

Université François-Rabelais de Tours

Institut Universitaire de Technologie de Tours

Département Génie Électrique et Informatique Industrielle



Robert MUGABUHAMYE
Damien RODRIGUES
2ème année – S2
Promotion 2005-2007

Enseignants
M. Thierry LEQUEU
Mme Sophie LAURENCEAU

Université François-Rabelais de Tours

Institut Universitaire de Technologie de Tours

Département Génie Électrique et Informatique Industrielle

Télémètre ultrasons

Robert MUGABUHAMYE
Damien RODRIGUES
2ème année – S2
Promotion 2005-2007

Enseignants
M. Thierry LEQUEU
Mme Sophie LAURENCEAU

Table des matières

Introduction.....	4
I.Présentation du projet.....	5
I.1.Planning et organisation du travail.....	6
I.2.Diagramme sagital.....	7
I.3.Cahier des Charges.....	8
II.Partie électronique.....	8
II.1.Émission du signal.....	8
II.2.Réception du signal.....	9
II.3.Amplification du signal.....	10
II.4.Le filtrage.....	11
II.5.Création d'un signal carré +5/-5V.....	12
II.6.Conception de la carte.....	13
II.7.Tests.....	14
II.8.Mode d'emploi.....	15
III.Partie informatique :.....	16
III.1.Le Programme :.....	16
III.2.Analyse du programme :.....	21
III.2.1.Interruption interne :.....	22
III.2.2.Interruption externe :.....	23
III.2.3.Affichage sur l'écran LCD :.....	23
IV.Problèmes rencontrés.....	25

Introduction

Depuis que nous avons intégré l' IUT, nous avons des projets à réaliser dans différentes matières, dans différents domaines .Ceci dans le but de nous familiariser avec les technologies qui sont utilisées dans le monde de l'industrie et d'acquérir une certaine expérience dans la pratique.

Lors du troisième semestre de notre formation , nous avons pour objectif de réaliser un projet touchant principalement à l'électronique .De ce fait , les projets du quatrième semestre devaient traiter d'un autre domaine en l'occurrence l'informatique .

En effet , il nous a été demandé par le professeur responsable du cours de Travaux de Réalisations (M Thierry LEQEU) de trouver des sujets ce rapportant principalement à l'informatique .Ce dernier étant également responsable du projet concernant le KART de l'IUT, il nous a plus ou moins inciter à travailler sur des projets pouvant profiter à l'amélioration du KART. C'est ainsi que nous avons plusieurs groupes qui traitent de sujet s'y rapportant.

Dans la suite du dossier , nous allons tenter de vous exposé le plus clairement possible le projet que nous avons réaliser tant bien que mal : *le télémètre arrière* .

I. Présentation du projet

Lors d'une course de KARTING ou de toutes autres sport, il est important de savoir à quelle distance se trouvent les concurrents qui sont devant nous mais également ceux qui sont derrière.

La première idée serait de mettre tout simplement un rétroviseur .Mais le KART étant présenté dans différents concours , le coté technologique et le coté esthétique sont important .Nous avons donc penser réaliser un dispositif discret et innovant d'où l'idée du télémètre arrière . Ce dernier permettra au conducteur de savoir à quelle distance ce trouve le poursuivant .

Parmi les méthodes qui nous paraissaient envisageable , celle de l'utilisation des ultrasons nous semblait la plus facile par sa mise en application et par sa fiabilité.

Les ultrasons sont des vibrations mécaniques, de même nature que le son, mais inaudibles à l'oreille humaine car leur fréquence est supérieure à 20kHz. De nos jours, les ultrasons sont fréquemment utilisés, l'exemple le plus connu est celui du sonar. En effet, un appareil génère des impulsions ultra-sonores et reçoit les ondes réfléchies par les obstacles qui sont ainsi détectés.

Les ultrasons sont également utilisés dans des domaines comme la médecine (traitement des névralgies, déterminations de lésions, recherche d'anomalies dans la boîte crânienne, etc...), la métallurgie (dégazage des métaux) ou la sono-chimie.

Nous allons utiliser le même principe que le sonar sauf qu'au lieu de simplement détecter l'obstacle (joué ici par l'adversaire) nous allons également calculer la distance nous séparant de ce dernier .

Notre projet se subdivise en deux grandes parties bien distinctes .La première partie va toucher à l'électronique et consistera en l' émission et la réception des ultrasons .Elle nous servira aussi à traiter le signal de retour pour le rendre utilisable .La deuxième partie est celle qui touchera à l'informatique . Elle nous permettra de calculer la distance et d'afficher sur un écran LCD cette distance .

Planning et organisation du travail

Afin de travailler dans les meilleures conditions possibles et de finir dans le temps imparti, nous avons fait un planning prévisionnel. Nous avons également comparé avec la réalité.

