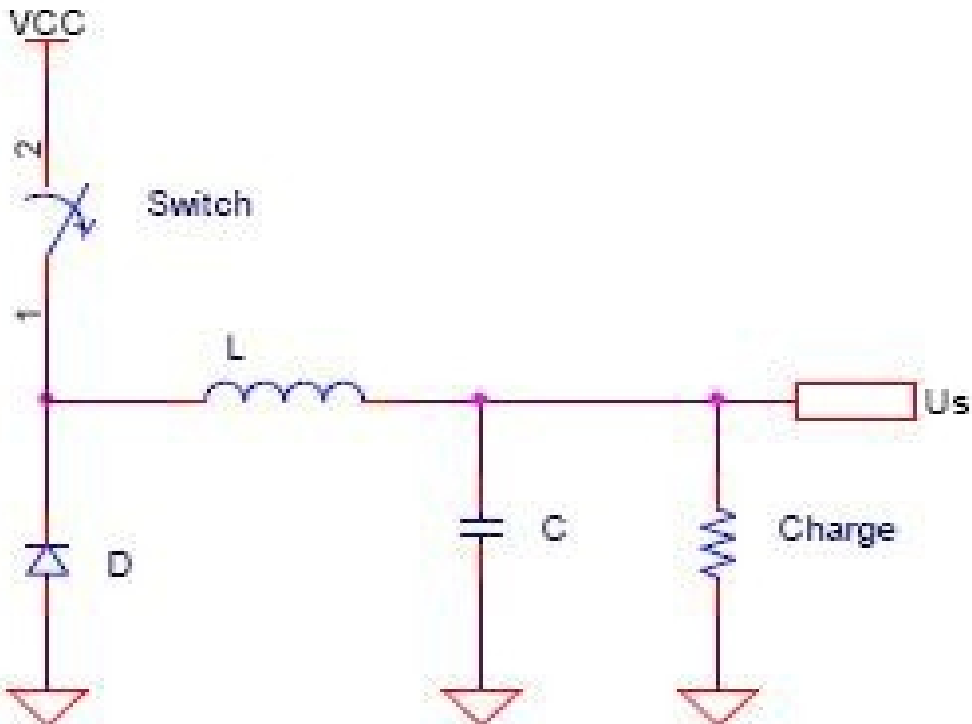


Illustration 8: schéma d'une alimentation de découpage de type BUCK

## 2.5. Les feux de stop et la photodiode

La commande des feux de stop sera réalisé grâce à une pédale de frein. Cette pédale est équipée d'un potentiomètre mécanique. La luminosité des feux de stop va varier au fur et à mesure de la pression exercée sur cette pédale. Ce potentiomètre nous permettra

d'obtenir une tension variable qui nous est fournie en entrée de L'ATMega8535 afin de varier la valeur moyenne de la tension d'alimentation des feux arrières. La résistivité de ce potentiomètre varie en fonction d'un mouvement mécanique. Sa valeur va diminuer en fonction de la pression qu'il subit, et plus la pression est forte et plus sa valeur deviendra faible.

La photodiode permettra la variation de l'éclairage en fonction de la luminosité ambiante. Donc on verra l'évolution de l'éclairage de nos phares en fonction du moment de la journée.



*Illustration 9: pédale de commande des feux de stop*

## 2.6. Liste des composants

Nom	Référence	Empreinte	Valeur	Quantité	Prix unitaire	Prix total
<b>Résistances</b>	R9, R11, R12, R13, R14, R15, R16, R17, R18, R19	RC04L	1,5k	10	0,05	0,5
'	R4, R5	RC04L	10k	2	0,05	0,1
'	R6, R8	RC04L	820 ohms	2	0,05	0,1
'	R1, R2, R7, R10	RC04L	1k	4	0,05	0,2
<b>Condensateurs</b>	C1, C7	RADIAL06	10µF	2	0,45	0,9
'	C2, C8	CK06	100nF	2	0,51	1,02
'	C3, C4	CK06	22pF	2	0,51	1,02
'	C5	RADIAL08	100µF	1	0,45	0,45
'	C6	RADIAL06L	470µF	1	0,45	0,45
<b>LEDs</b>	D1	LED3	3mm	1	0,09	0,09
'	D4	LED3	2mA	1	0,09	0,09
<b>Diodes</b>	D2	D041	1N5819	1	2	2
'	D3	D041	1N5819	1	2	2
<b>Connecteurs</b>	J1, J2, J3, J4, J5	10 broches	Molex – 5569	5	3,31	16,55
'	JP1	2 x 5 broches	CON ISP	1	2,4	2,4
'	JP2	2 broches	Alim	1	2,3	2,3
'	P1	DB15 femelle		1	2,53	2,53
<b>Inductances</b>	L1	RADIAL06L	10µH	1	0,73	0,73
'	L2	RADIAL06L	47µH	1	0,73	0,73
<b>Transistors</b>	Q1, Q2, Q3, Q4, Q5, Q6, Q7, Q8, Q10, Q11	TO220	IRL2203N	10	1,73	17,3
<b>Quartz</b>	Q9	HC18UV	16Mhz	1	0,39	0,39
<b>Réseau de résistance</b>	R3	09PL1	8 x 4,7k	1	0,28	0,28
<b>µContrôleur</b>	U1	40DIP600L	ATMega8535	1	5,93	5,93
<b>Régulateur</b>	U2	08DIP300L	LM2574M	1	1,8	1,8
<b>Boitier mâle</b>			484-1861	10	5,5	55
<b>Boitier femelle</b>			484-1798	10	4,64	46,4
<b>Contact à sertir femelle</b>			172-9134	10	0,17	1,7
<b>Contact M.5558 TL</b>			172-9140	10	0,23	2,3
<b>Total</b>				<b>94</b>		<b>165,26</b>

Illustration 10: schéma représentant les frais de notre projet

## 3. Réalisation et tests

Après avoir fait l'étude complète qui nous a permis de trouver la totalité des composants nécessaires pour le bon fonctionnement du projet, nous avons entamé la réalisation du prototype.

Grâce à l'aide de M.LEQUEU, nous avons pu ajouter certains composants et modifier certains autres sur la carte initiale réalisée par d'anciens étudiants. Sur cette nouvelle carte on a augmenté le nombre de transistor **Mosfet IRL2203N**, puis remplacé les anciens connecteurs par des connecteurs **MOLEX 5569**. Après tout ces changements, nous avons pu réaliser la nouvelle carte avec le logiciel **Orcad**.



