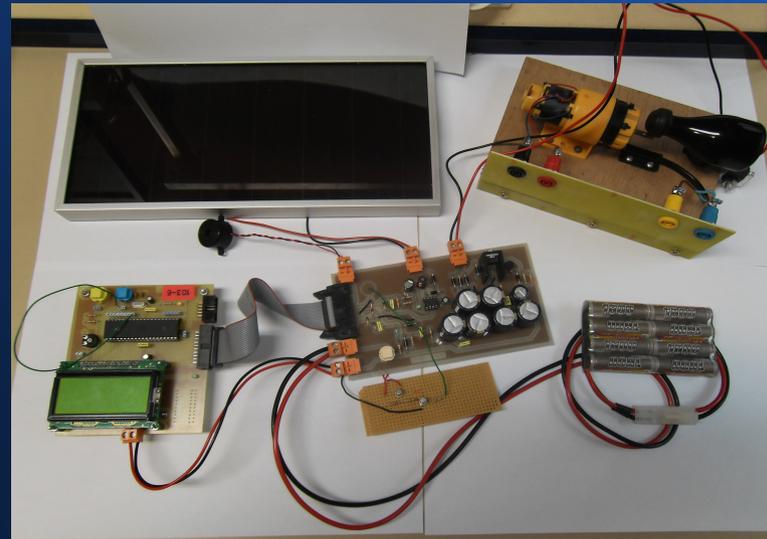


Éclairage à LED pour vélo avec autonomie



Thibault DELAROCHE

Vincent MOLLINIER

K4A

Thierry LEQUEU

Sofi RODIER

Plan

Introduction

Présentation du projet

Partie électronique

Partie programmation

Conclusion

Introduction

Gestion de l'éclairage à LED

- Batterie 9V
- Panneau solaire
- Dynamo
- ATmega8535

Présentation du projet

- Reprise de 2 projets existants
- Rendre le système autonome
- Encombrement et coût réduit

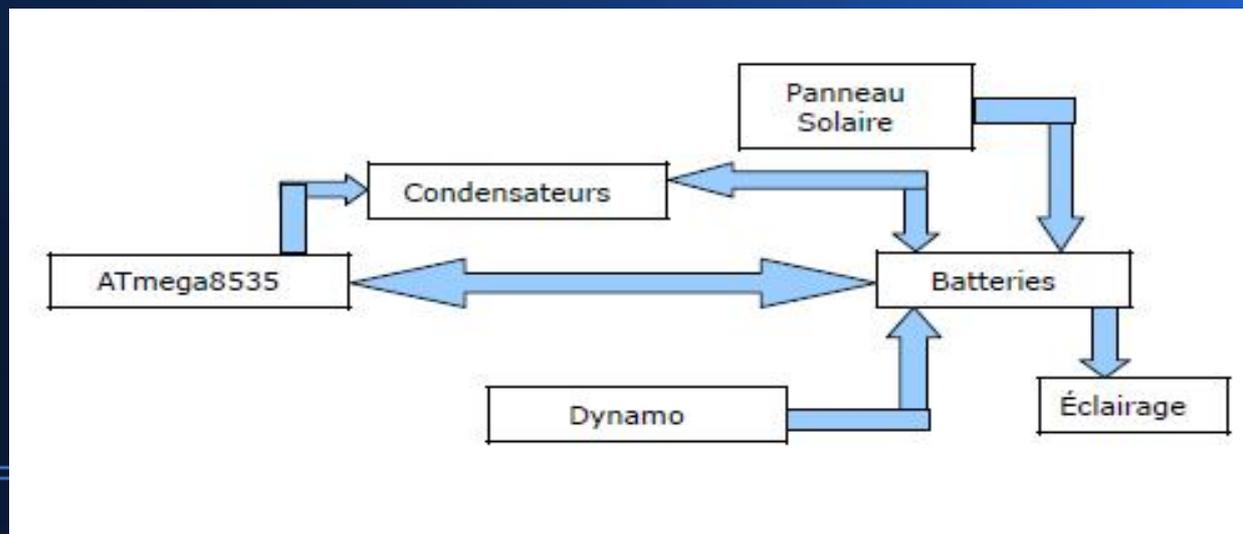
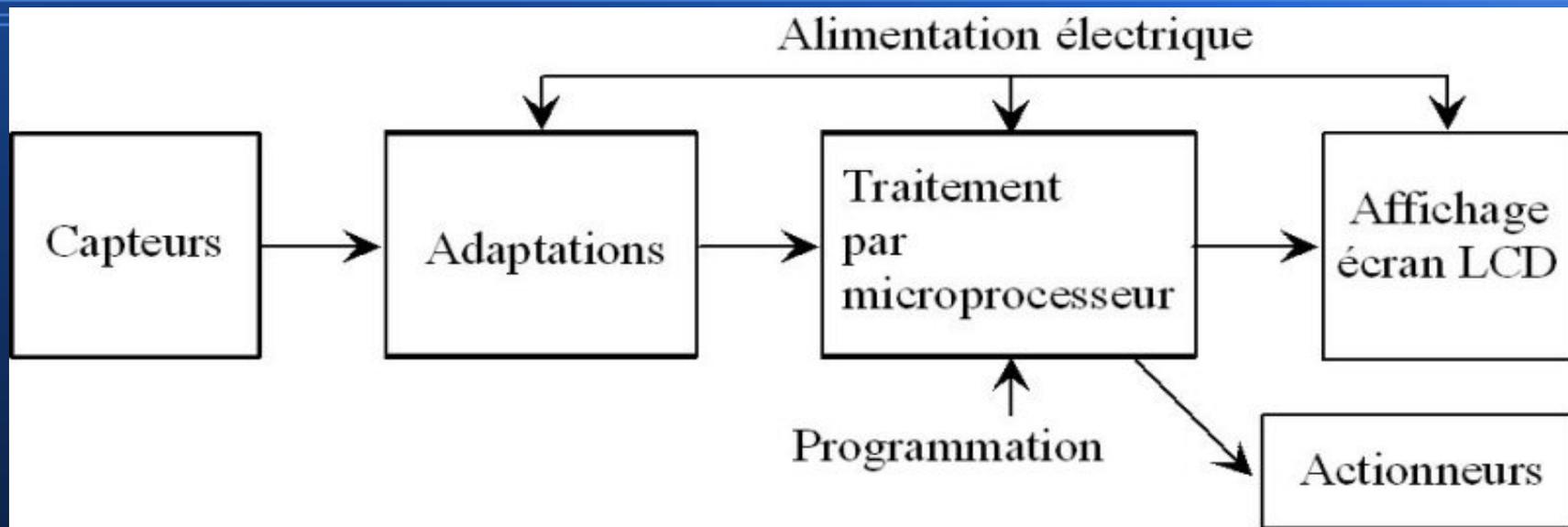
Présentation du projet

