



Université François Rabelais

Institut Universitaire et Technologique de Tours

Département Génie Électrique et Informatique Industrielle



## CHARGEUR SOLAIRE POUR TROTTINETTE ELECTRIQUE 24V

Etude et Réalisation GE & ISI

Jeremy THIAUX

Zhe FAN

Groupe K4A

Philippe AUGER

Thierry LEQUEU

Année 2014-2015





Université François Rabelais

Institut Universitaire et Technologique de Tours

Département Génie Électrique et Informatique Industrielle



*Illustration 1: Montage final de notre chargeur solaire*

## CHARGEUR SOLAIRE POUR TROTTINETTE ELECTRIQUE 24V

Etude et Réalisation GE & ISI

Jeremy THIAUX

Zhe FAN

Groupe K4A

Philippe AUGER

Thierry LEQUEU

Année 2014-2015

# Sommaire

Introduction.....	4
1.Présentation du projet.....	5
1.1.Cahier des charges.....	5
1.2.Etude du panneau solaire.....	6
1.3.Etude sur la batterie.....	7
1.4.Planning.....	8
2.Réalisation du chargeur.....	9
2.1.Présentation du hacheur BOOST.....	9
2.2.Fonctionnement du hacheur BOOST.....	10
2.3.Les principaux composants.....	13
2.4.Réalisation de la carte chargeur.....	17
3.Programmation pour ATMEGA 8535.....	19
3.1.Objectif.....	19
3.2.Etude sur la carte AT8535 et la carte AT8535 ANALOG.....	19
3.3.Etude sur l'ATMEGA 8535.....	21
3.4.Programmation sous Code Vision AVR.....	23
3.5.Mesurage et affichage de tension de la batterie.....	25
3.6.Mesurage et affichage de tension du panneau solaire.....	27
3.7.Mesurage et affichage de courant d'entrée du panneau solaire.....	29
4.Amélioration du chargeur.....	31
4.1.Pont de diviseur pour la tension d'entrée (la tension de panneau).....	31
4.2.Rajout des LED.....	31
4.3.Rajout d'un connecteur.....	32
4.4.Rajout d'un capteur de courant.....	32
4.5.Amélioration du programme.....	34
5.Problèmes rencontrés.....	39
6.Conclusion.....	40
7.Le résumé du projet.....	41
Mots clés.....	42
Bibliographie.....	44

## Introduction

Dans le cadre de l'étude et réalisation du quatrième semestre du DUT GEII, nous devons réaliser un projet personnel, en rapport avec nos matières (génie électrique, génie électronique, génie électrotechnique, informatique industrielle etc..). Un rapport écrit sera remis ainsi qu'une soutenance orale.

Nous avons organisé notre projet en plusieurs étapes et créé un emploi du temps, en fonction de nos contraintes et du temps qui nous était imparti. Notre projet consiste à mettre en place un chargeur solaire pour des batteries de trottinette. En effet l'IUT possède un panneau photovoltaïque de la marque PHOTOWATT PWX 500. Nous devons mettre en place un système permettant de régulariser la tension en sortie du panneau et un système permettant de mesurer les tensions aux bornes du panneau et des batteries

Ce projet fût l'occasion de pouvoir travailler en parfaite autonomie, avec les conseils de notre professeur référent Monsieur Thierry LEQUEU.

# 1. Présentation du projet

Ce projet fut une proposition faite par Monsieur LEQUEU. L'IUT a mis à notre disposition un panneau solaire et deux batteries 12V, 12Ah. Le panneau fournit une tension de 12V et les batteries ont besoin de 24V, c'est pour cela que nous devons mettre en place un système de régulation de tension.

## 1.1. Cahier des charges

Le système de puissance mis en place pour faire cette conversion est un Hacheur BOOST régulé en tension et en courant.

Une fois le hacheur mis en place, il faut trouver le moyen de pouvoir contrôler la tension au borne des batteries, nous allons donc passer par un microcontrôleur AT-8535.

Ce microcontrôleur va nous permettre de pouvoir contrôler la tension délivrée sur les batteries et la tension et le courant aux bornes du panneau solaire. Il va nous permettre aussi de mettre en place un afficheur pour afficher toutes ces grandeurs.

Au niveau du panneau solaire, il faut faire attention aux contraintes liées à la relation tension-courant. Il va falloir trouver le bon point pour avoir la puissance optimale.

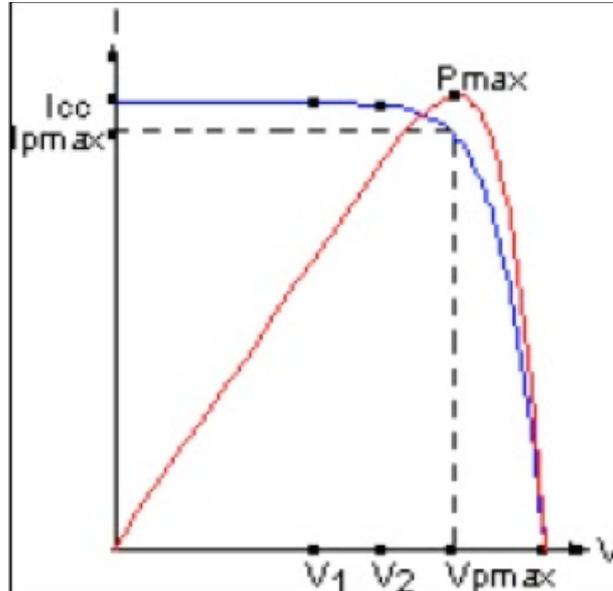
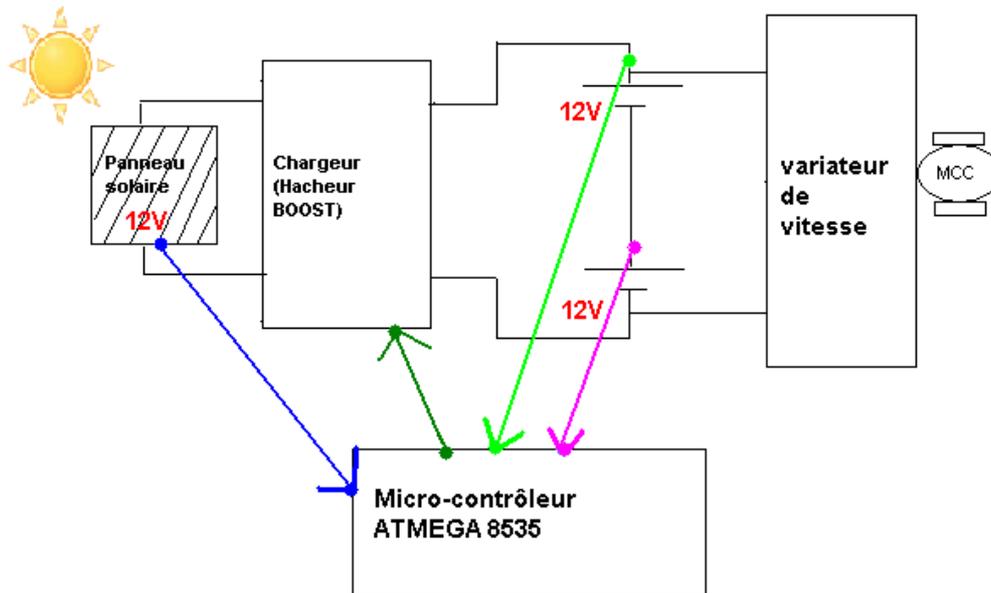


Illustration 2: Relation tension courant du panneau solaire

Nous allons essayer de faire une carte de la puissance la plus petite possible pour éviter tous les parasites éventuels. Les tests du micro-contrôleur seront faits sur une carte prêtée par le professeur. Nous allons ensuite, une fois les tests finis, créer une carte pour le microcontrôleur.

Voici un schéma synoptique du montage :



*Illustration 3: schéma synoptique du montage*

## 1.2. Etude du panneau solaire

Dans un premier temps, nous verrons ce qu'est un panneau solaire, il s'agit d'un dispositif technologique énergétique à base de capteurs solaires thermiques ou de capteurs solaires photovoltaïques destinés à convertir le rayonnement solaire en énergie thermique ou en énergie électrique.

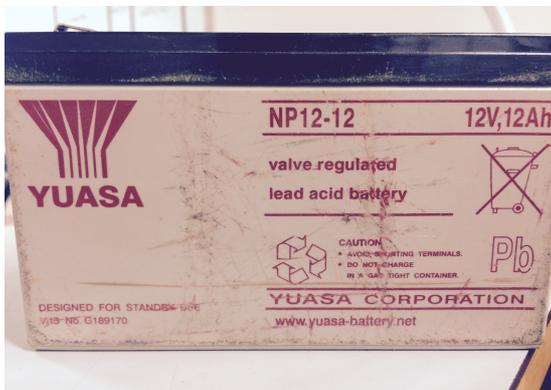
Pour notre projet, nous allons utiliser un panneau solaire de la marque PhotoWatt de référence Pwx 500. Lorsque l'on regarde la datasheet de ce panneau solaire, on voit que celui-ci est fait pour délivrer une tension de 12V DC et peut atteindre une tension de 17V DC ainsi qu'un courant de 3,4 A qui peut atteindre jusqu'à 4 A maximum.



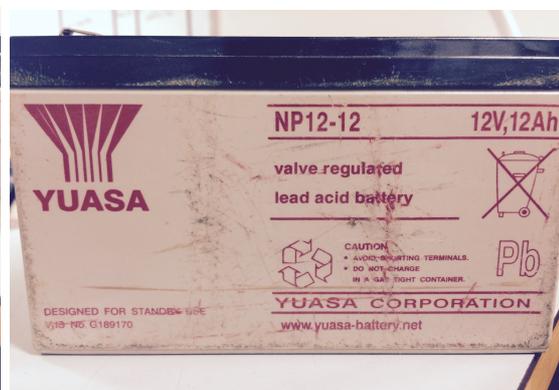
*Illustration 4: Panneau solaire PWX 500*

### 1.3. Etude sur la batterie

Dans cette partie, nous verrons ce qu'est une batterie ou plutôt une batterie d'accumulateurs. Il s'agit d'un ensemble d'accumulateurs électriques reliés entre eux de façon à créer un générateur électrique de tension et de capacité désirée. En sortie de notre hacheur, on obtient une tension de 24V DC avec un courant de 2A. Vu que notre projet consiste à recharger des batteries de trottinette, nous avons donc décidé de prendre deux batteries de 12V DC avec un courant de 12 A.h. Ces batteries sont déjà présentes sur la trottinette de l'IUT.



*Illustration 5: Batterie N°1 12V 12 Ah*



*Illustration 6: Batterie N°2 12V 12 Ah*

## 1.4. Planning

Rouge = Prévisionnel      Bleu = Réel

Semaine	37	38	39	40	41	42	43	44	45	46
Compréhension du sujet	Rouge	Rouge								
Choix des composants	Bleu	Rouge								
Création du circuit		Bleu	Rouge							
Hacheur sous kicad		Bleu	Bleu							
Gravure, soudure, tests du circuit		Rouge	Rouge							
Programmation du microcontrôleur		Bleu	Bleu	Rouge						
Test sur la carte existante				Rouge	Bleu					
Création du circuit microcontrôleur						Rouge			Rouge	
Test finals						Bleu			Rouge	Bleu
Rédaction du rapport	Rouge									
Soutenance										Bleu

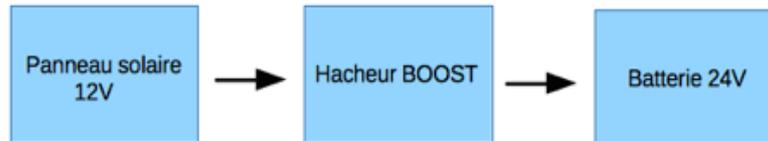
## 2. Réalisation du chargeur

Notre objectif lors de la conception de la carte a été divisé en 2 parties, dans un premier temps nous devons créer une carte test et vérifier que nous arrivons bien à transformer du 12V DC en 24V DC. Puis dans un second temps nous devons créer et recommencer la carte finale, mais en mettant en place un système permettant de régulariser le courant et de pouvoir afficher les tensions aux bornes du panneau solaire et des batteries.

### 2.1. Présentation du hacheur BOOST

Pour commencer, il y a trois types de hacheur, le hacheur BOOST pour élever une tension, le hacheur BUCK pour abaisser une tension et le hacheur BOOST-BUCK, qui permet d'élever une tension en négatif. 1

Le hacheur BOOST permet d'élever une tension. Par exemple du 12V en 24V DC, le hacheur BOOST fonctionne avec une tension continu. Si nous étions en sinusoïdale, nous utiliserions un transformateur.



*Illustration 7: Schéma structurelle d'une hacheur BOOST*

A travers ce schéma structurel de blocs simplifiés, on comprend bien le problème de notre sujet et l'intérêt de celui-ci.

A la sortie de notre panneau solaire qui fournit une tension de 12 V en continu, il faut trouver un moyen de transformer cette tension en 24V. C'est pour cela que l'on passe à travers notre hacheur BOOST.

Le hacheur BOOST fonctionne sur un système de commutation.

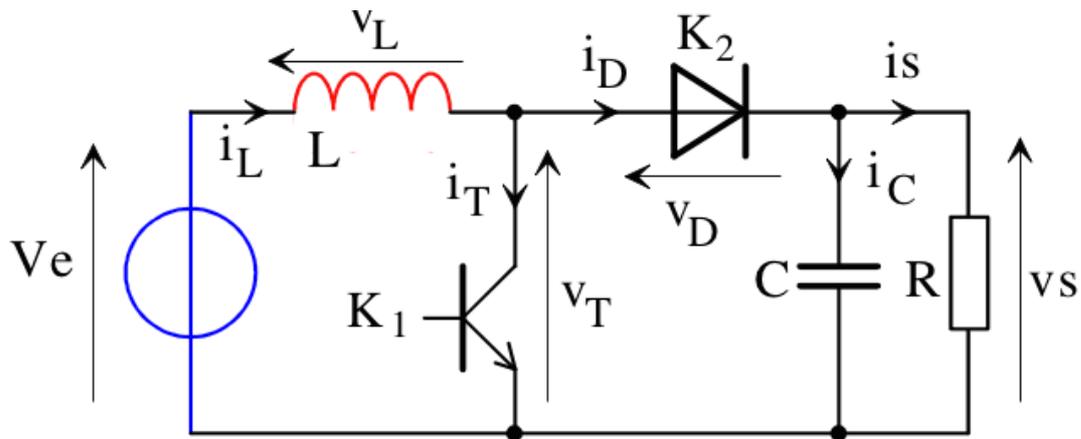


Illustration 8: Schéma de base d'un convertisseur BOOST

Le hacheur BOOST est constitué d'une bobine  $L$ , d'un transistor  $T$ , et d'une diode  $D$ . l'emplacement de chacun de ces composants est spécifique au hacheur BOOST, si ces composants avaient été mis dans un autre sens nous aurions eu un autre type de hacheur.

Nous pouvons voir l'utilité de chacun de ses composants comme le transistor  $S$  et la diode  $D$  qui jouent un rôle d'interrupteurs. Une bobine  $L$  qui permet d'emmagasiner l'énergie électrique et un condensateur en sortie pour filtrer la tension.