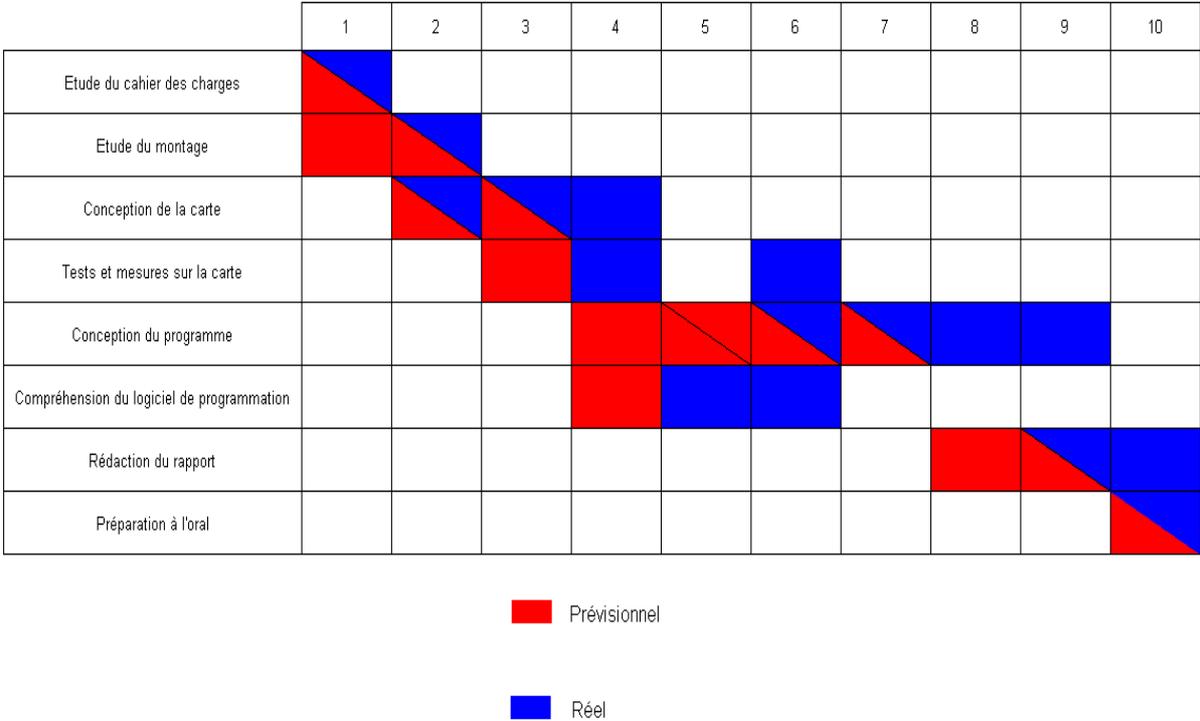


Illustration 1: Planning

Nous pouvons constater que les deux plannings sont légèrement différents mais sont assez similaires dans l'ensemble. Nous nous sommes plus attardés sur la partie programmation qui était « nouvelle » pour nous dans le sens où nous n'avions jamais programmé de micro-contrôleurs auparavant. Le travail a été réalisé en autonomie mais quelques renseignements nous ont été nécessaires et fournis par M.LEQUEU.

Diagramme sagittal

Le diagramme sagittal de notre projet est le suivant, il se décompose en quatre parties avec le micro-contrôleur au centre du projet.

