

# Télémètre ultrasons

Robert Mugabuhamyé  
Damien Rodrigues

Thierry Lequeu  
Sophie Laurenceau

# Sommaire



1. Présentation du projet
2. Partie électronique
3. Partie informatique

# 1. présentation






















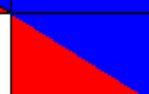
# Présentation du projet

**Solution retenue : les ultrasons (technique du sonar)**

**Cahier des charges :**

- rapport qualité/prix
- alimentation imposée
- 9 semaines
- précision allant de 0 à 10m
- adaptabilité à d'autres systèmes

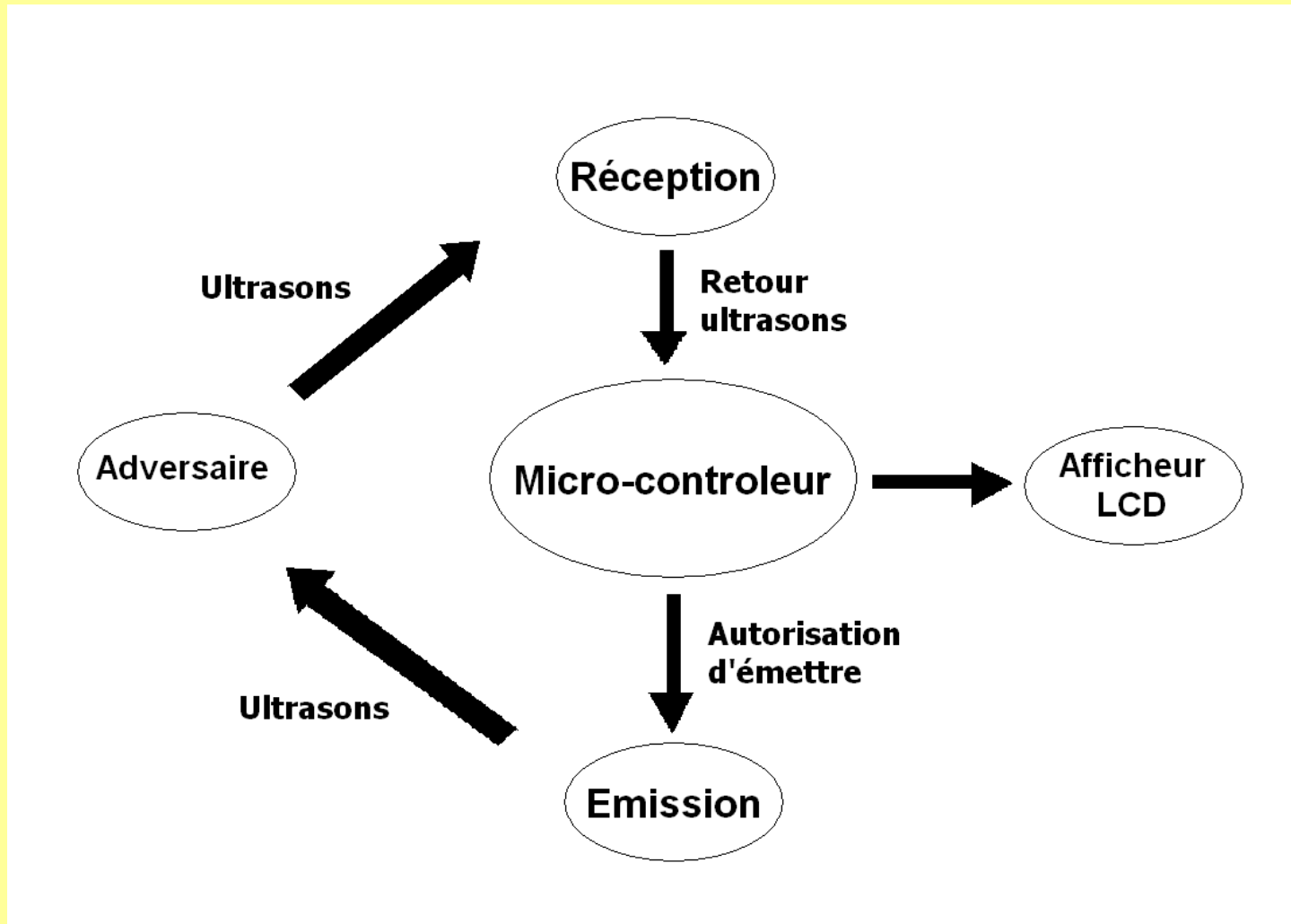
# Planning

	1	2	3	4	5	6	7	8	9	10
Etude du cahier des charges										
Etude du montage										
Conception de la carte										
Tests et mesures sur la carte										