## 2.2. Fonctionnement du hacheur BOOST

Le fonctionnement d'un convertisseur Boost peut être divisé en deux phases distinctes selon l'état de l'interrupteur.

° Une phase d'accumulation d'énergie : lorsque l'interrupteur est fermé (état passant), cela entraîne l'augmentation du courant dans l'inductance donc le stockage d'une quantité d'énergie sous forme d'énergie magnétique. La diode  $D$  est alors bloquée et la charge est alors déconnectée de l'alimentation.

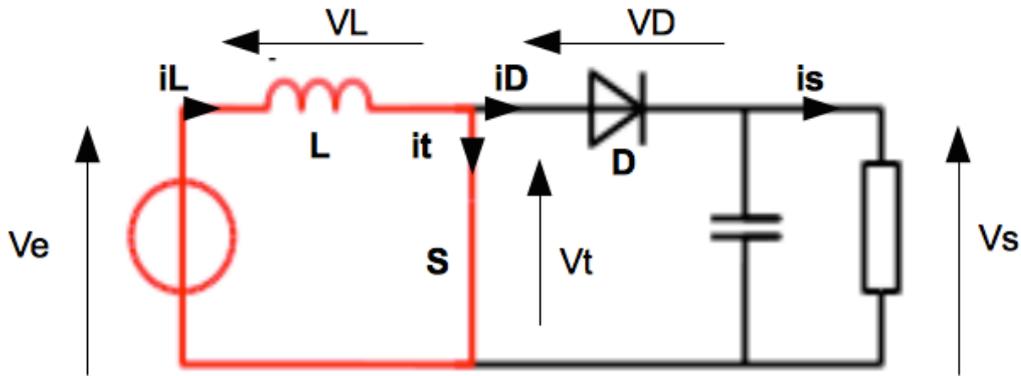


Illustration 9: Montage simplifié pour  $t$  appartenant à  $[0 \text{ et } ct]$

° Lorsque l'interrupteur est ouvert, l'inductance se trouve alors en série avec le générateur et sa F.E.M s'additionne à celle du générateur (effet survolteur). Le courant traversant l'inductance traverse ensuite la diode D, le condensateur C et la charge R. Il en résulte un transfert de l'énergie accumulée dans l'inductance vers la capacité. il s'agit de l'effet « Diode de roue libre ».

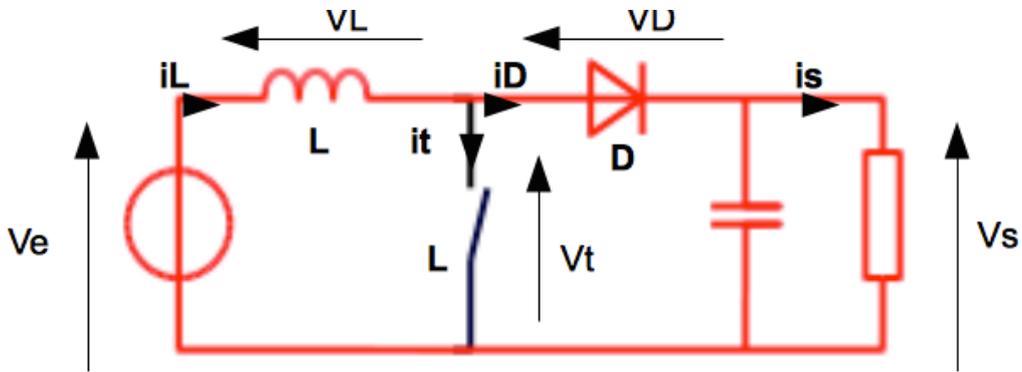


Illustration 10: Montage simplifié pour  $t$  appartenant à  $[ct \text{ et } T]$

Pour savoir si le montage est passant ou bloqué, il faut regarder le rapport cyclique

$\alpha$  étant le rapport cyclique, Il représente la fraction de la période  $T$  pendant laquelle l'interrupteur  $T$  conduit,  $\alpha$  est compris entre 0 et 1, lorsqu'il est égal à 0 il ne conduit jamais et lorsqu'il est égal à 1 il conduit tout le temps.

D'après la formule  $\frac{V_s}{V_e} = \frac{1}{1-\alpha}$ , on remarque que:

Lorsque  $\alpha = 0$  alors  $V_e = V_s$

Lorsque  $\alpha = 1$  alors  $V_s$  tend vers l'infini

Lorsque  $\alpha = 0,5$  alors  $V_s = 2 \times V_e$

On remarque alors que la tension de sortie augmente avec le rapport cyclique mais d'une façon non linéaire.

Nous avons fait une simulation du hacheur pour ensuite en tirer des courbes de mesure.

Caractéristiques Hacheur Boost 12V – 24V 50W						
Rapport Cyclique = 0;5						
Ve (V)	11,97	11,92	11,87	11,78	11,69	11,61
Ie (A)	1,1	1,58	2,15	3,15	4,13	5,02
Vs (V)	30,1	23,7	23,4	23,2	22,9	22,5
Is (A)	0,4	0,74	1,02	1,51	1,99	2,42
Pe (W)	13,167	18,8336	25,5205	37,107	48,2797	58,2822
Ps (W)	12,04	17,538	23,868	35,032	45,571	54,45
Rendement	0,914407	0,931208054	0,935248134	0,944080632	0,943895675	0,934247506

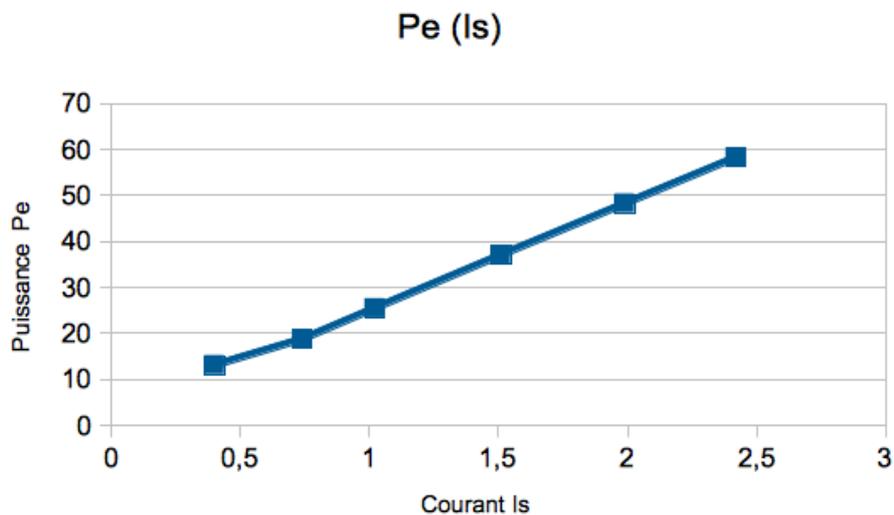
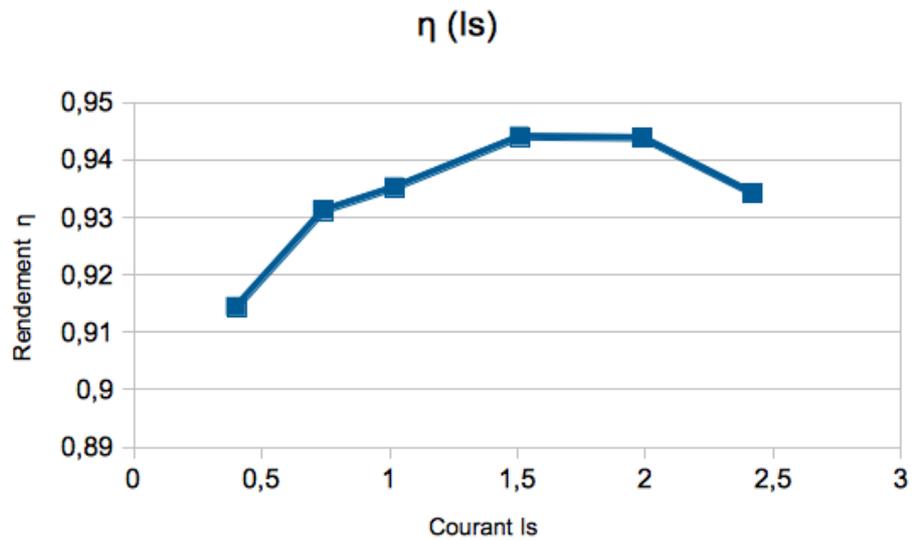
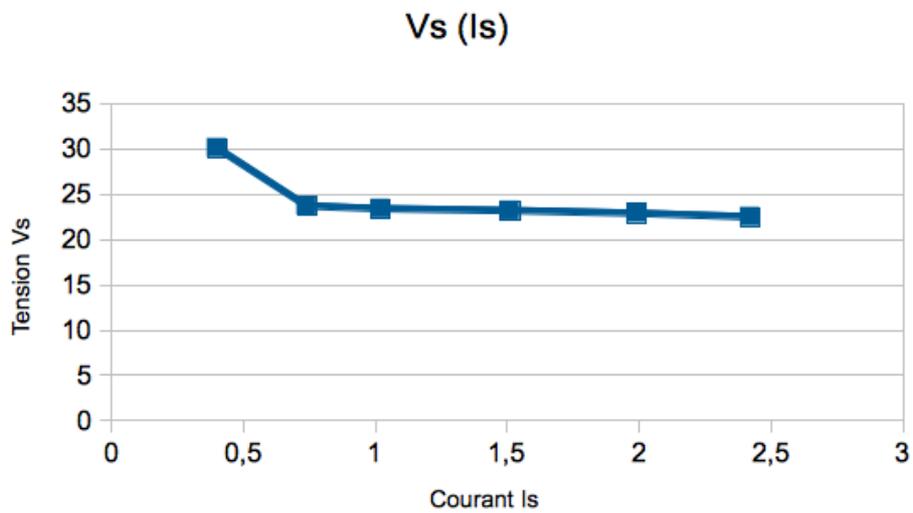


Illustration 11: Puissance d'entrée en fonction du courant de sortie



*Illustration 12: Rendement en fonction du courant de sortie*



*Illustration 13: Tension de sortie en fonction du courant de sortie*

### 2.3. Les principaux composants

On peut récapituler les composants utilisés au sein du tableau suivant.

	Composants	Fournisseurs	Désignation	Valeur	Code commande	Prix €
1	Mosfet	Radiospare	Q1	55V -80A	541-0008	2,53
2	Diode schottky	Radiospare	D1		301-259	4,39
3	Condensateur	Radiospare	C1	1000 $\mu$ 35V	571-127	1,21
4	Condensateur	Radiospare	C2	1000 $\mu$ 35V	571-127	1,21
5	Condensateur	Radiospare	C3	1000 $\mu$ 35V	571-127	1,21
6	Condensateur	Radiospare	C4	1000 $\mu$ 35V	571-127	1,21
7	Condensateur	Radiospare	C5	1000 $\mu$ 35V	571-127	1,21
8	Condensateur	Radiospare	C6	1000 $\mu$ 35V	571-127	1,21
9	Condensateur		C7	10 $\mu$		0,252
10	Condensateur		C8	100n		0,018
11	Condensateur		C9	100n		0,118
12	Inductance	Radiospare	L1	40 $\mu$ H 4A	736-1024	2,118
13	Bornier à vis		J1			0,67
14	Bornier à vis		J2			0,67
15	Support 2*4		P1	CONN_4x2		0,72
16	TC4422					1,601
17	Douille 4mm		PT-GND1			3,40
18	Douille 4mm		PT-GND2			3,40
19	Douille 4mm		PT1-12V1			3,40
20	Douille 4mm		PT2-24V2			3,40
21	Résistance		R1	10 $\Omega$		0,006
22	Résistance		R2	1000 $\Omega$		0,006
23	Résistance		R3	820 $\Omega$		0,006
24	Résistance		R4	5600 $\Omega$		0,006
25	Résistance		R5	1500 $\Omega$		0,006
26	Résistance		RV1	470 $\Omega$		0,60
27	Led		Led-jaune			1,26
28	Led		Led-rouge			1,26
29	Led		Led-bleu			1,26
30	Connecteur					3,20

31	LTS 6NP		5V 19,2 A			12,80
----	---------	--	-----------	--	--	-------

Le coût totale du projet nous revient à environ 54,36€ TTC au niveau de la mise en place du hacheur.

La nomenclature permet de classer les composants par type, il rassemble l'ensemble des composants utiles au fonctionnement de la carte électronique. On y fait figurer le nom des composants, leur désignation, leur valeur et/ou caractéristique. Le but de la nomenclature est de pouvoir changer un composant rapidement en se référant à ceci, pour trouver plus facilement la valeur de l'élément voulu.

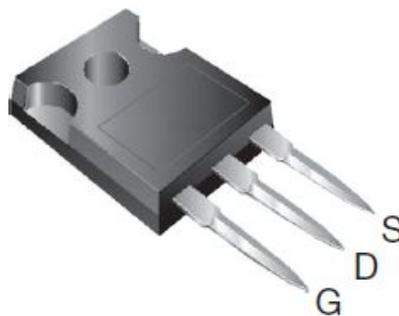
Pour le choix des composants, nous nous sommes inspirés d'un montage trouvé sur internet, seulement sur internet la personne voulait un hacheur pour transformer du 12V en 24V mais avec un courant de 10A alors que dans notre projet nous voulions seulement un courant de 2A en sortie dû à nos contraintes des batteries.

Nous pouvons donc voir brièvement les composants important à notre montage.

### 2.3.1. Transistor :

Nous avons choisi un transistor mosfet de référence IRFP064N avec un boîtier T0-247AC, il peut supporter un courant de 80A et une  $V_{dss} = 55V$ , et ayant un  $R_{DS}$  le plus petit possible comme ça nous étions sûr de ne pas provoquer une destruction du composants.

#### TO-247AC



*Illustration 14: Mosfet 55V-80A  
T0-247AC 2*

### 2.3.2. Diode schottky

Nous avons choisi une diode schottky de référence 40CPQ040PBF avec un boîtier qui est le même que le mosfet (TO-247AC). Ce composant peut supporter un courant de  $2 \times 20 \text{ A}$  et il peut supporter une tension entre 35V et 45V. Comme pour le Mosfet, nous avons décidé de surdimensionner nos composants pour éviter toute destruction des composants.



*Illustration 15: Diode schottky  
T0-247AC 3*

### 2.3.3. Inductance

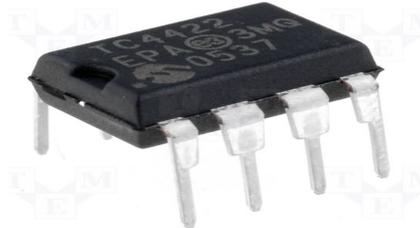
Pour l'inductance nous avons choisi une inductance de  $40 \mu\text{H}$  et avec une intensité de 4 A, à cause du courant fournit par le panneau solaire. Nous avons donc pris une inductance de la série 5900 car celle-ci à une taille très compacte, ce qui permet un gain de place sur la carte.



