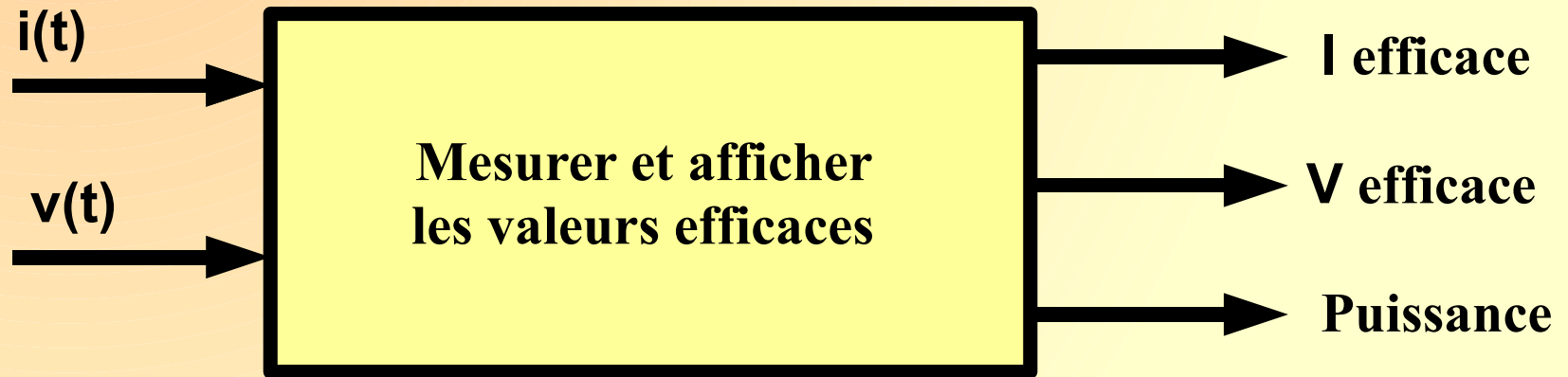


# Wattmètre secteur 220V/ 16A



# Objectif du projet



# Plan

**Cahier des charges**

**Atmega8535**

**Description de la démarche suivie**

**Programmation**

**Capteur de courant**

# 1.Cahier des charges

**Les contraintes :**

**Microprocesseur Atmega8535**

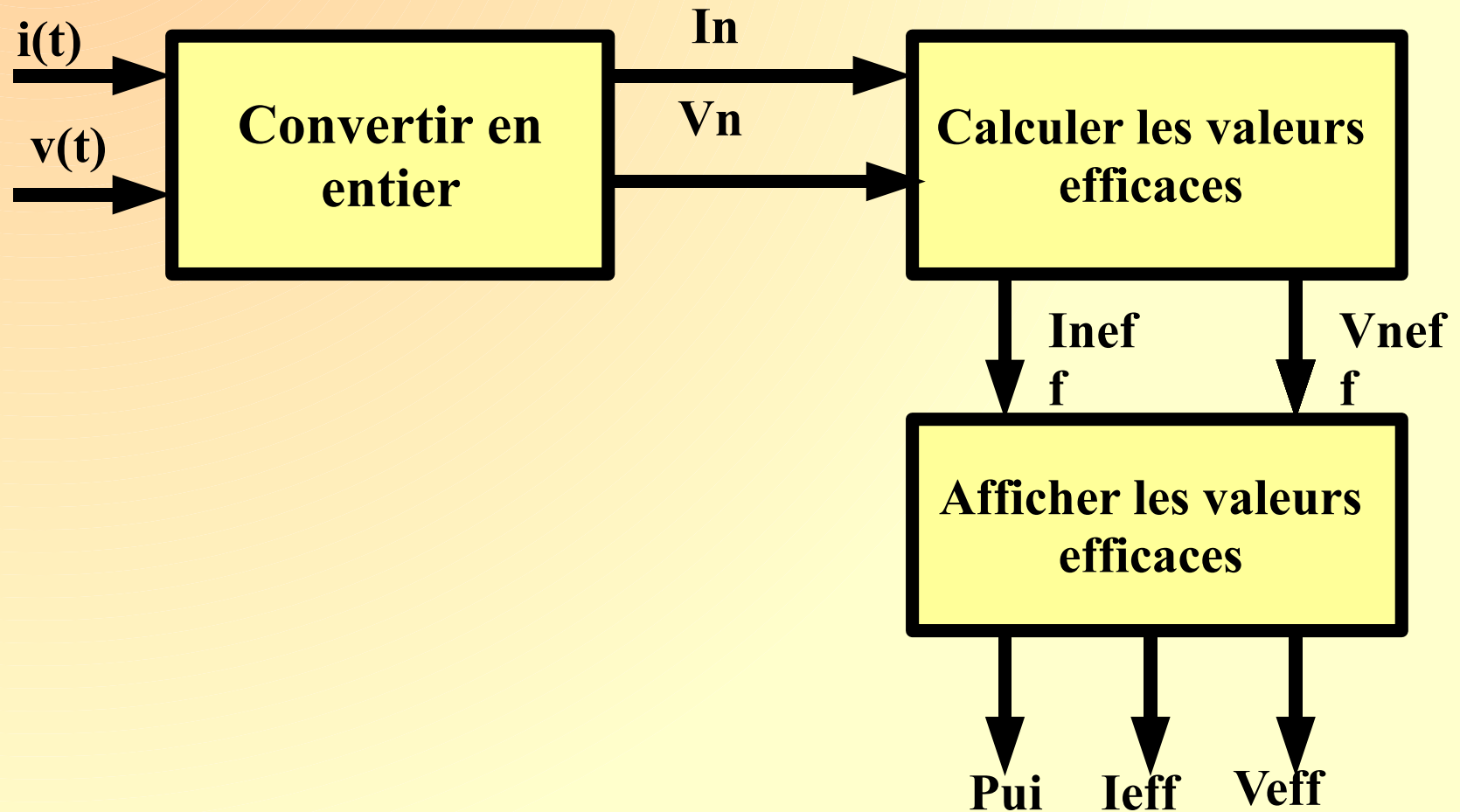
**Afficheur LCD 16\*4 Farnell 944-9019**