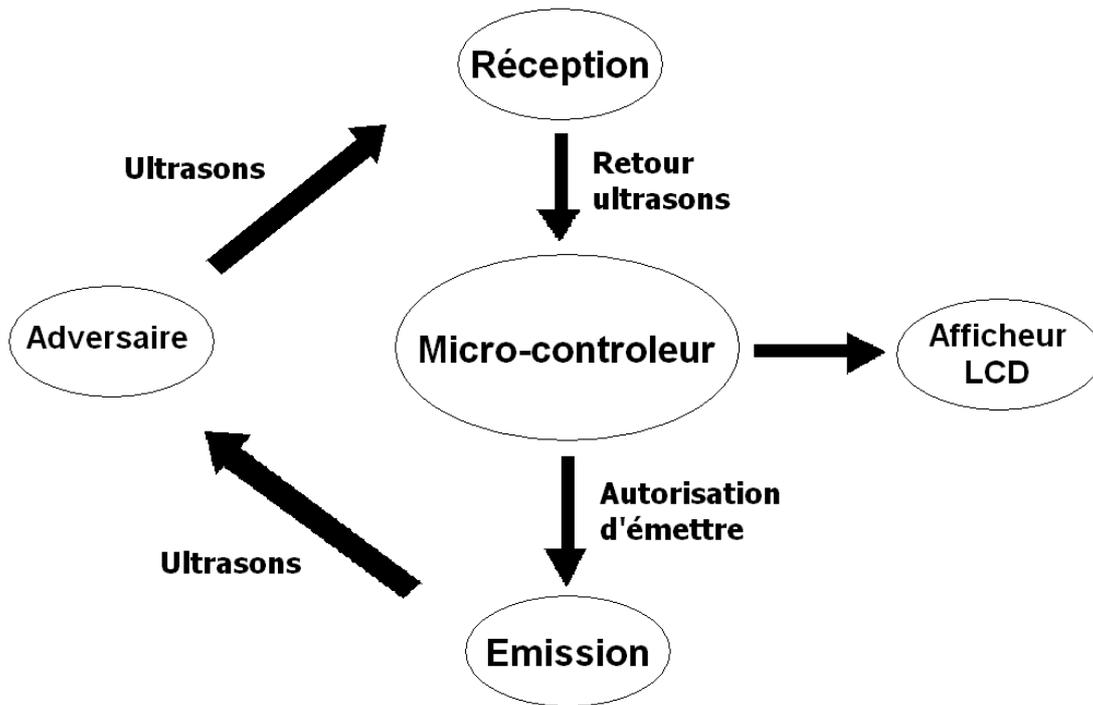


Illustration 2: Diagramme sagittal

Cahier des Charges

Comme pour tout projet que nous avons à réaliser, nous disposons d'un cahier des charges que nous devons essayer de suivre. Le cahier des charges nous impose différentes conditions qui sont autant de difficultés à surmonter.

Nous avons à suivre les directives suivantes :

- un temps limité de neuf semaines
- un rapport qualité prix raisonnable
- une capacité de détection allant de zéro à dix mètres
- un dispositif pouvant être utilisé dans des conditions difficiles
- l'utilisation du micro-contrôleur Atmega8535
- l'utilisation du logiciel Code Vision AVR
- la programmation en langage C

Vu le temps qui nous était imparti , la mise en place d'un planning prévisionnel nous a semblé nécessaire (voir annexes) .

II. Partie électronique

Pour rendre notre étude plus facile est plus efficace, nous avons décidé de diviser la partie électronique en plusieurs fonctions que nous traiterons les une après les autres .

Nous allons dans cette partie aborder les fonctions suivantes :

- émission d'un signal d'ultrasons à une fréquences de 40 Khz
- réception du signal de retour
- amplification de ce signal
- filtrage
- création d'un signal carré

Émission du signal

L'utilisation d'ultrasons à une fréquences de 40 Khz n'est pas dû au hasard . Durant notre parcours scolaire , nous avons déjà eu à utiliser des ultrasons et on a vu que c'est à cette fréquence qu'ils sont optimales .

Le but de cette fonction était donc de produire un signal carré d'une fréquence de 40 Khz . Pour cella , il nous a semblé judicieux d'utiliser un NE555 en astable en tant qu'oscillateur pour produire notre signal carré. Le NE555 est un composant largement utilisé dans l'industrie . Ceci est dû à son rapport qualité prix et sa facilité de mise en application.

Pour générer le signal , il suffit de monter le NE555 en astable et de bien choisir les composants mis aux bornes de ce derniers .Le calcul de ces composants se fait grâce à la formule suivante :

$$f = \frac{1,44}{(C1*(R1 + 2R2))}$$

Le montage en astable suit le schéma suivant :

Nous nous sommes imposés la valeur de C1 et de R1 et nous avons déduis R2. Soit

C1=1nF ; R1= 1KΩ ; R2 = 20KΩ .Pour avoir une grande possibilité de réglage, nous avons utilisé un potentiomètre de 22KΩ à la place de la résistance R2 de 20kΩ.

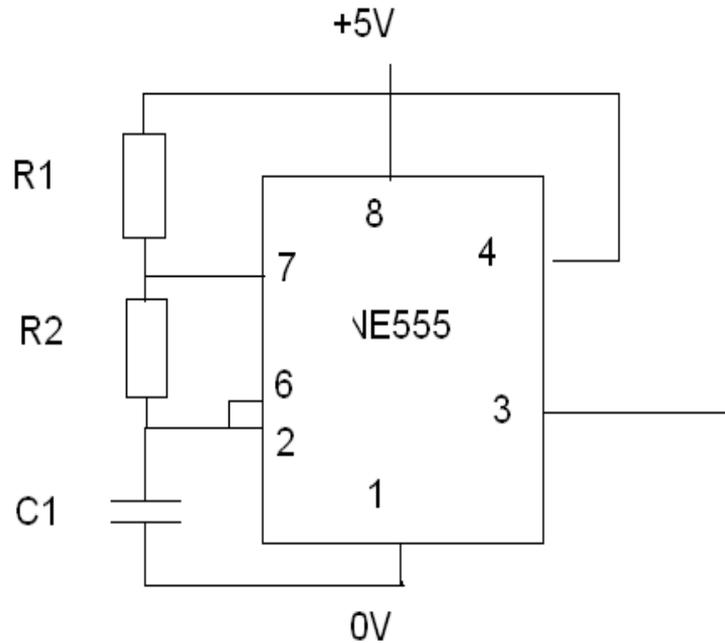


Illustration 3: NE555 monté en astable

La patte n° 4 du NE555 correspond au reset et sera relié au micro contrôleur qui de ce fait gèrera l'émission . Nous récupérerons notre signal de 40 KHz et d'amplitude 5 volts sur la patte n°3 qui correspond à la sortie .Il nous suffit par la suite de brancher un transducteur (émetteur) d'ultrasons sur cette patte pour émettre ce signal .