Conception du programme										
Compréhension du logiciel de programmation										
Rédaction du rapport										
Préparation à l'oral										

 Prévisionnel

 Réel

# Diagramme sagittal



## 2. partie électronique

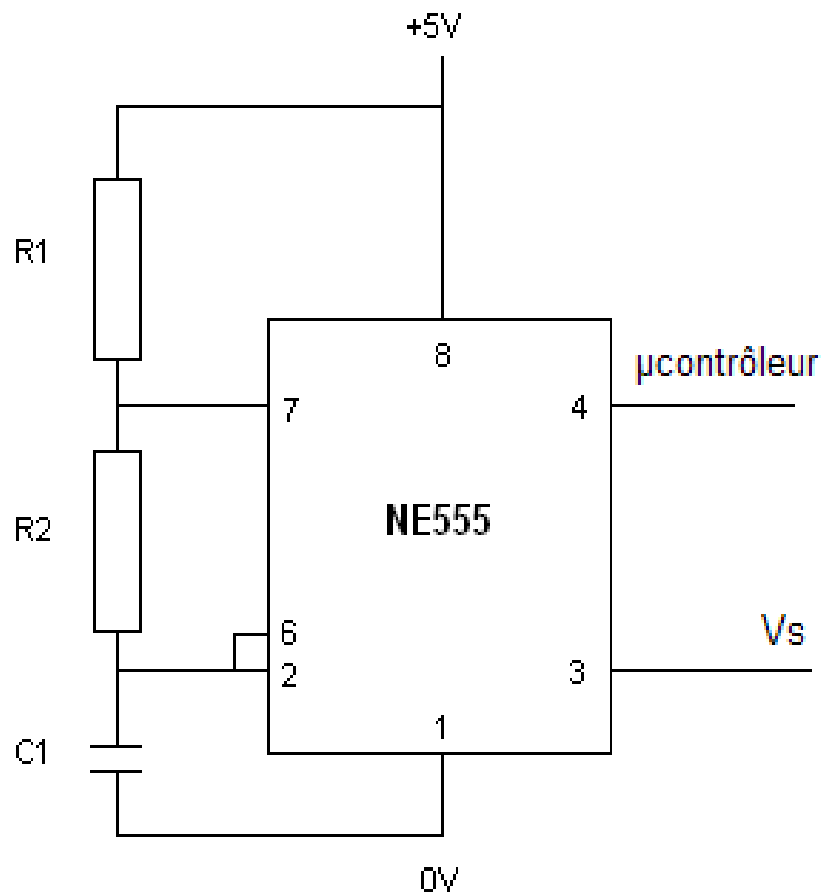
# Electronique

**La partie électronique se décompose en 5 étapes**

- 1. Emission d'un signal ultrasons de 40kHz**
- 2. Réception du signal de retour**
- 3. Amplification du signal**
- 4. Filtrage**
- 5. Création d'un signal carré**



# Emission du signal



La fréquence optimale pour utiliser les ultrasons est de 40kHz.

Il faut réaliser un signal carré ayant cette fréquence.

Nous utilisons un NE 555 monté en astable pour le créer.

