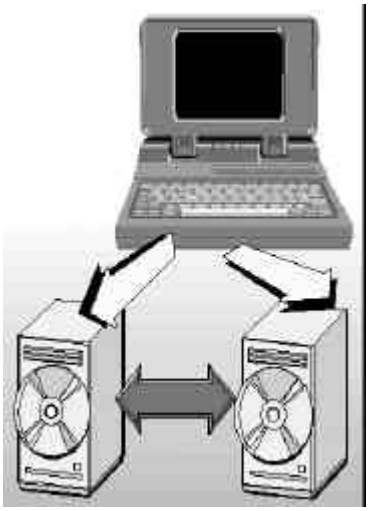
Chenillard Sonore
Projet 2002-2003

Claire Billaud & Benoît Lucardi
Professeur : F. Ghibaudo

SOMMAIRE

PRESENTATION DU PROJET

Sur le même principe que le "chenillard lumineux" utilisé sur les routes, le chenillard sonore a pour but de "véhiculer" un son entre plusieurs haut-parleurs, le tout sous le contrôle d'une unité centrale. Ici, l'unité centrale est une carte-mère de PC, utilisée sous DOS et munie d'un programme permettant de "piloter" le transfert de son. Les périphériques (désignés également par le nom de "satellites") sont constitués chacun d'un haut-parleur et d'un système permettant de recevoir et de décoder les ordres de l'unité centrale.



CHOIX DE CONCEPTION

Comme on l'a dit précédemment, l'unité centrale se base sur une carte-mère de PC (486) fonctionnant sous DOS. N'ayant pas la carte-mère, nous avons utilisé comme unité centrale un PC du labo sous DOS.

Le fonctionnement du chenillard se base sur deux circuits indépendants. L'un est destiné à la transmission analogique d'un son simple par l'intermédiaire d'une carte-son de type Sound Blaster incorporée à la carte-mère. L'autre est destiné à transmettre des ordres numériques de contrôle de volume permettant de constituer l'effet "chenillard".

À ce jour, seul le circuit numérique de contrôle de volume est réalisé, la partie analogique reste à faire.

La transmission, aussi bien du son que des ordres, devait se faire par ondes hertziennes. Il était prévu de monter un circuit AM à 433,3 Mhz pour la transmission des ordres et un circuit FM pour la transmission du son. Mais (sans doute à cause d'un problème dans le montage des antennes), la transmission radio AM n'a jamais bien marché et les ordres de l'unité centrale sont actuellement transmis par voie filaire.

Le chenillard sonore se compose donc de 2 éléments principaux : la carte émettrice (à brancher sur le port série du PC "Lefuneste" dans la salle C13) et la carte réceptrice. Nous avons également fabriqué une carte réceptrice test comportant une série de LED et destinée à vérifier la programmation du microcontrôleur PIC de la carte réceptrice (voir le chapitre consacré au récepteur).

L'EMETTEUR (L'UNITE CENTRALE)

Description générale

La plaque émettrice est destinée à être raccordée sur le port série du poste "Lefuneste" du labo C13. Pour cela, il faut redémarrer l'ordinateur en mode MS-DOS, débrancher la souris et brancher le connecteur série de la carte à sa place.

La plaque comporte 3 douilles à brancher :

- l'alimentation +5V (borne rouge)
- l'alimentation 0V (borne noire)
- la sortie signal (borne bleue).

Le but de la carte émettrice est le suivant : le signal issu du port série du PC est un signal +12V / -12V. Or, l'émetteur est prévu pour émettre du signal TTL/CMOS, soit du +5V / 0V, et le microcontrôleur PIC de la carte réceptrice est également prévu pour recevoir des signaux TTL/CMOS. Le travail essentiel de la carte consiste donc à convertir le signal série en TTL/CMOS et à l'émettre par l'intermédiaire de l'émetteur ou de la sortie signal.