Ce signal s'il rencontre un obstacle (un autre KART) sera renvoyé .

Réception du signal

Cette partie à été la plus facile à mettre en application puisqu'elle se réduit seulement au fait de connecter notre récepteur à la carte .Néanmoins, après réception du signal, un traitement assez important est nécessaire. En effet , le signal reviens avec une amplitude de quelque millivolts et fortement parasité. Il nous faut donc dans un premiers temps l'amplifier et dans un deuxième le filtrer .

Amplification du signal

La perte d'amplitude de notre signal est dû à la distance que ce dernier à parcourus . Plus la distance est grande , plus l'amplitude est faible .

Pour réaliser cette amplification , on utilise deux montages à AOP (amplificateur opérationnel) montés en cascade . L'utilisation de deux AOP à pour but de réaliser une adaptation d'impédance. Les deux montages sont non inverseur .Il est important que les deux montages soient non inverseur (ou inverseur) pour avoir un signal positif .

Pour arriver a des valeur utilisables , nous avons dû réalisé une amplification de 100. Le montage en amplificateur non inverseur se réalise à l'aide de deux résistance placé sur la réaction de l' AOP .

Le montage est le suivant :

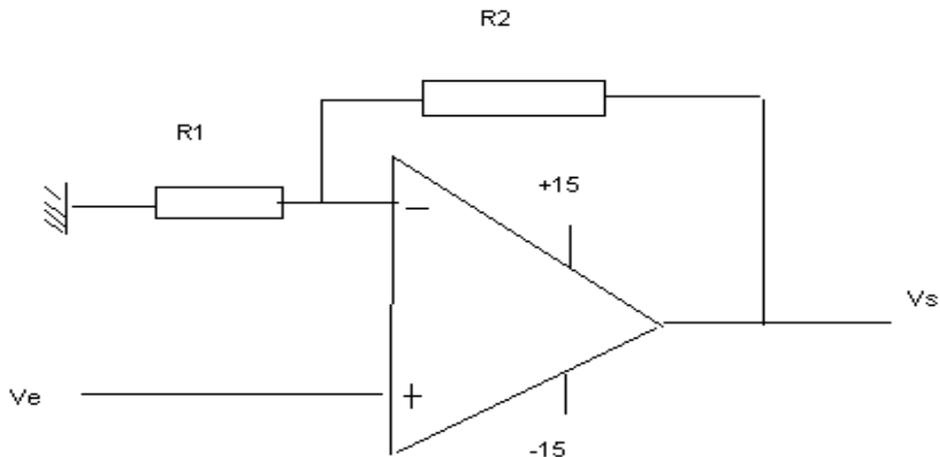


Illustration 4: AOP monté en amplificateur non inverseur

Le calcul des composants est relativement facile . Il nous suffit de calculer la fonction de transfert du montage . Sachant que nous sommes en régime linéaire , $V_+ = V_-$ or

$$V_- = V_s \left(R \frac{1}{R_1 + R_2} \right)$$

et $V_+ = 0$.

$$\text{Or : } A = \frac{V_s}{V_e}$$

$$\text{On en déduit donc que l'amplification } A = \left(1 + \left(\frac{R_2}{R_1} \right) \right)$$

Sur les deux étages d'amplification , les résistance on comme valeur $R_1 = 1\text{K}\Omega$, $R_2 = 75\text{K}\Omega$, $R_3 = 1\text{K}\Omega$, $R_4 = 10\text{K}\Omega$.

A ce stade notre signal de retour est amplifié mais il est toujours parasité . Le filtrage va nous permettre donc d'éliminer les parasites .

Le filtrage

Comme la fréquence du signal est de 40kHz, nous voulons récupérer cette même fréquence en sortie du filtre. Il va donc falloir éliminer toutes les fréquences en dessous de 39kHz et au delà de 42kHz. Ainsi, il est nécessaire de mettre en place un filtre passe-bande. Nous allons réaliser la mise en cascade d'un filtre passe bas avec un filtre passe haut ce qui nous permettra d'avoir le résultat souhaité, le filtre passe bas supprimera toutes les fréquences supérieures à 42kHz tandis que le filtre passe haut supprimera celles qui sont inférieures à 39kHz. Par conséquent, la fréquence de coupure du passe bas doit être supérieure à celle du passe haut.