Utilisations de différentes sources

- Batterie 9V
- Panneau solaire
- Dynamo

Gestion par ATmega8535

Présentation du projet

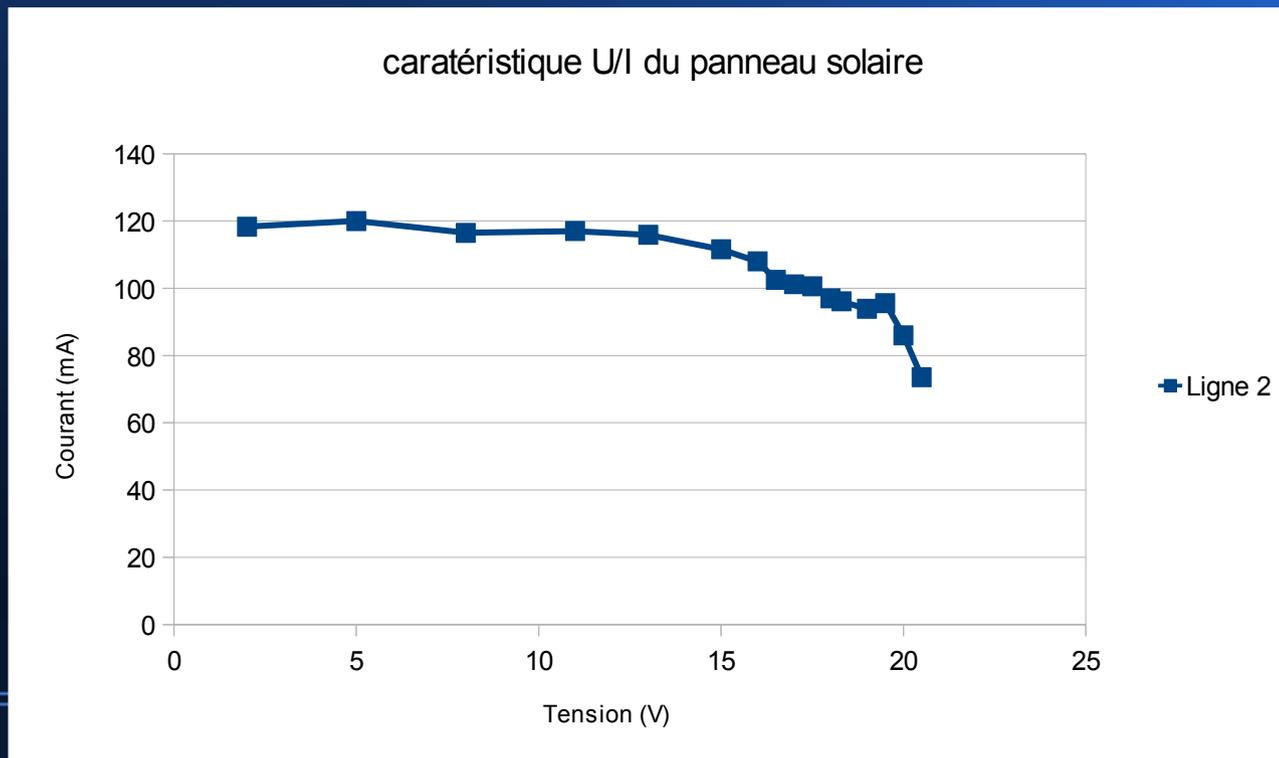


Partie électronique

Étude des différentes sources d'énergies

- Étude du panneau solaire

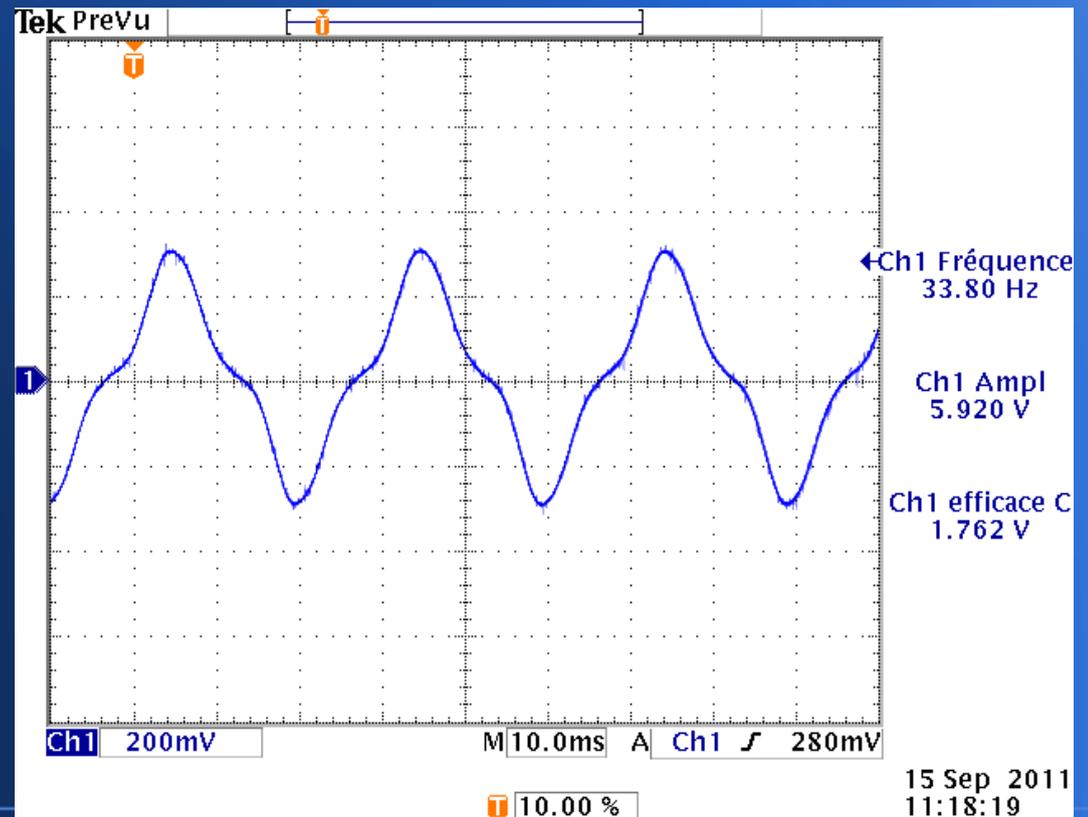
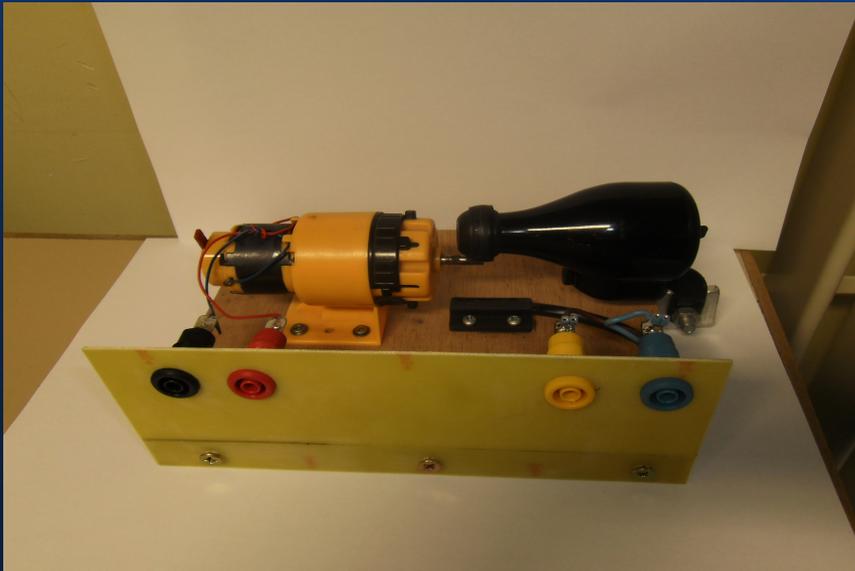
Tension (V)	20,5	20	19,5	19	18,3	18	17,5	17	16,5	16	15	13	11	8	5	2
Courant (mA)	73,6	86	95,6	93,9	96,1	97	100,6	101,2	102,5	108	111,6	115,9	117	116,5	120	118,3
Puissance (W)	1,51	1,72	1,86	1,78	1,75	1,74	1,76	1,72	1,69	1,73	1,67	1,5	1,29	0,93	0,6	0,24