*Illustration 16:  
Inductance  $47 \mu\text{H} - 4\text{A}$  4*

### 2.3.4. TC4422

Le TC4422 est un driver à haut courant pouvant commander des grands transistors, il est composé de 8pins ( 2Vdd, 2 GND, 2 Output, 1 Input, 1 NC), on relie les masses ensemble, les Vdd ensemble et également les Input.



*Illustration 17: TC4422 5*

### 2.3.5. Condensateur

Nous avons choisi de prendre des condensateurs 35V et 1000 $\mu$ F. Il faut impérativement utiliser des condensateurs à faible ESR (équivalent Series Resistor) car ils sont traversés par un courant efficace très élevé. L'utilisation de condensateurs électrochimiques ordinaires conduit à une très forte ondulation de la tension de sortie.



*Illustration 18: Condensateur  
35V - 1000 $\mu$ F 6*

## 2.4. Réalisation de la carte chargeur

Inspiré par un montage existant “convertisseur élévateur 12V – 10A”<sup>7</sup>, nous avons pu créer notre propre carte.

La réalisation de carte est faite sous le logiciel Kicad. Dans un premier temps, nous avons commencé par adapter le schéma existant à notre projet.

Nous avons créé le schéma à partir de l'application Eschéma sous Kicad.

Une fois que le schéma est réalisé, nous devons associer les composants à leurs boîtiers.

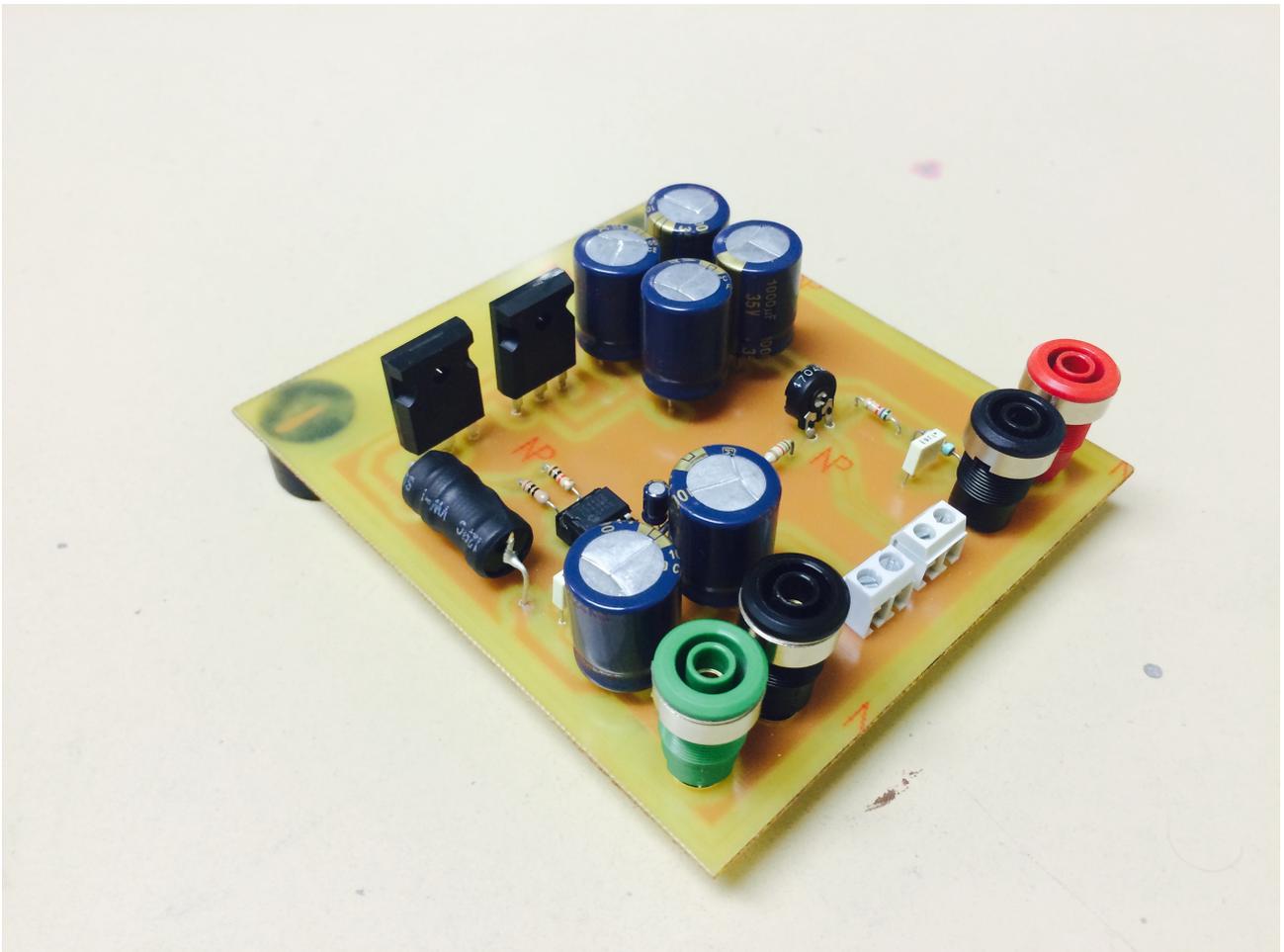
Une fois que tous les composants sont associés, nous pouvons alors passer au routage. Pour cela, il faut respecter quelques règles :

- Les douilles de tension d'entrée et de tension de sortie doivent être au bord de la carte et du même côté.

- Les pistes qui relient l'entrée et les condensateurs d'entrée doivent être directes et courtes, ainsi que celles qui relient la sortie et les condensateurs de sortie.

- Nous pouvons utiliser des vias pour la partie commande, mais pas pour la partie puissance.

Vous trouvez en annexe, les photos du routage, de la netlist et du schéma finale de la carte.



*Illustration 19: Carte de test de notre hacheur*

### 3. Programmation pour ATMEGA 8535

#### 3.1. Objectif

La tension de sortie du panneau varie en fonction de différents paramètres : la température, l'ensoleillement et de l'orientation du panneau. Elle peut atteindre 17V en valeur maximale.

En fonction de ces caractères, le panneau fournit une puissance variable. Il est donc important de mesurer sa tension et son courant afin de pouvoir calculer la puissance fournie. Le but est que le panneau puisse délivrer une puissance maximale à sa sortie.

La tension mesurée sur une batterie chargée, au repos à 20 °C, est égale à 2,275 V/élé , soit 13,65 V, pour une batterie qui a une tension nominale de 12V. Ici nous avons deux batteries en parallèle, donc cette valeur doit être  $13,65 \times 2 = 27,3$  V. Cette tension de 27,3 doit être lue aux bornes de la batterie lorsque celle-ci est bien chargée.

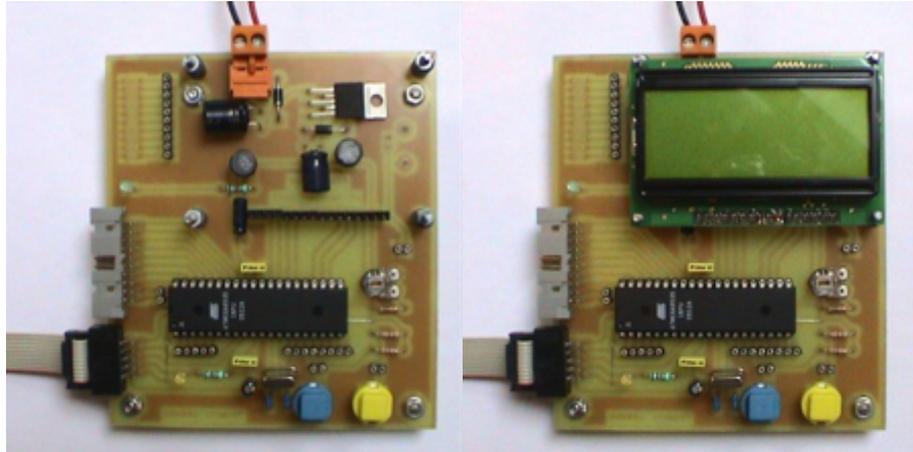
TENSION DE CHARGE		
Tension de charge en floating à 20°C	13.65 (±1%)	V
	2.275 (±1%)	V/élé

*Illustration 20: Tension de charge de la batterie*

Ce sont les raisons pour lesquelles nous allons mesurer et afficher la tension de panneau, le courant du panneau ainsi que le courant du panneau.

#### 3.2. Etude sur la carte AT8535 et la carte AT8535 ANALOG

Pour effectuer ces différentes mesures, nous allons utiliser une carte AT8535 développée par M.Thierry LEQUEU.



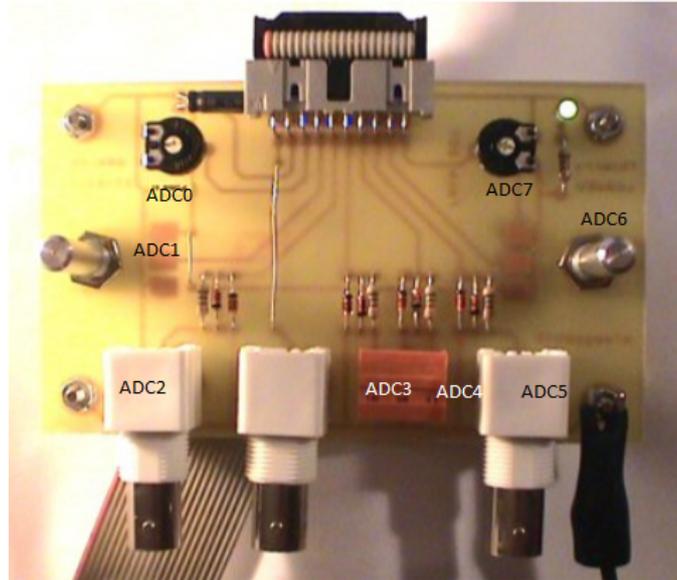
*Illustration 21: Vu de carte  
ATMEGA 8535*

Elle s'agit de plusieurs parties :

- Un microcontrôleur ATMEGA 8535 qui permet de programmer la partie intelligente de la carte.
- Un afficheur LCD 16 caractères \* 4 lignes.
- Un connecteur SPI (Serial Peripheral Interface).
- Un régulateur LM2575 qui permet d'alimenter la carte.
- Un quartz qui permet de fixer la fréquence de fonctionnement, soit 16MHz.
- Un connecteur 10 \* 2 relie les ports A du microcontrôleur pour effectuer des mesures.

Le mesurage d'une tension nécessite de relier la tension qu'on désire mesurer avec un port A (ADC - Convertisseur Analogique-Numérique) du microcontrôleur via un connecteur.

Dans un premier temps, nous n'avons prévu qu'un bornier sur notre carte de chargeur. Le mesurage se fait par l'intermédiaire d'une carte de mesure mise en disposition par l'IUT.

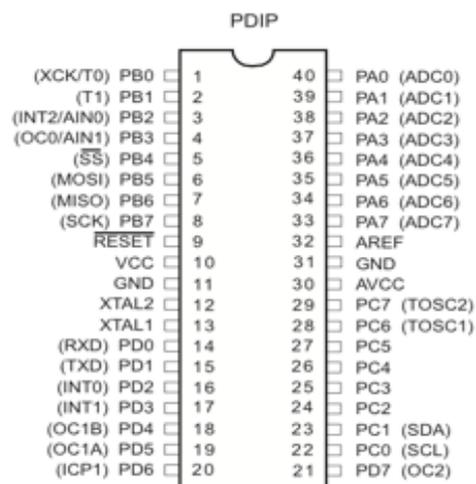


*Illustration 22: Carte de mesure : AT8535 - ANALOG*

Nous trouvons successivement ADC0 à ADC7 sur cette carte.

### 3.3. Etude sur l'ATMEGA 8535

Un ATMEGA est un microcontrôleur qui est cadencé par une horloge de 16 MHz. Ce composant permet de programmer la partie intelligente de la carte AT8535. Les données échangées sont stockées dans des mémoires.



*Illustration 23: ATMEGA 8535*

Il comporte plusieurs ports bidirectionnels qui peuvent être des entrées comme des sorties. Les principaux ports que nous allons utiliser sont :

Vcc : la tension d'alimentation

GND : la masse

Port A (PA0 à PA7):

Ce sont des ports analogiques utilisés pour la conversion analogique-numérique (mesurage de tension et de courant). Ils sont alimentés en 0 – 5V.

Port B (PB0 à PB7):

Ce sont des ports utilisés pour la programmation. La connexion entre le microcontrôleur se fait via un bus SPI. On utilise les ports PB5, PB6 et PB7 pour relier à un ordinateur.

Port C (PC0 à PC7):

Ces ports sont liés à l'afficheur LCD 8 bits afin de pouvoir afficher les différentes valeurs de tension et de courant.

Port D (PD0 à PD7):

Ports 8 bits bidirectionnels. Nous les utiliserons pour générer un signal MLI, qui est indispensable pour le fonctionnement du Hacheur.

Reset : Pour générer un redémarrage du système.

XTAL1 : C'est une entrée d'horloge qui permet le fonctionnement de l'ATMEGA.

XTAL2 : C'est une sortie de la patte inverse de l'amplificateur de l'oscillateur.

AVCC :

C'est une tension d'alimentation du port A et du convertisseur analogique-numérique. Cette patte doit être connectée à Vcc même si le convertisseur n'est pas utilisé. Si ce dernier est utilisé, il doit être tout de même relié à Vcc par l'intermédiaire d'un filtre passe-bas.

AREF : C'est une patte analogique de référence pour le convertisseur.

### 3.4. Programmation sous Code Vision AVR

Nous allons utiliser le logiciel Code Vision AVR pour programmer le micro-contrôleur. Le programme est écrit en langage C.

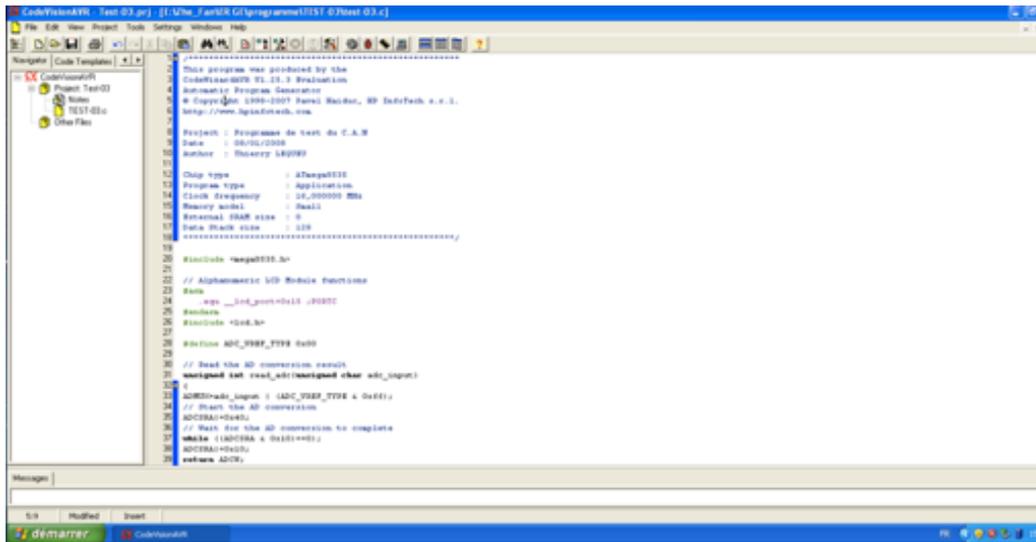


Illustration 24: Interface de Code Vision AVR

Dans un premier temps, il faut inclure la bibliothèque <mega8535.h>.

```
#include <mega8535.h>
```

Après il a fallu paramétrer les ports du microcontrôleur.