### 3.1. Schéma de la carte de contrôle

Tout le travail effectué dans la partie théorique, nous a permis de faire un schéma électrique final de la carte de contrôle. Maintenant il faut le saisir dans un logiciel dédié à cette fonction. Les différentes étapes de cette saisie sont les suivantes :

- la saisie du schéma sous Orcad Capture
- la recherche et la saisie des empreintes
- la mesure des dimensions de la carte
- le placement des composants sous Orcad Layout
- le routage des pistes
- Vérification avant impression du typon pour réalisation
- Soudure des différents composants sur la carte
- Essais de fonctionnement de la carte

#### 3.1.1. Logiciel Orcad Capture

Le logiciel Orcad nous permet de saisir le schéma électrique et de le transformer en un fichier Netlist. Ce fichier contient à lui seul la totalité des liaisons entre les composants.

Sans la pratique du cours, le logiciel aurait été difficile à manier. Ainsi pour le tracer on place en premier lieu tous les composants qui nous sont nécessaires sur la feuille de saisie et pour terminer on crée les liaisons entre les différents composants.



Illustration 11: logiciel Orcad

### 3.1.2. L'ATMega8535

Ce schéma correspond au micro-contrôleur ATMega8535, et celui ci commandera les sorties de la carte en fonction de l'état de ses entrées et du programme implanté au sein du micro-contrôleur, la programmation s'effectuant via le connecteur JP1( 2 x5 broches), également à gauche sur le schéma. Le quartz présent sert de base de temps au micro-contrôleur, ce qui permet entre autre, l'utilisation de la fonction de modulation de largeur d'impulsion.

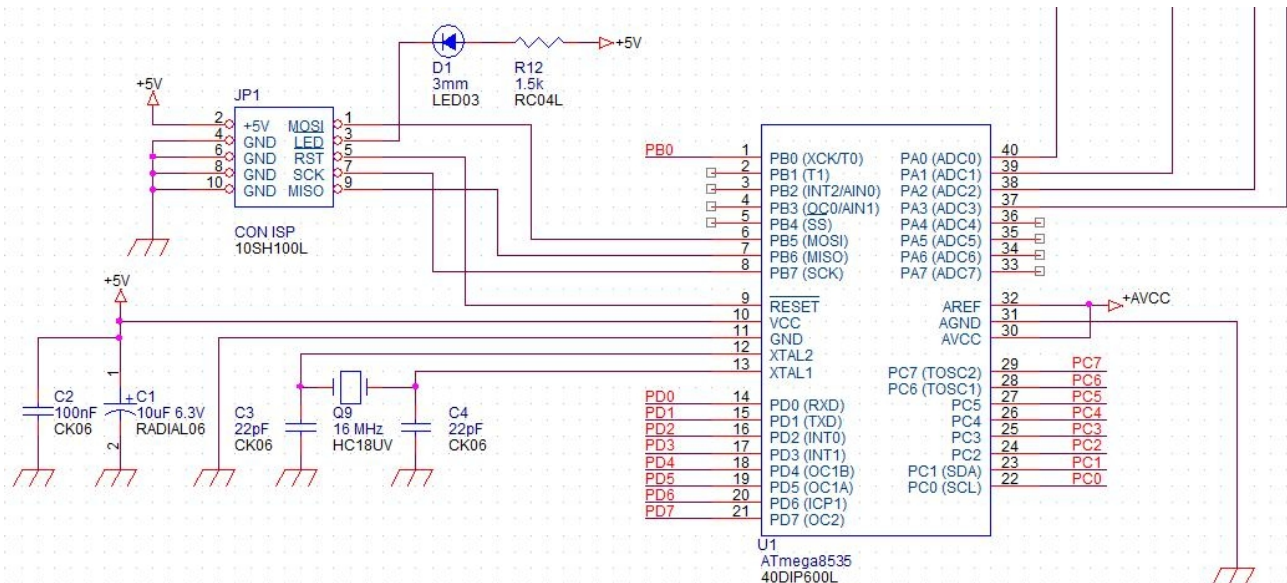


Illustration 12: schéma capture de L'ATMega 8535

### 3.1.3. L'alimentation à découpage

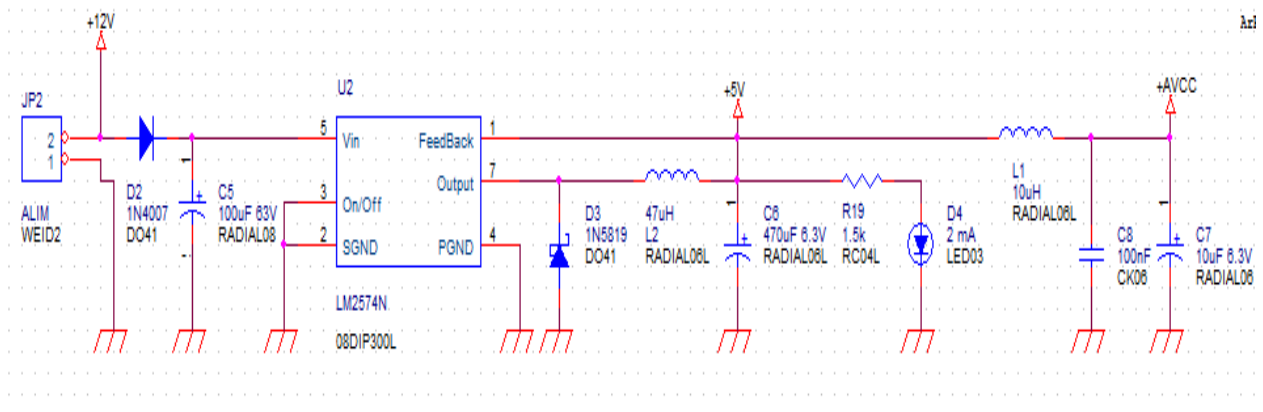


Illustration 13: schéma de capture de l'alimentation à découpage

L'utilisation d'une alimentation à découpage de type BUCK permet d'obtenir une tension



de sortie inférieur à la tension d'entrée tout en restant continue. Grâce au LM2574N, nous pouvons fournir une tension continue de 5 volts à L'ATMega8535 à partir d'une tension continue de 12 volts délivrées par les batteries du kart électrique. Cette alimentation convertit une tension continue en une autre tension continue de plus faible valeur.