Partie électronique

Étude des différentes sources d'énergies

- Étude de la dynamo

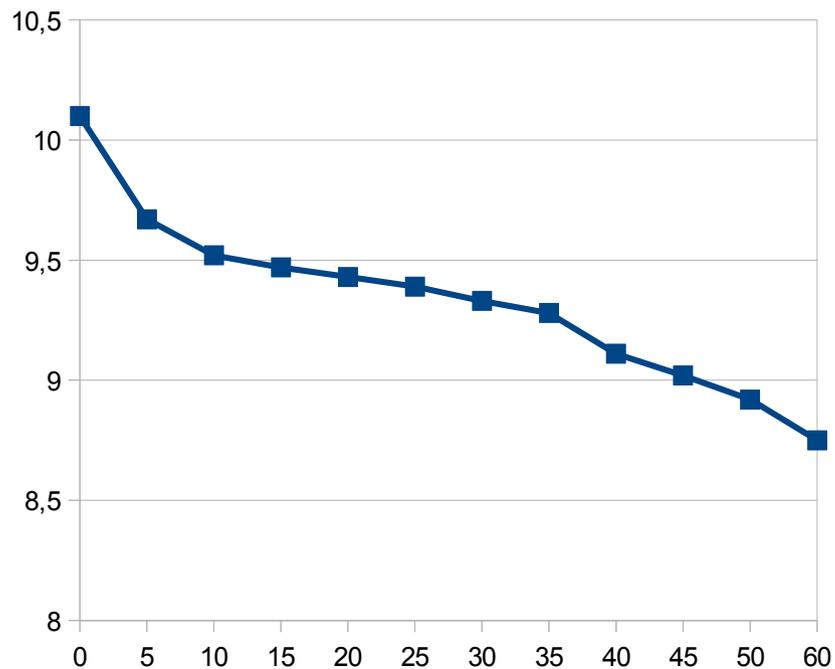


Partie électronique

Étude des différentes sources d'énergies

- Étude de la batterie

Tension U (V) en fonction du Temps (min)

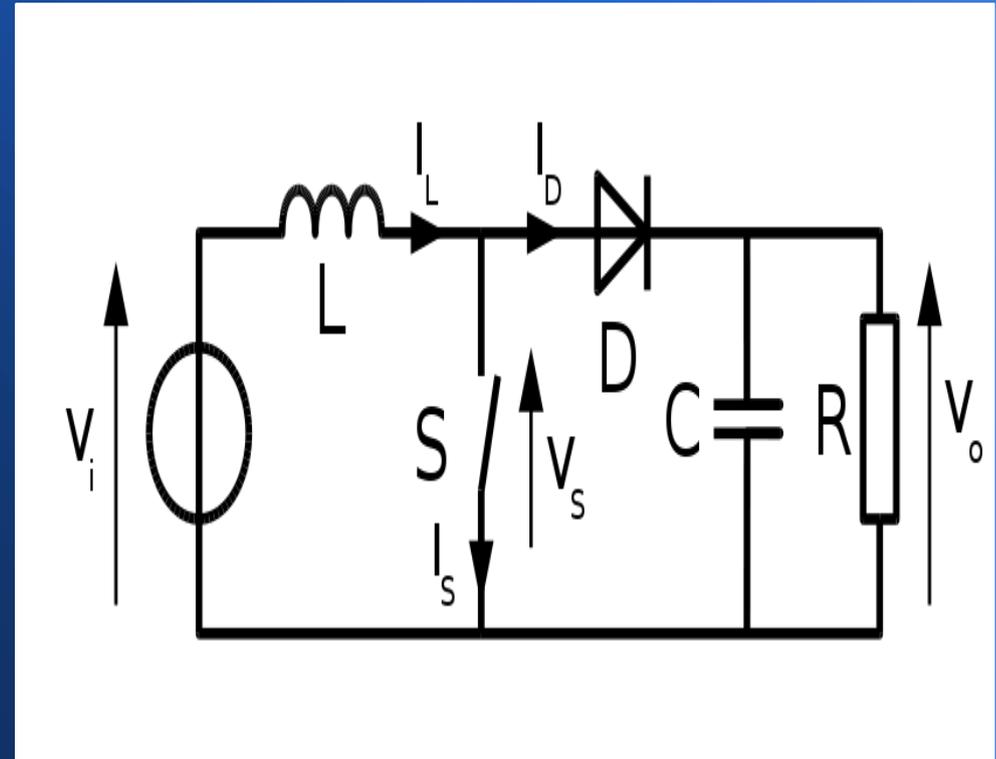
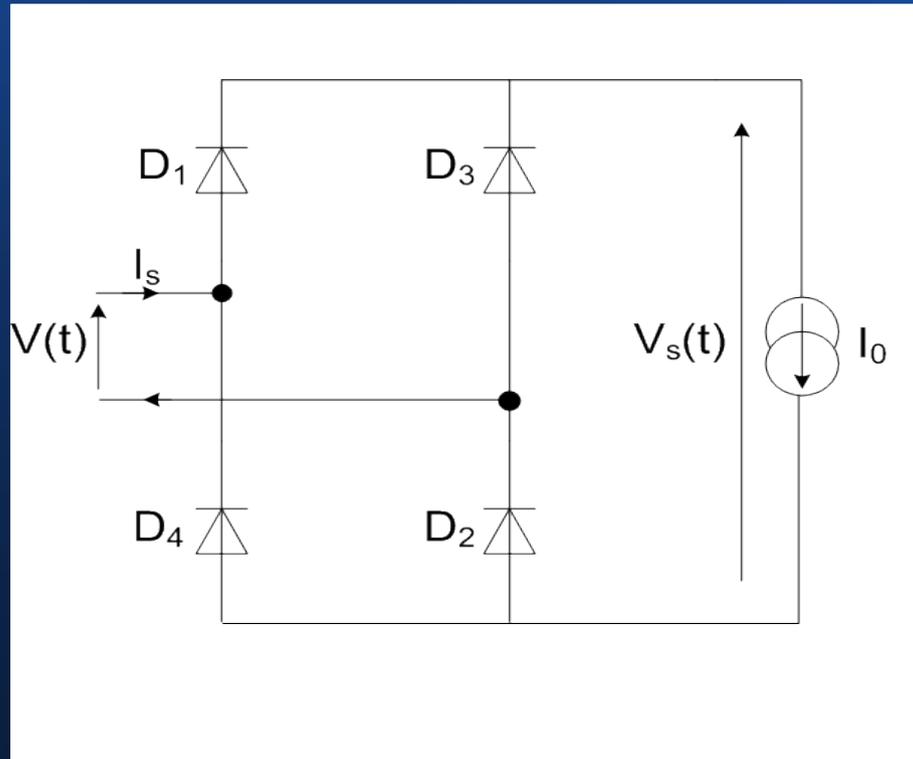