Le Port A s'agit d'un convertisseur analogique-numérique, il est paramétré en entrée.

Le Port B a pour but de communiquer avec l'ordinateur, il est également paramétré en entrée.

Le Port C est lié à l'afficheur 16\*4, il est paramétré en sortie.

Le Port D sort un signal MLI pour commander le transistor Mosfet, il est paramétré en sortie.

```
// Port A initialisation
```

```
PORTA=0x00;DDRA=0x00;
```

```
// Port B initialisation
```

```
PORTB=0x00;DDRB=0x00;
```

```
// Port C initialisation
```

```
PORTC=0x00;DDRC=0xFF;
```

```
// Port D initialisation
```

```
PORTD=0x00;DDRD=0xFF;
```

Nous devons également rappeler la fonction pour initialiser l'afficheur LCD.

```
lcd_init(16);
```

Les tensions mesurées sont analogiques. Pour les afficher, il nous a fallu les convertir en numériques. Pour cela nous avons utilisé un convertisseur analogique-numérique intégré dans l'ATMEGA.

```
// Convertisseur Analogique - Numérique
```

```
unsigned int read_adc(unsigned char adc_input){
```

```
// Début de la conversion
```

```
ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);
```

```
ADCSRA|=0x40;
```

```
//Attendre la fin de la conversion
```

```
while ((ADCSRA & 0x10)==0); ADCSRA|=0x10;
```

```
return ADCW; }
```

Le résultat de conversion est un nombre compris entre 0 et 1024 lorsqu'il s'agit d'un convertisseur de 10 bits.

### 3.5. Mesurage et affichage de tension de la batterie

La tension aux bornes de la batterie quand celle-ci est bien chargée peut monter jusqu'à environ 28V mais le convertisseur supporte une tension maximale de 5V. Il faut donc un pont diviseur de tension pour réduire cette tension de 28V.

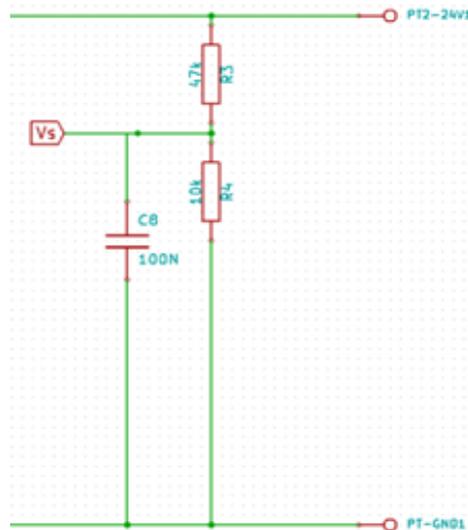


Illustration 25: Diviseur de tension en sortie

Soit :

$0 < V_{bat} < 5V$  dont  $V_{bat}$  signifie la tension qui entre dans le CAN.

$0 < V_{bat.reel} < 28V$  dont  $V_{bat.reel}$  signifie la tension réelle aux bornes de la batterie.

Nous pouvons établir la relation suivante :

$$\frac{V_{bat.max}}{V_{bat.reel.max}} = \frac{R4}{R3+R4}$$

On prend  $R4 = 10\text{ K}\Omega$  avec  $V_{bat.max} = 5V$  et  $V_{bat.reel.max} = 28V$ .

$$\text{soit : } \frac{5}{28} = \frac{10}{R3+10}$$

Donc  $R3 = 46\text{ K}\Omega$  .

On prend  $R3 = 47\text{ K}\Omega$  pour qu'elle soit une valeur de résistance normalisée.

Nous pouvons alors passer à la partie programmation. Il faut d'abord déclarer des variables utilisées.

```
unsigned int i; // i est le résultat de conversion
unsigned int j; // j est le nombre de boucle qu'il faut répéter
unsigned long Vbat; // Vbat est la valeur réelle de tension du panneau.
unsigned char tampon [20]; // un tampon pour afficher la valeur sur lcd
```

Dans la boucle, nous allons lire le résultat de conversion analogique-numérique, convertir ce résultat en valeur numérique puis, les afficher.

```
while (1)
{
    i = 0; // on initialise la valeur de i
    for (j=0;j<20;j++) // pour diminuer les erreurs, on fait répéter cette boucle 20 fois
    {
        i = i + read_adc(4); // on utilise le port ADC4 pour la conversion.
    }
    i = i / 20; // on calcule la valeur moyenne de i

    sprintf ( tampon, " i = %4u ", i ); // copier le texte qu'on veut afficher dans un tampon
    lcd_gotoxy(0,0); // le positionnement du texte
    lcd_puts(tampon) //afficher le texte
```

```

// on converti i en une valeur numérique
Vbat = ((unsigned long)i * 280 /1023);

//on affiche cette valeur
sprintf (tampon, "Vbat = %2u.%1u V", Vbat/10, Vbat%10);

lcd_gotoxy(0,1);

lcd_puts(tampon);

```

### 3.6. Mesurage et affichage de tension du panneau solaire

Le panneau solaire peut délivrer une tension maximale de 17V comme pour la mesure de tension de la batterie. Il nous a fallu un pont diviseur de tension lorsque le convertisseur fonctionne en 0 - 5V.

soit:

$0 < V_{pan} < 5V$  dont  $V_{pan}$  signifie la tension qui entre dans le CAN.

$0 < V_{pan.reel} < 28V$  dont  $V_{pan.reel}$  signifie la tension réelle aux bornes du panneau.

Nous pouvons établir la relation suivante :

$$\frac{V_{pan.max}}{V_{pan.reel.max}} = \frac{R_2}{R_1 + R_2}$$

On prend  $R_2 = 10 \text{ KOhm}$  avec  $V_{pan.max} = 5V$  et  $V_{pan.reel.max} = 17V$ .

soit :  $5 / 17 = 10 / (R_1 + 10)$

Donc  $R_1 = 24 \text{ KOhm}$ .

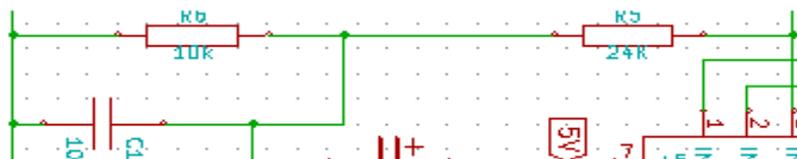


Illustration 26: Pont diviseur de tension en entrée

La partie programmation :

// Déclaration des variables

unsigned int k; // k est le résultat de conversion

unsigned int j; // j est le nombre de boucle qu'il faut répéter

unsigned long Vpan; // Vpan est la valeur réelle de tension du panneau.

unsigned char tampon [20]; // un tampon pour afficher la valeur sur lcd

while (1)

{

    k = 0; // on initialise la valeur de k

    for (j=0;j<20;j++) // pour diminuer les erreurs, on fait répéter cette boucle 20 fois

    {

        k = k + read\_adc(4); // on utilise le ADC4 pour la conversion.

    }

    k = k / 20; // on calcule la valeur moyenne de i

    sprintf ( tampon, “ k = %4u ”, k ); // copier le text qu'on veut afficher dans un tampon

    lcd\_gotoxy(0,0); // le positionnement du text

    lcd\_puts(tampon) //afficher le text

    // on converti i en valeur réelle

    Vpan = ((unsigned long)k\* 170 /1023);

    //on affiche cette valeur

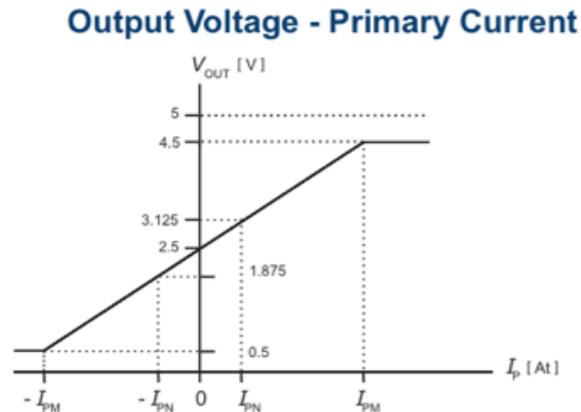
    sprintf (tampon, “Vpan = %2u.%1u V”, Vpan/10, Vpan%10);

    lcd\_gotoxy(0,1);

    lcd\_puts(tampon);

### 3.7. Mesurage et affichage de courant d'entrée du panneau solaire

La carte AT8535 ANALOG permet de mesurer que les tensions. Pour mesurer un courant, nous allons utiliser une carte «Interface BNC et alimentation pour le capteur LEM HASS ». Il s'agit d'un capteur de courant.



*Illustration 27: Tension en fonction de courant pour le capteur de courant*

En effet, le courant peut être une valeur négative. Quand il n'y pas de courant dans le capteur ( $I=0A$ ), il fournit une tension de 2,5 V, ce qui correspond à 512 en valeur numérique.

La partie programmation :

*// Déclaration des variables*

*unsigned int i; // k est le résultat de conversion*

*unsigned int j; // j est le nombre de boucle qu'il faut répéter*

*long Ipan; // Ipan est la valeur réelle de tension du panneau, il faut la déclarer en long car elle peut être une valeur négative*

*unsigned char tampon [20]; // un tampon pour afficher la valeur sur lcd*

*while (1)*

*{*

```

l = 0; // on initialise la valeur de l
for (j=0;j<20;j++) // pour diminuer les erreurs, on fait répéter cette boucle 20 fois
{
    l = l + read_adc(4); // on utilise le ADC4 pour la conversion.
}
l = l / 20; // on calcule la valeur moyenne de i

sprintf ( tampon, “ l = %4u ”, kl); // copier le text qu’on veut afficher dans un tampon
lcd_gotoxy(0,0); // le positionnement du text
lcd_puts(tampon) //afficher le text

// on converti l en valeur réelle
Ipan = ((float)(l-512)/6.656) ;
//on affiche cette valeur
sprintf (tampon, “Ipan = %2u.%1u V”, Ipan/10, Ipan%10);
lcd_gotoxy(0,1);
lcd_puts(tampon);

```

## 4. Amélioration du chargeur

### 4.1. Pont de diviseur pour la tension d'entrée (la tension de panneau)

Nous avons mis à la sortie de notre hacheur un pont diviseur de tension pour pouvoir permettre de mesurer la tension aux bornes des batteries. Cependant, il faut également envisager de mettre un pont diviseur de tension à la sortie du panneau voltaïque pour pouvoir mesurer la tension aux bornes du panneau voltaïque, car à l'entrée du hacheur nous avons du 12V, mais l'ATMEGA 8535 qui permet de mesurer la tension fonctionne en 5V.

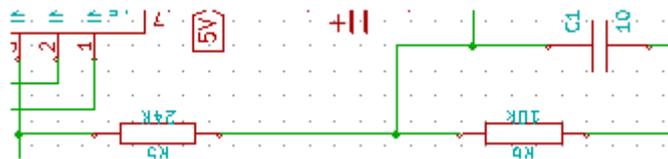


Illustration 28: Diviseur de tension en entrée

### 4.2. Rajout des LED

Sur notre carte, nous avons décidé d'ajouter 3 diodes électroluminescentes, une telle diode électroluminescente pour la tension d'entrée, une telle diode électroluminescente pour la tension de sortie et une telle diode électroluminescente qui sert de témoin pour la connexion avec l' ATMEGA 8535.

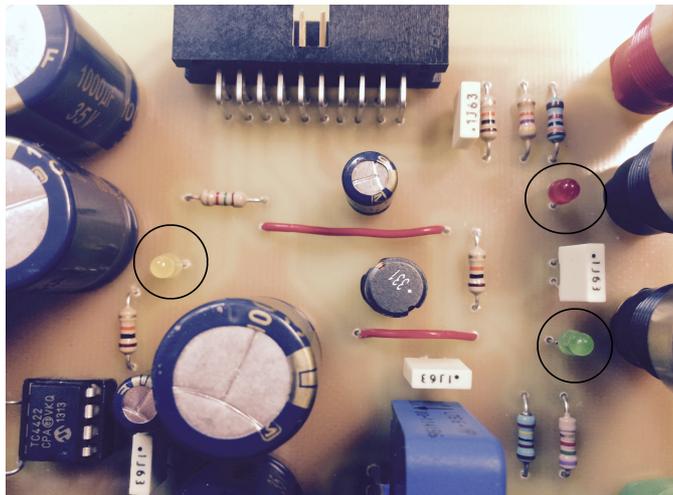
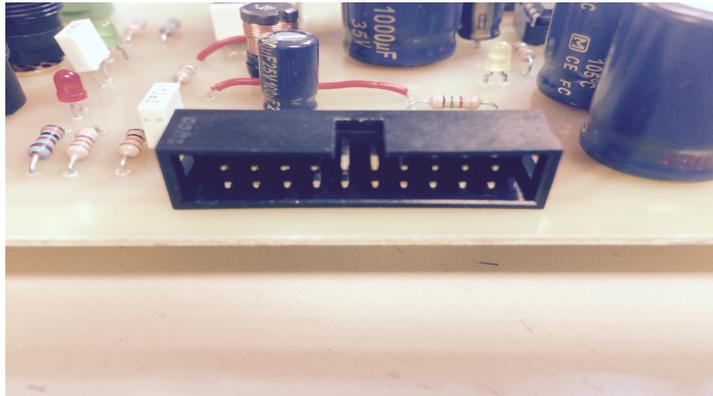


Illustration 29: diode électroluminescent témoin sur la carte

### 4.3. Rajout d'un connecteur

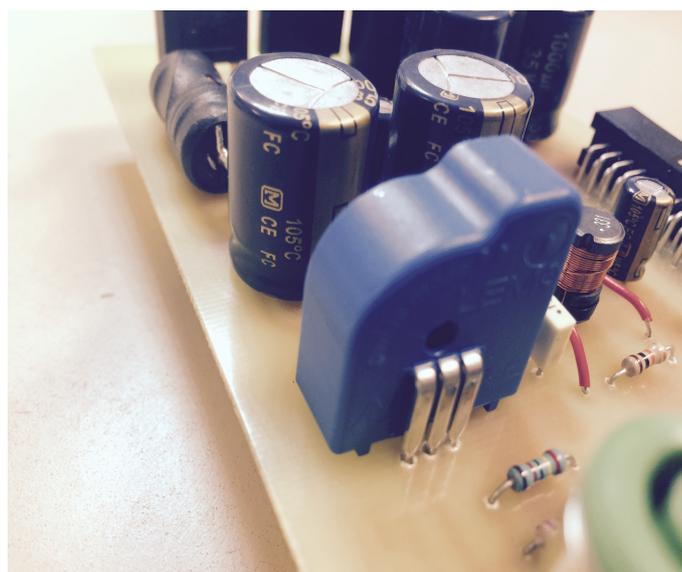
Pour la partie test, nous avons mis des borniers à vis sur notre carte que nous avons relié sur une carte test de l'ATMEGA8535 par des fils, mais il y avait différent bornier à vis en fonction de ce que l'on souhaitait commander, comme la MLI ou l'affichage de tension. Nous avons donc décidé d'ajouter un connecteur qui permet d'avoir une connexion directe avec la carte test de l'ATMEGA 8535. Ce connecteur permet de générer toutes les options de la carte test.