**Alimentation 0 à 5V**

**Logiciel CodeVisionAVR**

# 1. Cahier des charges

Schéma fonctionnel de niveau 2 :



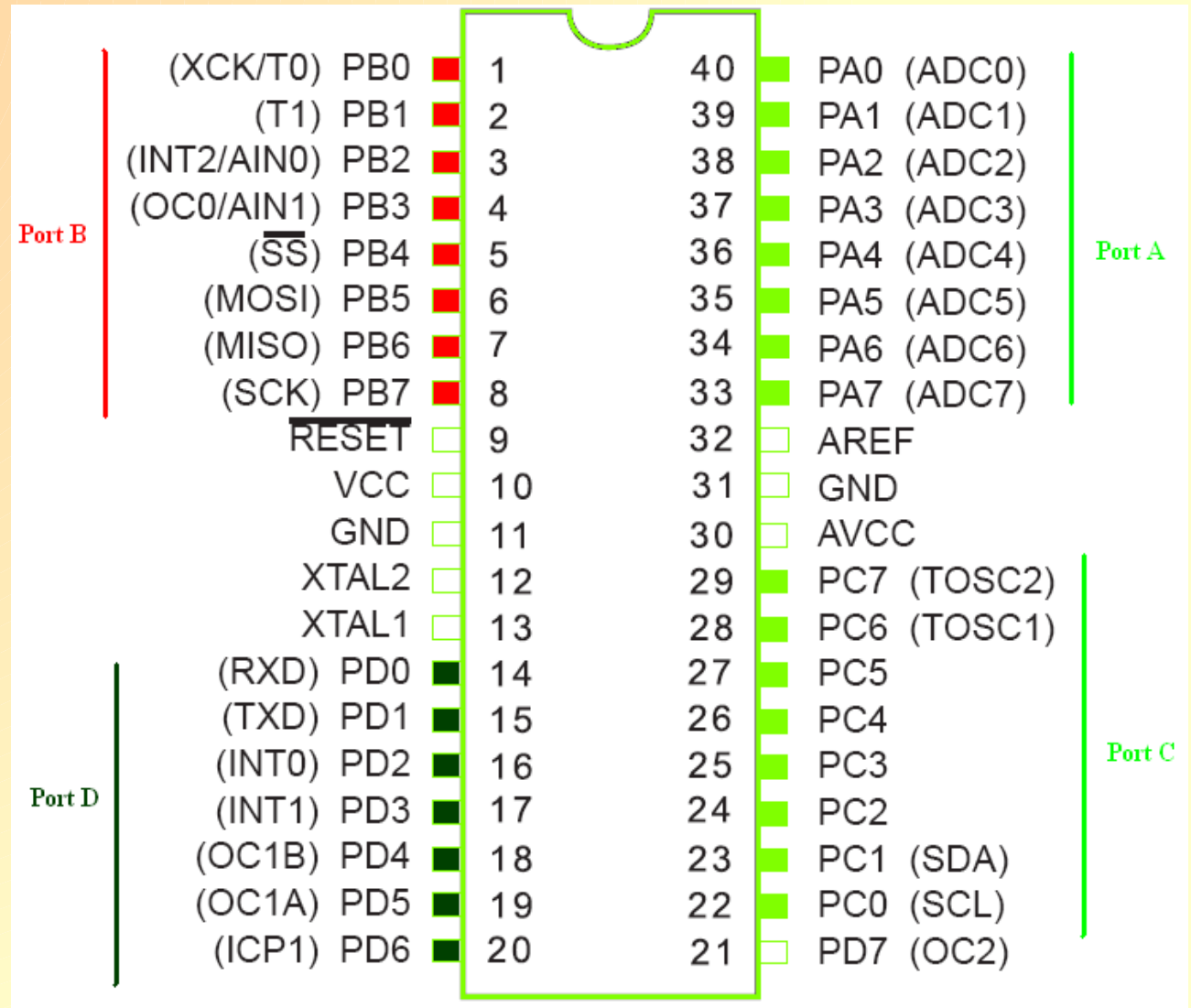
# 2. Atmega 8535

**Port A** : l'entrée du signal alternatif pour le convertir avec un CAN.

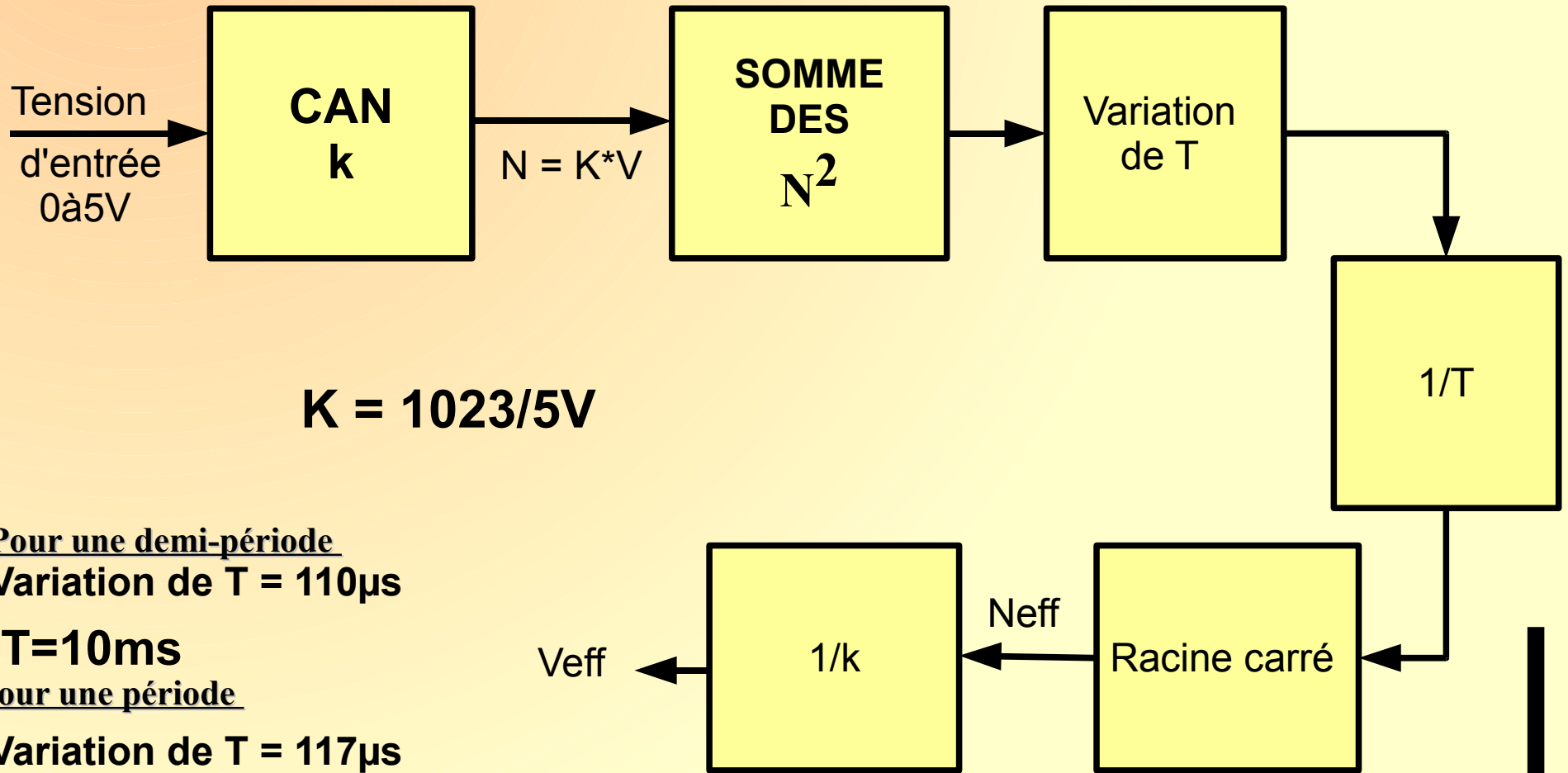
**Port B** : pour la programmation

**Port C** : connecter à l'afficheur LCD..