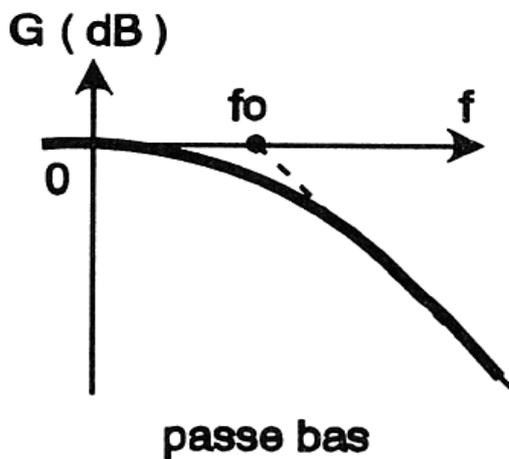


Illustration 5: diagramme de bode d'un filtre passe-bas

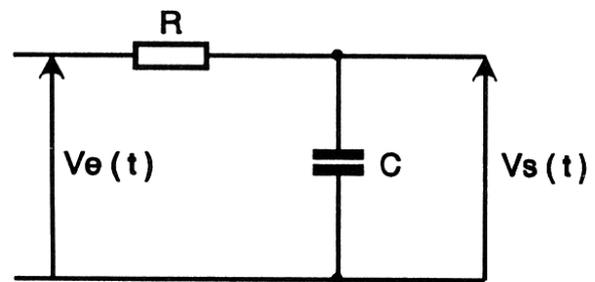


Illustration 6: filtre passe-bas

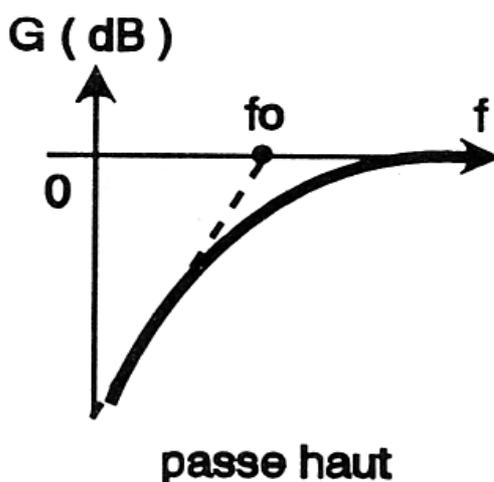


Illustration 7: Diagramme de bode d'un filtre passe-haut

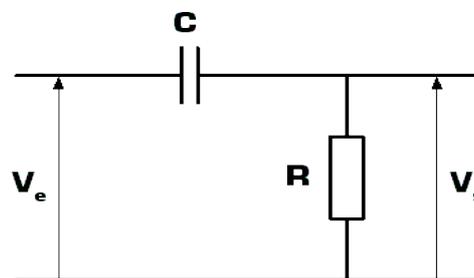


Illustration 8: filtre passe-haut

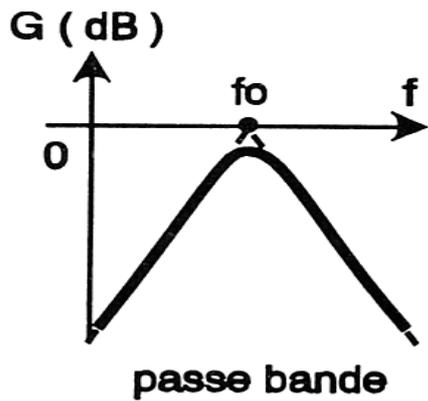


Illustration 9: Diagramme de bode d'un filtre passe-haut

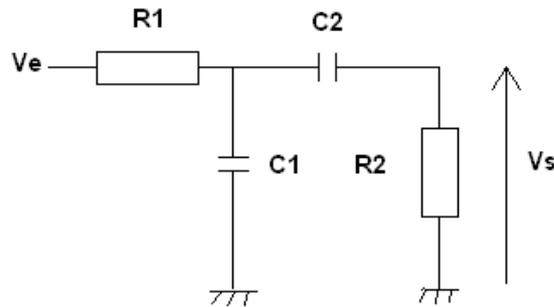


Illustration 10: filtre passe-bande

Nous savons que la fréquence de coupure d'un filtre du premier ordre est : $f_c = 1/2\pi RC$. Nous imposons les valeurs de C et C2 à 1nF, ce qui nous permet de calculer les résistances des deux filtres. Les valeurs trouvées sont $R1 = 3,7k\Omega$ et $R2 = 4,08k\Omega$.

Le signal est désormais filtré mais il va maintenant falloir le transformer afin que ce soit un signal carré.