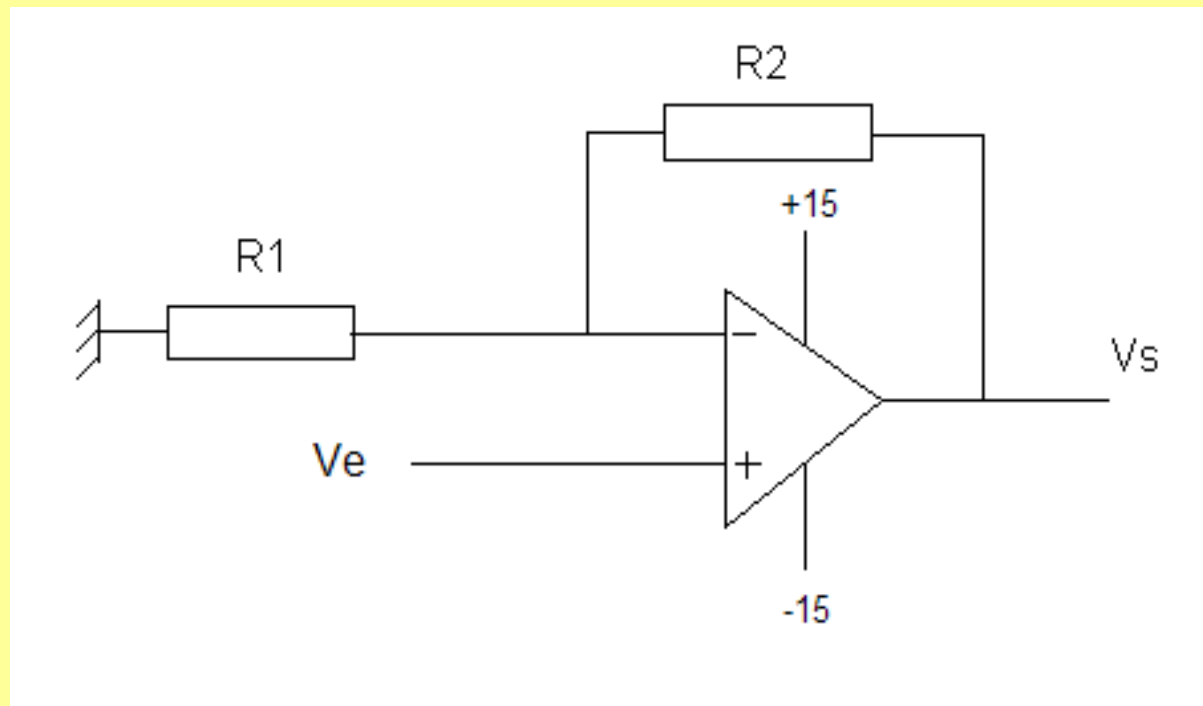
C'est la patte 4 qui sera reliée au micro-contrôleur (reset)