**Port D** : connecter au bornier de programmation du microprocesseur.



# 3. Description de la démarche suivie



$$K = 1023/5V$$

Pour une demi-période  
Variation de T = 110µs

**T=10ms**

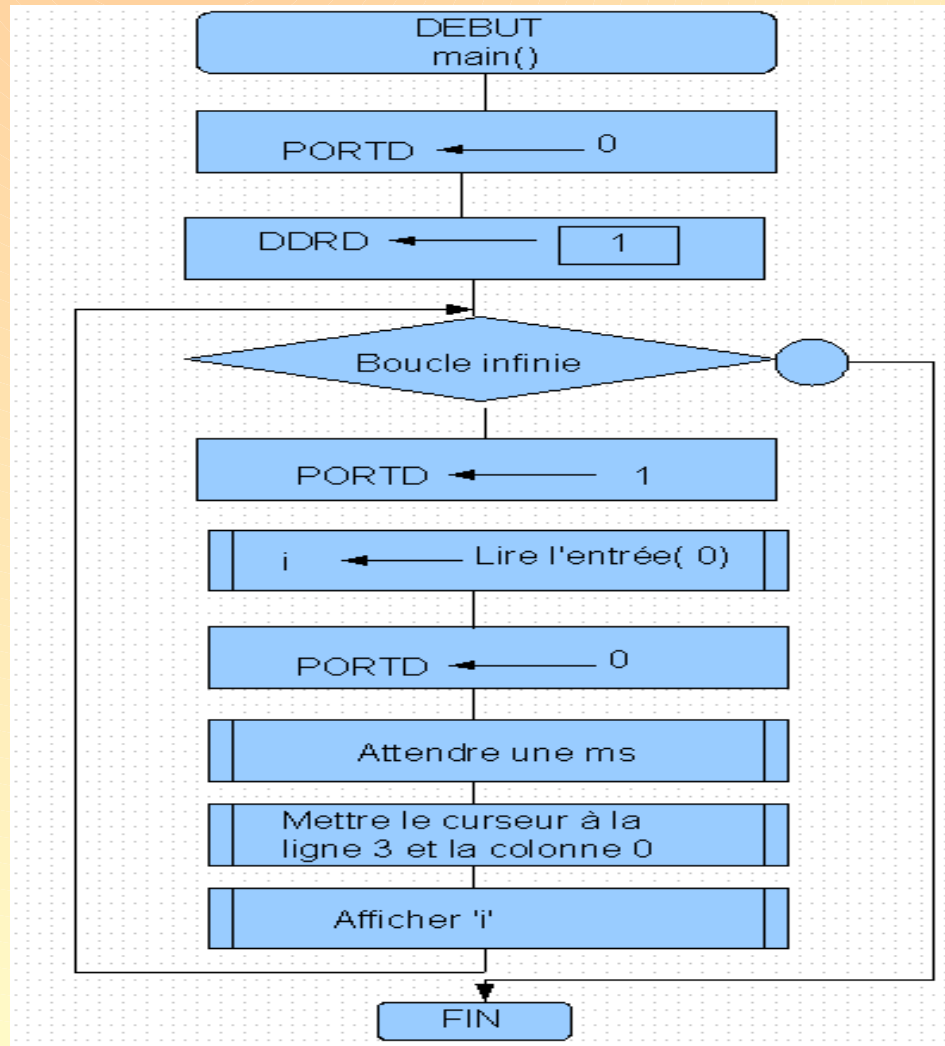
Pour une période

Variation de T = 117µs

**T=20ms**

# 4. Programmation

Calcul de temps de conversion