Utilisation du port série

Le port série envoie des données octet par octet en série (d'où son nom). Lorsqu'il n'y a pas de données, la ligne est à l'état haut (+12V). L'émission d'un octet en série commence par un bit de départ à l'état bas (-12V), puis l'envoi de l'octet bit par bit en commençant par le bit de poids faible (LSB), puis l'ajout (ou non) d'un bit de parité destiné à vérifier l'exactitude de l'octet, puis un ou plusieurs bits de stop à l'état haut.

L'émission de ces octets est configurable, que ce soit au niveau de la vitesse, du nombre de bits envoyés, de la présence ou non d'un bit de parité et du nombre de bits de stop. Toute cette configuration se fait ici par l'intermédiaire d'un programme en C écrit sous Turbo C. Côté matériel, on utilise ici le port série le plus simplement possible, en ne câblant que les broches "Signal émis" et "Masse du signal". On n'utilise pas les fonctions spéciales du port série car elles sont inutiles pour ce qu'on a à faire.

Configuration de l'émission série

Comme on l'a dit précédemment, le mode d'émission du port série est configurable. En langage C, il existe un moyen simple de programmer le port série, à condition de connaître l'emplacement de ses registres de contrôle et d'utiliser la fonction `outportb(numport, octet)` qui envoie un octet *octet* sur le port numéroté *numport*, qui peut être un port matériel ou un registre. À partir de cela, on peut construire une fonction d'initialisation du port série qui configure l'émission de 8 bits à 1200 bauds, sans parité et avec 1 bit de stop :

```
#include <stdio.h>
#include <conio.h>
#include <dos.h>
/* dos.h est la librairie contenant les prototypes des
  fonctions inportb et outportb (sous Borland)*/
```

```
#define COM1 0x3F8

// on définit ici le port courant
#define COMx COM1

/* Adresses des registres
  Pour plus de modularité, on fait
  un offset par rapport à l'adresse de base */
#define TxD COMx
#define RxD COMx
#define DLLB COMx
#define IER (COMx+1)
#define DLHB (COMx+1)
#define IIR (COMx+2)
#define FCR (COMx+2)
#define LCR (COMx+3)
#define MCR (COMx+4)
#define LSR (COMx+5)
#define MSR (COMx+6)


void Init_COM(void)
{
outportb(LCR,0x80); // DLAB = 1

// 1200 bauds
outportb(DLLB,0x60); // Vitesse adresse basse
outportb(DLHB,0x00); // Vitesse adresse haute

outportb(LCR,0x03);
/* DLAB = 0
  Pas de parité
  8 bits de données
  1 bit de stop */
}
```

Vous pouvez lire l'explication complète de ce programme et de la configuration de l'émission série sur Internet :
http://www.programmationworld.com/site/Cours.asp?Action=cours&Numero=26

Programmation C de l'émission d'ordres

Le principe de l'émission d'ordres de volume est le suivant : un ordre est composé de 2 octets.

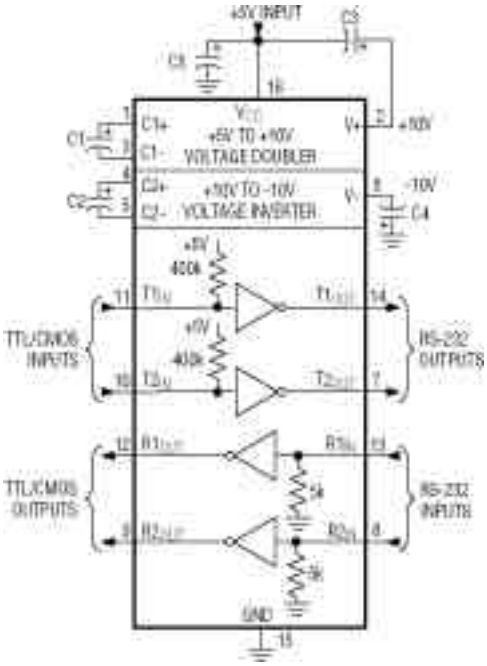
- Un octet divisé en 2 : les 4 MSB pour l'adresse du satellite destinataire (0001 car on n'a ici qu'un seul satellite, ou 0000 qui correspond à tous les satellites) et les 4 LSB pour le code de la fonction (ici 0001 car on n'a qu'une seule fonction : l'ordre de volume simple). Le code de la fonction doit être impair (LSB=1) pour que le satellite distingue l'octet adresse/fonction de l'octet d'ordre de volume. On peut donc créer 3 fonctions différentes.