### 3.1.4. Les entrées

Ainsi pour mieux respecter notre cahier de charge, nous avons besoin de 8 entrées qui sont :

- Signal marche avant et arrière
- Clignotants ( 2 entrées)
- Feux de position / route
- Feux de détresse ( warnings)
- Mode automatique / manuel
- Interrupteur général
- Potentiomètre de frein
- Photorésistance

Pour l' interrupteur d'entrée nous avons choisi un connecteur de type DBUB femelle avec 15 broches. Ce connecteur sera relié à un boîtier annexe et permettra ainsi la commande de tout l'éclairage et de toute la signalisation. Le pont de résistance permet d'assurer le

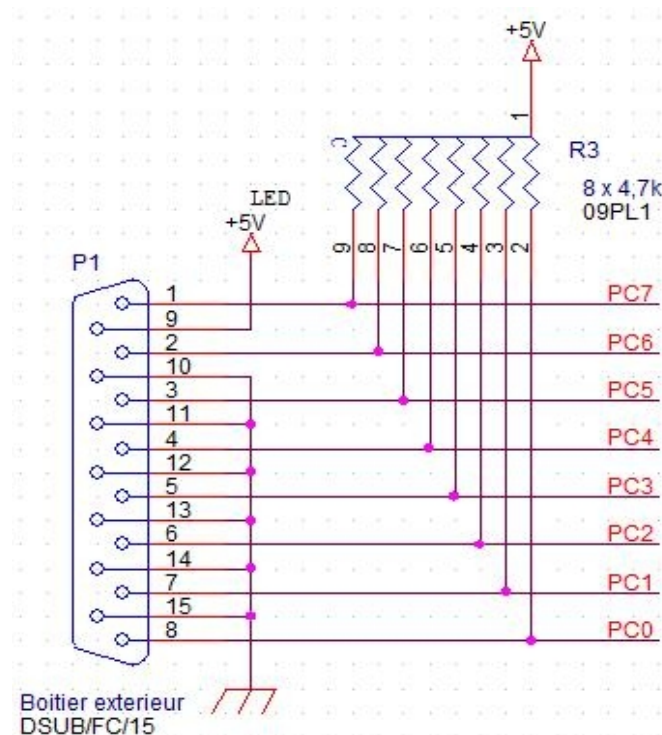


Illustration 14: schéma capture des entrées tout ou rien

passage du courant du boîtier vers L'ATMega8535.

Par contre le deuxième connecteur J1 est composé de 2x5 broches pour les 2 entrées variables.

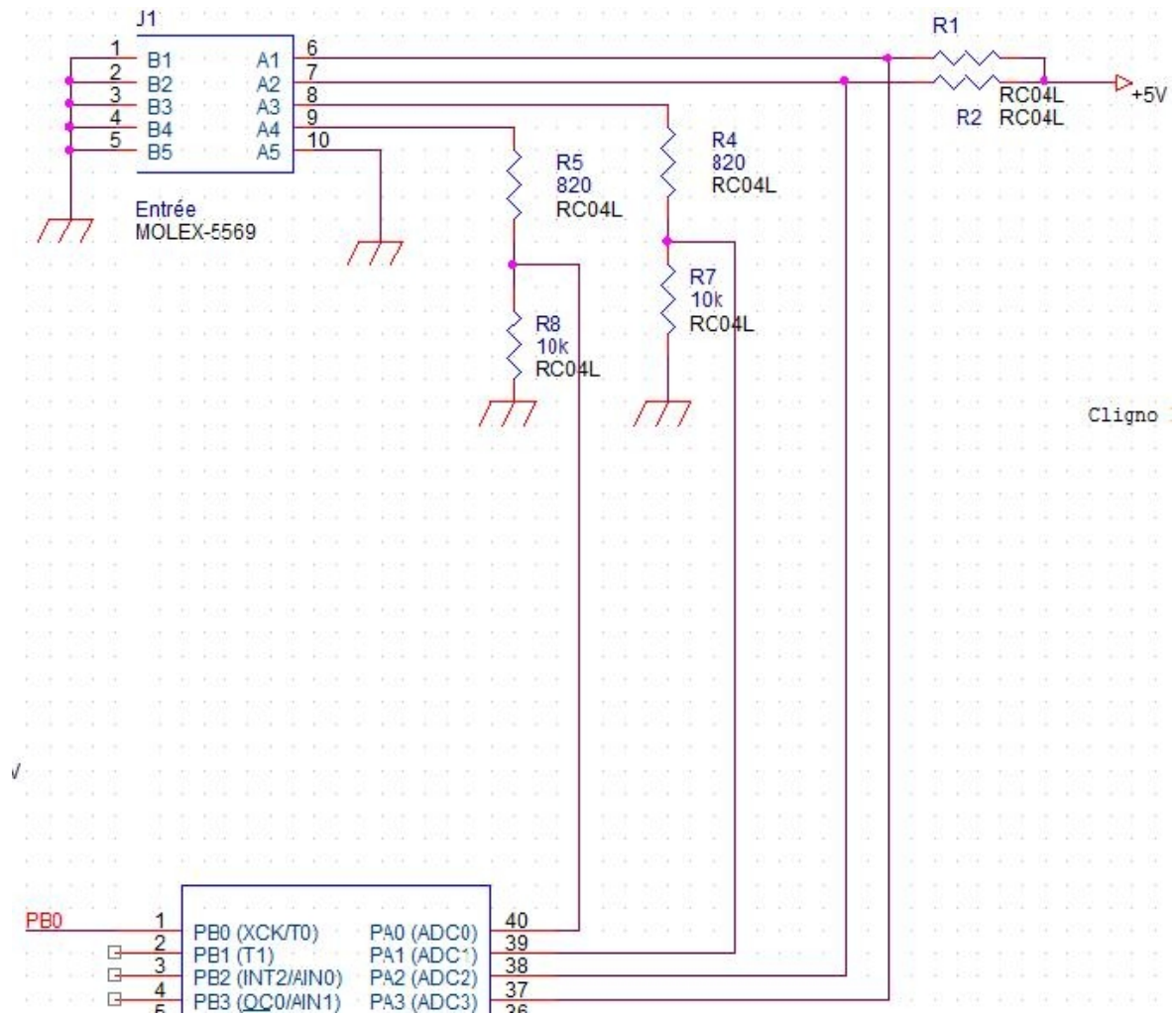


Illustration 15: schéma capture des entrées variables

### 3.1.5. Les sorties

Sur ce schéma sont alignés 4 connecteurs Molex 5569 qui représentent les sorties, qui seront par la suite reliés aux lampes. L'ensemble des 4 sorties est relié à 10 transistors qui auront pour rôle de commander les lampes.