Organigramme

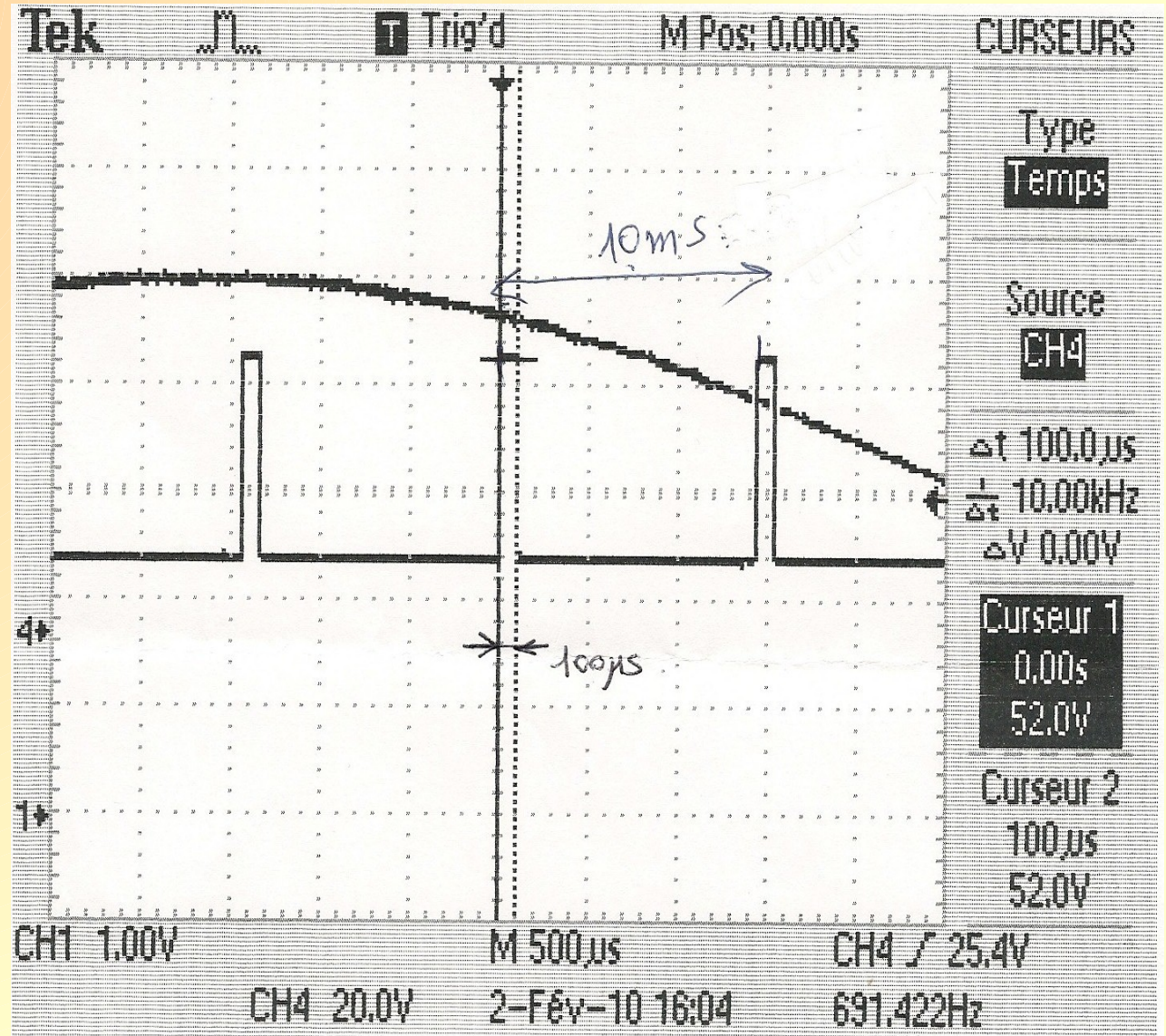
Le programme pour faire le calcul de temps de conversion

```
PORTD=0x00;
DDRD=0xFF; //port D en sortie.
while (1)
{
    PORTD=0xFF;
    i=read_adc(0);
    PORTD=0x00;
    delay_ms(1);
    sprintf(tampon,"%4d",i);
    lcd_gotoxy(3,0);
    lcd_puts(tampon);
};
}
```