- Un octet de donnée, pour le contrôle du volume : pour des raisons liées à la conception du récepteur (voir plus loin), ici *FE* correspond au volume le plus faible et *00* au volume le plus fort.

Par ailleurs, l'ordre de volume doit être un octet pair (LSB=0) pour que le satellite le distingue de l'octet adresse/fonction qui le précède.

Le listing du programme est donné en annexe. Ce programme envoie un octet adresse/fonction (adresse 1 et fonction 0) et un octet de volume définis dans le programme.

La conversion en TTL/CMOS : le driver de ligne série MAX 232



Les ordres délivrés ainsi par le port série sont en -12V / +12V. Afin de les rendre utilisables pour l'émetteur et pour le microcontrôleur PIC des satellites, ils doivent être convertis en signal TTL/CMOS, soit 0 / +5V. Il existe des circuits intégrés qui effectuent directement cette conversion, qu'on appelle drivers de ligne série. Le MAX-232 est l'un d'entre eux. Nous l'avons choisi pour notre projet en raison de sa grande simplicité d'utilisation.

Le schéma de brochage du MAX-232 est donné ci-dessous. Pour plus de précisions, voir la datasheet du MAX-232 donnée en annexe.

Adresse (4 bits)	Fonction (4 bits)	Donnée de volume (8 bits)

On voit que le MAX-232 ne demande que 4 condensateurs de 1 µF pour fonctionner. Le signal RS-232 issu du port série est branché sur les broches 8 et 13 *(RS-232 Inputs)* et les broches 9 et 12 *(TTL/CMOS Outputs)* alimentent en TTL/CMOS soit l'émetteur, soit la borne de sortie signal.

L'émetteur AM

Même si nous avons dû abandonner l'émission radio au profit d'une émission filaire suite à des problèmes d'antenne, l'émetteur radio de la carte émettrice est toujours en place et théoriquement en état de fonctionnement (ce qui n'est pas le cas de son antenne, les composants RF, notamment, étant probablement débranchés). Il s'agit d'un émetteur radio AM à 433 Mhz destiné à émettre un signal TTL/CMOS dont la fréquence va jusqu'à 1200 bauds, d'où le choix de la vitesse d'émission au niveau du port série. Il s'agit d'un AM-TX1-433 de RF Solutions dont la datasheet est donnée en annexe.

Le montage de cet élément est donné dans sa datasheet ; la résistance à utiliser pour un signal 0V / 5V est d'environ 470 ohms (minimum 435 ohms). La datasheet de cet émetteur inclut également la description des antennes utilisables avec les modules émetteur/récepteur de RF Solutions. Pour le projet, nous avons essayé de monter des antennes fouet (Short Whip Antenna) pour leur caractère omnidirectionnel, mais l'émission ne se faisait plus correctement au-delà de 30 cm d'écart entre les cartes émettrice et réceptrice, si bien que nous avons abandonné l'émission radio faute de pouvoir résoudre ce problème.

LE RECEPTEUR (LE "SATELLITE")

Description générale

La carte réceptrice est destinée à être intégrée à un "satellite". Sa fonction est le contrôle du volume du son transmis aux haut-parleurs.

Elle comporte 7 douilles à brancher :

- l'alimentation +5V (borne rouge)
- l'alimentation 0V (borne noire)
- l'alimentation +15V (borne verte)
- l'alimentation -15V (borne jaune)
- le signal issu de la carte émettrice (borne bleue marquée "Sig", dans le coin)
- l'entrée de signal analogique à amplifier (borne bleue marquée "Vin", la plus proche du VCA)
- la sortie de signal analogique amplifié (borne blanche)

Le fonctionnement de la carte est le suivant : le signal de commande issu de la carte émettrice est envoyée sur un microcontrôleur PIC. Celui-ci, en décodant les informations d'adresse, de fonction et de volume, envoie un octet d'ordre de volume sur un VCA (Voltage Controlled Amplifier, amplificateur contrôlé en tension) par l'intermédiaire d'un CNA (Convertisseur Numérique vers Analogique).