Création d'un signal carré +5/-5V

Lors des tests nous avons réalisé qu'il nous fallait un signal carré en sortie du montage. Pour cela, nous allons intégrer deux AOP montés en comparateur, le premier permet d'obtenir le signal carré, et le deuxième permet d'atténuer le signal carré d'amplitude +/- 15V et le rabaisser à +/- 5V afin qu'il soit compatible avec le micro-contrôleur. Cette partie n'est pas sur le typon d'origine mais nous espérons pouvoir le modifier avant la fin du planning.

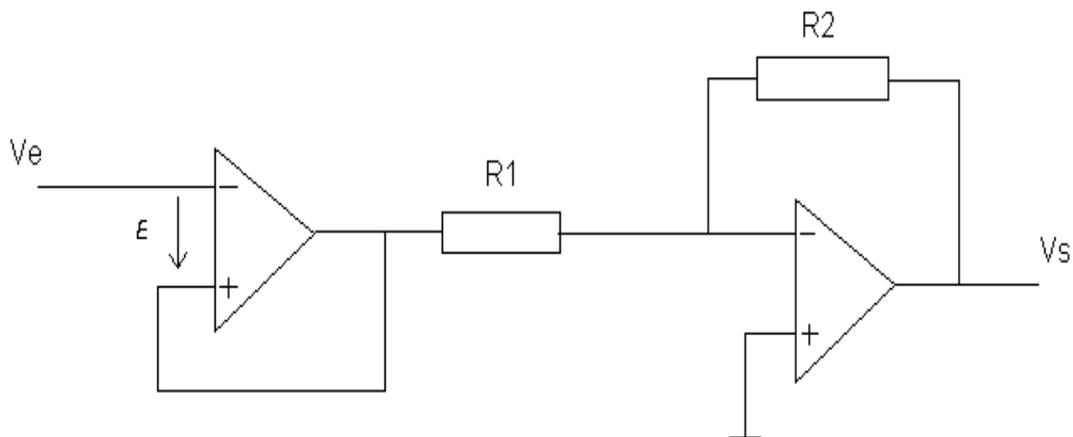


Illustration 11: Montage comparateur

A la sortie du premier comparateur, nous avons un signal carré d'amplitude +/- 15V et grâce au rapport 1/3 introduit par les résistances R1 et R2 (respectivement 300 et 100kΩ). Nous obtenons ainsi un signal carré d'amplitude +/- 5V ce qui correspond à nos attentes.

Conception de la carte

Grâce aux calculs effectués et présentés précédemment, nous pouvons désormais réaliser le typon puis, la carte électronique. Le typon a été réalisé à l'aide du logiciel Wincircuit.

Voici le typon du télémètre ultrasons :

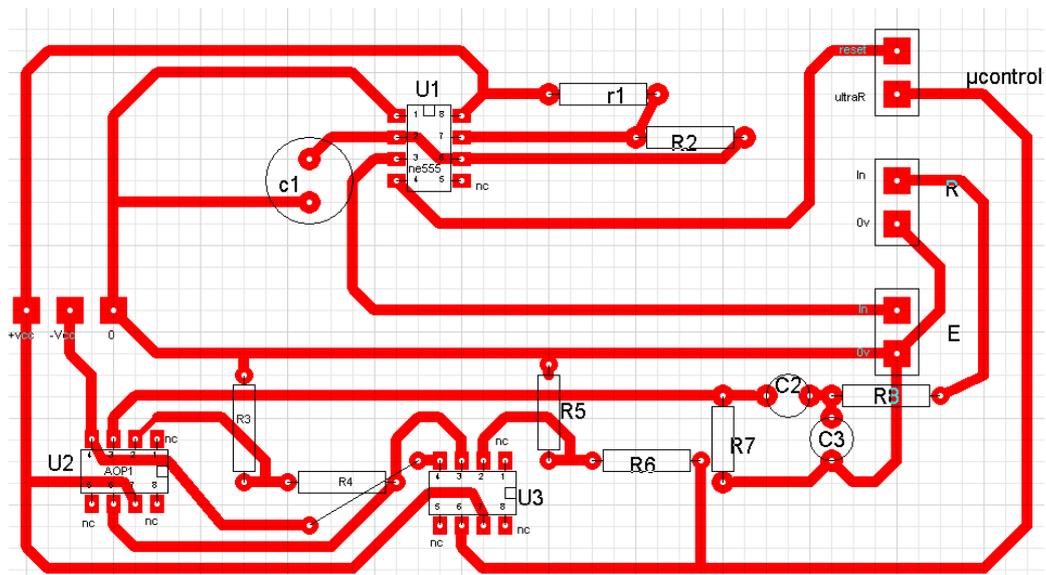


Illustration 12: Typon du télémètre

Nomenclature des différents composants utilisés :

Référence	Valeurs	Prix en €
R1	1kΩ	0,01
R2(potentiomètre)	22kΩ	0,50
R3	1kΩ	0,01
R4	75kΩ	0,01
R5	1kΩ	0,01
R6	10kΩ	0,01
R7	3,7kΩ	0,01
R8	4,08kΩ	0,01
R9	10kΩ	0,01
C1	1nF	0,20

<i>Référence</i>	<i>Valeurs</i>	<i>Prix en €</i>
C2	1nF	0,20
C3	1nF	0,20
U1	NE 555	0,24
U2/U3	TL081	0,48
Émetteur/Récepteur	S/R	7,06

Tests

FP1

Nous obtenons un signal carré 0-5V en sortie du NE 555 de fréquence 40kHz.