*Illustration 30: Connecteur de la carte*

### 4.4. Rajout d'un capteur de courant

Auparavant pour mesurer le courant fourni par le panneau, nous avons utilisé une carte sur laquelle il y a un capteur de courant qui converti une valeur de courant en une valeur de tension de façon linéaire. Pour faciliter la mesure, nous allons intégrer un capteur de courant “LTS 6-PN” sur notre carte de chargeur. La sortie de ce capteur va être reliée au connecteur.

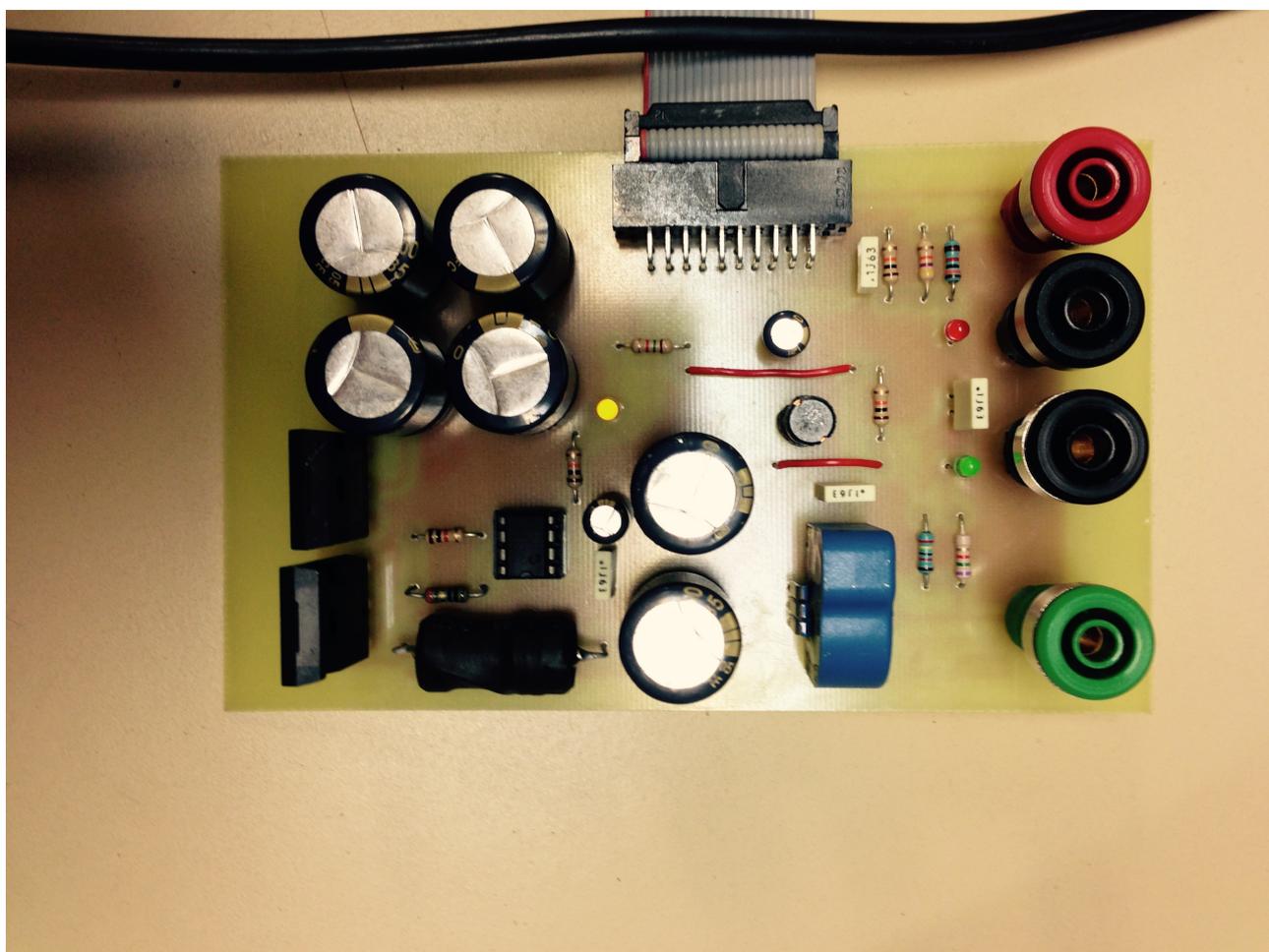


*Illustration 31: Capteur de courant LTS 6-NP*

Ensuite nous avons recommencé la nouvelle carte sur Kicad, où nous y avons intégré tous les nouveaux éléments (Pont diviseur de tension, led, connecteur et le capteur de courant). Pour commencer il a fallu modifier le schéma de base, en y insérant ces composants, en regardant leurs empreintes et en observant comme ces composants étaient reliés entre-eux.

Ceci est en fait, il a fallu leur associer les boîtiers pour que les composants au moment du routage aient la taille désirée. Ceci ne nous empêche pas de pouvoir les installer au moment des soudures.

Vous trouvez en annexe, les photos du routage, de la netlist et du schéma finale de la carte.



*Illustration 32: Vue de dessus de la carte finale*

## 4.5. Amélioration du programme

### 4.5.1. Mesurage simultané

La carte AT8535-ANALOG ne possède que deux ports de CAN. Les mesures que nous avons effectuées avec la première carte de chargeur étaient faites de façon séparées. En ajoutant un connecteur, cela nous permet de mesurer toutes les grandeurs de façon simultanée.

Nous avons distribué le port ADC0 à la tension du panneau, le port ADC2 au courant du panneau et le port ADC4 à la tension des batteries.

Le programme modifié est donc le suivant :

```
while (1)
{
i=0;
k=0;
l=0;
p0=p=0;

for(j=0;j<20;j++)
{
i=i+read_adc(4); // la tension des batteries est sur ADC4
k=k+read_adc(0); // la tension du panneau est sur ADC0
l=l+read_adc(2); // le courant du panneau est sur ADC2
}
i=i/20; // On calcul la valeur moyenne de conversion
k=k/20;
l=l/20;

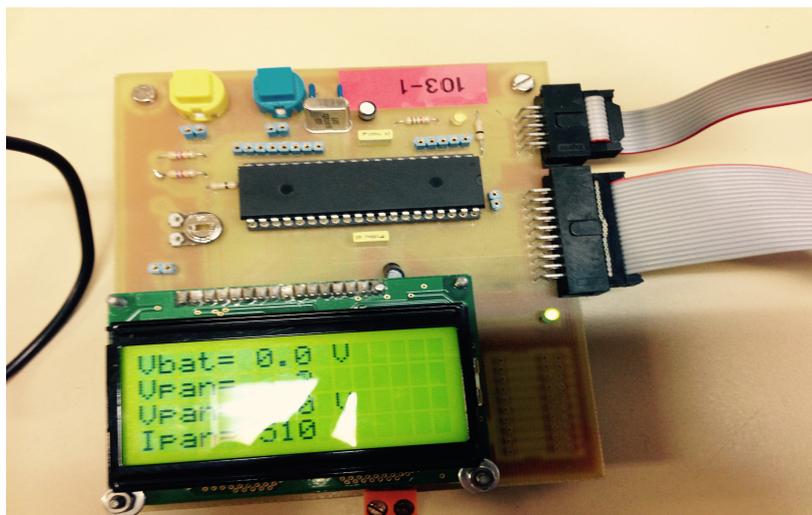
// on affiche la valeur de tension des batteries
Vbat=((unsigned long)i*290/1023);
sprintf(tampon,"Vbat=%2u.%1u V",Vbat/10,Vbat%10);
lcd_gotoxy(0,1);
lcd_puts(tampon);
```

```

// On affiche la valeur de tension du panneau
Vpan=((unsigned long)k*170/1023);
sprintf(tampon,"Vpan=%2u.%1u V",Vpan/10,Vpan%10);
lcd_gotoxy(0,2);
lcd_puts(tampon);

//On affiche la valeur de courant du panneau
//Ipan=((unsigned long)l*60/1023);
Ipan=((float)(l-512)/6.656);
sprintf(tampon,"I=%4u %2d.%1dA",l,Ipan/10,labs(Ipan)%10);
lcd_gotoxy(0,3);
lcd_puts(tampon);

```



*Illustration 33: Affichage de tension et de courant*

#### 4.5.2. Générer un signal MLI

Pour faire fonctionner un hacheur, il faut commander le transistor mosfet par un signal carré avec un rapport cyclique entre 0 et 1, ce qui est aussi appelé un signal MLI.

Le mosfet était commandé par un signal MLI délivré par un générateur de tension dans le test auparavant. Nous avons voulu générer ce signal MLI à partir du microcontrôleur.

Le rapport cyclique d'un signal MLI est entre 0 et 1, cela correspond 0 à 256 en valeur numérique pour le microcontrôleur.

```
//commander MLI
```

```
unsigned char m=0x20; //déclaration de m, m étant la valeur numérique du rapport cyclique
```

```
PORTD.0=1;
```

```
if (PIND.1 == 0) // modifier le rapport cyclique avec deux boutons
```

```
{
```

```
  m=m+1;
```

```
};
```

```
if (PIND.2 == 0)
```

```
{m=m-1;
```

```
};
```

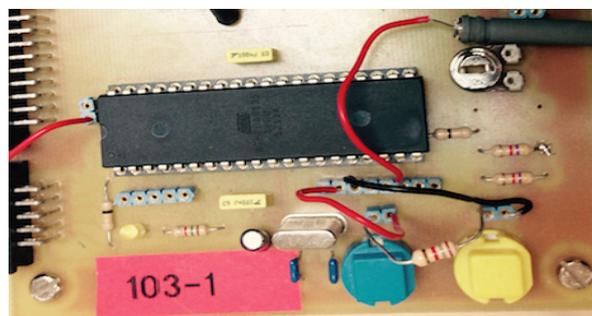
```
OCR1A=m;
```

```
// afficher la valeur de m
```

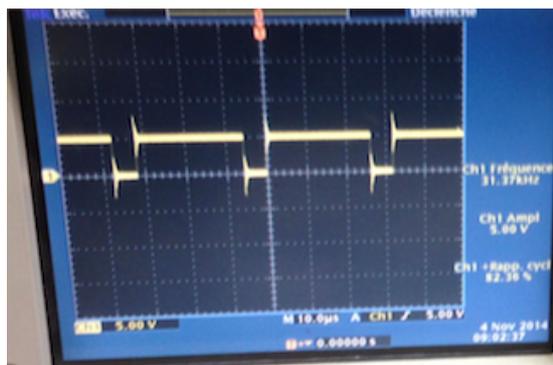
```
sprintf(tampon,"m=%d<---",m);
```

```
lcd_gotoxy(0,0);
```

```
lcd_puts(tampon);
```



*Illustration 34: Générer un signal MLI à partir de l'ATMEGA8535*



*Illustration 35: Signal MLI*

### 4.5.3. Recherche de puissance maximale du panneau solaire

Nous avons précédemment mesuré et affiché la tension ainsi que le courant. Nous allons ensuite nous intéresser à la puissance du panneau solaire.

$P = V \cdot I$  avec P:la puissance, V : la tension, I : le courant

Partie programmation :

```
long p ;  
p = (long) Ipan * Vpan; // puissance avec sa valeur réelle  
sprintf(tampon, " p=%5d/ %3d", p, Vpan); // afficher la puissance p  
lcd_gotoxy(0,1);  
lcd_puts(tampon);
```

A chaque cycle, il doit toujours comparer la puissance avec celle du cycle précédente (P0)

Si  $P > P_0$ , c'est-à-dire que la puissance est en train d'augmenter, il faut augmenter le rapport cyclique.

Si  $P < P_0$ , c'est-à-dire que la puissance est en train de diminuer, il faut donc diminuer le rapport cyclique.

La tension fournie par le panneau (Vpan) varie, mais en sortie, aux bornes des batteries elle ne doivent jamais dépasser sa valeur maximale, soit 28V. Il faut donc limiter la valeur maximale de rapport cyclique en fonction de la tension de panneau.

Soit :  $V_{bat.max} / V_{pan} = m / 1 - m$  avec  $V_{bat.max} = 28V$

Nous pouvons en déduire la valeur maximale de m :

$m_{max} = 256 - 8.53 \cdot V_{pan}$

Pour une raison de sécurité, m doit être toujours entre 40 % et 60 % quelque soit les autres conditions.

Partie programmation :

```
m_max = 256 - 8.53 * Vpan; // valeur maximale du rapport cyclique en fonction de la  
                             tension du panneau
```

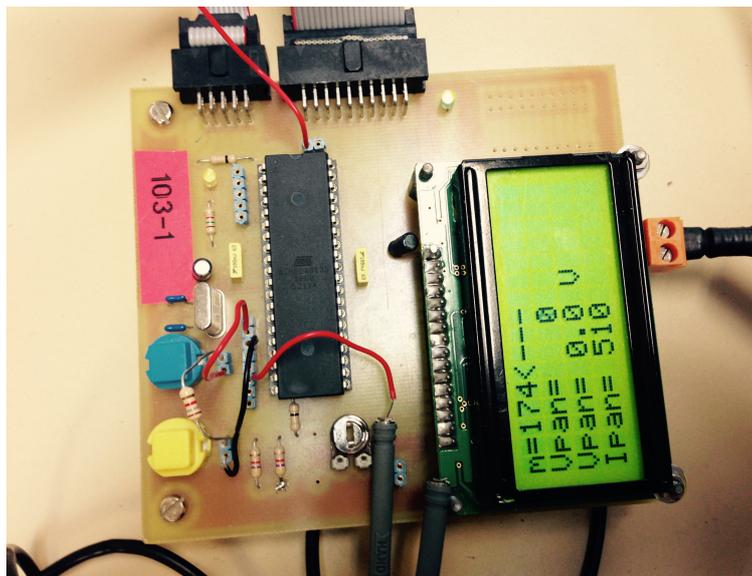
```
if (p > p0)
```

```
// si  $p > p_0$  et m n'a pas encore atteint sa valeur maximale, on augmente le rapport cyclique.
```

```

{
if (m<m_max)
{
m++;
}
}
// si p<p0, on diminue le rapport cyclique
if (p < p0)
{
m--;
}
p0 = p; // on stocke le valeur de p dans un mémoire p0 pour les comparer plus tard
PORTD.0=1;
if (m>153) m=153; // limitation du rapport cyclique entre 40 % et 60 % pour pour la sécurité
if (m<102) m=102;
OCR1A=m; //changer le rapport cyclique
sprintf(tampon,"m=%d<---",m); // afficher le rapport cyclique
lcd_gotoxy(0,0);
lcd_puts(tampon);

```



*Illustration 36: Comparaison de puissance et ajustement du rapport cycle*

## 5. Problèmes rencontrés

Lors de ce projet, nous n'avons pas rencontré de grosses difficultés. Nous n'avons pas eu de difficulté au niveau de la réalisation de la carte, à part au niveau de la conception de la carte test. Nous ne nous sommes pas rendu compte de la largeur des pistes à mettre pour les douilles et pour certains composants. Lors du soudage, les pistes étaient tellement fines que l'on a préféré recommencer une nouvelle carte en augmentant les pistes pour être sur que la carte fonctionne du premier coup, et ne pas produire de court-circuit.