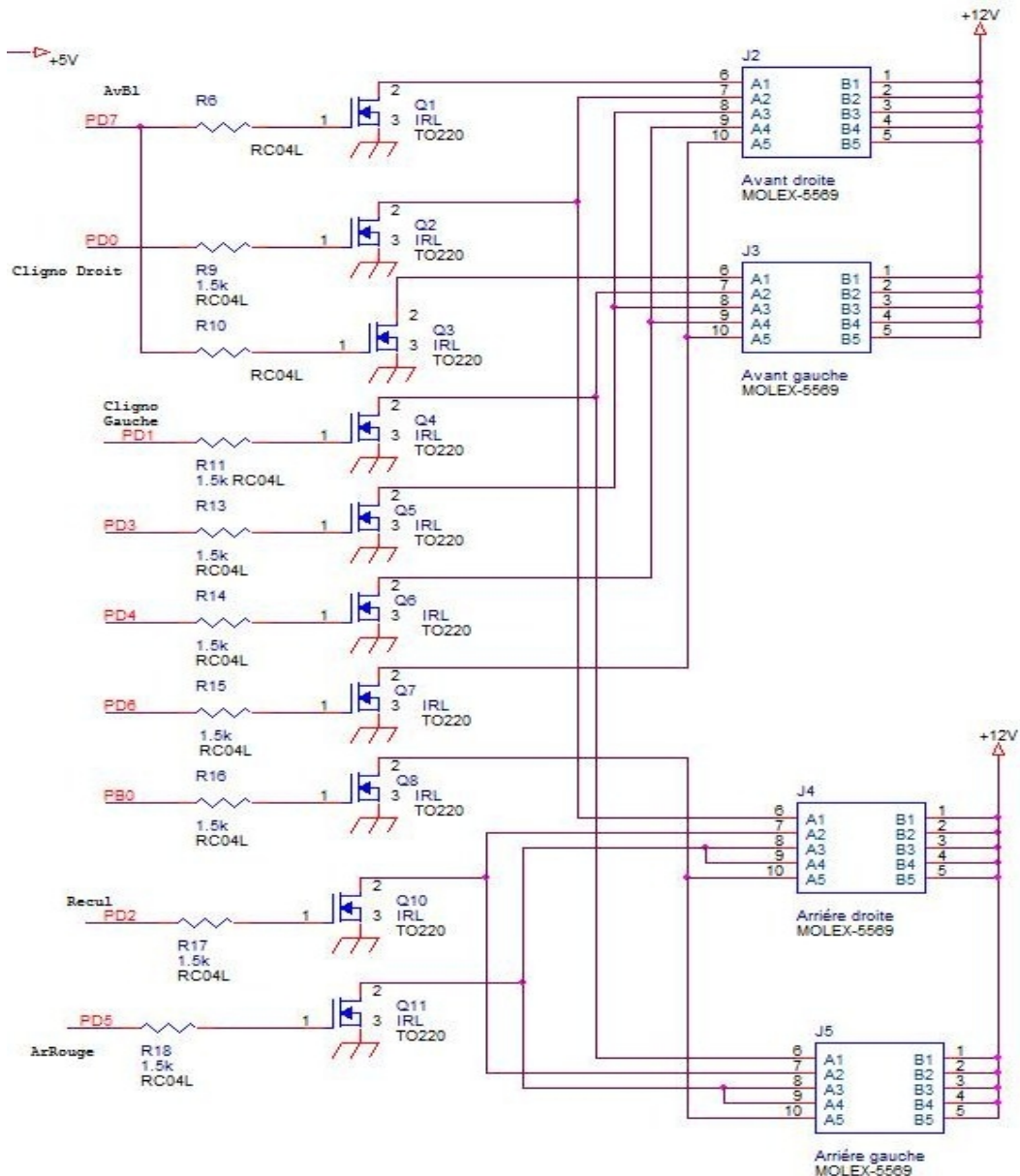


Illustration 16: schéma capture des sorties du montage

## 3.2. Le typon

Une fois le schéma électrique réalisé, il faut créer la Netlist dans Capture, pour pouvoir l'envoyer dans le logiciel Layout afin de réaliser le typon général que nous utiliserons pour faire notre carte.