temps(min)	0	5	10	15	20	25	30	35	40	45	50	60
U(V)	10,1	9,67	9,52	9,47	9,43	9,39	9,33	9,28	9,11	9,02	8,92	8,75
I(A)	1,08	1,04	1,02	1,02	1,01	1,01	1	1	0,98	0,97	0,96	0,94



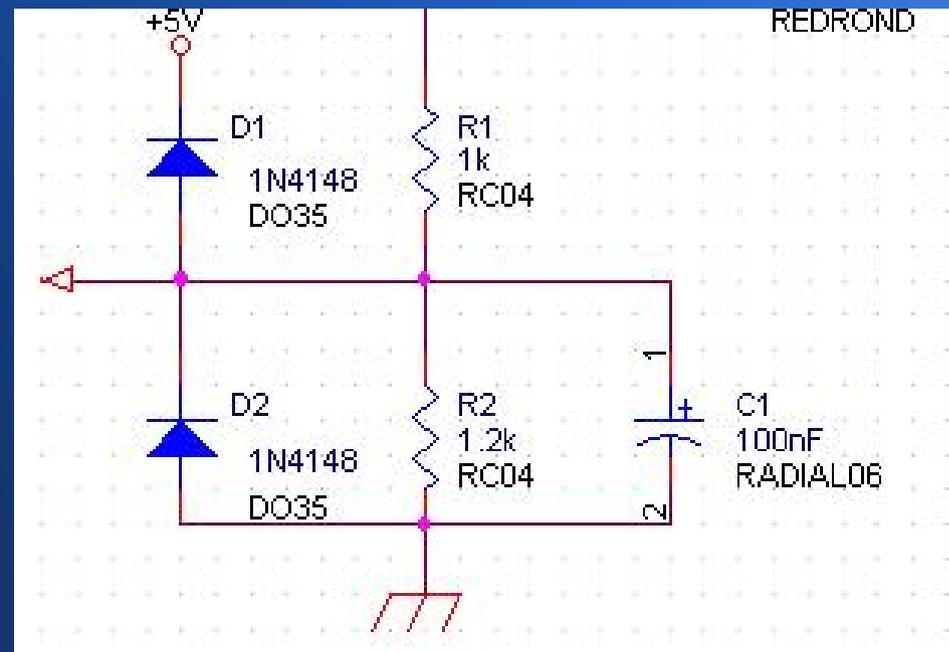
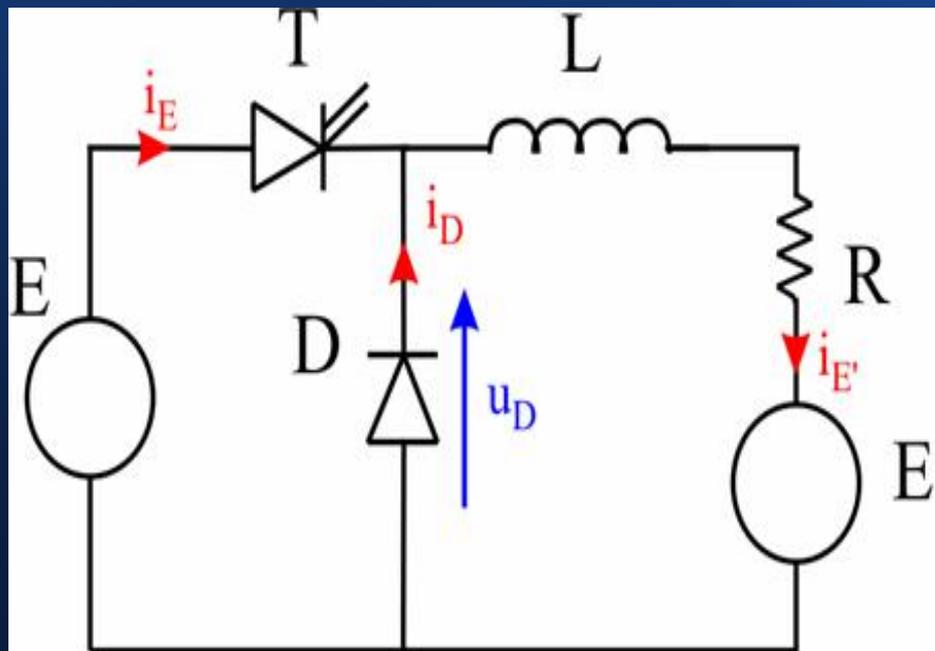
Partie électronique