FP2

Nous obtenons un signal sinusoïdal de quelques mV d'amplitude et qui est parasité aux bornes du récepteur ultrason.

FP3

Le signal est de la même forme que vu précédemment mais amplifié de 100.

FP4

A la sortie du filtre, le signal est parfaitement sinusoïdal et de plus, n'est plus parasité.

FP5

Les test n'ont pas encore été réalisés pour cette fonction, mais vu la simplicité du montage, il ne devrait pas y avoir de problèmes particuliers.

Mode d'emploi

Réglages des potentiomètres :

Le potentiomètre P1 permet de régler la fréquence d'émission qui doit absolument être de 40kHz afin d'être compatible avec les autres éléments du montage.

Placement des récepteur/émetteur ultrasons :

L'émetteur et le récepteur doivent être placés perpendiculairement à l'arrière du karting et doivent avoir la même orientation. Un souci se pose cependant : dans un virage l'afficheur indiquera la distance entre la barrière de derrière et le kart mais c'est un problème insignifiant.

III. Partie informatique :

Comme dit précédemment , c'est dans cette partie que nous allons calculer la distance nous séparant de notre adversaire .

Au premier abord , on pensait que cette partie allait être la plus facile à mettre en oeuvre mais il n'en est rien.

En effet ,la prise en main du logiciel Code AVR studio a été beaucoup plus difficile que prévue .Nous avons perdu quelques séances a comprendre le fonctionnement de ce dernier.

Le Programme :

```
#include <mega8535.h>

#include<delay.h>

#include<stdio.h>

// Alphanumeric LCD Module functions

#asm

    .equ __lcd_port=0x15 ;PORTC

#endasm

#include <lcd.h>

// prototype des fonctions utilisées

interrupt [TIM1_COMPA] void timer1_compa_isr(void);

interrupt [EXT_INT0] void ext_int0_isr(void);

//déclaration des variables

int cpt;

int d;

int temps ;
```

```

//déclaration des tampons pour l'affichage ;
unsigned char tampon[20];
unsigned char tampon1[20];

// Timer 1 output compare A interrupt service routine
interrupt [TIM1_COMPA] void timer1_compa_isr(void)
{
    cpt=cpt+1;    //incréméntation du compteur
}

// External Interrupt 0 service routine
interrupt [EXT_INT0] void ext_int0_isr(void)
{

    temps=cpt/1000;    //conversion en secondes
    d=340*temps;    //calcul de distance
    cpt=0;    //raz du compteur

}

void main(void)
{

// Declare your local variables here

// initialisation des ports Entrées/sorties
//Port A en sortie
PORTA=0x00;

```

```
DDRA=0xFF;

// Port B en sortie
PORTB=0x00;
DDRB=0xFF;

// Port C en sortie
PORTC=0x00;
DDRC=0xFF;

// Port D en en entrée .
PORTD=0x00;
DDRD=0x00;

// 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: 250,000 kHz
// Mode: CTC top=OCR1A
// 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: On
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x0B;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0xFA; //valeur de comparaison (250Hz)
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      interruption sur un niveau haut sur le INT0

```

```

// INT0: On
// INT0 Mode: Rising Edge
// INT1: Off
// INT2: Off
GICR|=0x40;
MCUCR=0x03;
MCUCSR=0x00;
GIFR=0x40;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x10;

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

// Global enable interrupts
#asm("sei")

// LCD module initialization
lcd_init(16);
lcd_clear();

while (1)
{
    PORTD=0xFF;
    //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();

};

}

```

Analyse du programme :

Pour être claire dans nos explications ; nous allons expliquer le programme fonction après fonction .

Dans la première partie , on peut retrouver les différentes bibliothèques utilisées (#include ...) et les prototypes des fonctions développées dans la suite .

Les prototypes sont les derniers :

```
interrupt [TIM1_COMPA] void timer1_compa_isr(void);
```

```
interrupt [EXT_INT0] void ext_int0_isr(void);
```

Le premier correspond à une interruption interne qui nous permettra d'incrémenter notre compteur .Le deuxième quant à lui correspond à l'interruption externe qui correspond aux signal d'ultrasons venant de la carte électronique .

Interruption interne :

Nous avons passé beaucoup de temps à comprendre le mécanisme des interruptions.

Le principe est d'utiliser une horloge du Micro contrôleur afin d'exécuter une instruction précise .