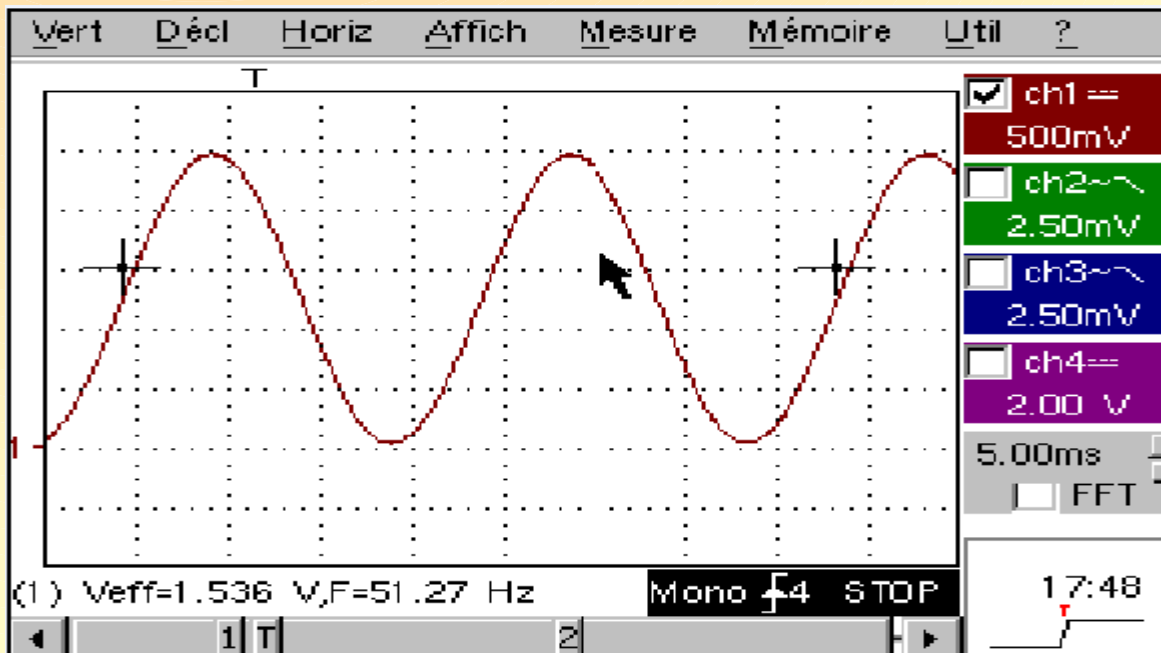
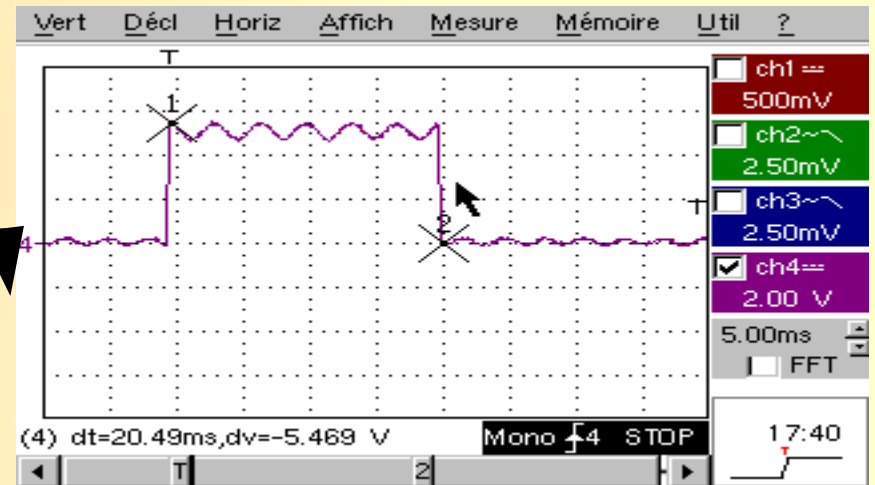
# 4. Programmation