# Réception du signal

**Ce fut la partie la plus simple, il fallait simplement connecter le récepteur à la carte.**

**Malheureusement deux problèmes se posent :**  
**Amplitude de quelques mV**  
**Signal parasite**

# Amplification

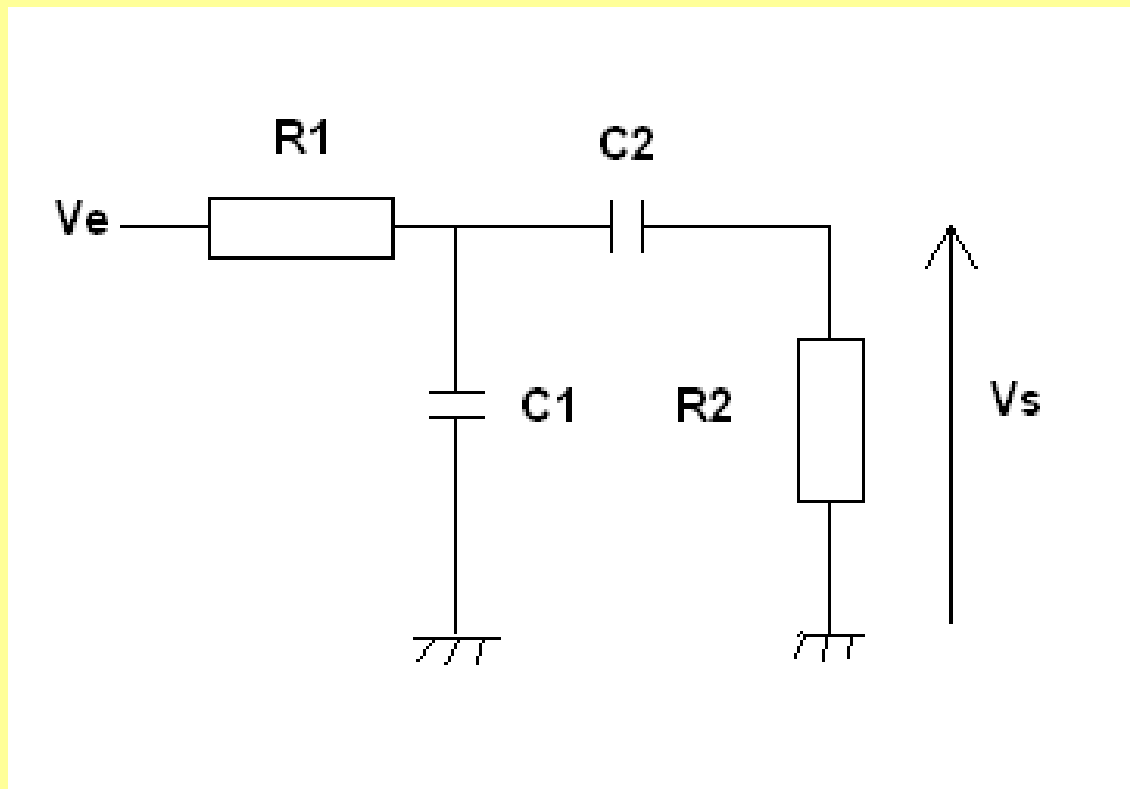


**Vu la petitesse du signal il est nécessaire de l'amplifier, cela est dû à la distance.**

**Mise en cascade de deux AOP non-inverseurs.**

# Filtrage

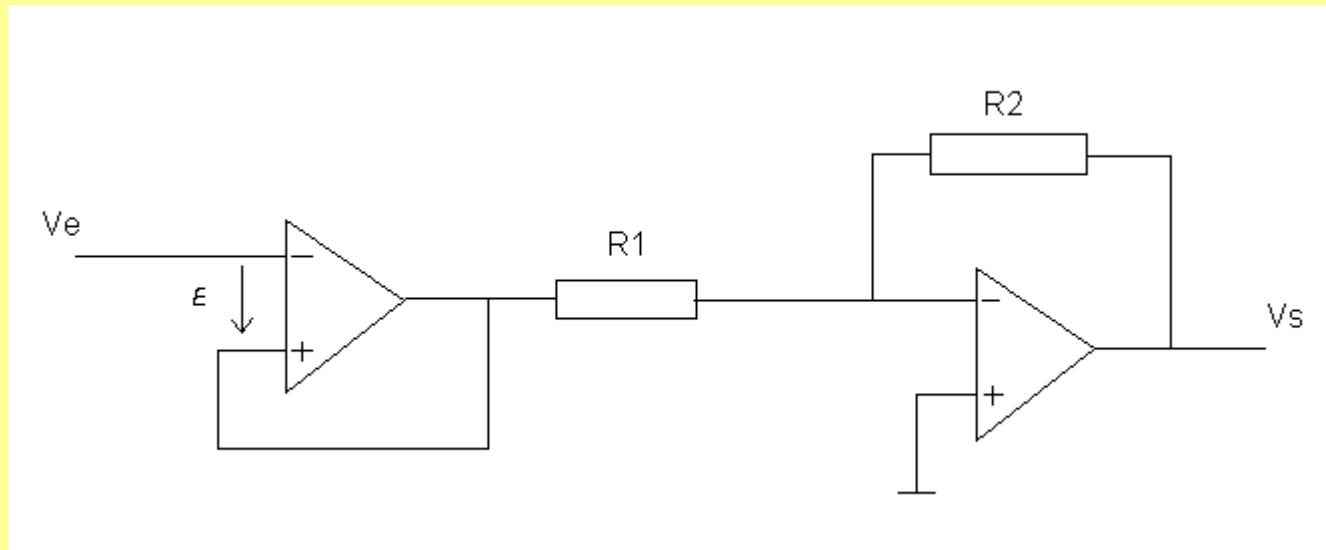
Mise en cascade d'un filtre passe-bas avec un filtre passe-haut.  
Le filtre va supprimer les fréquences inférieures à 39kHz et  
supérieures à 42kHz.