Dans notre cas ; nous voulions incrémenter une variable nommée cpt (compteur) toutes les milliseconde .Nous avons choisi d'utiliser le Timer1 initialisé à 250 Khz.

Les réglages sont les suivants :

```
// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: 250,000 kHz
// Mode: CTC top=OCR1A
// 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: On
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x0B;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0xFA;
OCR1BH=0x00;
OCR1BL=0x00;
```

Nous allons comparer la valeur de notre horloge à une autre.

Avec une initialisation à 250 kHz; le compteur nous réaliser $250 \cdot 10^3$ interruptions en une seconde. En divisant par 10^3 , on trouve 250 impulsions par milliseconde. Il suffit donc

dans les réglages de préciser au micro contrôleur que lorsque le Timer1 aura réaliser 250 interruptions , notre cpt se verra incrémenté de 1.

OCR1AL=0xFA; cette ligne nous permet de faire la comparaison entre la valeur de Timer1 et une valeur A égale à 250 (ce qui correspond à 0xFA en hexadécimale).

la partie correspondante à cette instruction est la suivante :

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

Interruption externe :

Dans cette fonction , on demande au micro contrôleur de mémoriser la valeur de notre compteur dans une variable appelée temps lorsqu'on à un retour d'ultrasons .

Ceci se fait par un interruption externe lorsque nous aurons un front montant au niveau de l'entrée INT0.

Nous allons également calculer la distance qui est égale au temps multiplié par la vitesse du son (340m/s).

```
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
}
```

Affichage sur l'écran LCD :

Nous disposons d'un écran LCD de 4 lignes et de 16 colonnes .

Par réglage , le micro contrôleur assigne le port C à l'écran .

Nous nous sommes inspiré d'un programme existant pour réaliser cette dernière fonction .

On initialise l'écran, on efface le texte existant .

Pour l'écriture sur l'écran , on utilise la fonction

```
lcd_puts(tampon);
```

cette dernière affiche ce qu'il y a dans tampon .

Tampon est une variable char de 20 caractère dans lequel nous copions les valeurs du temps .Ceci se fait grace à la fonction

```
sprintf(tampon,"temps =%5d s",temps).
```

```
// LCD module initialization
```

```
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();
```

```
};
```

N.B. : On peut noter au début du « while » dans le « main » la ligne suivante « PORTD=0xFF ». Elle nous permet de mettre un signal de 5V au niveau du reset du NE 555, ce qui autorisera son fonctionnement.

IV. Problèmes rencontrés

Le problème majeure que nous avons rencontré est la partie programmation de micro-contrôleur. En effet, nous n'avions jamais programmé ce genre de composant auparavant et la compréhension du langage et des différentes instructions du micro-contrôleur nous ont faits buter et il nous a fallut rechercher des documentations et des cours sur internet. Certes nos connaissances en langage C nous ont fortement aidés car les bases sont les mêmes.

L'imprévu majeure de la partie électronique fut le rajout à la dernière minute de montage comparateur. Nous avons dû refaire le typon et donc créer une nouvelle carte, ce qui nous a ralenti fortement dans la finalisation du projet.

Conclusion

En conclusion ce projet fut pour nous un défi personnel puisque nous sommes parti de zéro et avons pu réaliser en entier le projet . Malgré cela , nous avons réussi à mettre au point un système fiable et simple par sa structure .

Nous pensons que notre système réponds parfaitement aux attentes du cahier des charges puisqu'il opérationnel et peut désormais être installé sur la karting ou tout autre véhicule.

Ce projet nous a été bénéfique puisque nous avons mené une étude complète (théorie, essai, réalisation, programmation et test) . Nous avons pu mettre en pratique les compétence acquises lors des deux années du DUT (montages électroniques, langage C...). De plus nous avons découvert la programmation de micro-contrôleurs, composant essentiel dans l'industrie mais que nous n'avions pas étudiés durant ces deux ans.

Nous tenons à remercier M.Paul LEREBOURG et M. Florent PROU pour leur contribution à la partie programmation, notamment dans la compréhension des différentes fonctions liées aux interruptions et au logiciel Code Vision AVR.

Index des illustrations

Illustration 1: Planning.....	6
Illustration 2: Diagramme sagittal.....	7
Illustration 3: NE555 monté en astable.....	9
Illustration 4: AOP monté en amplificateur non inverseur.....	10
Illustration 5: diagramme de bode d'un filtre passe-bas.....	11
Illustration 6: filtre passe-bas.....	11
Illustration 7: Diagramme de bode d'un filtre passe-haut.....	11
Illustration 8: filtre passe-haut.....	11
Illustration 9: Diagramme de bode d'un filtre passe-haut.....	12
Illustration 10: filtre passe-bande.....	12
Illustration 11: Montage comparateur.....	12
Illustration 12: Typon du télémètre.....	13

Annexes