Définitions des besoins de la carte



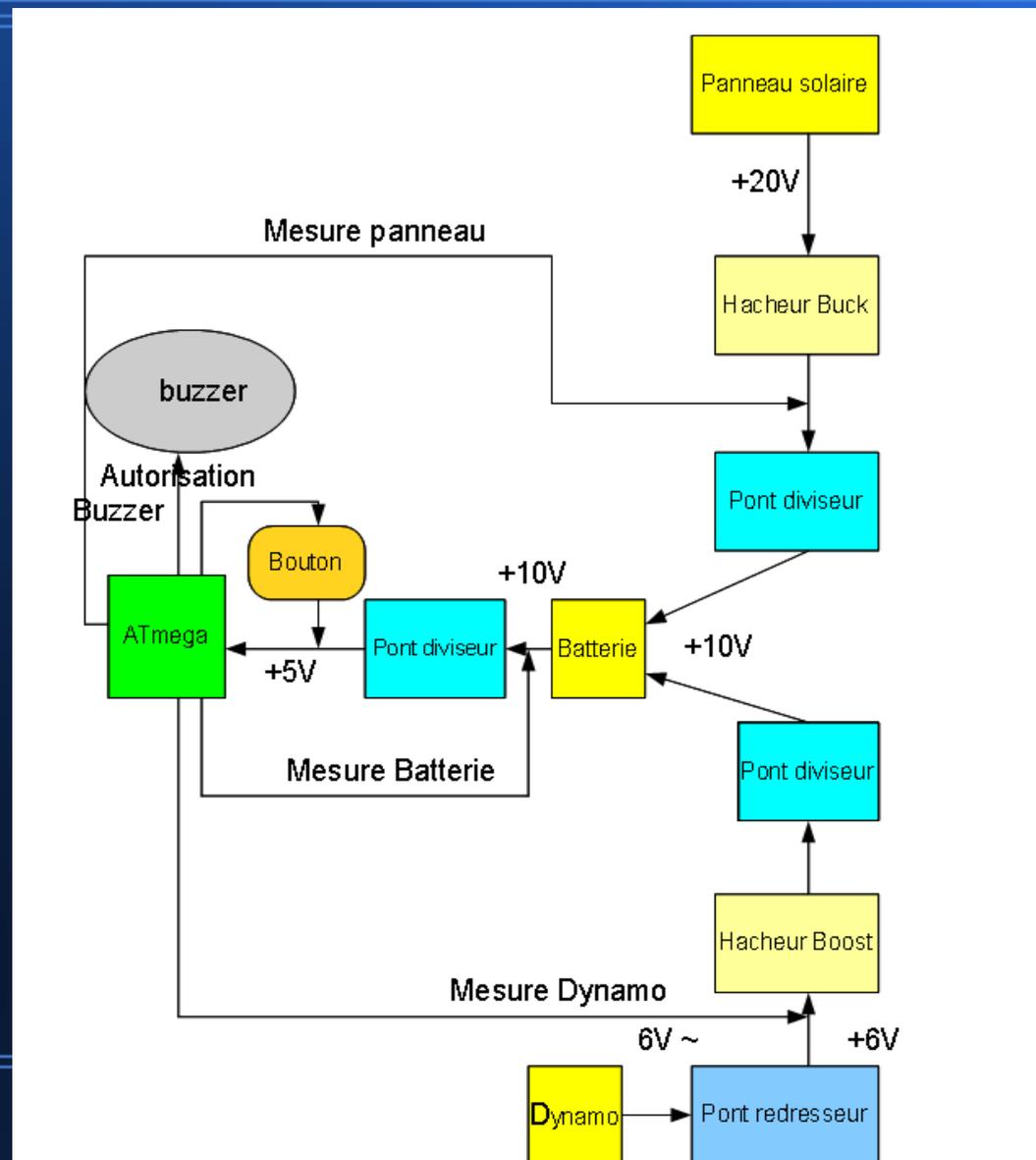
Partie électronique

Définitions des besoins de la carte



Partie électronique

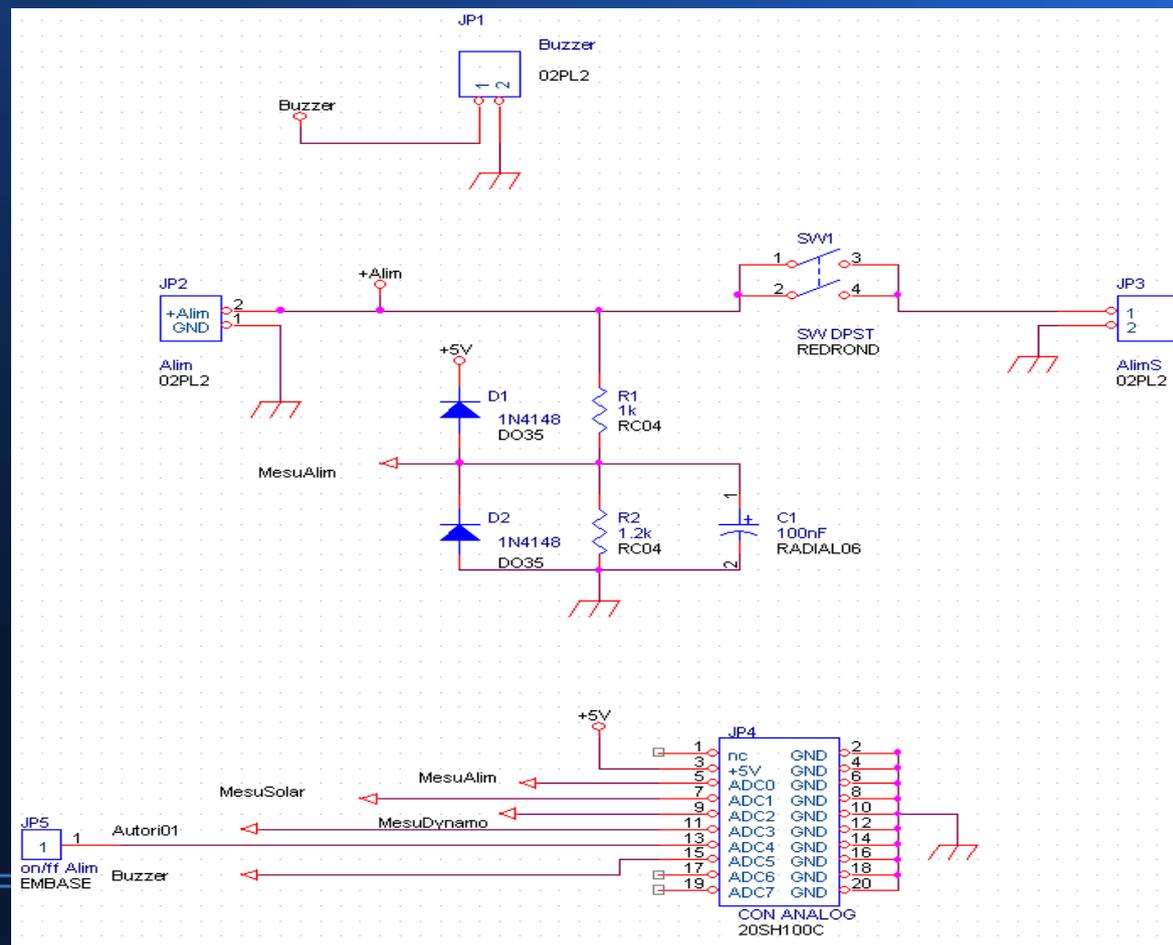