Ce petit incident nous a retardé seulement d'une séance car nous nous sommes rendu compte de notre erreur tout de suite.

Après les seules difficultés que nous avons rencontrées sont au niveau de la programmation de l'afficheur. Le mesurage de tension et de courant se fait via un convertisseur analogique numérique. Le résultat est un nombre compris entre 0 et 1024. Pour afficher la valeur réelle d'une grandeur, il faut faire des calculs.

Quand nous avons essayé d'afficher la puissance sur l'écran, nous avons eu un problème d'affichage car il dépassait le nombre d'octets autorisés. Pour résoudre ce problème, nous avons pris la valeur réelle de la tension et du courant pour calculer la puissance. Ce résultat était assez petit donc cela ne nous a pas posé de problème.

Mais dans l'ensemble du projet, nous n'avons pas eu de soucis majeur et nous avons pu finir notre projet dans le temps imparti.

## 6. Conclusion

Pour conclure ce projet d'Etude et Réalisation du semestre 4 dans le cadre du génie électrique et du génie informatique, nous pouvons dire que ce projet nous a permis de mettre en œuvre, toutes nos connaissances acquises lors de ces deux ans de formation au sein de l'IUT GEII.

Ce projet a été divisé en plusieurs parties. Dans un premier temps, nous avons mis en place un hacheur, ce qui nous a permis de mettre en pratique toute nos connaissances vues au semestre 3 dans la matière de MC-ET2, car dans cette matière nous avons seulement vu les différents types de Hacheur.

Ensuite, nous nous sommes intéressés à la partie programmation, avec essentiellement la programmation de microcontrôleur, ce qui a permis de mettre nos connaissances du semestre 2 et 3 en avant.

Lors de la conception de notre carte du chargeur, nous avons mis en avant nos connaissances vues en électronique et fabrication de circuits imprimés tout au long de notre formation.

Ce projet nous a permis de travailler en binôme et travailler en autonomie, et pour finir nous voudrions remercier Mr Thierry LEQUEU qui a été notre professeur référent pendant ce projet et qui nous a aidé avec ses diverses connaissances.

## 7. Le résumé du projet

Au sein du 4ème semestre de l'IUT GEII, nous devions réaliser un projet pour le cours de l'étude et réalisation. Nous avons choisi de fabriquer un chargeur de panneau solaire pour charger des batteries d'une trottinette.

Pour cela, nous avons d'abord fait des études sur le panneau et la batterie pour connaître leurs caractéristiques puisque le panneau fournit une tension de 12V et les deux batteries en parallèle demande une tension de 24V. Il nous a fallu un montage qui convertit une tension de 12V à 24V. Inspiré par un montage existant, nous avons décidé de créer un hacheur Boost 12V – 24V 50W. Après avoir calculé la partie puissance, nous avons sélectionné et commandé les composants essentiels pour notre projet.

La conception du schéma et du typon se fait sous le logiciel Kicad. Nous avons donc fabriqué notre première carte. Puisque le panneau fournit une puissance variable et les batteries ne doivent pas être surchargées. Il est donc important de mesurer la tension et le courant du panneau, et des batteries.

Cette partie de programmation est fait avec un microcontrôleur l'ATMEGA8535 sous code vision AVR. En utilisant un convertisseur analogique et numérique, nous avons pu mesurer et afficher différentes grandeurs. Ensuite nous nous sommes rendu compte que notre première carte avait quelques défauts : nous n'avions pas prévu un connecteur pour la programmation, le signal MLI était géré par un générateur, le capteur de courant n'était pas sur la carte etc.

Nous avons donc re-modifié notre schéma et refait une nouvelle carte. Avec cette nouvelle carte, nous avons pu mesurer et afficher les valeurs de tension et de courant, calculer la puissance du panneau et ajuster le rapport cyclique en conséquence.

278 mots

## Mots clés

- \* Chargeur
- \* Panneau
- \* Batterie
- \* Trottinette
- \* Solaire
- \* Hacheur
- \* BOOST
- \* Inductance
- \* LTS 6-NP
- \* Condensateur
- \* Diode schottky
- \* Transistor
- \* Mosfet
- \* LED
- \* Pont divise de tension
- \* ATMEGA 8535
- \* Connecteur
- \* Programmation
- \* Kicad
- \* CodeVisionAVR
- \* Convertisseur Analogique-Numérique
- \* Vision
- \* Afficheur
- \* Création carte
- \* Insolation
- \* Perçage
- \* Soudage

## Index des illustrations

Illustration 1: Montage final de notre chargeur solaire.....	2
Illustration 2: Relation tension courant du panneau solaire.....	5
Illustration 3: schéma synoptique du montage.....	6
Illustration 4: Panneau solaire PWX 500.....	7
Illustration 5: Batterie N°1 12V 12 Ah.....	7
Illustration 6: Batterie N°2 12V 12 Ah.....	7
Illustration 7: Schéma structurelle d'une hacheur BOOST.....	9
Illustration 8: Schéma de base d'un convertisseur BOOST.....	10
Illustration 9: Montage simplifié pour $t$ appartenant à $[0$ et $\alpha]$ .....	11
Illustration 10: Montage simplifié pour $t$ appartenant à $[\alpha$ et $T]$ .....	11
Illustration 11: Puissance d'entrée en fonction du courant de sortie.....	12
Illustration 12: Rendement en fonction du courant de sortie.....	13
Illustration 13: Tension de sortie en fonction du courant de sortie.....	13
Illustration 14: Mosfet 55V-80A T0-247AC.....	15
Illustration 15: Diode schottky T0-247AC.....	16
Illustration 16: Inductance 47 $\mu$ H - 4A.....	16
Illustration 17: TC4422.....	17
Illustration 18: Condensateur 35V - 1000 $\mu$ F.....	17
Illustration 19: Carte de test de notre hacheur.....	18
Illustration 20: Tension de charge de la batterie.....	19
Illustration 21: Vu de carte ATMEGA 8535.....	20
Illustration 22: Carte de mesure : AT8535 - ANALOG.....	21
Illustration 23: ATMEGA 8535.....	21
Illustration 24: Interface de Code Vision AVR.....	23
Illustration 25: Diviseur de tension en sortie.....	25
Illustration 26: Pont diviseur de tension en entrée.....	27
Illustration 27: Tension en fonction de courant pour le capteur de courant.....	29
Illustration 28: Diviseur de tension en entrée.....	31
Illustration 29: diode électroluminescent témoin sur la carte.....	31
Illustration 30: Connecteur de la carte.....	32
Illustration 31: Capteur de courant LTS 6-NP.....	32
Illustration 32: Vue de dessus de la carte finale.....	33
Illustration 33: Affichage de tension et de courant.....	35
Illustration 34: Générer un signal MLI à partir de l'ATMEGA8535.....	36
Illustration 35: Signal MLI.....	36
Illustration 36: Comparaison de puissance et ajustement du rapport cycle.....	38

## Bibliographie

[1] **Cours MC-ET2**, *Hacheur*, semestre 3

[2] **Radiospare**, Transistor , [En Ligne ]. (Page consultée le 1/11/2014)

<<http://img-europe.electrocomponents.com/largeimages/RBIPOLAR-10.jpg>>

[3] **Radiospare**, diode, [En Ligne ]. (Page consultée le 1/11/2014)

<<http://img-europe.electrocomponents.com/largeimages/RBIPOLAR-10.jpg>>

[4] **Radiospare**, inductance, [En Ligne ]. (Page consultée le 1/11/2014)

<<http://img-europe.electrocomponents.com/largeimages/R7361015-01.jpg>>

[5] **Radiospare**, TC4422 [En Ligne ]. (Page consultée le 1/11/2014)

<<http://img-europe.electrocomponents.com/largeimages/R8211179-01.jpg>>

[6] **Radiospare**, condensateur : [En Ligne ]. (Page consultée le 1/11/2014)

<<http://img-europe.electrocomponents.com/largeimages/R0571127-01.jpg>>

[7] **Hacheur Boost**, Cnvertisseur élévateur 12V → 24V-10A, 2010, [En Ligne ]. (Page consultée le 1/11/2014)

<[http://f6csx.free.fr/PROJETS/BOOST/Boost\\_qro/Boost\\_qro.htm](http://f6csx.free.fr/PROJETS/BOOST/Boost_qro/Boost_qro.htm)>

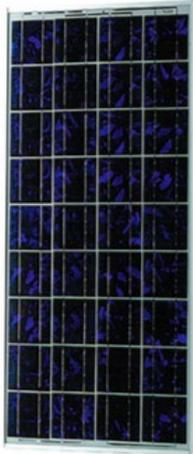


# **Index des annexes**

1.Datasheet.....	46
2.Datasheet batterie .....	49
3.Schéma de référence.....	50
4.Schéma carte test.....	51
5.Schéma carte finale.....	52
6.Netlist test.....	53
7.Netlist finale.....	54
8.Typon test.....	55
9.Typon final.....	56
10.Programme complet .....	57

# 1. Datasheet

## PHOTOWATT PWX500 - 12V MODULE PHOTOVOLTAÏQUE- JBox



APPLICATIONS
- Télécommunications
- Protection cathodique
- Pompage d'eau
- Signalisation
- Electrification rurale
- Applications résidentielles
- Bâtiments industriels
- Connexion réseau

- 4x9 cellules multicristallines (125,50 x 125,50 mm)
- Garantie produit : 5 ans\*
- Garantie puissance : 25 ans\*
- Tolérance de puissance : +/- 5Wc
- Certificats qualité : ESTI (61215), ISO 9001...

EMBALLAGE		
Poids du module	Kg	9,2
Dimensions du module	mm	1042 x 462 x 45
Type d'emballage	modules	4 per carton
Dimensions de l'emballage	mm	1100 x 490 x 183
Poids emballé	Kg	40
Dimension max d'une palette (72 modules)	mm	1150 x 1050 x 1600
Poids maximum d'une palette (72 modules)	Kg	650



Ce module PWX500 présente la technologie bi-verre, optimisé pour répondre à l'ensemble des applications PV. Grâce à ses 2 plaques de verre le PWX500 assure avec une tenue mécanique et une protection électrique accrues et répond particulièrement aux applications en environnement marin et tropical.

Le PWX500 utilise la technologie des cellules multicristallines PHOTOWATT. Les cellules solaires sont mesurées individuellement et triées électroniquement avant d'être interconnectées. L'encapsulation des cellules est réalisée entre une plaque de verre trempé et une feuille de Tedlar. L'encapsulant, de l'EVA résistant aux UV, enrobe les cellules photovoltaïques à l'intérieur des laminés et protège les cellules de la corrosion. La face arrière du module est étanche et protégée des dommages mécaniques par une feuille polymère continue et résistante.

Avec un centrage des tolérances à +/-5Wc, le PWX500 garantit l'homogénéité de puissance de vos installations, et un investissement financier correspondant réellement aux watts produits.

	PWX500		Configuration 12 V	
	<b>Puissance typique</b>	<b>W</b>	45	50
<b>Puissance minimale</b>	<b>W</b>	40,1	45,1	50,1
<b>Tension à la puissance typique</b>	<b>V</b>	17	17,2	17,3
<b>Intensité à la puissance typique</b>	<b>A</b>	2,65	2,9	3,2
<b>Intensité de court circuit</b>	<b>A</b>	2,9	3,2	3,4
<b>Tension en circuit ouvert</b>	<b>V</b>	21,4	21,6	21,7
<b>Tension maximum du circuit</b>	<b>V</b>	<b>600V DC</b>		
<b>Coefficient de température</b>		$\alpha = +1,46 \text{ mA/}^\circ\text{C}$ ; $\beta = -79 \text{ mV/}^\circ\text{C}$ ; $\gamma \text{ P/P} = -0,43 \text{ \% /}^\circ\text{C}$		

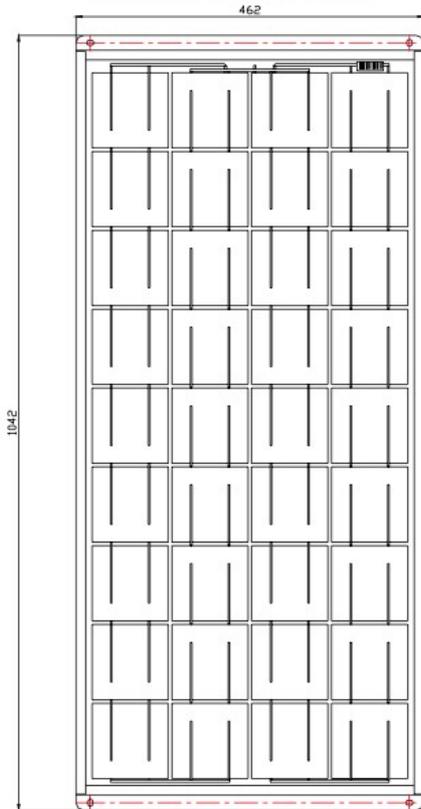
Spécifications de puissance à 1000 W/m<sup>2</sup> : 25°C : AM 1,5

\* Selon les conditions générales de garantie



33 Rue Saint Honoré - Z.I. Champfleuri  
38300 Bourgoin-Jallieu - FRANCE  
Téléphone +33 (0)4 74 93 80 20 - Fax +33 (0)4 74 93 80 40  
www.photowatt.com - info.fr@photowatt.com

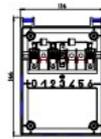
D15-P03-01 FR indice 0  
Les caractéristiques des produits de ce document sont données à titre indicatif et n'ont pas de valeur contractuelle. Sous réserve de la qualité de ses produits, la société Photowatt se réserve la possibilité d'en modifier les caractéristiques sans préavis.