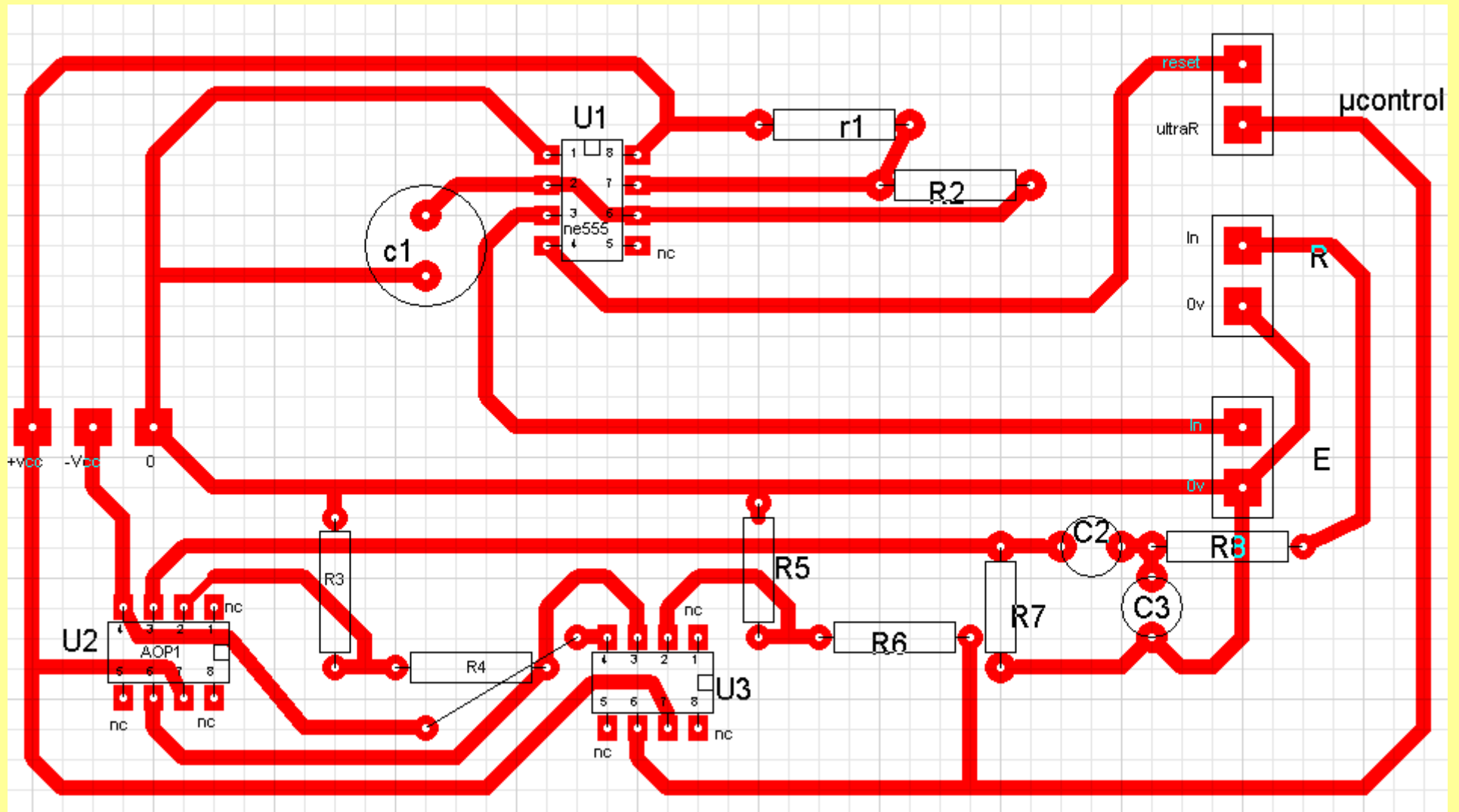
# Signal carré

Cette partie n'a pas été entièrement réalisée car nous nous sommes aperçus