La carte test

Elle a été conçue pour tester le fonctionnement du microcontrôleur PIC avant de continuer le montage de la carte réceptrice. Elle comprend 3 broches :

- l'alimentation +5V (borne rouge)
- l'alimentation 0V (borne noire)
- le signal issu de la carte émettrice (borne bleue)

Elle contient un microcontrôleur PIC dont le port D est branché sur une série de 8 diodes électroluminescentes, ce qui permet de visualiser la réponse du PIC aux signaux de la carte émettrice quand la borne signal est branchée.

Important : le programme du PIC est conçu pour fonctionner avec une horloge de 8 Mhz, mais nous ne disposons que d'une horloge 8Mhz pour 2 cartes à PIC. Il est donc nécessaire de s'assurer que l'horloge 8 Mhz est branchée avant de tester le PIC.

Le récepteur AM

Il s'agit du récepteur couplé à l'émetteur AM vu précédemment, un AM-HRR3-433 de RF Solutions. Sa datasheet est donnée en annexe.

Il est monté sur la carte test, pas sur la carte réceptrice. De plus, sur la carte test, son montage a été plusieurs fois défait et refait, et il sera probablement nécessaire de le refaire entièrement, et de même pour l'antenne qui se trouve à côté (les composants RF de l'antenne, en particulier, sont pour la plupart débranchés). Le signal issu de la carte émettrice est actuellement transmis par fil (borne signal).

Le microcontrôleur PIC

Il s'agit d'un microcontrôleur doté d'une mémoire Flash/EEPROM, de plusieurs ports d'entrée/sortie dont un port série asynchrone, et très facile à programmer grâce au logiciel MPLAB installé sur les PC du labo C13. Pour programmer le PIC, il faut brancher le programmateur PICStart sur un port série du PC, activer le PICStart à partir de MPLAB et lancer la commande "Program". Les bits de configuration utilisés pour le programme sont :

Oscillator : XT - Watchdog : Off - PowerUp Timer : On - Code Protect : Off - Brown Out Detect : On - Low Voltage Program : On - Data EE Protect : Off - Flash Program Write : Enable - Background Debug : Disable

Le microcontrôleur PIC utilisé ici est un PIC16F877-04/P. Les extraits de sa datasheet utiles pour le projet sont donnés en annexe ; la datasheet complète est téléchargeable gratuitement sur le site de Microchip *http://www.microchip.com* ou dans le CD-ROM du projet.

Remarque : Pour brancher le PIC correctement, il est nécessaire de découpler ses alimentations avec 2 condensateurs (typiquement de 100 nF) et de brancher l'entrée RESET (broche n°1) sur le 5V afin d'éviter des remises à zéro accidentelles.

La réception série asynchrone du microcontrôleur PIC

Le microcontrôleur PIC16F877 dispose d'un port série asynchrone appelé USART (Universal Serial Asynchronous Receiver/Transmitter). C'est ce port que nous utilisons pour recevoir les données de la carte émettrice.

Une partie importante de la programmation du PIC consiste à configurer le port USART :

```
bsf         STATUS,RP0  ; passage BANK1
bcf         TXSTA,BRGH  ; baud rate lent
bcf         TXSTA,SYNC  ; mode asynchrone
movlw       103
movwf       SPBRG       ; définition du baud rate (1200)
bsf         TRISC,7
bsf         TRISC,6     ; activation du port série (RC6
                        ;                   et RC7)
bcf         STATUS,RP0  ; passage BANK0
bsf         RCSTA,SPEN  ; activation port série
bcf         RCSTA,RX9   ; 8 bits
bsf         RCSTA,CREN  ; réception continue
bcf         RCSTA,FERR
bcf         RCSTA,OERR  ; reset des bits d'erreur
```

Programmation du microcontrôleur PIC pour la réception des ordres

Le programme passé au PIC pour la réception des ordres fonctionne par interruptions. Le port de sortie destiné à recueillir l'octet de contrôle du volume est le port D.