T : demi-période = 10ms



# 4. Programmation

Période du signal

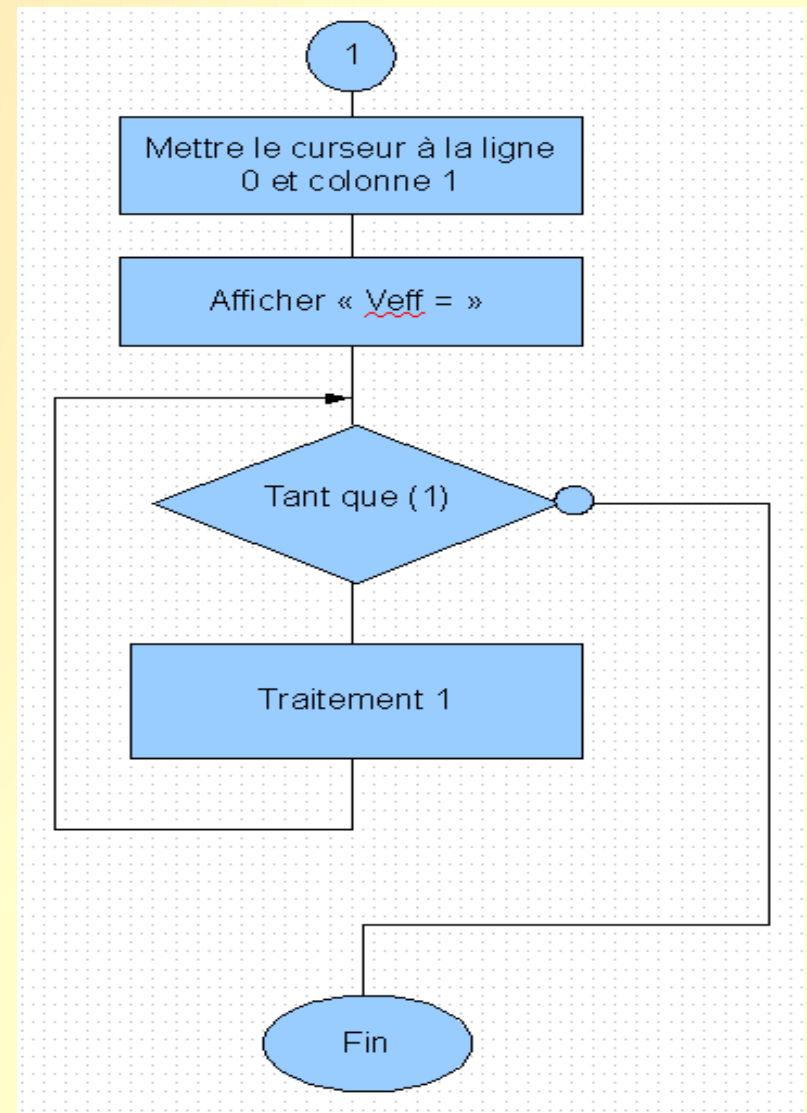
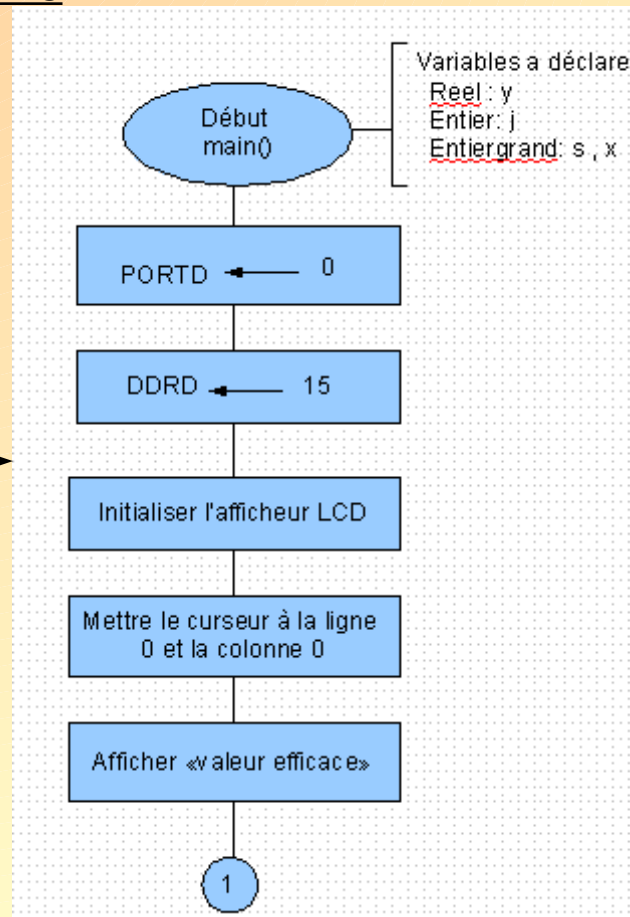