### 3.2.1. Le logiciel Orcad Layout plus

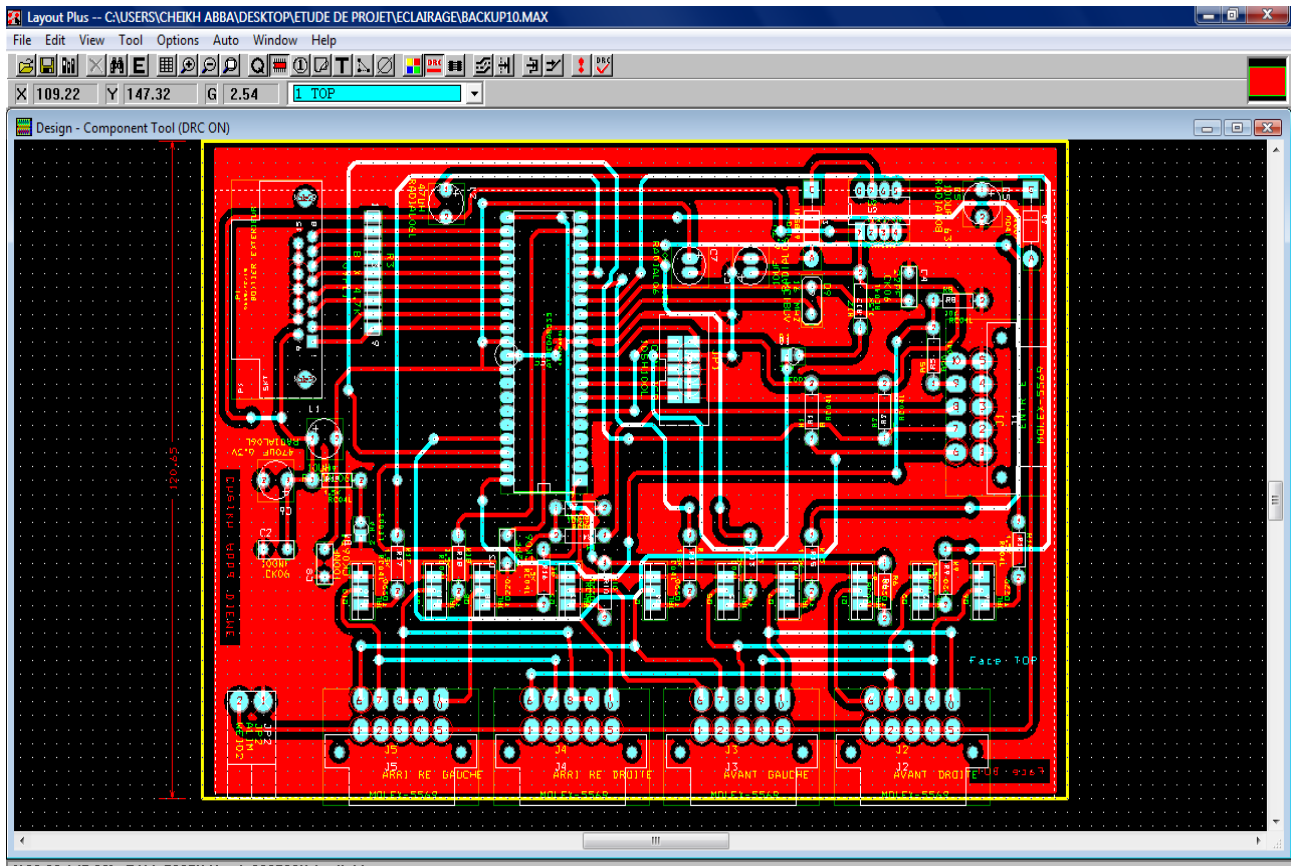


Illustration 17: impression d'écran du logiciel Layout plus

### 3.2.2. Les contraintes

Pour mieux réussir notre typon, il y a un certain nombre de règles à respecter. Tout d'abord on a utilisé une carte qui fait 180 mm sur 120 mm pour pouvoir mieux placer les composants et avoir une mise en œuvre facile. Nous avons opté pour une carte double face ce qui permet de l'améliorer et aussi de l'aérer. Les composants doivent être placés de telle sorte qu'on puisse avoir des connections les plus courtes possibles afin de pouvoir limiter les problèmes de compatibilité électromagnétique.

Tout d'abord il faudra en priorité relier les pistes transportant un fort courant. Les connecteurs doivent être placés de tels sortes qu'ils soient plus facilement accessibles par les câbles électriques qui seront par la suite reliés aux différentes lampes de la voiture électrique qui font l'objet de notre projet. Le connecteur de programmation doit être facilement accessible.

Les transistors doivent être tout près des connecteurs correspondants. Le quartz devra être tout près du micro-contrôleur, ainsi que l'alimentation de découplage afin d'éviter toutes sortes de perturbations entre ces éléments qui sont essentiels au bon fonctionnement de L'ATMega8535. Pour finir le placement d'un plan de masse sur un côté des 2 faces de la carte va permettre de lutter contre les problèmes de compatibilité électromagnétique.

Toutes ces petites contraintes nous montrent la complexité de la conception du typon et cela explique le temps relativement consacré à cette étape.

### 3.2.3. Le résultat final

Après avoir fait le routage des composants sur Layout avec l'option double face, on a obtenu ces deux schémas qui représentent ici les deux faces de la carte.