à la dernière minute.

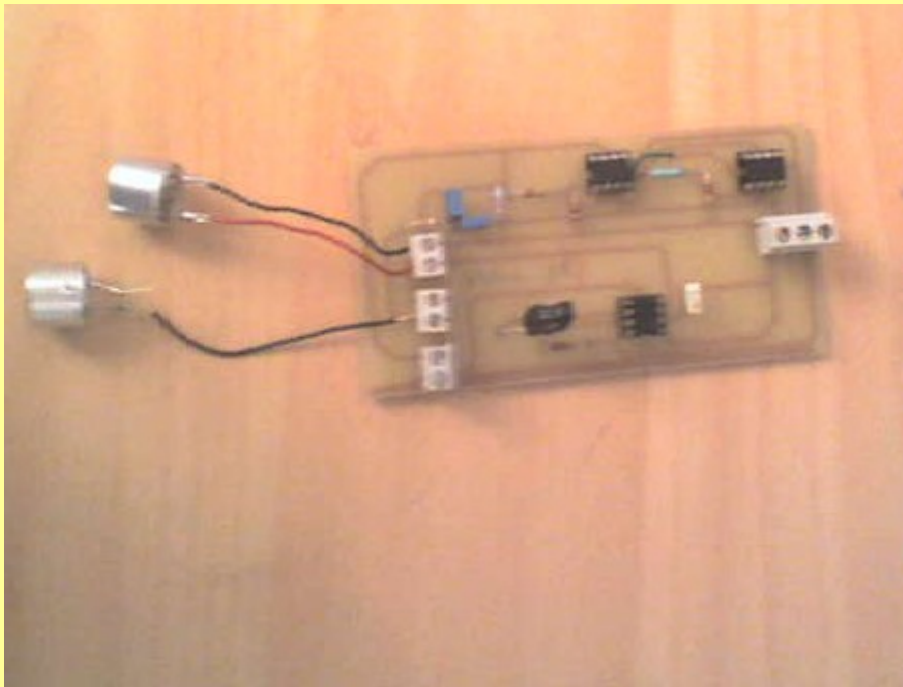
Nous n'avons donc pas eu le temps de l'intégrer au montage.