Signal d'entrée de fréquence  $F = 50\text{Hz}$

# 4. Programmation

La fonction principale

**organigramme**



# 4. Programmation

Déclaration des variables  
le port D en sortie



```
void main(void)
{
// Declare your local variables here
long int x ;
float y;
int j;
long int S ;

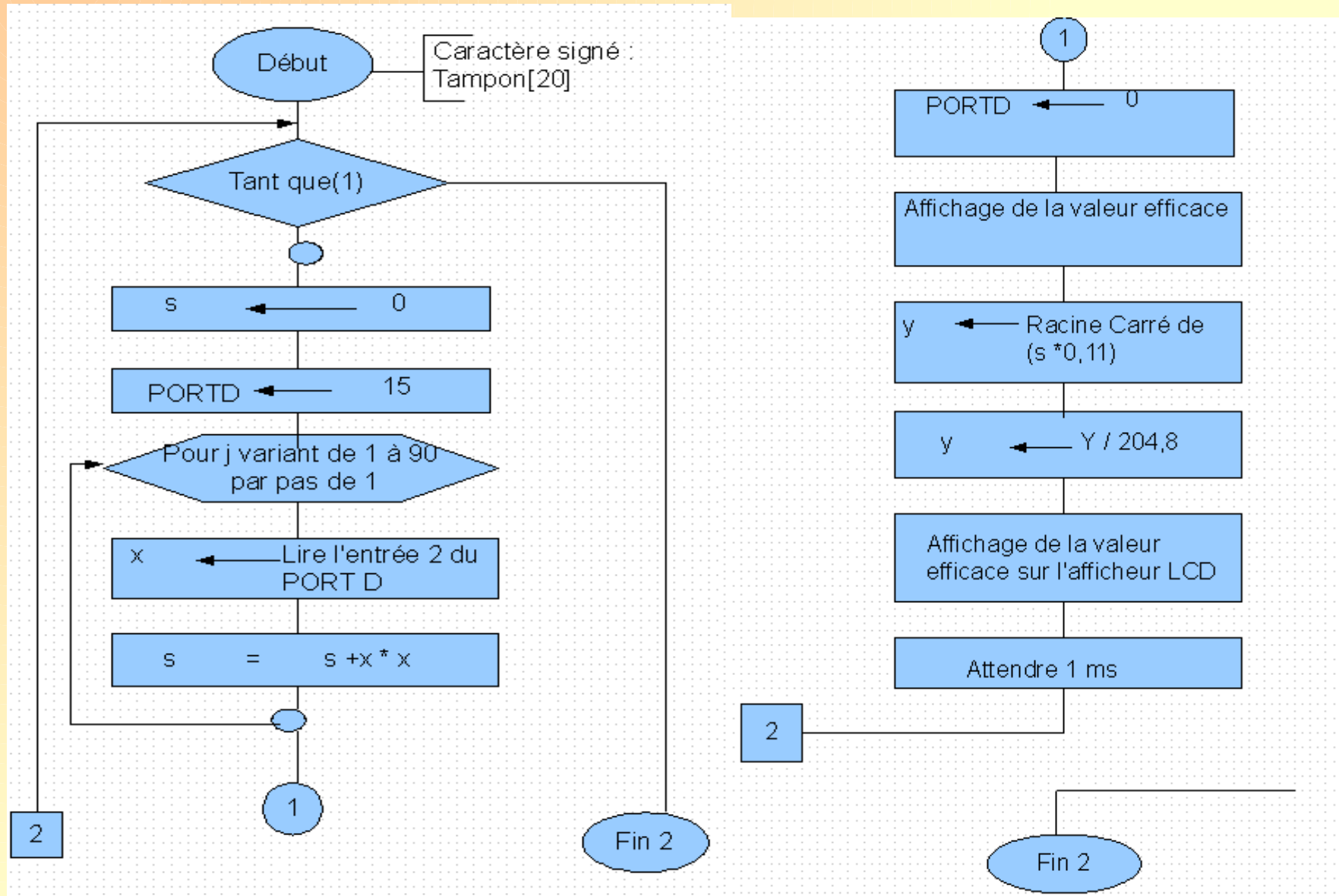
PORTD=0x00;
DDRD=0xFF; //port D en sortie .
```

Écrire sur l'afficheur  
LCD



```
// LCD module initialization
lcd_init(16);
lcd_gotoxy(0,0);
lcd_putsf("valeur efficace"); //Pour écrire sur l'afficheur LCD
lcd_gotoxy(0,1);
lcd_putsf("Veff = ");
```