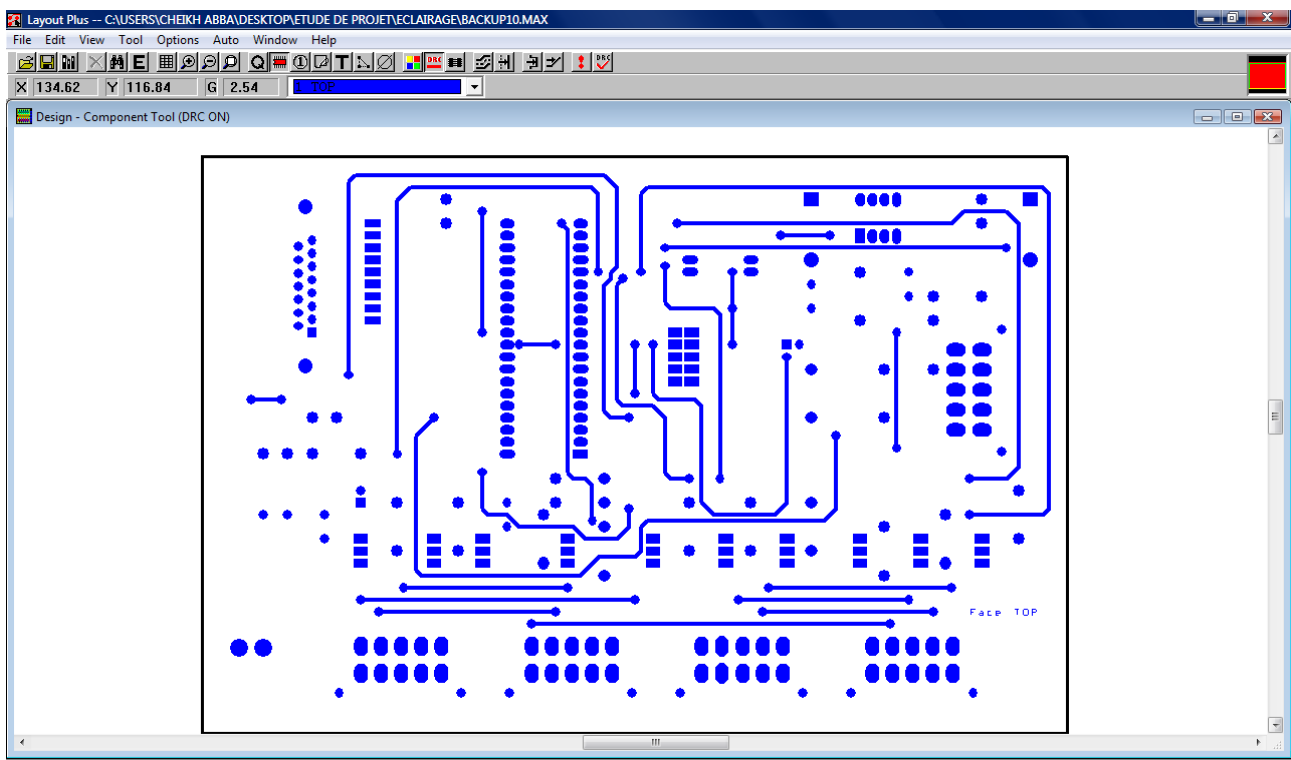


Illustration 18: typon de la carte côté composant

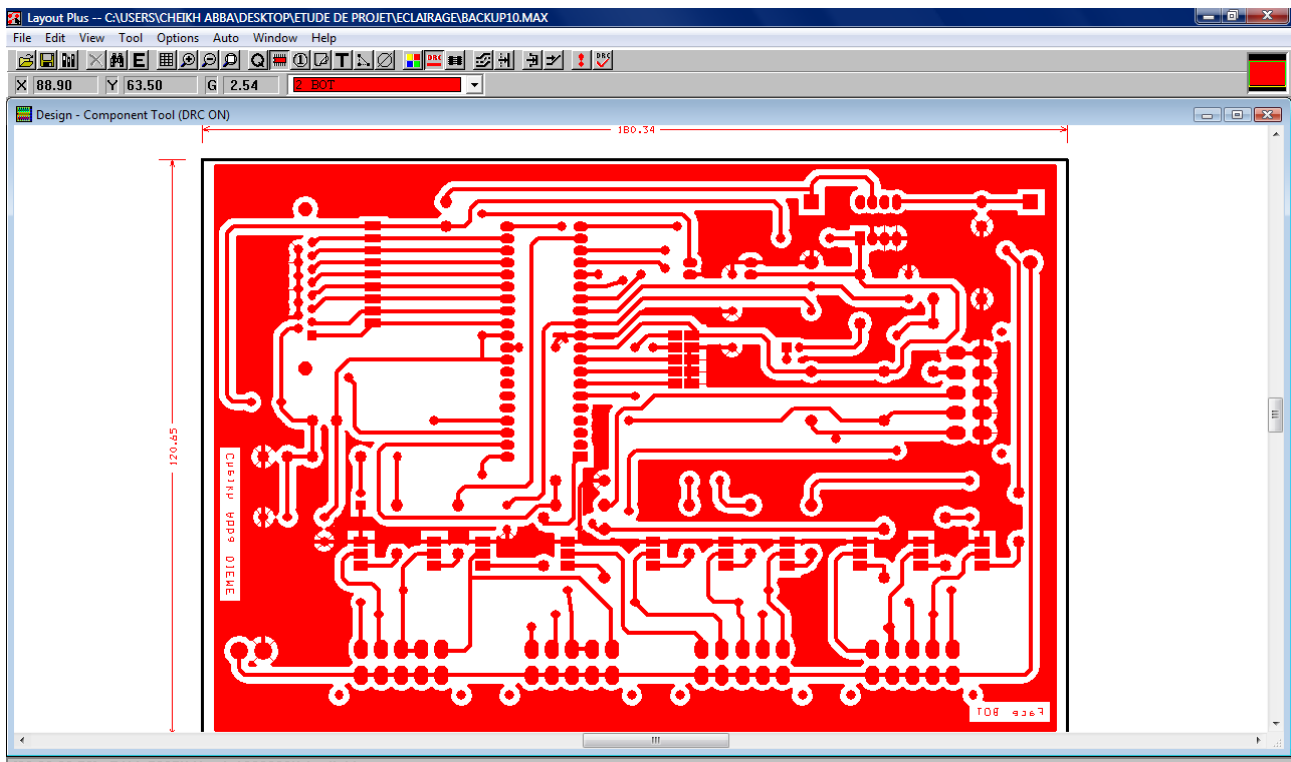


Illustration 19: typon de la carte partie commande

On a imprimé ces deux schémas en double, chacun sur du papier calque. Cette impression double nous permettra de mieux accentuer l'apparition des pistes et des pastilles sur notre plaque. Ainsi chaque schéma en double va être positionné sur les deux parties vierges de la plaque. Cette plaque va subir une série de traitements : elle passera en premier par l'insolation, puis par le révélateur et enfin par la gravure et l'éliminateur. Toutes ces étapes vont nous permettre d'obtenir la carte électronique gravée. Après la réception de cette carte électronique, on passera par le perçage, puis par la soudure des différents composants. Pour terminer nous effectuerons les premiers tests afin de valider le bon fonctionnement de cette carte électronique.

### 3.2.4. Le test d'alimentation

Ce test consiste à vérifier le bon fonctionnement de l'alimentation à découpage. Pour cela nous avons alimenté la carte avec une tension de 12 volts sans la présence du LM2574N( régulateur de tension ). Nous avons constaté que dans les supports du régulateur de tension on obtient à la sortie une tension de 12 volts. Puis sur la même tension d'alimentation et cette fois ci avec la présence du LM2575N on obtient à la sortie les 5 volts attendu. Ces 5 volts sont aussi présents sur les pattes de L'ATMega8535

(micro-contrôleur). On a constaté que la LED verte utilisée dans cette partie du montage s'est allumée ce qui montre la non présence de court circuit et le bon fonctionnement de notre carte électronique. Après le test concluant de la carte électronique, nous sommes donc passé à la phase suivante de notre projet, la programmation du micro-contrôleur.

### **3.3. La programmation**

La programmation du micro-contrôleur permettra enfin de finaliser notre projet. Cette programmation s'effectuera sur le logiciel Code Vision AVR qui est un peu complexe. Cette programmation a débuté par le test du bon fonctionnement de L'ATMega8535. Ce programme de test nous est fournis par le M.LEQUEU. Après cette petite vérification nous pourrons ainsi commencer le programme qui permettra de faire fonctionner les feux en fonction des besoins de la voiture électrique.

#### **3.3.1. Le programme de test**

Nous avons pu réaliser le programme complet de L'ATMega8535. Ce programme après le test fait à partir de deux roues codeuses nous a permis d'obtenir le résultat espéré. Il ne restera qu'a implanter le programme dans L'ATMega8535 afin d'effectuer le test final sur la voiture électrique pour pouvoir conclure de sa fiabilité. Ce dernier test n'a pas pu être réalisé avec le professeur responsable M. LEQUEU avant de rendre le rapport. Mais il sera surement effectué avant le passage à l'oral.

Le programme ainsi réalisé pour toutes les parties du projet est affiché ci-dessous.