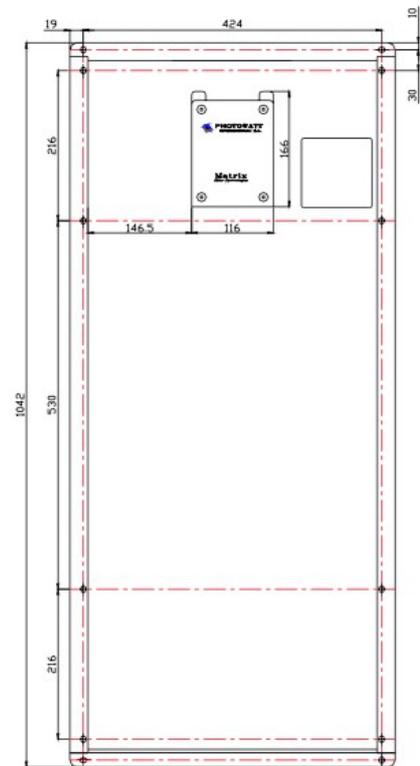
**DETAILS DE LA BOITE**



La boîte universelle accepte des câbles de 1,5 mm<sup>2</sup> à 4 mm<sup>2</sup> (AWG 11 à AWG16)

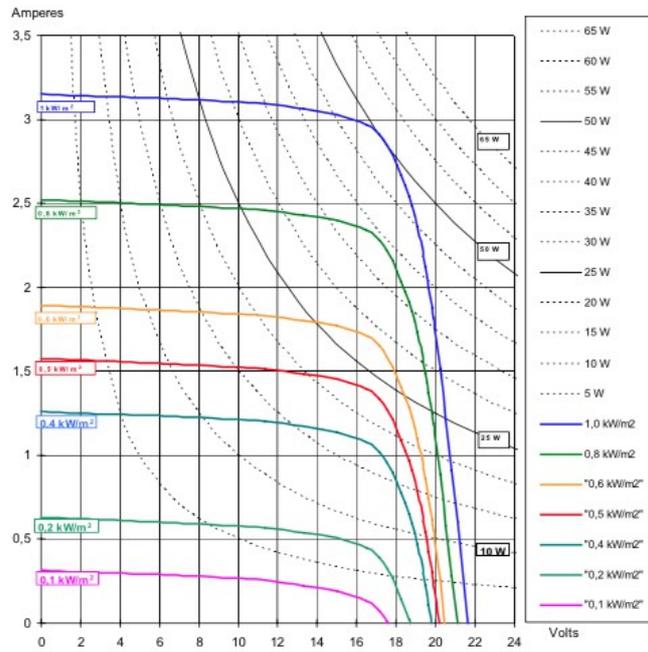
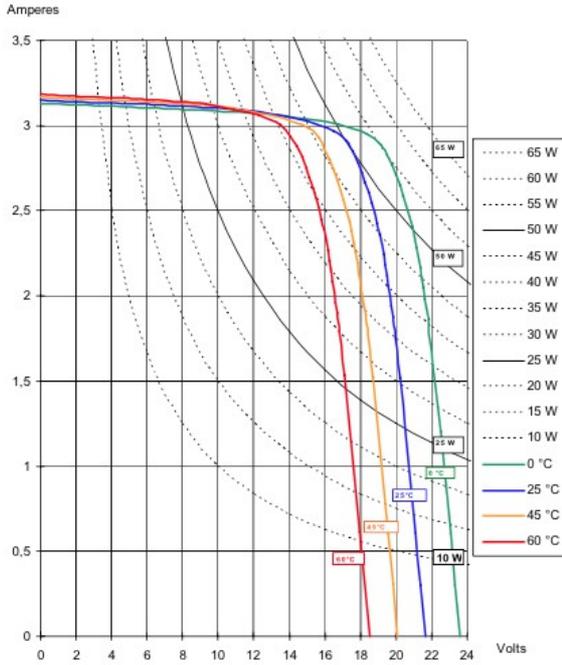


Ce module est protégé par 2 diodes schotky (1 diode bypass pour 18 cellules)



I=F(V) à E=1 kW/m<sup>2</sup>, AM=1,5 en fonction de la température

I=F(V) à T = 25°C en fonction de l'irradiance E (kW / m<sup>2</sup>), AM 1,5.



## 2. Datasheet batterie

### Gamme NP - Batteries plomb sans entretien

#### NP12-12

SPECIFICATIONS		
Tension nominale	12	V
Capacité en 20h (C20) à 1.75V/éle. (20°C)	12	Ah
Capacité en 10h (C10) à 1.75V/éle. (20°C)	11.1	Ah
DIMENSIONS		
Longueur	151 (±1)	mm
Largeur	98 (±1)	mm
Hauteur		mm
Hauteur bornes incluses	97.5 (±2)	mm
Poids (typique)	4.05	kg
TYPE DE BORNES		
Coque à languette (type Fixator, montage rapide, démontable)	6.35	mm
PLAGE DE TEMPERATURE DE FONCTIONNEMENT		
Stockage	-20°C à +60°C	
Charge	-15°C à +50°C	
Décharge	-20°C à +60°C	
STOCKAGE		
Perte de capacité par mois à 20°C (approximatif)	3	%
MATERIAU DU BAC		
Standard	ABS (UL 94-HB)	
Option flamme retardante (FR)	ABS (UL94-V0)	
TENSION DE CHARGE		
Tension de charge en floating à 20°C	13.65 (±1%)	V
	2.275 (±1%)	V/éle
Coefficient de correction de tension de charge en floating en fonction de la température (à partir de 20°C)	-3	mV/éle/°C
Charge en cyclage (ou rapide) à 20°C	14.5 (±3%)	V
	2.42 (±3%)	V/éle
Coefficient de correction de tension de charge en cyclage en fonction de la température (à partir de 20°C)	-4	mV/éle/°C
COURANT DE CHARGE		
Limite de courant de charge en floating	Pas de limite	A
Limite de courant de charge en cyclage (ou charge rapide)	3	A
COURANT MAXIMUM DE DECHARGE		
1 seconde	350	A
1 minute	75	A
COURANT DE COURT-CIRCUIT ET RESISTANCE INTERNE (selon la norme EN CEI 60896-21)		
Résistance interne	44.39	mΩ
Courant de court-circuit	320	A
IMPEDANCE		
Mesurée à 1 kHz	18	mΩ
PERFORMANCES ET CARACTERISTIQUES		
Voir manuel technique	NP	
DUREE DE VIE		
Classification EUROBAT: Commercial Standard	3 à 5	ans
Durée de vie Yuasa à 20°C	5	ans
SECURITE		
<b>Installation</b>	Peut être installée et utilisée dans toutes les positions, sauf à l'envers en permanence.	
<b>Poignées</b>	Les batteries ne doivent pas être suspendues par les poignées si poignées.	
<b>Souppes</b>	Chaque élément batterie est équipé de soupape pour permettre aux gaz de s'échapper et aussi assurer l'étanchéité.	
<b>Dégazage</b>	Les batteries VRLA produisent de l'hydrogène qui mélangé avec de l'air peut devenir explosif. Ne pas installer les batteries dans une enceinte étanche.	
<b>Recyclage</b>	Les batteries VRLA YUASA en fin de vie, doivent être recyclées selon la législation nationale en vigueur.	

### Fiche Technique



#### CERTIFICATIONS PAR ORGANISMES INDEPENDANTS

ISO 9001 - Système d'organisation qualité  
 ISO 14001 - Système d'organisation environnementale  
 EN 18001 - Système d'organisation hygiène et sécurité  
 UNDERWRITERS LABORATORIES (UL)  
 VdS (Germany) - VdS No. G189170



#### NORMES

IEC61056



TOUTES LES DONNEES PEUVENT ETRE MODIFIEES  
 SANS INFORMATION PREALABLE  
 Version N°: V.2 / Date de version: Mars 2010



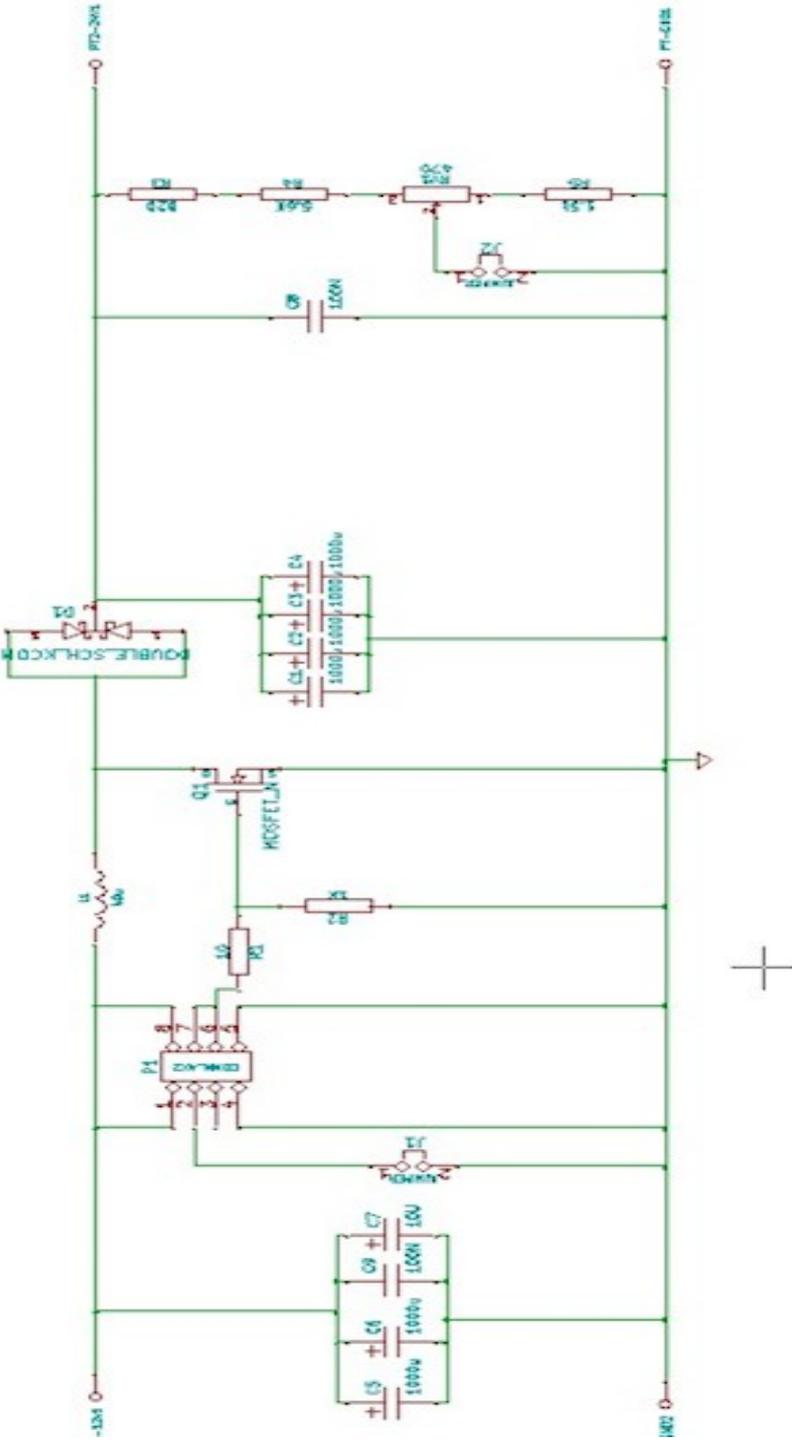
YUASA BATTERIES  
 FRANCE  
 Zac des Chesnes Ouest  
 13 rue du Morelton  
 36070 Saint-Quentin  
 Fallavier

[www.yuasaeurope.com](http://www.yuasaeurope.com)