# Calcul de la valeur efficace



# 4. Programmation

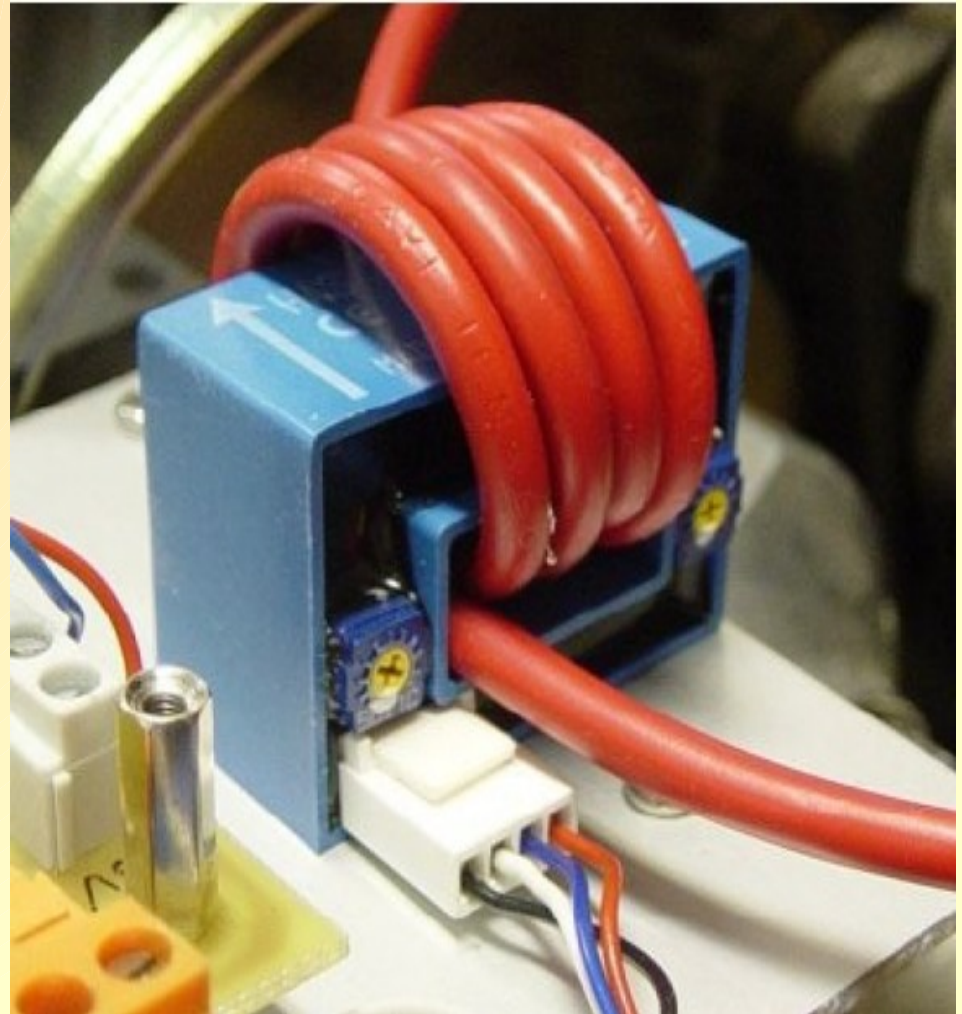
```
while (1)
{
    unsigned char tampon[20];
    S=0;
    PORTD=0xFF; // calculer le temps de conversion et le temps d'affichage en mettant la sortie DO à 1
    for (j=1; j<=90
    {
        x=read_adc(2); // converti l'entrée analogique en valeur numérique et le met à l'intérieur de tableau.
        // 2 est le numéro de l'entrée de la carte d'interface analogique
        S=S+x*x; // l'opération qui permet le calcul des valeurs moyennes
    }
    PORTD=0x00; // un créneau qui nous permet de calculer le temps de conversion
    sprintf(tampon, "%4d", x); // afficher la valeur efficace
    lcd_gotoxy(7,2);
    lcd_puts(tampon); // afficher la valeur sur l'afficheur

    sprintf(tampon, "%6ld", S); // Pour afficher en long (int) il faut le "l" à ne pas oublier
    lcd_gotoxy(0,3); // on se place à la 3ième ligne et à la 0 colonne.
    lcd_puts(tampon);

    y= sqrt(S*0.011) //110µs/10ms=0.011
    y = y/204.8; // pour afficher la valeur en Volts
    sprintf(tampon, "%.2f V", y);
    lcd_gotoxy(7,1);
    lcd_puts(tampon);
    delay_ms(1000);
};
}
```

# 5. Capteur de courant

Expérience réalisée :



# 5. Capteur de courant





# 5. Capteur de courant



# Planning prévisionnel et réel

Tâches \ Semaines	3	4	5	6	7	8	9	10	11	12	13
Le choix du projet à étudier	Red										
L'étude et l'analyse du sujet		Red	Red								
Rédaction et modification du programme						Red	Red				
Affichage de la valeur efficace de la tension							Red				
Test et validation								Red			
Affichage des puissances									Red	Red	
l'oral											Red
Rédaction du rapport	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red	Red
			Red								
			Green								

planning prévisionnel

planning réel