#### **3.3.2. Programme implanté sur les ports entrées et sorties**

```
// Input/Output Ports initialization  
// Port A initialization  
// Func3=In Func2=In Func1=In Func0=In  
// State3=T State2=T State1=T State0=T  
PORTA=0x00;
```

```
DDRA=0x00;
```

```
// Port B initialization
```

```
// Func8=Out Func7=Out Func6=Out Func0=In
```

```
// State8=0 State7=0 State6=0 State0=T
```

```
PORTB=0x00;
```

```
DDRB=0x08;
```

```
// Port C initialization
```

```
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
```

```
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
```

```
PORTC=0x00;
```

```
DDRC=0x00;
```

```
// Port D initialization
```

```
// Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out  
Func0=Out
```

```
// State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0
```

```
PORTD=0x00;
```

```
DDRD=0xFF; //mettre en sortie
```

Ces lignes de code en C vont nous permettre de paramétrer les ports du micro-contrôleur. Le port C sera paramétré en sortie, alors que les broches 0,1,2,3 du port A sont paramétrées en entrées, de même que le broche 0 du port B. Les autres broches 6,7,8 du Port B sont paramétrées en sortie. Le port D est aussi paramétré en sortie.

### **3.3.3. Programme de la fonction MLI**

La fonction MLI sera utilisée pour l'éclairage variable des feux de stop. Donc pour cela nous utiliserons la fonction MLI du micro-contrôleur pour pouvoir avoir un éclairage variable pour les phares de la voiture électrique.

```
// Timer/Counter 0 initialization
```

```
// Clock source: System Clock
```



```

// Clock value: 16000,000 kHz
// Mode: Phase correct PWM top=FFh
// OC0 output: Non-Inverted PWM
TCCR0=0x61;
TCNT0=0x00;
OCR0=0x00;

```

Ce programme ci-dessous nous permet de faire clignoter toutes les lampes.

```

//if(comodoD==1)
    clignoD = !clignoD;
/*else
    clignoD = 0;
if(comodoG==1)*/
    //clignoG = 1;
/* else
    clignoG = 0;*/

```

Dans cette partie du programme nous avons géré le codage contrôlant les feux de stop, Phare et code. Ce programme nous permet de déterminer l'intensité de la lumière en fonction de la luminosité.

```

while (1)
{
    //FEUX STOP

    if(cmdstop==1)
    {
        a=read_adc(1);    //lecture de la valeur du potard
        sprintf(tampon,"%4d",a);
        lcd_gotoxy(0,1);
        lcd_puts(tampon);
        OCR0=a;
    }
    else

```

```

{
    OCR0=0;
}

// PHARE & CODE

if(cmdcode!=0)          //commande code
{
    b=0;
    b=read_adc(6);      //lecture de la valeur du capteur de lumière
    sprintf(tampon,"%4d",b);
    lcd_gotoxy(0,2);
    lcd_puts(tampon);

    if(cmdphare!=0)     //commande phare
        OCR2=b;
    else
    {
        b=10;
        OCR2 = b;      //cst de code
    }
}
else
{
    OCR2=0;
}
}
}

```

Après le test de notre programme nous avons constaté que celui-ci répondait à nos attentes. Le test a pu être réalisé grâce à un programme basé sur l'utilisation des potentiomètres, qui nous a donné un intervalle variant de 0 à 250. Cet intervalle va déterminer la variation des états que doivent avoir nos lampes. Cet intervalle devra être décomposé de 4 parties : les phares sont éteints, les feux de positions sont allumés, les

feux de croisements sont allumés et les feux de route sont allumés. Sans le test final ces plages d'intervalles ne pourront pas être déterminées. Ce test n'a pas encore été effectué, mais le sera avant la soutenance. De ce fait certains résultats apparaitront après la remise du dossier.

### 3.3.4. Planning finale

Planning Prévisionnel																									
Taches/Semaines	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24								
Prise de connaissance du sujet	P																								
Recherche d'information	R																								
Élaboration du cahier des charges et du planning	P	P																							
Formation Orcad	R	R																							
Recherche de solution		P																							
Réalisation du typon		R	R																						
Test et vérification			P	P	P				P																
Rédaction du document de synthèse										P	P														
Essai du prototype	R	R	R	R	R	R				R	R	R						P	P	P					
Remise des dossiers																									
Soutenance orale																									
<b>Planning Prévisionnel</b>																									
<b>Prévisionnel</b>																									
<b>Réel</b>																									

Illustration 20: planning prévisionnel et réel

## Conclusion

L'étude que nous avons menée nous a permis de revoir certains fondements de notre formation. Elle nous a permis de bien comprendre le fonctionnement d'une alimentation à découpage que nous avons vu en cours de MC-ET2. Elle nous a incité à revoir certaines bases d'informatique pour la programmation du micro-contrôleur et d'électronique pour la réalisation de la carte électronique.

Pour pouvoir réaliser cette carte nous avons utilisé le logiciel Orcad dans lequel nous avons saisi le schéma électrique avant d'utiliser layout pour le routage. Du fait de la complexité de cette carte nous avons choisi l'option double face pour lutter contre l'utilisation de straps ( sorte de liaison électrique faite au moyen d'un fil gainé et permettant ainsi de relier un point de la carte à un autre point dont l'accès serait impossible ) pour mieux l'aérer.

Pour la partie informatique nous avons utilisé un logiciel Code Vision AVR en utilisant le langage C. Ce programme après réalisation a été testé par l'utilisation de deux potentiomètres, il fonctionne. A ce jour le test final n'a pas encore été réalisé avec le professeur chargé de l'encadrement M.LEQUEU. Nous espérons passer le test final qui nous fournira les réponses à nos attentes avant le passage à l'oral. Le programme ainsi réalisé va être implanté dans le micro-contrôleur et est censé contrôler toutes les fonctionnalités des lampes de la voiture électrique selon la commande de l'utilisateur.

Ce projet nous a permis de mettre en œuvre certaines de nos compétences. Il permet de voir si nous avons les capacités, à respecter un cahier des charges, à travailler en équipe, l'apprentissage et les contraintes dues à la durée. Tous ces petits points auxquels nous avons été confrontés, nous permettront de mieux nous préparer pour notre stage de fin de cycle.