Définitions des besoins de la carte



Partie électronique

réalisation du schéma électrique sous Orcad

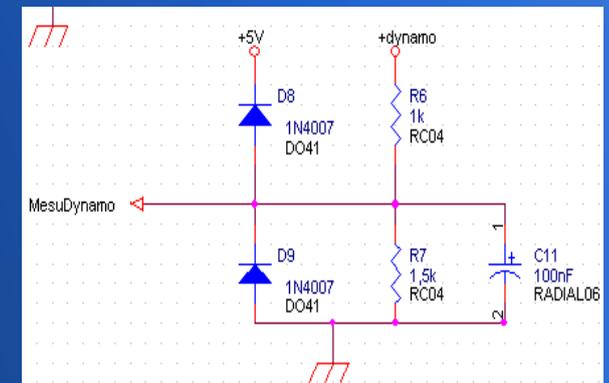
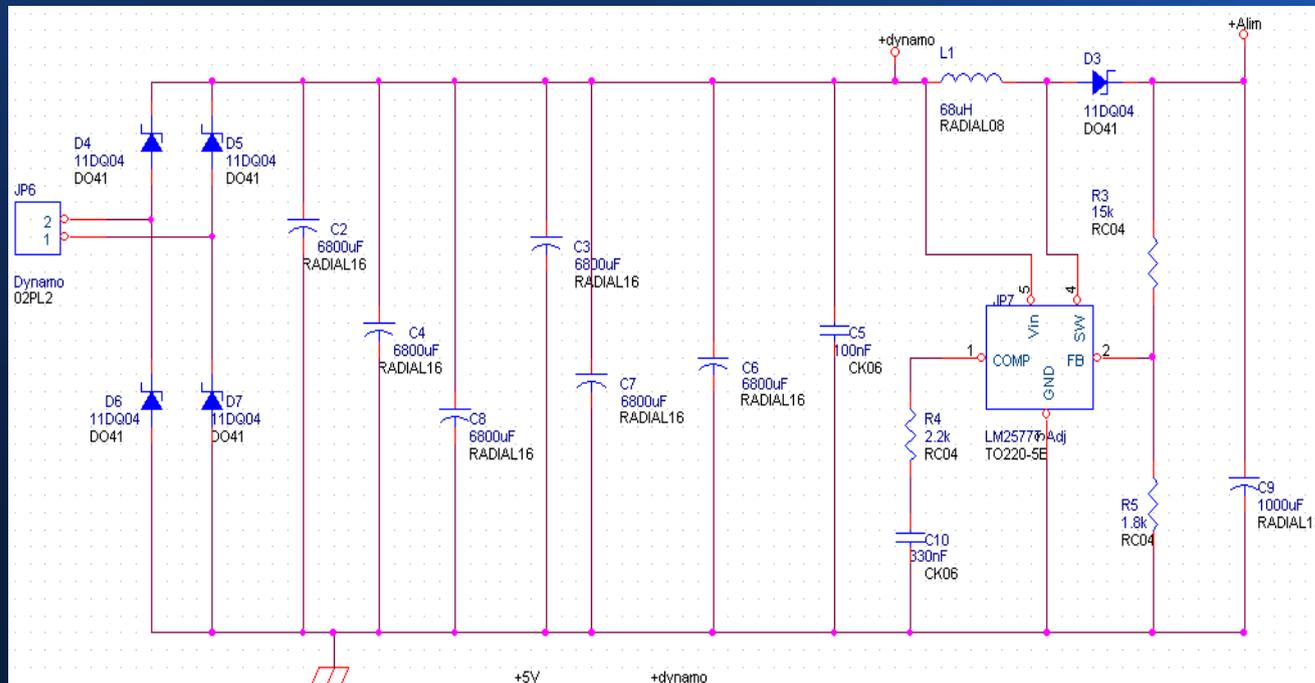
- Alimentation