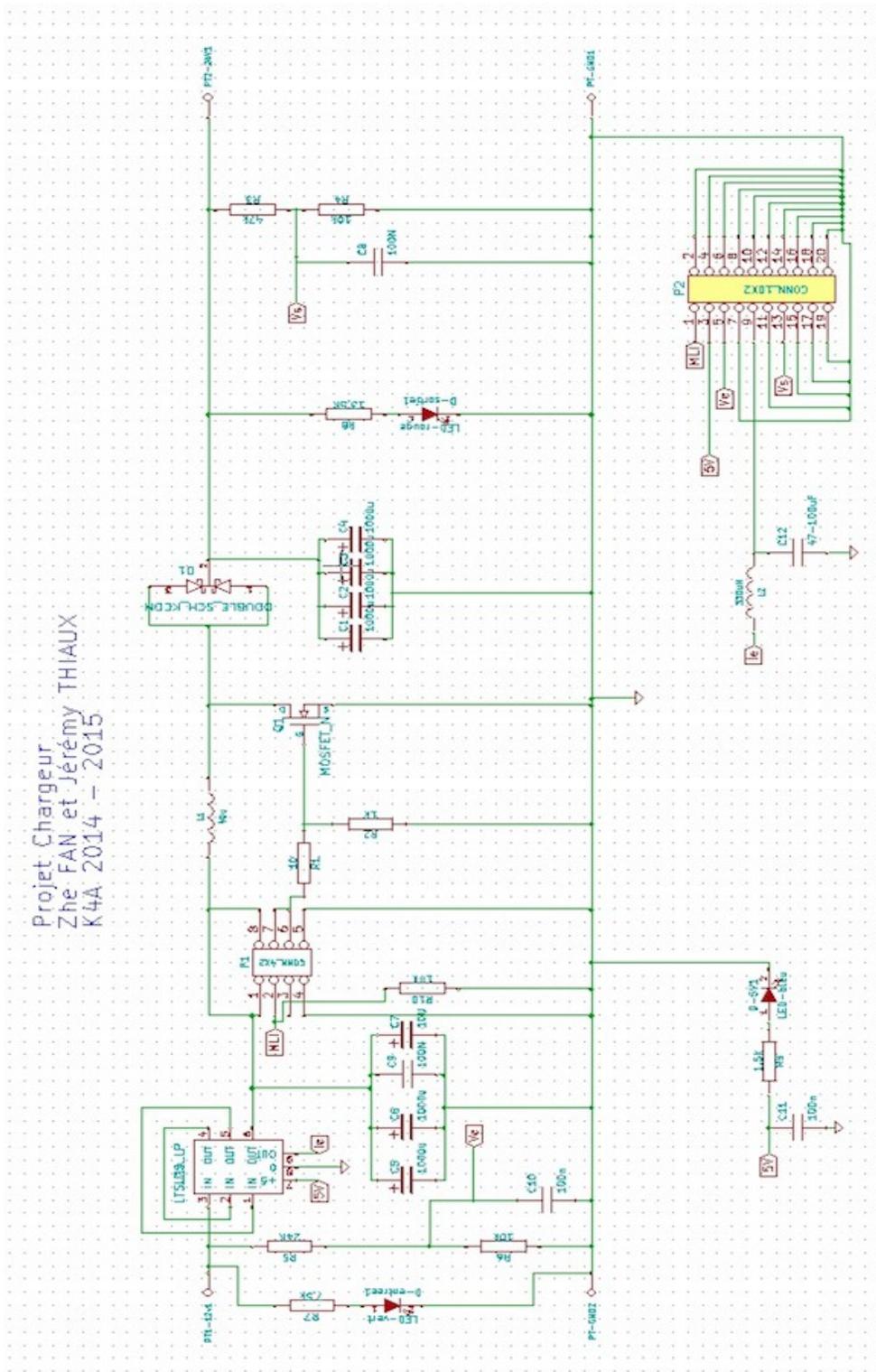
NP



# 4. Schéma carte test



# 5. Schéma carte finale



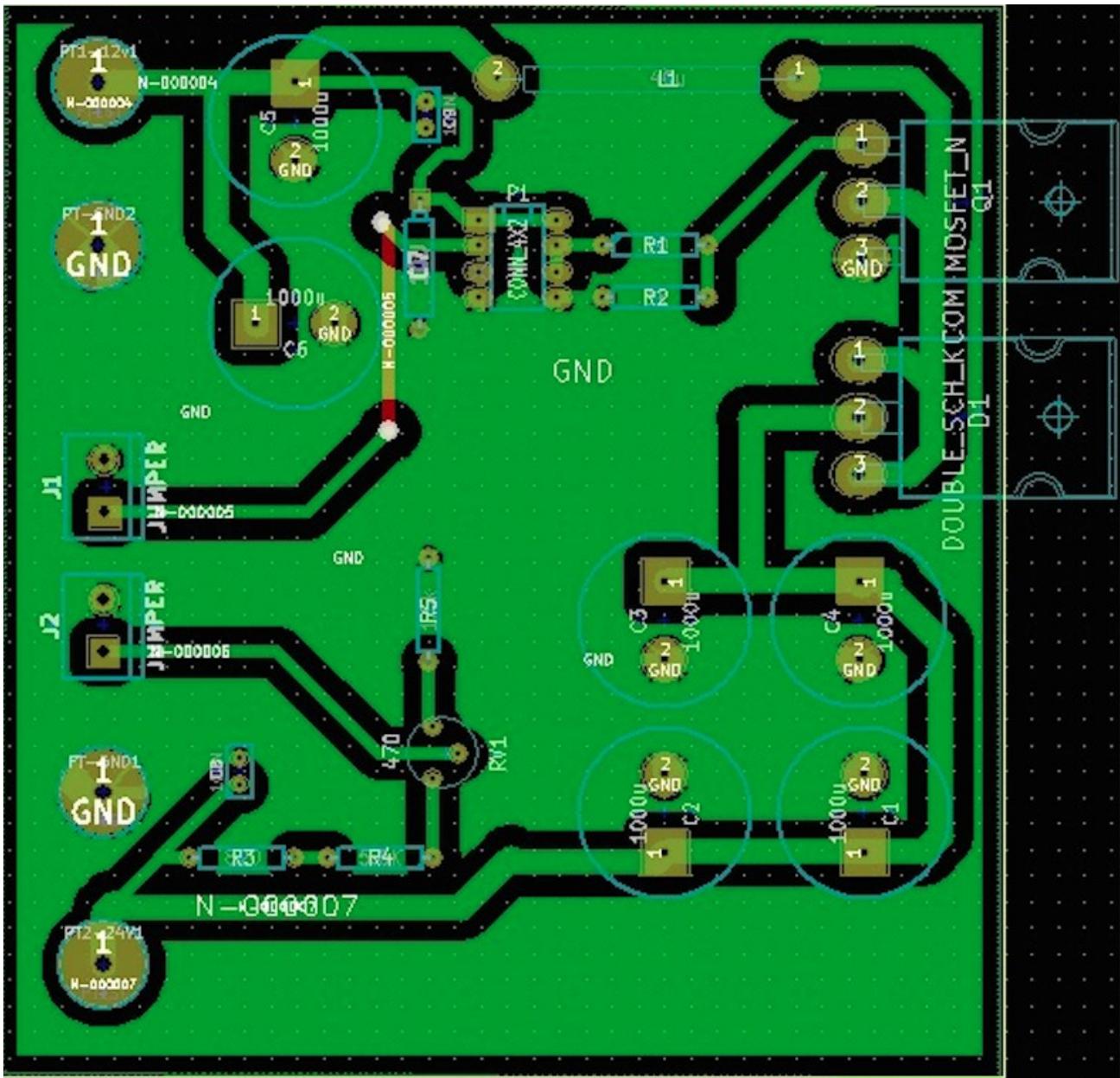
## 6. Netlist test

1	C1 -	1000u : C2V16
2	C2 -	1000u : C2V16
3	C3 -	1000u : C2V16
4	C4 -	1000u : C2V16
5	C5 -	1000u : C2V16
6	C6 -	1000u : C2V16
7	C7 -	10U : CP5
8	C8 -	100N : C1
9	C9 -	100N : C1
10	D1 -	DOUBLE_SCH_KCOM : linear-technology-2-T0247L
11	J1 -	JUMPER : bornier2
12	J2 -	JUMPER : bornier2
13	L1 -	40u : R5BIS
14	P1 -	CONN_4X2 : 8DIP300L
15	PT-GND1 -	PTEST : PTSORTIE
16	PT-GND2 -	PTEST : PTSORTIE
17	PT1-12v1 -	PTEST : PTSORTIE
18	PT2-24V1 -	PTEST : PTSORTIE
19	Q1 -	MOSFET_N : linear-technology-2-T0247L
20	R1 -	10 : R4
21	R2 -	1k : R4
22	R3 -	820 : R4
23	R4 -	5.6K : R4
24	R5 -	1.5k : R4
25	RV1 -	470 : RV2

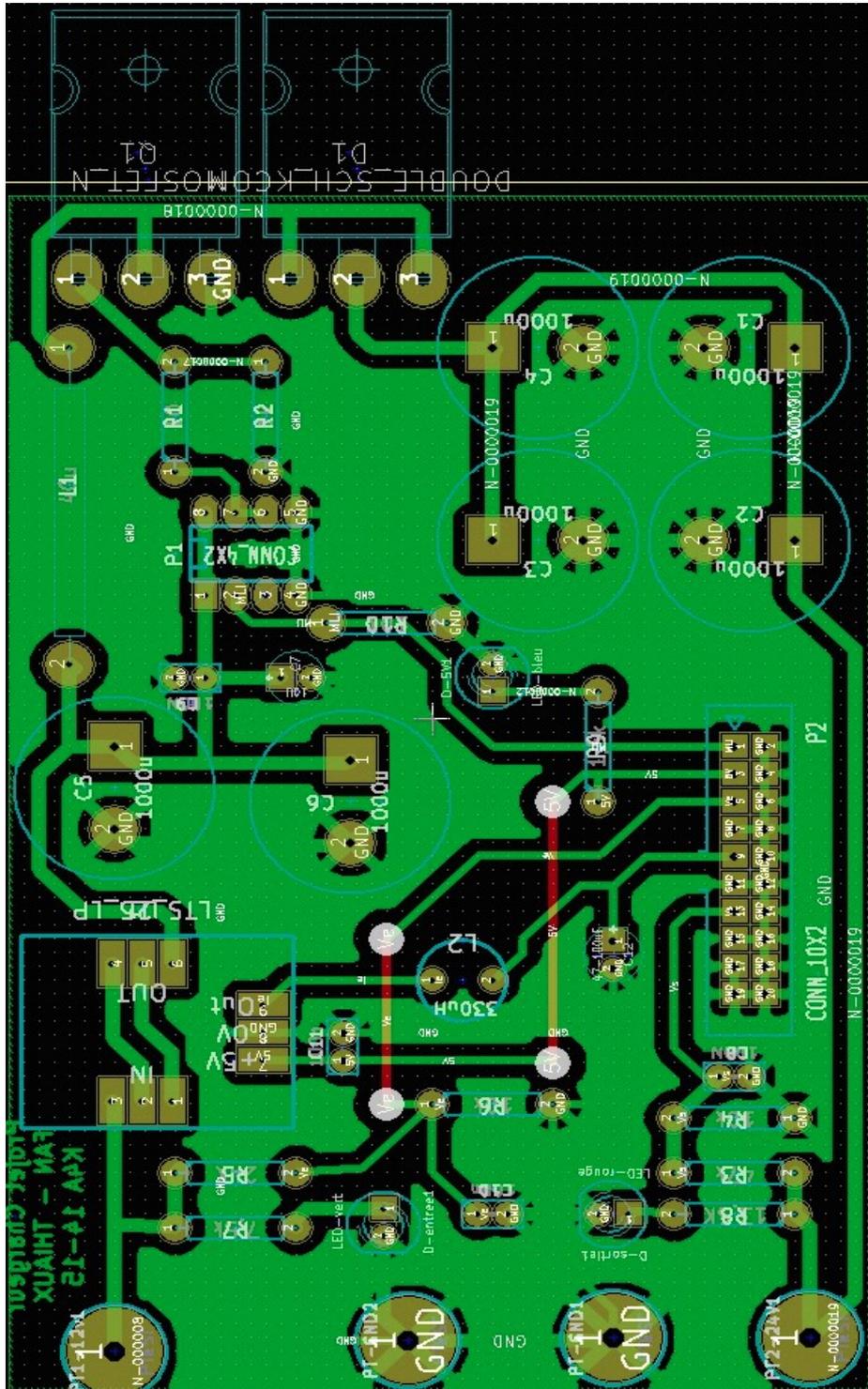
## 7. Netlist finale

1	C1 -	1000u : C2V16
2	C2 -	1000u : C2V16
3	C3 -	1000u : C2V16
4	C4 -	1000u : C2V16
5	C5 -	1000u : C2V16
6	C6 -	1000u : C2V16
7	C7 -	10U : C1V5
8	C8 -	100N : C1
9	C9 -	100N : C1
10	C10 -	100n : C1
11	C11 -	100n : C1
12	C12 -	47-100uF : C1V5
13	D-5V1 -	LED-bleu : LED-5mm
14	D-entree1 -	LED-vert : LED-5mm
15	D-sortie1 -	LED-rouge : LED-5mm
16	D1 -	DOUBLE_SCH_KCOM : linear-technology-2-TO247L
17	L1 -	40u : R5BIS
18	L2 -	330uH : INDUCTANCE
19	P1 -	CONN_4X2 : 8DIP300L
20	P2 -	CONN_10X2 : HE10_20D_new
21	PT-GND1 -	PTEST : PTSORTIE
22	PT-GND2 -	PTEST : PTSORTIE
23	PT1-12v1 -	PTEST : PTSORTIE
24	PT2-24V1 -	PTEST : PTSORTIE
25	Q1 -	MOSFET_N : linear-technology-2-TO247L
26	R1 -	10 : R4
27	R2 -	1k : R4
28	R3 -	47k : R4
29	R4 -	10k : R4
30	R5 -	24K : R4
31	R6 -	10k : R4
32	R7 -	7,5k : R4
33	R8 -	13,5K : R4
34	R9 -	1,5k : R4
35	R10 -	10K : R4
36	U1 -	LTS_25_LP : LTS_25_LP

## 8. Typon test



# 9. Typon final



## 10. Programme complet

```
/******
```

```
This program was produced by the  
CodeWizardAVR V1.25.3 Evaluation  
Automatic Program Generator  
© Copyright 1998-2007 Pavel Haiduc, HP InfoTech s.r.l.  
http://www.hpinfotech.com
```

```
Chip type      : ATmega8535  
Program type   : Application  
Clock frequency : 16,000000 MHz  
Memory model   : Small  
External SRAM size : 0  
Data Stack size : 128  
*****/
```

```
#include <mega8535.h>  
  
// Alphanumeric LCD Module functions  
#asm  
    .equ __lcd_port=0x15 ;PORTC  
#endasm  
#include <lcd.h>  
  
#define ADC_VREF_TYPE 0x00  
  
// Read the AD conversion result  
unsigned int read_adc(unsigned char adc_input)  
{  
    ADMUX=adc_input | (ADC_VREF_TYPE & 0xff);  
    // Start the AD conversion  
    ADCSRA|=0x40;  
    // Wait for the AD conversion to complete  
    while ((ADCSRA & 0x10)==0);  
    ADCSRA|=0x10;  
    return ADCW;  
}  
  
#include <stdio.h>  
#include <delay.h>  
#include <stdlib.h>  
  
// Declare your global variables here  
unsigned int i,j,k,l,m_max;
```

```

long p,p0;
unsigned long Vbat,Vpan;
long Ipan;
unsigned char m=0x20;
unsigned char tampon[20];

void main(void)
{
// Declare your local variables here

// Input/Output Ports initialization
// Port A 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
PORTA=0x00;
DDRA=0x00;

// Port B 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
PORTB=0x00;
DDRB=0x00;

// 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=0xff;

// Port D 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
PORTD=0x00;
DDRD=0x31;

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=FFh
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer 1 Stopped

```

```

// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer 1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0xB1;
TCCR1B=0x01;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x30;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer 2 Stopped
// Mode: Normal top=FFh
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

```

```

// ADC initialization
// ADC Clock frequency: 1000,000 kHz
// ADC Voltage Reference: AREF pin
// ADC High Speed Mode: Off
// ADC Auto Trigger Source: None
ADMUX=ADC_VREF_TYPE & 0xff;
ADCSRA=0x84;
SFIOR&=0xEF;

// LCD module initialization
lcd_init(16);
m=128; // rapport cyclique du hacheur = 50%
//lcd_gotoxy(0,0);
//lcd_putsf("Projet Chargeur");

p0=p=0;

while (1)
{
    i=0;
    k=0;
    l=0;

    for(j=0;j<20;j++)
    {
        i=i+read_adc(4); // tension de sortie
        k=k+read_adc(0); // tension d'entrÈe (panneau)
        l=l+read_adc(2); // courant d'entrÈe (panneau) + offset 512 !!
    }
    i=i/20;
    k=k/20;
    l=l/20;

    //on affiche la valeur de i
    sprintf(tampon,"Vbat=%4u",i);
    //lcd_gotoxy(0,0);
    //lcd_puts(tampon);

    // on affiche la valeur de tension de sortie de chargeur
    Vbat=((unsigned long)i*290/1023);
    sprintf(tampon,"Vbat=%2u.%1u V",Vbat/10,Vbat%10);
    //lcd_gotoxy(0,1);
    //lcd_puts(tampon);

```

```
//on affiche la valeur de k
sprintf(tampon,"Vpan=%4u",k);
//lcd_gotoxy(0,1);
//lcd_puts(tampon);
```

```
// On affiche la valeur de tension d'entrÈe de chargeur.
Vpan=((unsigned long)k*170/1023);
sprintf(tampon,"Vpan=%2u.%1u V",Vpan/10,Vpan%10);
//lcd_gotoxy(0,2);
//lcd_puts(tampon);
```

```
// on affiche la valeur de l
sprintf(tampon,"Ipan=%4u",l);
//lcd_gotoxy(0,3);
//lcd_puts(tampon);
```

```
//On affiche la valeur de courant d'entrÈe de chargeur.
//Ipan=((unsigned long)l*60/1023);
Ipan=((float)(l-512)/6.656);
sprintf(tampon,"I=%4u %2d.%1dA",l,Ipan/10,labs(Ipan)%10);
lcd_gotoxy(0,3);
lcd_puts(tampon);
```

//p = (long) k \* (long) (l-512); // puissance du panneau avec k = Vpan et l = Ipan + offset de 512.

p = (long) Ipan \* Vpan; // puissance avec valeur rÈelle

```
sprintf(tampon," p=%5d/ %3d",p,Vpan); // afficher la puissance p
lcd_gotoxy(0,1);
lcd_puts(tampon);
```

```
sprintf(tampon,"p0=%5d/ %3d",p0,Ipan); // afficher la puissance p0
lcd_gotoxy(0,2);
lcd_puts(tampon);
```

m\_max = 256 - 8.53\*Vpan; // valeur maximale du rapport cyclique en fonction de tension du panneau

```

if (p > p0)
{
if (m<m_max)
{
m++;
}
}

if (p < p0)
{
m--;
}

p0 = p; // on stocke le valeur de p dans un mÈmoire p0 pour les comparer plus tard

PORTD.0=1;
if (PIND.1 == 0) // modifier le rapport cyclique avec deux boutons
{m=m+1;
};
if (PIND.2 == 0)
{m=m-1;
};

if (m>153) m=153; // limitation du rapport cycliquepour pour la sÈcuritÈ
if (m<102) m=102;

OCR1AL=m; //changer le rapport cyclique

sprintf(tampon,"m=%d<---",m);
lcd_gotoxy(0,0);
lcd_puts(tampon);

delay_ms(1000);

};
}

```