# Résumé

## **mots**

Pour mieux entreprendre notre projet, nous avons fait des recherches consistant à trouver des solutions pour son bon fonctionnement. Ces recherches nous ont poussé à créer une carte électronique. Cette carte est équipée d'un micro-contrôleur programmable, élément centrale de notre projet. Ce micro-contrôleur permet la commande des diverses sorties du montage en fonction des différentes entrées. Pour les entrées on a la présence de la photodiode qui permet de commander l'intensité lumineuse des feux de la voiture électrique en fonction de la luminosité ambiante, puis on a la pédale de frein qui aura pour rôle de commander l'intensité lumineuse des feux de stop en fonction de la pression sur cette pédale pour le freinage de la voiture. Avec l'étude théorique on a su déterminer les différents composants qui nous sont nécessaires pour le bon fonctionnement de notre projet. Ainsi pour obtenir une tension de sortie de 5 volts qui va servir à alimenter notre micro-contrôleur sur une tension d'alimentation de 12 volts, donc on a utilisé un régulateur de tension et une alimentation de découpage de type BUCK. Pour la réalisation de notre carte électrique on a utilisé un logiciel Orcad pour la saisie et Layout pour le routage des liaisons entre les différents composants. La réalisation du programme a été réalisée par le logiciel Code Vision AVR, et ce programme a été testé avec l'utilisation de deux potentiomètres. Lors du test de notre programme en C on a constaté que celui fonctionnait, et ceci pourra nous permettre de l'implanter dans le micro-contrôleur pour pouvoir faire le test final sur la voiture électrique. Du à une mauvaise gestion de notre planning nous n'avons pas pu faire le test final sur la voiture électrique afin de valider le bon fonctionnement de notre projet. Cependant la réponse vous sera présentée lors du passage à l'orale.

## Index des illustrations

Illustration 1: schéma fonctionnel de niveau 1.....	6
Illustration 2: schéma fonctionnel de niveau 2.....	7
Illustration 3: planning prévisionnel.....	8
Illustration 4: comparatif des lampes disponibles.....	9
Illustration 5: schéma du micro-contrôleur ATmega 8535.....	10
Illustration 6: schéma des transistors MOSFET.....	11
Illustration 7: schéma du régulateur de tension.....	12
Illustration 8: schéma d'une alimentation de découpage de type BUCK.....	12
Illustration 9: pédale de commande des feux de stop.....	13
Illustration 10: schéma représentant les frais de notre projet.....	14
Illustration 11: logiciel Orcad.....	15
Illustration 12: schéma capture de L'ATmega 8535.....	16
Illustration 13: schéma de capture de l'alimentation à découpage.....	16
Illustration 14: schéma capture des entrées tout ou rien.....	17
Illustration 15: schéma capture des entrées variables.....	18
Illustration 16: schéma capture des sorties du montage.....	19
Illustration 17: impression d'écran du logiciel Layout plus.....	20
Illustration 18: typon de la carte côté composant.....	21
Illustration 19: typon de la carte partie commande.....	22
Illustration 20: planning prévisionnel et réel.....	27
Illustration 21: capture de la carte électrique.....	32

## **Bibliographie**

[Http://www.thierry-lequeu.fr](http://www.thierry-lequeu.fr) : site internet de M.LEQUEU

- Cours de MC-ET2 et Cours D'informatique sur la programmation en C
- Ressources d'anciens collègues

# Annexes :

## schéma de la carte électrique

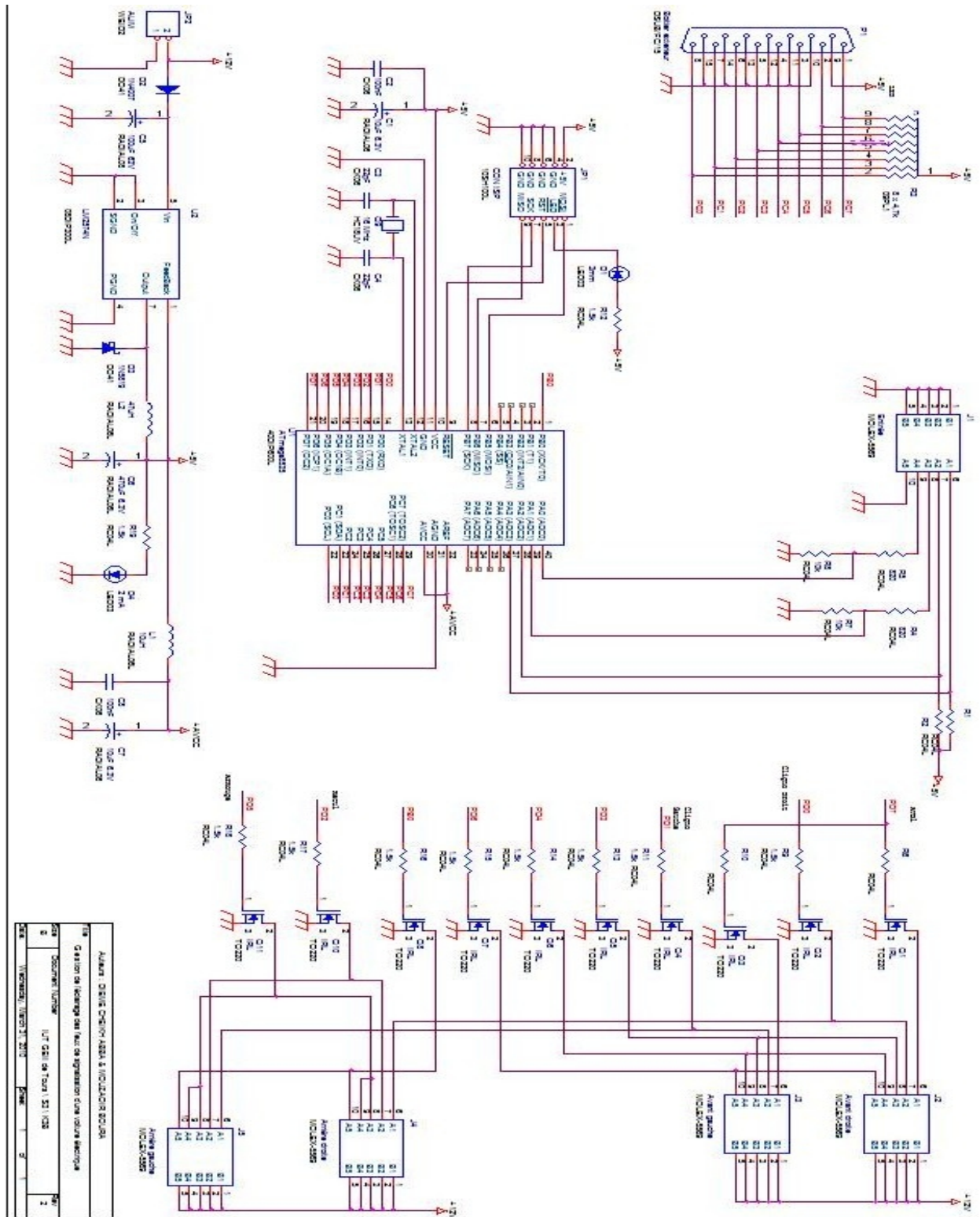


Illustration 21: capture de la carte électrique