# Турон

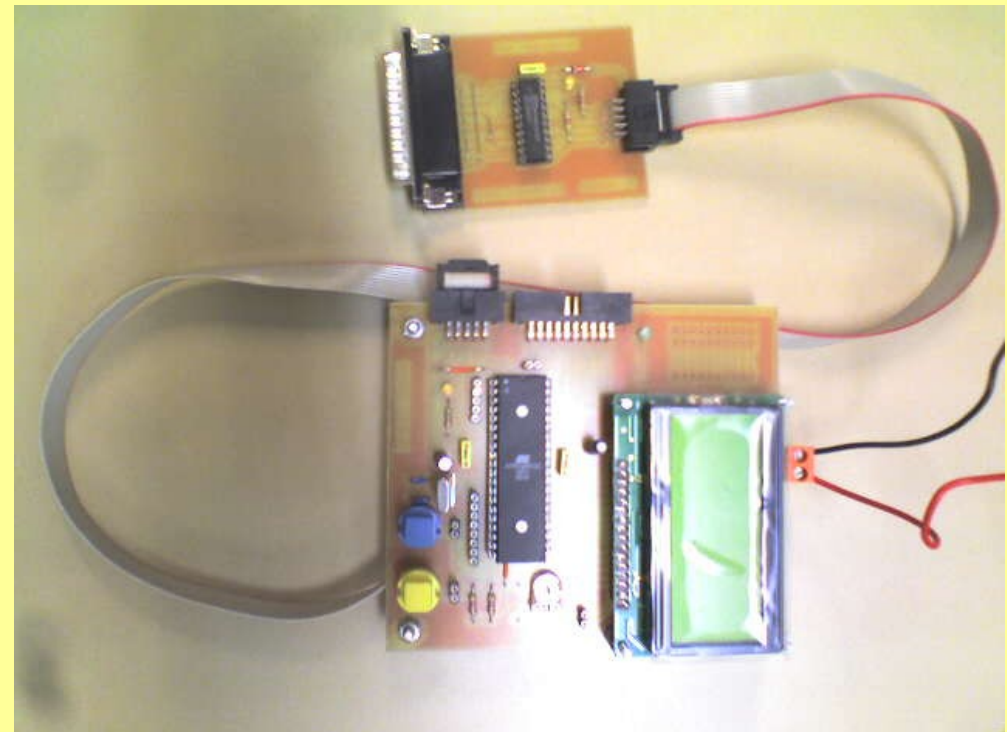


# Cartes



**Carte émission/réception**

**Carte de développement**



# 3. partie informatique



# Informatique

**Programme réalisé avec Code Vision AVR**

**Langage C**

**Programmation d'un micro-côntroleur ATmega8535**

# Programme

Déclaration des bibliothèques

Interruptions

Gestion de l'affichage sur l'écran LCD

# Librairies

Trois librairies nous ont été nécessaires pour ce programme

```
#include <mega8535.h>           //permet d'utiliser les fonctions du  $\mu$ contrôleur  
#include <delay.h>           //permet d'utiliser les fonctions liées au temps  
#include <stdio.h>           //permet d'utiliser les fonctions liées au chaînes  
                               de caractères
```

# Interruptions

**Interruption interne :**

```
interrupt [TIM1_COMPA] void timer1_compa_isr(void);    //déclaration de fonction
```

```
interrupt [TIM1_COMPA] void timer1_compa_isr(void)  
{  
    cpt=cpt+1;    //incréméntation du compteur  
}
```

**Le but de ces lignes de codes sont de réaliser l'incréméntation du compteur toutes les millisecondes.**

# Interruptions

**Interruption externe :**

```
interrupt [EXT_INT0] void ext_int0_isr(void); //déclaration de fonction
```

```
interrupt [EXT_INT0] void ext_int0_isr(void)  
{  
    temps=cpt/1000; // conversion en secondes  
    d=340*temps; // calcul de la distance  
    cpt=0; // raz du compteur  
}
```

**Grâce à cette interruption nous allons pouvoir mémoriser la valeur du compteur dans la variable “temps”, puis calculer la distance avec le karting qui suit.**

# Affichage sur l'écran

```
lcd_init(16);  
lcd_clear();  
while (1)  
{  
    //affichage du temps  
    lcd_gotoxy(0,0);  
    sprintf(tampon,"temps =%5d s",temps);  
    lcd_puts(tampon);  
    //affichage de la distance  
    lcd_gotoxy(0,1);  
    sprintf(tampon1,"distance= %3d m",d);  
    lcd_puts(tampon1);  
    delay_us(1000);  
    lcd_clear();  
};
```

Cette partie réalise l'affichage des différentes informations sur l'écran lcd. L'affichage est permanent grace au « while(1) ».

# Remerciements

Thierry Lequeu

Paul Lerebourg

Florent Prou