Le principe du programme est le suivant :

- le PIC reçoit 2 types d'octets qu'il distingue par leur LSB (1 : adresse/fonction, 0 : donnée de volume)

- après l'initialisation de toutes les variables, le programme entre dans la boucle Bo ; il ne quittera plus cette boucle jusqu'à l'arrêt du PIC. La seule fonction de cette boucle est de faire sortir sur le port D la valeur de la variable de volume "vol".

- Cette boucle est coupée par une interruption lors de la réception d'un octet sur le port série du PIC.

- Si l'octet est du 1er type (xxxxxxx1), il lance un sous-programme et teste si l'adresse (les 4 MSB) est celle du pic, si c'est le cas la variable "flag" passe à 1 et les 3 autres bits (b3,b2,b1) de l'octet sont sauvegardés dans la varible "fonc" (pour fonction). Puis on sort de l'interruption.

- Si l'octet est du type xxxxxxx0 et le flag à 0, le PIC sort immédiatement de l'interruption.

- Si l'octet est du type xxxxxxx0 et le flag à 1, on lance le sous-programme correspondant à la fonction appelée (seule la fonction 0 (ordre de volume simple) a été réalisée). Dans le sous-programme f0, le flag repasse à 0 et on enregistre la valeur de la variable "com" dans "vol". Les 7 msb sont sauvegardé dans la variable "com" puis on sort de l'interruption.

Le listing de ce programme est donné en annexe.

Conversion des ordres issus du PIC : l'ensemble CNA/VCA

Une fois que le PIC a généré l'octet de contrôle du volume, il faut l'envoyer sur un circuit qui modifie l'amplitude d'un signal en fonction de cet octet. Pour cela, on utilise un VCA (Voltage Controlled Amplifier) qui reçoit l'octet par l'intermédiaire d'un CNA.

Le CNA : il s'agit d'un DAC08 d'Analog Devices dont la datasheet est donnée en annexe. Il fonctionne en mode "Basic Positive Référence Operation". Dans ce mode, le courant qu'il délivre est donné par la formule

$$I_0=\frac{Vref}{Rref}*\frac{Octet}{256}$$

La sortie du CNA est un drain de courant : afin de transformer le courant négatif I_0 en une tension positive, on utilise un AO inverseur classique avec une résistance, de manière à avoir V_c (tension de contrôle pour le VCA)=I_0*R (ce montage est indiqué dans la datasheet du CNA sous le nom "Positive Low Impedance Output Operation").
Avec *Vref=5V, Rref=1,2 Kohms* et *R=1 Kohm*, le CNA délivre une tension entre 0V et 4,15V environ.

Le VCA : afin d'assurer le maximum de compatibilité entre le CNA et le VCA, nous avons choisi un VCA d'Analog Devices ; il s'agit du SSM2018T dont la datasheet est donnée en annexe.

Ce VCA est en fait un atténuateur contrôlé en tension : 0V en entrée correspond à aucune atténuation, 4V à l'atténuation maximale (100 dB). C'est pour cette raison que l'octet de volume maximal (*FE*) correspond au volume minimum.

Les branchements du VCA par rapport au CNA sont les mêmes que ceux données dans la datasheet du VCA (*Figure 44 : DAC controls the VCA gain*)

En testant ce montage, nous nous sommes rendu compte que si V_c=4V, le VCA produit une atténuation de 100 dB. Selon le signal fourni en entrée, cette atténuation peut être trop forte ; en

testant avec un signal d'entrée de 5V, on a vu que l'atténuation est si rapide qu'avec un volume au-dessous de 200 (le volume allant de 0 à 256) on n'a déjà plus rien en sortie. Selon le signal fourni par le circuit analogique au niveau du VCA, il pourra être nécessaire de modifier la gamme de valeurs de V_c. Pour cela, il suffit de changer la résistance R placée sur l'AO : diviser R par 2 permet de réduire de moitié la gamme de valeurs de V_c, par exemple.