Partie électronique

réalisation du schéma électrique sous Orcad

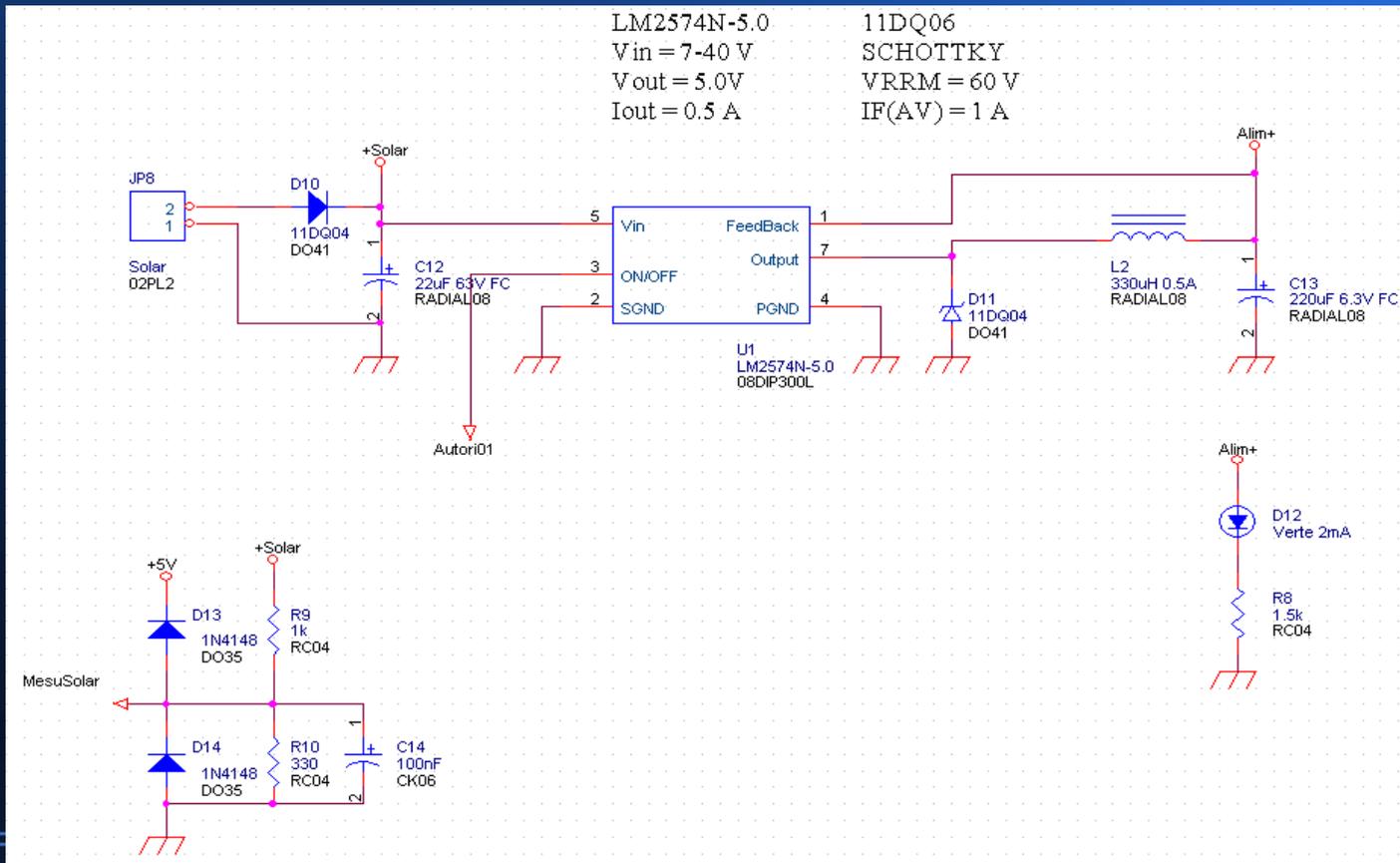
- Dynamo



Partie électronique

réalisation du schéma électrique sous Orcad

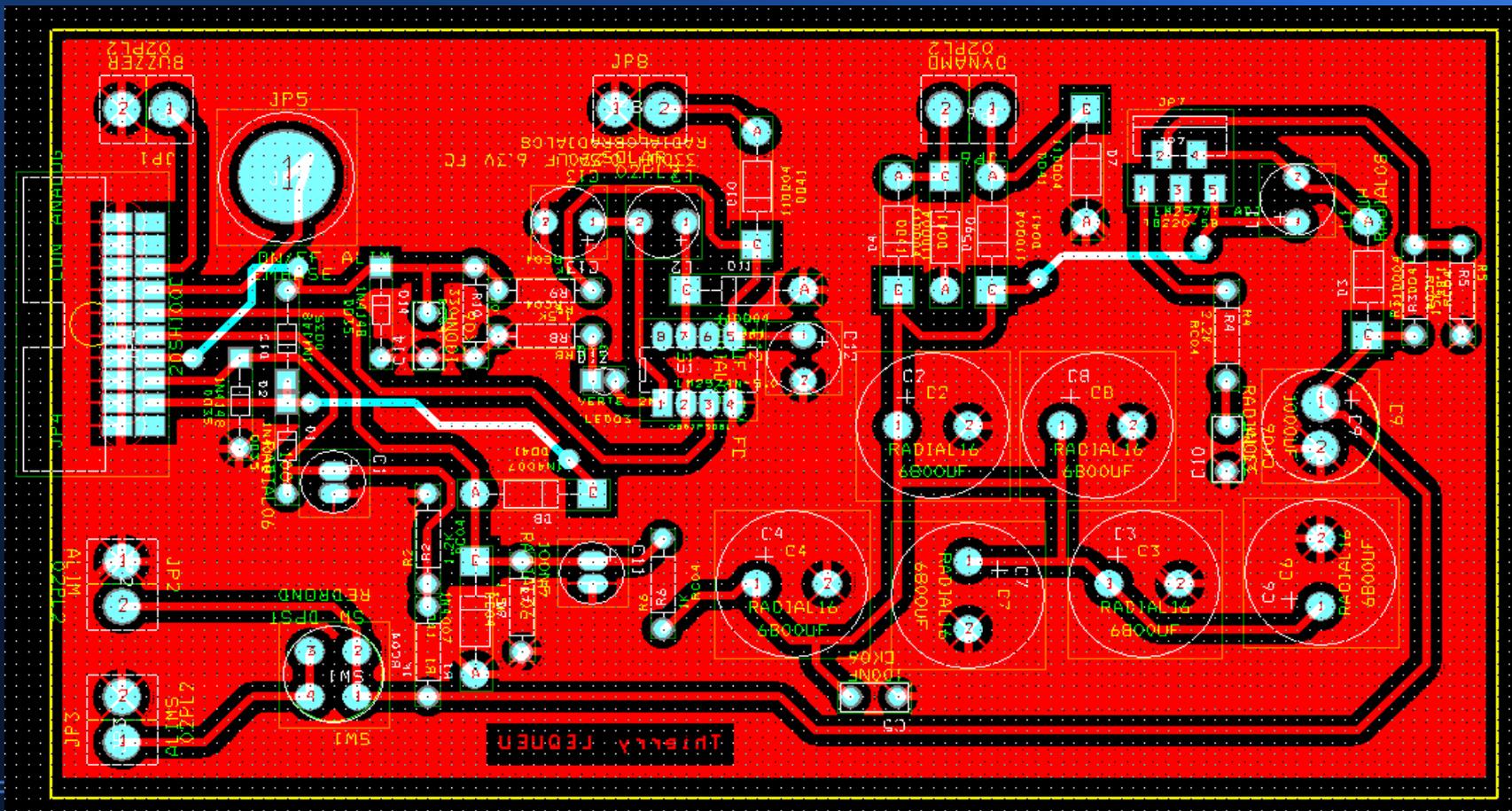
- Panneau solaire



Partie électronique

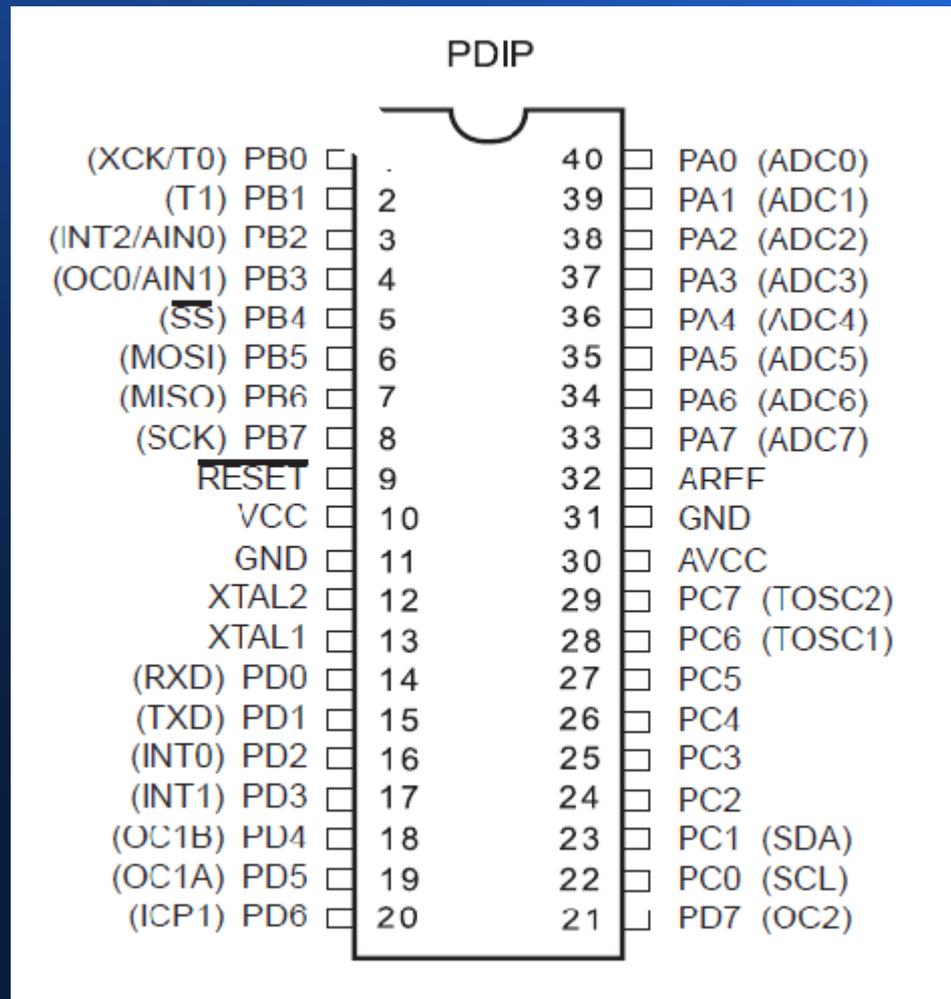
réalisation du schéma électrique sous Orcad

- Typon



Partie programmation

Présentation de l'ATMEGA8535



Partie programmation

Explications du programme

- Les entrées / sorties

```
// Entrées analogiques :
#define ADC_VREF_TYPE 0x00

#define MesBat 0      // Normalement entrée N°0
#define MesSol 1     // Normalement entrée N°1
#define MesDyn 2     // Normalement entrée N°2

// Entrées numériques :
#define BP1      PIND.0

// Sorties numériques :
#define PanToBat PORTA.3
#define Alim     PORTA.4
#define Buzzer   PORTA.5
```

- 3 entrées analogiques
- 1 entrée numérique
- 3 sorties

Partie programmation

Explications du programme

- Au démarrage

```
// ADC initialization
// ADC Clock frequency: 125,000 kHz
// ADC Voltage Reference: AREF pin
// ADC High Speed Mode: Off
// ADC Auto Trigger Source: Free Running
ADMUX=ADC_VREF_TYPE;
ADCSRA=0x87;
SFIOR&=0x0F;

// LCD module initialization
lcd_init(16);
lcd_putsf("Bonjour");
lcd_gotoxy(0,0); //x,y

//Mesurer la tension de la batterie au démarrage
ibat = read_adc(MesBat);
kbat = (float)ibat*2*5/1024;
delay_ms(2000);
```

- Initialisation LCD et ADC
- Mesure de la tension de la batterie
- Affichage sur l'écran LCD

Partie programmation

Explications du programme

- Début de la boucle

```
while (1)
{
//TEST AFFICHAGE INCREMENTATION j
/* sprintf(TensionBat,"Bonjour - j = %d",j);
lcd_gotoxy(0,0); //x,y
lcd_puts(TensionBat); */

ibat = read_adc(MesBat); //mesure de la tension batterie
idyn = read_adc(MesDyn); //mesure tension dynamo
ipan = read_adc(MesSol); //mesure tension panneau solaire
```

- Mesure des tensions analogiques

- Appuie sur BP1

```
//Incrémentatation de j lors d'un appuie sur BP
if(BP1==0)
{
j++; // j = j++ NE FAIT RIEN!!!!!!! ATTENTION!!!!!!
delay_ms(200);
}
if(j>=4)
j=1;
```

- Incrémentatation de « j »

Partie programmation

Explications du programme

- Conversion et affichage

```
//MESURE TENSION BATTERIE
if(j==1)
{
    lcd_clear();
    lcd_gotoxy(0,0);
    lcd_putsf("batterie");

    kbat = (float)ibat*2*5/1024;    //valeur de la tension réelle, calcul forcé en float
    sprintf(TensionBat, "%4.3f", kbat);
    lcd_gotoxy(0,1);
    lcd_puts(TensionBat); //on affiche la valeur de k
    delay_ms(50);
}

//MESURE TENSION PANNEAU
if(j==2)
{
    lcd_clear();
    lcd_gotoxy(0,0);
    lcd_putsf("panneau ");

    kpan = (float)ipan*24/1024;    //valeur de la tension réelle
    sprintf(TensionPan, "%4.3f", kpan);
    lcd_gotoxy(0,1);
    lcd_puts(TensionPan); //on affiche la valeur de k
    delay_ms(50);
}

//MESURE TENSION DYNAMO
if(j==3)
{
    lcd_clear();
    lcd_gotoxy(0,0);
    lcd_putsf("dynamo ");

    kdyn = (float)idyn*2*5/1024;    //valeur de la tension réelle
    sprintf(TensionDyn, "%4.3f", kdyn);
    lcd_gotoxy(0,1);
    lcd_puts(TensionDyn); //on affiche la valeur de k
    delay_ms(50);
}
```

Partie programmation

Explications du programme

- Traitement des données

```
if(kbat<=8.4)
{
  PanToBat = 1;      //recharger batterie
  Buzzer = 1;       //buzzer en marche pendant 2s
  delay_ms(500);
  Buzzer = 0;       // éteindre buzzer
}

else                //sinon ne rien faire
{
  PanToBat = 0;
  Buzzer = 0;
}

if(kbat<=8)
  Alim = 0;
else
  Alim = 1;
```

- Charger la batterie
- Activer le buzzer
- Couper l'alimentation

Conclusion

- Sujet complet
- Complexité
- Association électronique analogique et numérique
- Organisation